



개발자 안내서

# AWS RoboMaker



# AWS RoboMaker: 개발자 안내서

Copyright © 2024 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

상표 및 브랜드 디자인은 타사 제품 또는 서비스와 함께, Amazon 브랜드 이미지를 떨어뜨리거나 고객에게 혼동을 일으킬 수 있는 방식으로 사용할 수 없습니다. Amazon이 소유하지 않은 기타 모든 상표는 Amazon과 제휴 관계이거나 관련이 있거나 후원 관계 여부에 관계없이 해당 소유자의 자산입니다.

# Table of Contents

AWS RoboMaker란 무엇입니까? .....	1
기능 .....	1
요금 .....	2
시작하기 .....	3
개념 .....	3
컨테이너 .....	3
시뮬레이션 작업 .....	4
Simulation WorldForge .....	4
Environment .....	4
애플리케이션 .....	4
애플리케이션 구성 .....	5
설정 .....	5
가입하세요. AWS 계정 .....	5
관리 권한이 있는 사용자 생성 .....	6
첫 시뮬레이션 실행 .....	7
개발 .....	8
애플리케이션 컨테이너 빌딩 .....	8
필수 조건 .....	8
ROS 작업 영역에서 애플리케이션 컨테이너 구축 .....	9
컨테이너 테스트 .....	14
애플리케이션 컨테이너 게시 .....	14
애플리케이션 버전 관리 .....	16
로봇 애플리케이션 작업 .....	17
로봇 애플리케이션 생성 .....	17
로봇 애플리케이션 버전 생성 .....	18
로봇 애플리케이션 보기 .....	19
로봇 애플리케이션 업데이트 .....	19
로봇 애플리케이션 삭제 .....	20
로봇 애플리케이션 버전 삭제 .....	20
시뮬레이션 애플리케이션 작업 .....	21
시뮬레이션 애플리케이션 생성 .....	21
시뮬레이션 애플리케이션 버전 생성 .....	22
시뮬레이션 애플리케이션 보기 .....	22
시뮬레이션 애플리케이션 업데이트 .....	23

시뮬레이션 애플리케이션 삭제 .....	23
시뮬레이션 애플리케이션 버전 삭제 .....	24
애플리케이션 버전 지정 .....	24
이미지를 사용한 애플리케이션 버전 관리 .....	25
\$LATEST 버전 .....	26
애플리케이션 버전 업데이트 .....	26
애플리케이션 버전 삭제 .....	27
이미지를 사용하여 애플리케이션 개발 .....	27
ROS 애플리케이션을 컨테이너로 마이그레이션 .....	28
ROS 컨테이너 FAQ .....	28
AWS RoboMaker 호환 컨테이너에 대한 요구 사항 .....	33
GPU 애플리케이션을 실행하기 위한 이미지 생성 .....	52
hello world 샘플 애플리케이션을 실행하기 위한 이미지 생성 .....	53
Simulation .....	73
시뮬레이션 실행 .....	73
시뮬레이션 구성 .....	76
Amazon VPC 액세스를 위한 시뮬레이션 작업 구성 .....	77
시뮬레이션 작업에 대한 인터넷 액세스 .....	78
SimulationJob 컴퓨팅 구성 .....	79
사용자 지정 시뮬레이션 도구 구성 .....	79
루트 액세스 및 시스템 기능 .....	80
시뮬레이션 관리 .....	82
시뮬레이션 작업 생성 .....	82
시뮬레이션 작업 보기 .....	87
시뮬레이션 작업 취소 .....	88
시뮬레이션 작업 복제 .....	88
시뮬레이션 작업 다시 시작 .....	89
시뮬레이션 로깅 .....	90
사용자 지정 업로드 구성 추가 .....	91
AWS RoboMaker에 의해 생성된 환경 변수 .....	92
배치 시뮬레이션 .....	93
시뮬레이션 작업 배치 시작 .....	93
시뮬레이션 작업 배치 보기 .....	95
시뮬레이션 작업 배치 취소 .....	96
시뮬레이션 작업 배치 복제 .....	96
월드 생성 .....	98

Simulation WorldForge 개념 .....	98
시뮬레이션 월드 템플릿 알아보기 .....	99
평면도 .....	99
인테리어 .....	100
일반적인 작업 .....	103
한 층의 방 목록 지정 .....	104
긴 복도 요청 .....	105
방 사이 출입구 요청 .....	106
모든 방에 구성 적용 .....	107
출입구에 문 요청 .....	108
출입구에 문 없음 요청 .....	110
넓은 평면도 공간 요청 .....	111
사용자 지정 천장 높이 요청 .....	111
여러 방의 바닥에 동일한 자재 유형 지정 .....	112
같은 유형의 방 사이의 바닥에 서로 다른 자재 유형 지정 .....	113
방에 가구를 더 많이, 적게 지정 .....	114
모든 침실 및 단일 공용 거실/식당에 특정 가구 유형 추가 .....	116
가구가 없는 방 지정 .....	118
시뮬레이션 월드 템플릿 본문을 위한 JSON 스키마 .....	119
JSON의 샘플 월드 템플릿 .....	157
침실이 하나인 집 .....	158
방 한 개 .....	162
방 두 개 .....	163
시뮬레이션 월드 템플릿 관리 .....	165
템플릿 생성 .....	165
템플릿 보기 .....	183
템플릿 수정 .....	184
템플릿 삭제 .....	185
템플릿 릴리스 .....	186
월드 생성 작업 관리 .....	188
작업 생성 .....	188
작업 보기 .....	189
작업 취소 .....	190
월드 내보내기 작업 관리 .....	191
내보내기 작업 생성 .....	191
내보내기 작업 보기 .....	192

시뮬레이션에서 내보낸 월드 사용 .....	193
내보낸 월드를 데이터 소스로 사용 .....	194
ROS와 Gazebo에서 내보낸 월드 사용 .....	196
사용자 지정 물리, 조명, 모델이 포함된 내보낸 월드 사용 .....	197
보안 .....	199
데이터 보호 .....	199
인증 및 액세스 제어 .....	200
권한 부여 및 액세스 제어 소개 .....	200
필요한 권한 .....	201
AWS RoboMaker에서 IAM을 사용하는 방식 .....	208
인증 및 액세스 제어 문제 해결 .....	209
정책이란 무엇입니까? .....	210
AWS 관리형 정책 .....	213
서비스 연결 역할 사용 .....	217
IAM 시작하기 .....	220
로깅 및 모니터링 .....	222
Amazon CloudWatch를 사용한 AWS RoboMaker 모니터링 .....	222
AWS CloudTrail을 사용한 호출 로깅 .....	225
리소스에 태그 지정 .....	227
태그 기본 사항 .....	227
태그 규제 및 제한 .....	228
IAM 정책에 태그 사용 .....	229
보안 규정 준수 .....	231
복원성 .....	231
인프라 보안 .....	232
VPC 엔드포인트(AWS PrivateLink) .....	232
AWS RoboMaker VPC 엔드포인트에 대한 고려사항 .....	232
AWS RoboMaker에 대한 인터페이스 VPC 엔드포인트 생성 .....	233
AWS RoboMaker에 대한 VPC 엔드포인트 정책 생성 .....	233
API 참조 .....	235
작업 .....	235
BatchDeleteWorlds .....	238
BatchDescribeSimulationJob .....	241
CancelDeploymentJob .....	247
CancelSimulationJob .....	250
CancelSimulationJobBatch .....	253

CancelWorldExportJob .....	256
CancelWorldGenerationJob .....	259
CreateDeploymentJob .....	262
CreateFleet .....	271
CreateRobot .....	275
CreateRobotApplication .....	280
CreateRobotApplicationVersion .....	286
CreateSimulationApplication .....	291
CreateSimulationApplicationVersion .....	298
CreateSimulationJob .....	304
CreateWorldExportJob .....	320
CreateWorldGenerationJob .....	327
CreateWorldTemplate .....	335
DeleteFleet .....	340
DeleteRobot .....	343
DeleteRobotApplication .....	346
DeleteSimulationApplication .....	349
DeleteWorldTemplate .....	352
DeregisterRobot .....	355
DescribeDeploymentJob .....	359
DescribeFleet .....	365
DescribeRobot .....	370
DescribeRobotApplication .....	375
DescribeSimulationApplication .....	380
DescribeSimulationJob .....	386
DescribeSimulationJobBatch .....	397
DescribeWorld .....	409
DescribeWorldExportJob .....	413
DescribeWorldGenerationJob .....	419
DescribeWorldTemplate .....	426
GetWorldTemplateBody .....	430
ListDeploymentJobs .....	433
ListFleets .....	438
ListRobotApplications .....	443
ListRobots .....	447
ListSimulationApplications .....	451

ListSimulationJobBatches .....	456
ListSimulationJobs .....	460
ListTagsForResource .....	464
ListWorldExportJobs .....	467
ListWorldGenerationJobs .....	471
ListWorlds .....	475
ListWorldTemplates .....	479
RegisterRobot .....	483
RestartSimulationJob .....	487
StartSimulationJobBatch .....	490
SyncDeploymentJob .....	506
TagResource .....	513
UntagResource .....	516
UpdateRobotApplication .....	519
UpdateSimulationApplication .....	525
UpdateWorldTemplate .....	532
데이터 유형 .....	535
BatchPolicy .....	538
Compute .....	539
ComputeResponse .....	541
DataSource .....	543
DataSourceConfig .....	545
DeploymentApplicationConfig .....	548
DeploymentConfig .....	550
DeploymentJob .....	552
DeploymentLaunchConfig .....	555
Environment .....	557
FailedCreateSimulationJobRequest .....	558
FailureSummary .....	560
Filter .....	561
FinishedWorldsSummary .....	562
Fleet .....	564
LaunchConfig .....	566
LoggingConfig .....	569
NetworkInterface .....	570
OutputLocation .....	572

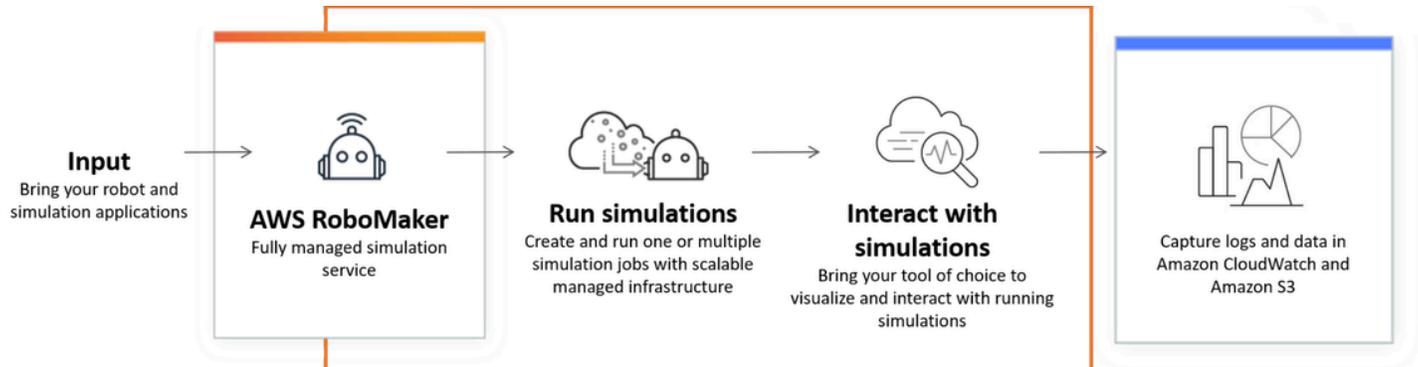
PortForwardingConfig .....	573
PortMapping .....	574
ProgressDetail .....	575
RenderingEngine .....	577
Robot .....	578
RobotApplicationConfig .....	581
RobotApplicationSummary .....	584
RobotDeployment .....	586
RobotSoftwareSuite .....	589
S3KeyOutput .....	590
S3Object .....	591
SimulationApplicationConfig .....	593
SimulationApplicationSummary .....	596
SimulationJob .....	598
SimulationJobBatchSummary .....	604
SimulationJobRequest .....	607
SimulationJobSummary .....	611
SimulationSoftwareSuite .....	614
Source .....	615
SourceConfig .....	617
TemplateLocation .....	619
TemplateSummary .....	620
Tool .....	622
UploadConfiguration .....	624
VPCConfig .....	626
VPCConfigResponse .....	628
WorldConfig .....	630
WorldCount .....	631
WorldExportJobSummary .....	632
WorldFailure .....	634
WorldGenerationJobSummary .....	636
WorldSummary .....	639
일반적인 오류 .....	640
공통 파라미터 .....	641
엔드포인트 및 할당량 .....	644
Service endpoints .....	644

---

서비스 할당량 .....	645
문제 해결 .....	649
시뮬레이션 작업 .....	649
Simulation WorldForge .....	653
지원 정책 .....	656
지원 변경 사항: 2022년 12월 15일 .....	656
지원 사항 변경: 2022년 5월 2일 .....	656
지원 변경 사항: 2022년 3월 15일 .....	657
지원 종료일: 2022년 1월 31일 .....	658
지원 종료일: 2021년 4월 30일 .....	659
문서 기록 .....	661
.....	dclxiii

# AWS RoboMaker란 무엇입니까?

AWS RoboMaker는 로봇 개발자가 인프라를 관리하지 않고도 시뮬레이션을 실행하고, 규모를 조정하고, 자동화할 수 있는 클라우드 기반 시뮬레이션 서비스입니다. AWS RoboMaker를 통해 로봇 개발자는 시뮬레이션 워크로드를 비용 효율적으로 규모 조정 및 자동화하고, 단일 API 호출로 대규모 및 병렬 시뮬레이션을 실행하고, 사용자 지정되고 무작위화된 3D 가상 환경을 만들 수 있습니다. 시뮬레이션 서비스를 사용하면 애플리케이션 테스트 속도를 높이고 정의한 템플릿으로 수백 개의 새로운 월드를 만들 수 있습니다.



AWS RoboMaker는 지속적 통합 및 지속적 전달(CI/CD) 파이프라인 내에서 테스트를 자동화하고, 대량의 반복 시험으로 강화 모델을 교육하고, 테스트를 위해 여러 동시 시뮬레이션을 플릿 관리 소프트웨어에 연결할 수 있습니다. AWS 기계 학습, 모니터링 및 분석 서비스와 결합하면 로봇이 데이터를 스트리밍하고, 탐색하고, 통신하고, 이해하고, 학습할 수 있습니다.

[AWS RoboMaker 리소스](#) 페이지에는 시뮬레이션 교육 리소스, 시뮬레이션 월드 에셋, 샘플 애플리케이션, 워크샵 및 튜토리얼 라이브러리, 하드웨어 개발자 키트에 대한 링크가 포함되어 있습니다.

## AWS RoboMaker 기능

AWS RoboMaker에는 다음 기능이 포함됩니다.

- [AWS RoboMaker를 이용한 시뮬레이션](#) – 인프라를 프로비저닝하거나 관리하지 않고도 시뮬레이션 작업을 실행할 수 있는 완전 관리형 시뮬레이션 서비스입니다. 이 서비스는 대규모 및 병렬 시뮬레이션을 지원하며 테스트 대상 시나리오의 복잡성에 따라 자동으로 규모를 조정합니다. AWS RoboMaker 시뮬레이션은 ROS, 사용자 지정 로봇 애플리케이션, Gazebo, Unity, Unreal 및 Nvidia Isaac 기반 시뮬레이션 등 원하는 로봇 소프트웨어와 시뮬레이터를 실행하는 데 사용할 수 있습니다.
- [Simulation WorldForge로 월드 생성](#) – 엔지니어링 투자나 월드 생성 인프라를 관리하지 않고도 실제 조건을 모방한 수백 개의 사전 정의된 무작위 시뮬레이션 월드를 자동으로 생성할 수 있습니다. 오늘

날 Simulation WorldForge는 구성 가능한 평면도와 가구로 구성된 실내 가정 환경을 위한 월드를 제공합니다.

## AWS RoboMaker 요금

다른 AWS 제품과 마찬가지로 AWS RoboMaker에도 사용 계약이나 최소 약정이 없습니다. AWS RoboMaker 사용 비용에 대한 자세한 내용은 [AWS RoboMaker 요금](#)을 참조하세요.

AWS RoboMaker를 시작하고 서비스에 대해 자세히 알아보려면 [AWS RoboMaker 시작](#) 섹션을 계속 진행하세요.

# AWS RoboMaker 시작

AWS RoboMaker는 클라우드에서 로봇 시뮬레이션을 실행합니다. 시작하려면 콘솔에서 시뮬레이션된 로봇과 환경을 볼 수 있도록 [IAM](#) 역할이 있는 AWS 계정을 생성해야 합니다. 다음으로 시뮬레이션 작업을 실행할 환경 및 로봇 애플리케이션 모두를 위한 컨테이너를 구축합니다. 그런 다음 시뮬레이션 작업에서 로그와 데이터를 캡처합니다.

## 주제

- [AWS RoboMaker 개념](#)
- [설 AWS RoboMaker정](#)
- [첫 시뮬레이션 실행](#)

# AWS RoboMaker 개념

이 단원에서는 AWS RoboMaker를 효과적으로 사용하기 위해 이해해야 할 핵심 개념 및 용어에 대해 설명합니다. 자세한 내용은 [AWS RoboMaker FAQ](#)를 참조하십시오.

## 개념

- [컨테이너](#)
- [시뮬레이션 작업](#)
- [Simulation WorldForge](#)
- [Environment](#)
- [애플리케이션](#)
- [애플리케이션 구성](#)

# 컨테이너

Amazon ECR 리포지토리에 컨테이너 이미지가 저장됩니다. 컨테이너는 서비스에서 실행될 때 이미지에서 생성됩니다. 일반적인 시뮬레이션에서는 로봇 운영 체제(ROS)를 사용하여 하나의 컨테이너는 Gazebo의 환경을 시뮬레이션하고 다른 컨테이너는 로봇을 시뮬레이션할 수 있습니다. 자세한 내용은 Amazon VPC 사용 설명서의 [Amazon VPC란 무엇인가요?](#)를 참조하세요.

## 시뮬레이션 작업

단일 시뮬레이션 작업으로 하나 또는 두 개의 애플리케이션이 실행됩니다. 일반적인 시뮬레이션 작업에는 로봇 애플리케이션(환경 데이터에 응답하는 사용자 지정 로직)을 환경(로봇이 있는 월드 모델)과 페어링하는 작업이 포함됩니다. 시뮬레이션 작업은 결과 및 지표를 제공합니다. 자세한 내용은 [AWS RoboMaker를 이용한 시뮬레이션](#) 섹션을 참조하세요.

## Simulation WorldForge

Simulation WorldForge를 사용하면 사용자가 정의한 템플릿으로 시뮬레이션 월드를 더 쉽고 빠르게 생성할 수 있습니다. 또한 도메인 무작위화를 통해 수많은 시뮬레이션 월드가 필요한 시뮬레이션 워크로드를 관리하는 데 도움이 될 수 있습니다. 자세한 내용은 [Simulation WorldForge로 월드 생성](#) 섹션을 참조하세요.

## Environment

애플리케이션은 환경의 구성과 환경에서 실행할 도구를 지정합니다. 환경에서 실행되는 도구는 동일한 파일 시스템, 환경 변수 및 네트워킹을 공유합니다. 환경에서 실행되는 애플리케이션 및 도구는 환경의 파일 변경 사항을 반영해야 하며 환경은 사용 가능한 도구를 제공해야 합니다. 환경에 맞는 컨테이너 이미지를 제공해야 합니다. 자세한 내용은 [이미지를 사용하여 AWS RoboMaker 애플리케이션 개발](#) 섹션을 참조하세요.

## 애플리케이션

시뮬레이션 작업을 생성하기 전에 AWS RoboMaker에서 로봇 애플리케이션을 생성해야 합니다. 로봇 애플리케이션에는 탐색 및 인식을 위한 로봇 코드가 포함되어 있습니다. 시뮬레이션 애플리케이션에는 환경을 시뮬레이션하는 데 필요한 모든 자산과 로직이 포함되어 있습니다. AWS RoboMaker는 로봇 애플리케이션과 시뮬레이션 애플리케이션의 버전을 두 개 이상 만들 수 있도록 지원합니다. 자세한 내용은 [애플리케이션 버전 지정](#) 섹션을 참조하세요.

당사의 애플리케이션은 두 가지 (주요) 구성 요소로 구성되어 있습니다.

- 컨테이너는 애플리케이션이 한 컴퓨팅 환경에서 다른 컴퓨팅 환경으로 빠르고 안정적으로 실행되도록 코드와 종속성을 함께 패키징하는 표준 소프트웨어 단위입니다.
- 소프트웨어 제품군: 번들의 콘텐츠를 추출, 소싱, 검증 및 실행할 수 있는 환경을 나타냅니다. 현재 지원되는 소프트웨어 제품군은 일반(로봇 애플리케이션용) 및 SimulationRuntime(시뮬레이션 애플리케이션용)입니다.

## 애플리케이션 구성

[CreateSimulationJob](#)에서 시뮬레이션 또는 로봇 애플리케이션을 제공할 때는 실제로 [RobotApplicationConfig](#)와 [SimulationApplicationConfig](#)를 지정합니다. 즉, 실제 애플리케이션 ARN 및 버전과 함께 다음 시작 구성, 업로드 구성 및 도구를 지정합니다.

- [LaunchConfig](#) – 환경에서 애플리케이션 코드를 어떻게 실행할지 시뮬레이션 서비스에 알려줍니다.
- [UploadConfiguration](#) – 애플리케이션당 최대 10개의 업로드 구성을 전달할 수 있습니다. AWS RoboMaker는 업로드 구성 경로에 쓴 파일을 출력 버킷에 업로드합니다.
- [Tool](#) – 애플리케이션 컨테이너에서 실행할 사용자 지정 프로세스 목록

자세한 내용은 [AWS RoboMaker를 이용한 시뮬레이션](#) 섹션을 참조하세요.

## 설 AWS RoboMaker정

AWS RoboMaker설정하려면 먼저 AWS 계정과 IAM 관리자 사용자를 생성해야 합니다.

### 가입하세요. AWS 계정

계정이 없는 경우 다음 단계를 완료하여 계정을 만드세요. AWS 계정

가입하려면 AWS 계정

1. <https://portal.aws.amazon.com/billing/signup>을 여세요.
2. 온라인 지시 사항을 따르세요.

등록 절차 중에는 전화를 받고 키패드로 인증 코드를 입력하는 과정이 있습니다.

에 AWS 계정가입하면 AWS 계정 루트 사용자a가 생성됩니다. 루트 사용자에게는 계정의 모든 AWS 서비스 및 리소스 액세스 권한이 있습니다. 보안 모범 사례로, 사용자에게 관리자 액세스 권한을 할당하고 루트 사용자 [액세스가 필요한 작업을 수행할 때는 루트 사용자만](#) 사용하십시오.

AWS 가입 프로세스가 완료된 후 확인 이메일을 보냅니다. 언제든지 <https://aws.amazon.com/>으로 가서 내 계정(My Account)을 선택하여 현재 계정 활동을 보고 계정을 관리할 수 있습니다.

## 관리 권한이 있는 사용자 생성

가입한 AWS 계정후에는 일상적인 작업에 루트 사용자를 사용하지 않도록 관리 사용자를 보호하고 AWS IAM Identity Center활성화하고 생성하십시오. AWS 계정 루트 사용자

보안을 유지하세요. AWS 계정 루트 사용자

1. 루트 사용자를 선택하고 AWS 계정 이메일 주소를 입력하여 계정 [AWS Management Console](#)소유자로 로그인합니다. 다음 페이지에서 비밀번호를 입력합니다.

루트 사용자를 사용하여 로그인하는 데 도움이 필요하면AWS 로그인 사용 설명서의 [루트 사용자 로 로그인](#)을 참조하세요.

2. 루트 사용자의 다중 인증(MFA)을 활성화합니다.

지침은 IAM [사용 설명서의 AWS 계정 루트 사용자 \(콘솔\)에 대한 가상 MFA 디바이스 활성화를 참조하십시오.](#)

### 관리 액세스 권한이 있는 사용자 생성

1. IAM Identity Center를 활성화합니다.

지침은 AWS IAM Identity Center 사용 설명서의 [AWS IAM Identity Center설정](#)을 참조하세요.

2. IAM ID 센터에서 사용자에게 관리 액세스 권한을 부여하십시오.

를 ID 소스로 사용하는 방법에 대한 자습서는 사용 [설명서의 기본값으로 IAM Identity Center 디렉터리사용자 액세스 구성](#)을 참조하십시오. IAM Identity Center 디렉터리 AWS IAM Identity Center

### 관리 권한이 있는 사용자 로 로그인

- IAM IDentity Center 사용자 로 로그인하려면 IAM IDentity Center 사용자를 생성할 때 이메일 주소로 전송된 로그인 URL을 사용합니다.

IAM Identity Center 사용자를 사용하여 [로그인하는 데 도움이 필요하면 사용 설명서의 AWS 액세스 포털에 로그인](#)을 참조하십시오.AWS 로그인

### 추가 사용자에게 액세스 권한 할당

1. IAM Identity Center에서 최소 권한 권한 적용 모범 사례를 따르는 권한 세트를 생성합니다.

지침은 사용 설명서의 [권한 집합 생성](#)을 참조하십시오.AWS IAM Identity Center

2. 사용자를 그룹에 배정한 다음 그룹에 Single Sign-On 액세스 권한을 할당하십시오.

자세한 지침은 사용 설명서의 [그룹 추가](#)를 참조하십시오.AWS IAM Identity Center

## 첫 시뮬레이션 실행

이 가이드의 다음 섹션에서는 첫 번째 시뮬레이션을 실행하는 방법을 설명합니다. 이는 순서대로 실행해야 합니다.

첫 번째 시뮬레이션을 실행하려면

1. [컨테이너화된 애플리케이션 빌드](#)
2. [Amazon ECR에 게시](#)
3. [시뮬레이션 실행](#)

# AWS RoboMaker를 이용한 개발

이 섹션은 AWS RoboMaker를 사용하여 개발할 수 있도록 설정하는 데 도움이 됩니다. 이미지를 구축하여 Amazon ECR에 게시하고 이미지를 사용하여 애플리케이션을 개발하는 방법을 알아봅니다.

## 주제

- [애플리케이션 컨테이너 빌딩](#)
- [Amazon ECR에 애플리케이션 컨테이너 게시](#)
- [로봇 애플리케이션 작업](#)
- [시뮬레이션 애플리케이션 작업](#)
- [애플리케이션 버전 지정](#)
- [이미지를 사용하여 AWS RoboMaker 애플리케이션 개발](#)

## 애플리케이션 컨테이너 빌딩

AWS RoboMaker에서 시뮬레이션 작업을 제출하는 단계는 다음과 같습니다. 애플리케이션 컨테이너를 빌드하고, 컨테이너를 AWS RoboMaker 애플리케이션에 연결하고, 컨테이너를 사용하여 시뮬레이션 작업을 제출합니다. 이 섹션에서는 AWS RoboMaker에 대한 Docker를 사용하여 애플리케이션 컨테이너를 빌드하는 방법을 설명합니다. [hello-world 샘플 애플리케이션](#)을 사용하여 ROS 기반 예제를 위한 샘플 로봇 및 시뮬레이션 애플리케이션 컨테이너를 빌드하는 데 필요한 단계를 보여줍니다. 이 페이지에서는 컨테이너를 로컬에서 테스트하는 방법도 보여줍니다.

ROS를 사용하지 않는 경우, [GPU 및 컨테이너 지원을 통해 AWS RoboMaker에서 충실도가 높은 시뮬레이션을 실행하는 방법](#)을 설명하는 블로그 게시물을 참조하세요.

## 섹션

- [필수 조건](#)
- [ROS 작업 영역에서 애플리케이션 컨테이너 구축](#)
- [컨테이너 테스트](#)

## 필수 조건

시작하기 전에 개발 환경에 필요한 종속성이 있는지 확인하세요. 시스템에 Docker, AWS CLI, VCS 가져오기 도구가 설치되어 있어야 합니다.

- AWS CLI 설치: <https://docs.aws.amazon.com/cli/latest/userguide/getting-started-install.html>
- Docker 설치: <https://docs.docker.com/get-docker/>
- [VCS 가져오기 도구](#)를 설치합니다(워크플로에 필요한 경우).

```
sudo pip3 install vcstool
```

또한 [다음 권한이 포함된 IAM 역할](#)을 가진 AWS 계정이 있어야 합니다.

- IAM 역할 생성
- AWS RoboMaker 리소스 생성(시뮬레이션 작업, 로봇, 시뮬레이션 애플리케이션)
- Amazon ECR 리포지토리 생성 및 업로드

마지막으로, 계정 번호를 알아야 하고, 시뮬레이션을 실행할 리전을 선택해야 합니다. AWS RoboMaker은 [AWS RoboMaker 엔드포인트 및 할당량](#)에 나열된 다음 리전에서 지원됩니다.

## ROS 작업 영역에서 애플리케이션 컨테이너 구축

AWS RoboMaker 시뮬레이션은 시뮬레이션 애플리케이션과 선택적 로봇 애플리케이션으로 구성됩니다. 각 애플리케이션은 이름과 컨테이너 이미지로 정의됩니다. 이 섹션에서는 시뮬레이션 애플리케이션과 로봇 애플리케이션 모두에 사용할 컨테이너 이미지를 빌드하는 방법을 보여줍니다. 다음 예제에서 두 애플리케이션은 모두 단일 작업 영역 내에 구축되어 있습니다. 다음 접근 방식은 모든 ROS 프로젝트에 쉽게 일반화할 수 있습니다.

시작하려면 hello world 리포지토리를 복제하고 소스를 가져오세요.

```
git clone https://github.com/aws-robotics/aws-robomaker-sample-application-helloworld.git helloworld
cd helloworld
vcs import robot_ws < robot_ws/.rosinstall
vcs import simulation_ws < simulation_ws/.rosinstall
```

다음으로 helloworld 디렉터리에 새 파일을 생성하고 이름을 Dockerfile로 지정합니다. 다음 명령을 복사하여 붙여넣습니다.

```
# ===== ROS/Colcon Dockerfile =====
# This sample Dockerfile will build a Docker image for AWS RoboMaker
# in any ROS workspace where all of the dependencies are managed by rosdep.
#
# Adapt the file below to include your additional dependencies/configuration
```

```

# outside of rosdep.
# =====

# ==== Arguments ====
# Override the below arguments to match your application configuration.
# =====

# ROS Distribution (ex: melodic, foxy, etc.)
ARG ROS_DISTRO=melodic
# Application Name (ex: helloworld)
ARG APP_NAME=robomaker_app
# Path to workspace directory on the host (ex: ./robot_ws)
ARG LOCAL_WS_DIR=workspace
# User to create and use (default: robomaker)
ARG USERNAME=robomaker
# The gazebo version to use if applicable (ex: gazebo-9, gazebo-11)
ARG GAZEBO_VERSION=gazebo-9
# Where to store the built application in the runtime image.
ARG IMAGE_WS_DIR=/home/$USERNAME/workspace

# ===== ROS Build Stages =====
# ${ROS_DISTRO}-ros-base
#   -> ros-robomaker-base
#     -> ros-robomaker-application-base
#       -> ros-robomaker-build-stage
#         -> ros-robomaker-app-runtime-image
# =====

# ==== ROS Base Image =====
# If running in production, you may choose to build the ROS base image
# from the source instruction-set to prevent impact from upstream changes.
# ARG UBUNTU_DISTRO=focal
# FROM public.ecr.aws/lts/ubuntu:${UBUNTU_DISTRO} as ros-base
# Instruction for each ROS release maintained by OSRF can be found here:
# https://github.com/osrf/docker_images
# =====

# ==== Build Stage with AWS RoboMaker Dependencies ====
# This stage creates the robomaker user and installs dependencies required
# to run applications in RoboMaker.
# =====

FROM public.ecr.aws/docker/library/ros:${ROS_DISTRO}-ros-base AS ros-robomaker-base
ARG USERNAME

```

```
ARG IMAGE_WS_DIR

RUN apt-get clean
RUN apt-get update && apt-get install -y \
  lsb \
  unzip \
  wget \
  curl \
  xterm \
  python3-colcon-common-extensions \
  devilspie \
  xfce4-terminal

RUN groupadd $USERNAME && \
  useradd -ms /bin/bash -g $USERNAME $USERNAME && \
  sh -c 'echo "$USERNAME ALL=(root) NOPASSWD:ALL" >> /etc/sudoers'

USER $USERNAME
WORKDIR /home/$USERNAME

RUN mkdir -p $IMAGE_WS_DIR

# ==== ROS Application Base ====
# This section installs exec dependencies for your ROS application.
# Note: Make sure you have defined 'exec' and 'build' dependencies correctly
# in your package.xml files.
# =====
FROM ros-robomaker-base as ros-robomaker-application-base
ARG LOCAL_WS_DIR
ARG IMAGE_WS_DIR
ARG ROS_DISTRO
ARG USERNAME

WORKDIR $IMAGE_WS_DIR
COPY --chown=$USERNAME:$USERNAME $LOCAL_WS_DIR/src $IMAGE_WS_DIR/src

RUN sudo apt update && \
  rosdep update && \
  rosdep fix-permissions

# Note: This will install all dependencies.
# You could further optimize this by only defining the exec dependencies.
# Then, install the build dependencies in the build image.
RUN rosdep install --from-paths src --ignore-src -r -y
```

```

# ==== ROS Workspace Build Stage ====
# In this stage, we will install copy source files, install build dependencies
# and run a build.
# =====
FROM ros-robomaker-application-base AS ros-robomaker-build-stage
LABEL build_step="${APP_NAME}Workspace_Build"
ARG APP_NAME
ARG LOCAL_WS_DIR
ARG IMAGE_WS_DIR

RUN . /opt/ros/$ROS_DISTRO/setup.sh && \
    colcon build \
    --install-base $IMAGE_WS_DIR/$APP_NAME

# ==== ROS Robot Runtime Image ====
# In the final stage, we will copy the staged install directory to the runtime
# image.
# =====
FROM ros-robomaker-application-base AS ros-robomaker-app-runtime-image
ARG APP_NAME
ARG USERNAME
ARG GAZEBO_VERSION

ENV USERNAME=$USERNAME
ENV APP_NAME=$APP_NAME
ENV GAZEBO_VERSION=$GAZEBO_VERSION

RUN rm -rf $IMAGE_WS_DIR/src

COPY --from=ros-robomaker-build-stage $IMAGE_WS_DIR/$APP_NAME $IMAGE_WS_DIR/$APP_NAME

# Add the application source file to the entrypoint.
WORKDIR /
COPY entrypoint.sh /entrypoint.sh
RUN sudo chmod +x /entrypoint.sh && \
    sudo chown -R $USERNAME /entrypoint.sh && \
    sudo chown -R $USERNAME $IMAGE_WS_DIR/$APP_NAME

ENTRYPOINT ["/entrypoint.sh"]

```

방금 생성한 Dockerfile은 도커 이미지를 빌드하는 데 사용되는 명령 세트입니다. Dockerfile에 있는 설명을 읽고 무엇이 구축되고 있는지 파악하고 필요에 따라 조정합니다. 쉽게 개발할 수 있도록

Dockerfile은 [Open Source Robotics Foundation\(OSRF\)](#)에서 관리하는 공식 ROS 도커 이미지를 기반으로 합니다. 하지만 프로덕션 환경에서 실행할 때는 업스트림 변경으로 인한 영향을 방지하기 위해 [GitHub의 OSRF 소스 명령 세트](#)를 사용하여 ROS 기본 이미지를 빌드하도록 선택할 수 있습니다.

entrypoint.sh라는 파일을 새로 생성합니다.

```
#!/bin/bash
set -e
source "/home/$USERNAME/workspace/$APP_NAME/setup.bash"
if [[ -f "/usr/share/$GAZEBO_VERSION/setup.sh" ]]
then
    source /usr/share/$GAZEBO_VERSION/setup.sh
fi
printenv
exec "${@:1}"
```

ENTRYPOINT 파일은 Docker 컨테이너가 생성될 때 실행되는 실행 파일입니다. 엔트리 포인트를 사용하여 ROS 작업 공간을 소싱하므로 AWS RoboMaker에서 roslaunch 명령을 쉽게 실행할 수 있습니다. 이 ENTRYPOINT 파일에 사용자 고유의 환경 구성 단계를 추가할 수도 있습니다.

Dockerfile은 다단계 빌드와 Docker BuildKit과의 통합 캐싱을 사용합니다. 다단계 빌드는 별도의 빌드 단계가 있는 워크플로를 허용하므로 빌드 종속성과 소스 코드가 런타임 이미지에 복사되지 않습니다. 이렇게 하면 도커 이미지 크기가 줄어들고 성능이 향상됩니다. 캐싱 작업은 이전에 빌드된 파일을 저장하여 향후 빌드 속도를 높입니다.

다음 명령어를 사용하여 로봇 애플리케이션을 빌드합니다.

```
DOCKER_BUILDKIT=1 docker build . \
--build-arg ROS_DISTRO=melodic \
--build-arg LOCAL_WS_DIR=./robot_ws \
--build-arg APP_NAME=helloworld-robot-app \
-t robomaker-helloworld-robot-app
```

로봇 애플리케이션을 빌드한 후 다음과 같이 시뮬레이션 애플리케이션을 빌드할 수 있습니다.

```
DOCKER_BUILDKIT=1 docker build . \
--build-arg GAZEBO_VERSION=gazebo-9 \
--build-arg ROS_DISTRO=melodic \
--build-arg LOCAL_WS_DIR=./simulation_ws \
--build-arg APP_NAME=helloworld-sim-app \
-t robomaker-helloworld-sim-app
```

`docker images` 명령을 실행하여 도커 이미지가 성공적으로 빌드되었는지 확인합니다. 출력은 다음과 같을 것입니다.

```
Administrator:~/environment/helloworld (ros1) $ docker images
REPOSITORY              TAG          IMAGE ID          CREATED           SIZE
robomaker-helloworld-sim-app  latest      5cb08816b6b3     6 minutes ago    2.8GB
robomaker-helloworld-robot-app latest      b5f6f755feec     10 minutes ago   2.79GB
```

이 시점에서 도커 이미지를 성공적으로 빌드했습니다. AWS RoboMaker와 사용하기 위해 이미지를 업로드하기 전에 로컬에서 테스트하는 것이 좋습니다. 다음 단계에서는 그 방법을 설명합니다.

## 컨테이너 테스트

다음 명령을 사용하면 로컬 개발 환경에서 애플리케이션을 실행할 수 있습니다.

로봇 애플리케이션 실행:

```
docker run -it -v /tmp/.X11-unix/:/tmp/.X11-unix/ \
-u robomaker -e ROBOMAKER_GAZEBO_MASTER_URI=http://localhost:5555 \
-e ROBOMAKER_ROS_MASTER_URI=http://localhost:11311 \
robomaker-helloworld-robot-app:latest roslaunch hello_world_robot rotate.launch
```

시뮬레이션 애플리케이션 실행:

```
docker run -it -v /tmp/.X11-unix/:/tmp/.X11-unix/ \
-u robomaker -e ROBOMAKER_GAZEBO_MASTER_URI=http://localhost:5555 \
-e ROBOMAKER_ROS_MASTER_URI=http://localhost:11311 \
robomaker-helloworld-sim-app:latest roslaunch hello_world_simulation empty_world.launch
```

컨테이너가 제대로 작동하는지 확인했으면 [애플리케이션 컨테이너를 Amazon ECR에 게시](#)한 다음 [시뮬레이션 작업 제출](#)을 수행할 수 있습니다.

## Amazon ECR에 애플리케이션 컨테이너 게시

시뮬레이션 작업에서 AWS RoboMaker이 사용하는 컨테이너는 완전 관리형 컨테이너 레지스트리인 [Amazon Elastic Container Registry\(ECR\)](#)에 저장되어야 합니다. 성공적으로 [애플리케이션 컨테이너를 구축](#)했으면 Amazon ECR로 푸시해야 합니다. 이 단원에서 그 방법을 설명합니다.

시작하려면 다음 명령에서 재사용되는 몇 가지 환경 변수를 설정하여 반복적인 입력을 줄일 수 있습니다.

```
export robotapp=robomaker-helloworld-robot-app
export simapp=robomaker-helloworld-sim-app
export account=<YOUR AWS ACCOUNT NUMBER>
export region=<YOUR AWS REGION>
export ecruri=$account.dkr.ecr.$region.amazonaws.com
```

그런 다음 로그인해 두 개의 새 리포지토리를 만듭니다.

```
aws ecr get-login-password --region $region | docker login --username AWS --password-stdin $ecruri
aws ecr create-repository --repository-name $robotapp
aws ecr create-repository --repository-name $simapp
```

Amazon ECR 리포지토리의 URI를 사용하여 도커 이미지에 태그를 지정할 수 있습니다.

```
docker tag $robotapp $ecruri/$robotapp:latest
docker tag $simapp $ecruri/$simapp:latest
```

다음으로 Amazon ECR에 도커 이미지를 푸시합니다.

```
docker push $ecruri/$robotapp
docker push $ecruri/$simapp
```

마지막으로 다음 명령을 실행하여 Amazon ECR에 업로드된 이미지를 확인할 수 있습니다.

```
aws ecr list-images --repository-name $simapp
aws ecr list-images --repository-name $robotapp
```

다음 코드 조각은 예상 출력을 보여줍니다.

```
Administrator:~/environment/helloworld (ros1) $ aws ecr list-images --repository-name
$simapp
{
  "imageIds": [
    {
      "imageDigest": "sha256:28cad40230402343024kf303f30fk20f2f2fa0a8148",
      "imageTag": "latest"
    }
  ]
}
```

```
Administrator:~/environment/helloworld (ros1) $ aws ecr list-images --repository-name
$robotapp
{
  "imageIds": [
    {
      "imageDigest": "sha256:28cad40230402343024kf303f30fk20f2f2fa0a8148",
      "imageTag": "latest"
    }
  ]
}
```

이제 로봇 및 시뮬레이션 도커 이미지가 Amazon ECR 내에서 호스팅되고 있습니다. [시뮬레이션 작업을 제출](#)하기 전에 이러한 이미지를 [로봇 애플리케이션](#) 또는 [시뮬레이션 애플리케이션](#)에 연결해야 합니다.

## 애플리케이션 버전 관리

AWS RoboMaker는 로봇 애플리케이션과 시뮬레이션 애플리케이션을 두 가지 이상의 버전으로 생성하는 것을 지원합니다. 따라서 로봇 및 시뮬레이션이 사용하는 코드를 제어할 수 있습니다. 버전은 \$LATEST 버전 애플리케이션의 번호가 지정된 스냅샷입니다. 개발 워크플로의 여러 부분에서 사용할 버전을 생성할 수 있습니다. 예를 들어 개발, 베타 배포 또는 프로덕션 단계가 있습니다.

AWS RoboMaker 로봇 애플리케이션 또는 시뮬레이션 애플리케이션의 버전을 관리할 때 애플리케이션의 스냅샷을 생성합니다. Amazon ECR은 이미지 다이제스트를 사용하여 애플리케이션 버전을 나눕니다. AWS RoboMaker은 각 버전의 이미지 다이제스트를 기억합니다.

Amazon ECR에 이미지를 업로드하고 이미지 다이제스트를 변경하지 않은 경우, 해당 버전의 애플리케이션에 액세스하여 사용할 수 있습니다. 애플리케이션당 최대 40개의 버전을 생성할 수 있습니다.

이미지를 생성할 때 이미지에 태그를 적용할 수도 있습니다. \$LATEST 버전의 latest로 태그 필드의 값을 지정할 수 있습니다. 이 값들은 서로 다릅니다.

이미지에 latest 태그를 가져오는 방법에는 두 가지가 있습니다.

- 값이 latest인 태그를 지정했습니다.
- 태그가 없는 이미지를 푸시하면 Amazon ECR이 해당 latest 태그로 이미지를 업데이트합니다.

AWS RoboMaker에서 이미지에 태그를 지정하면 이미지가 항상 \$LATEST 버전으로 선택됩니다. 예를 들어 이미지 이름 myImage, 태그 xyz 및 이미지 다이제스트 123을 사용하여 로봇 애플리케이션을 만들면 \$LATEST 버전은 다이제스트 123인 myImage:xyz가 됩니다.

태그를 추가해야 하는 시나리오는 다음과 같습니다.

- 새 태그를 사용하도록 \$LATEST 버전을 업데이트합니다. 예를 들어 이미지 myImage가 있는 경우 abc 태그를 사용하여 이미지를 업데이트할 수 있습니다. 이미지의 \$LATEST 버전은 myImage:abc를 가리킵니다.
- 이미지를 업데이트하고 태그를 다시 지정합니다. 예를 들어, abc 태그가 있는 이미지를 변경할 수 있습니다. xyz 태그를 업데이트한 후 사용할 수 있습니다. \$LATEST 버전은 myImage:xyz를 가리킵니다.

자세한 내용은 [애플리케이션 버전 지정](#) 섹션을 참조하세요.

## 로봇 애플리케이션 작업

AWS RoboMaker 로봇 애플리케이션은 로봇의 애플리케이션 스택 실행을 담당하는 컨테이너 이미지입니다. 로봇 애플리케이션 이미지는 Amazon ECR에서 호스팅되어야 합니다. 로봇 애플리케이션을 시뮬레이션 애플리케이션과 함께 사용하여 시뮬레이션 작업을 생성하는 경우가 많습니다.

### 섹션

- [로봇 애플리케이션 생성](#)
- [로봇 애플리케이션 버전 생성](#)
- [로봇 애플리케이션 보기](#)
- [로봇 애플리케이션 업데이트](#)
- [로봇 애플리케이션 삭제](#)
- [로봇 애플리케이션 버전 삭제](#)

## 로봇 애플리케이션 생성

### Using the console

1. <https://console.aws.amazon.com/robomaker/> 에서 AWS RoboMaker 콘솔에 로그인합니다.
2. 왼쪽 창에서 개발을 선택한 다음 로봇 애플리케이션을 선택합니다.
3. 로봇 애플리케이션 생성을 선택합니다.

4. 로봇 애플리케이션 생성 페이지에서 로봇 애플리케이션의 이름을 입력합니다. 로봇을 식별하는 데 도움이 되는 이름을 선택합니다.
5. Amazon ECR 컨테이너 이미지를 제공합니다. Amazon ECR에 푸시한 이미지를 사용할 수 있습니다. 자세한 내용은 [Amazon Elastic Container Registry란?](#)을 참조하세요.
6. 태깅에 대한 자세한 내용은 [AWS RoboMaker 리소스 태깅](#) 단원을 참조하세요.
7. 생성을 선택합니다.

## Using the AWS CLI

```
aws robomaker create-robot-application \
--name my-robot-app \
--robot-software-suite name=General \
--environment uri=<ACCOUNT>.dkr.ecr.<REGION>.amazonaws.com/my-robot-app:latest
```

## 로봇 애플리케이션 버전 생성

### Using the console

1. <https://console.aws.amazon.com/robomaker/> 에서 AWS RoboMaker 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 개발을 선택한 다음 로봇 애플리케이션을 선택합니다.
3. 로봇 애플리케이션 이름을 선택합니다.
4. 로봇 애플리케이션 세부 정보 페이지에서 새로운 버전 생성을 선택한 후 생성을 선택합니다.

### Using the AWS CLI

```
aws robomaker create-robot-application-version --name my-robot-app-arn
```

## 로봇 애플리케이션 보기

### Using the console

1. <https://console.aws.amazon.com/robomaker/> 에서 AWS RoboMaker 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 개발을 선택한 다음 로봇 애플리케이션을 선택합니다.
3. 로봇 애플리케이션 이름을 선택합니다.

### Using the AWS CLI

```
aws robomaker describe-robot-application --application my-robot-application-arn
```

## 로봇 애플리케이션 업데이트

### Using the console

1. <https://console.aws.amazon.com/robomaker/> 에서 AWS RoboMaker 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 개발을 선택한 다음 로봇 애플리케이션을 선택합니다.
3. 업데이트할 로봇 애플리케이션 옆에 있는 상자를 선택합니다.
4. 작업을 선택한 다음 업데이트를 선택합니다.
5. 소스를 추가 또는 제거할 수 있지만, 소스 로봇 애플리케이션 파일이 하나 이상 있어야 합니다.
6. 업데이트를 선택하여 로봇 애플리케이션을 업데이트합니다.

### Using the AWS CLI

```
aws robomaker update-robot-application \  
--application my-robot-application-arn \  
--robot-software-suite name=General \  
--environment uri=:<ACCOUNT>.dkr.ecr.<REGION>.amazonaws.com/my-robot-app:latest
```

## 로봇 애플리케이션 삭제

### Using the console

1. <https://console.aws.amazon.com/robomaker/> 에서 AWS RoboMaker 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 개발을 선택한 다음 로봇 애플리케이션을 선택합니다.
3. 로봇 애플리케이션의 이름을 선택하고 생성된 시간 및 마지막 업데이트 시간 등 세부 정보를 확인합니다.
4. Robot applications details(로봇 애플리케이션 세부 정보) 페이지에서 삭제를 선택한 다음, 다시 삭제를 선택해 확인합니다.

### Using the AWS CLI

```
aws robomaker delete-robot-application --application my-robot-application-arn
```

## 로봇 애플리케이션 버전 삭제

### Using the console

1. <https://console.aws.amazon.com/robomaker/> 에서 AWS RoboMaker 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 개발을 선택한 다음 로봇 애플리케이션을 선택합니다.
3. 버전을 보려는 로봇 애플리케이션의 이름을 선택합니다.
4. 로봇 세부 정보 페이지에서 버전 세부 정보를 보려는 버전을 선택합니다.
5. Robot applications details(로봇 애플리케이션 세부 정보) 페이지에서 삭제를 선택하고 삭제를 선택해 확인합니다.

### Using the AWS CLI

```
aws robomaker delete-robot-application-version \
```

```
--application my-robot-application-arn \  
--version 2
```

## 시뮬레이션 애플리케이션 작업

AWS RoboMaker 시뮬레이션 애플리케이션은 로봇의 시뮬레이션 스택 실행을 담당하는 컨테이너 이미지입니다. 시뮬레이션 애플리케이션 이미지는 Amazon ECR에서 호스팅되어야 합니다. 시뮬레이션 애플리케이션을 로봇 애플리케이션과 함께 사용하여 시뮬레이션 작업을 생성하는 경우가 많습니다.

### 섹션

- [시뮬레이션 애플리케이션 생성](#)
- [시뮬레이션 애플리케이션 버전 생성](#)
- [시뮬레이션 애플리케이션 보기](#)
- [시뮬레이션 애플리케이션 업데이트](#)
- [시뮬레이션 애플리케이션 삭제](#)
- [시뮬레이션 애플리케이션 버전 삭제](#)

## 시뮬레이션 애플리케이션 생성

### Using the console

1. AWS RoboMaker 콘솔(<https://console.aws.amazon.com/robomaker/>)에 로그인합니다.
2. 왼쪽 탐색 창에서 개발을 선택한 다음 시뮬레이션 애플리케이션을 선택합니다.
3. 시뮬레이션 애플리케이션 생성을 선택합니다.
4. 시뮬레이션 애플리케이션 생성성 페이지에서 시뮬레이션 애플리케이션의 이름을 입력합니다. 시뮬레이션을 식별하는 데 도움이 되는 이름을 선택합니다.
5. Amazon ECR 컨테이너 이미지를 제공합니다. Amazon ECR에 푸시한 이미지를 사용할 수 있습니다. 자세한 내용은 [Amazon ECR이란 무엇인가?](#)를 참조하세요.
6. 태깅에 대한 자세한 내용은 [AWS RoboMaker 리소스 태깅](#) 단원을 참조하십시오.
7. 생성을 선택합니다.

### Using the AWS CLI

```
aws robomaker create-simulation-application \  
--name my-sim-app \  
--simulation-software-suite name=SimulationRuntime \  
--robot-software-suite name=General \  
--environment uri=<ACCOUNT>.dkr.ecr.<REGION>.amazonaws.com/my-sim-app:latest
```

## 시뮬레이션 애플리케이션 버전 생성

### Using the console

1. AWS RoboMaker 콘솔(<https://console.aws.amazon.com/robomaker/>)에 로그인합니다.
2. 왼쪽 탐색 창에서 개발을 선택한 다음 시뮬레이션 애플리케이션을 선택합니다.
3. 시뮬레이션 애플리케이션 name을 선택합니다.
4. 애플리케이션 세부 정보 페이지에서 새로운 버전 생성을 선택한 후 생성을 선택합니다.

### Using the AWS CLI

```
aws robomaker create-simulation-application-version --name my-simulation-  
application-arn
```

## 시뮬레이션 애플리케이션 보기

### Using the console

1. AWS RoboMaker 콘솔(<https://console.aws.amazon.com/robomaker/>)에 로그인합니다.
2. 왼쪽 탐색 창에서 개발을 선택한 다음 시뮬레이션 애플리케이션을 선택합니다.
3. 시뮬레이션 애플리케이션의 이름을 선택하고 생성된 시간 및 마지막 업데이트 시간 등 세부 정보를 확인합니다.

### Using the AWS CLI

```
aws robomaker describe-simulation-application --job my-simulation-job-arn
```

## 시뮬레이션 애플리케이션 업데이트

### Using the console

1. AWS RoboMaker 콘솔(<https://console.aws.amazon.com/robomaker/>)에 로그인합니다.
2. 왼쪽 탐색 창에서 개발을 선택한 다음 시뮬레이션 애플리케이션을 선택합니다.
3. 업데이트할 시뮬레이션 애플리케이션 옆에 있는 상자를 선택합니다.
4. 작업을 선택하고 업데이트를 선택합니다.
5. 소스를 추가 또는 제거할 수 있지만, 소스 시뮬레이션 애플리케이션 파일이 하나 이상 있어야 합니다.
6. 업데이트를 선택하여 시뮬레이션 애플리케이션을 업데이트합니다.

### Using the AWS CLI

```
aws robomaker update-simulation-application \  
--application my-simulation-application-arn \  
--robot-software-suite name=General \  
--simulation-software-suite name=SimulationRuntime \  
--environment uri=:<ACCOUNT>.dkr.ecr.<REGION>.amazonaws.com/my-simulation-app:latest
```

## 시뮬레이션 애플리케이션 삭제

### Using the console

1. AWS RoboMaker 콘솔(<https://console.aws.amazon.com/robomaker/>)에 로그인합니다.
2. 왼쪽 탐색 창에서 개발을 선택한 다음 시뮬레이션 애플리케이션을 선택합니다.
3. 시뮬레이션 애플리케이션의 이름을 선택합니다. 그러면 세부 정보(예: 생성된 시간, 마지막 업데이트 시간)가 표시됩니다.

4. 시뮬레이션 애플리케이션 세부 정보 페이지에서 삭제를 선택한 다음, 다시 삭제를 선택해 확인합니다.

### Using the AWS CLI

```
aws robomaker delete-simulation-application --application my-simulation-application-arn
```

## 시뮬레이션 애플리케이션 버전 삭제

### Using the console

1. AWS RoboMaker 콘솔(<https://console.aws.amazon.com/robomaker/>)에 로그인합니다.
2. 왼쪽 탐색 창에서 개발을 선택한 다음 시뮬레이션 애플리케이션을 선택합니다.
3. 버전을 보려는 시뮬레이션 애플리케이션의 이름을 선택합니다.
4. 시뮬레이션 세부 정보 페이지에서 버전을 선택하여 세부 정보를 확인합니다.
5. 세부 정보 페이지에서 삭제를 선택한 다음 다시 삭제를 선택해 확인합니다.

### Using the AWS CLI

```
aws robomaker delete-simulation-application-version \  
--application my-simulation-application-arn \  
--version 2
```

## 애플리케이션 버전 지정

AWS RoboMaker는 로봇 애플리케이션과 시뮬레이션 애플리케이션을 두 가지 이상의 버전으로 생성하는 것을 지원합니다. 따라서 로봇 및 시뮬레이션이 사용하는 코드를 제어할 수 있습니다. 버전은 \$LATEST 버전 애플리케이션의 번호가 지정된 스냅샷입니다. 개발, 베타 버전 개발, 프로덕션과 같은 개발 워크플로의 여러 부분에서 사용할 버전을 생성할 수 있습니다.

AWS RoboMaker 로봇 애플리케이션 또는 시뮬레이션 애플리케이션의 버전을 관리할 때 애플리케이션의 스냅샷을 생성합니다.

colcon를 사용하여 애플리케이션을 빌드하는 경우 AWS RoboMaker는 각 버전에 대한 파일의 Amazon S3 경로와 ETag를 기억합니다. Amazon S3 경로에 여전히 존재하고 변경되지 않은 경우 (ETag는 변경되지 않음) 버전이 만들어졌을 때 존재하는 그대로 애플리케이션 버전을 사용할 수 있습니다.

애플리케이션에 컨테이너 이미지를 사용하는 경우 이미지를 Amazon ECR에 업로드합니다. Amazon ECR은 이미지 다이제스트를 사용하여 애플리케이션 버전을 표시합니다. AWS RoboMaker는 각 버전의 이미지 다이제스트를 기억합니다.

Amazon ECR에 이미지를 업로드하고 이미지 다이제스트를 변경하지 않은 경우, 해당 버전의 애플리케이션에 액세스하여 사용할 수 있습니다.

애플리케이션당 최대 40개의 버전을 생성할 수 있습니다.

## 주제

- [이미지를 사용한 애플리케이션 버전 관리](#)
- [\\$LATEST 버전](#)
- [애플리케이션 버전 업데이트](#)
- [애플리케이션 버전 삭제](#)

## 이미지를 사용한 애플리케이션 버전 관리

애플리케이션을 개발할 때 컨테이너 이미지의 \$LATEST 버전을 업데이트 할 수 있습니다. \$LATEST 버전을 선택하면 지정한 Amazon ECR 위치에서 해당 버전을 가져올 수 있습니다.

이미지를 생성할 때 이미지에 태그를 적용할 수도 있습니다. \$LATEST 버전에 대해 태그 필드 값을 "latest"로 지정할 수 있습니다. 이 값들은 서로 다릅니다.

이미지에 "latest" 태그를 가져오는 방법에는 두 가지가 있습니다.

- 값이 "latest"인 태그를 지정했습니다.
- 태그가 없는 이미지를 푸시하면 Amazon ECR이 해당 "latest" 태그로 이미지를 업데이트합니다.

AWS RoboMaker에서 이미지에 태그를 지정하면 해당 이미지는 항상 \$LATEST 버전으로 선택됩니다. 예를 들어 이미지 이름 "myImage", 태그 "xyz", 이미지 다이제스트 "123"를 사용하여 로봇 애플리케이션을 만드는 경우, \$LATEST 버전은 다이제스트 "123"이 있는 myImage:xyz입니다.

태그를 추가하려는 경우의 시나리오는 다음과 같습니다.

- 새 태그를 사용하도록 \$LATEST 버전을 업데이트하려고 합니다. 예를 들어 이미지 "myImage"가 있는 경우 "abc" 태그를 사용하여 이미지를 업데이트할 수 있습니다. \$LATEST 버전의 이미지는 myImage:abc를 가리킵니다.
- 이미지를 업데이트하고 태그를 다시 지정하고자 합니다. 예를 들어, "abc" 태그가 있는 이미지를 변경할 수 있습니다. "xyz" 태그를 업데이트한 후 사용할 수 있습니다. \$LATEST 버전은 myImage:xyz를 가리킵니다.

## \$LATEST 버전

버전을 만들 때 AWS RoboMaker는 \$LATEST 버전의 스냅샷을 만들고 버전 번호를 1씩 증가시킵니다. AWS RoboMaker는 파일의 아마존 S3 경로와 ETag를 기억합니다. 경로는 파일을 검색하는 데 사용됩니다. ETag는 변경되지 않았음을 확인하는 데 사용됩니다. 버전 번호는 절대 재사용되지 않습니다. 예를 들어 최신 버전이 10인 경우 제거한 다음 새 버전을 만들면 새 버전이 버전 11이 됩니다.

애플리케이션을 개발할 때 \$LATEST 버전을 업데이트 할 수 있습니다. \$LATEST 버전을 선택하면 지정한 위치에서 버전이 검색됩니다. 예를 들어, 최신 버전의 로봇 애플리케이션과 시뮬레이션 애플리케이션을 사용하여 시뮬레이션 작업을 시작한 다음 경로에서 로봇 애플리케이션을 변경하고 시뮬레이션 작업을 다시 시작하면 업데이트된 로봇 애플리케이션이 사용됩니다.

로봇 애플리케이션을 배포하는 경우 배포할 번호가 지정된 특정 버전을 선택해야 합니다. 로봇 애플리케이션 버전을 생성하는 방법에 대한 자세한 내용은 [로봇 애플리케이션 버전 생성](#) 단원을 참조하십시오.

시뮬레이션 애플리케이션 버전을 생성하는 방법에 대한 자세한 내용은 [시뮬레이션 애플리케이션 버전 생성](#) 단원을 참조하십시오. ETag에 대한 자세한 내용은 [일반 응답 헤더](#)를 참조하십시오.

## 애플리케이션 버전 업데이트

AWS RoboMaker 애플리케이션의 \$LATEST 버전만 업데이트할 수 있습니다. 이와 같이 할 때 AWS RoboMaker에서 해당 버전을 사용할 수 있습니다. 예를 들어, 시뮬레이션 작업을 다시 시작하면 애플리케이션의 최신 버전이 시뮬레이션에 사용됩니다.

자세한 정보는 [로봇 애플리케이션 업데이트 및 시뮬레이션 애플리케이션 업데이트](#) 섹션을 참조하세요.

## 애플리케이션 버전 삭제

애플리케이션 버전이 더 이상 필요 없는 경우에는 삭제합니다. 자세한 정보는 [로봇 애플리케이션 버전 삭제](#) 및 [시뮬레이션 애플리케이션 버전 삭제](#) 섹션을 참조하세요.

## 이미지를 사용하여 AWS RoboMaker 애플리케이션 개발

### ⚠ Important

2022년 3월 15일부터 기존 시뮬레이션 작업에 영향을 미칠 수 있는 AWS RoboMaker 시뮬레이션을 변경했습니다. 로봇 애플리케이션, 시뮬레이션 애플리케이션 및 시뮬레이션 작업에 따라 수행할 수 있는 이러한 변경 사항과 마이그레이션 단계에 대한 자세한 내용은 [ROS 애플리케이션을 컨테이너로 마이그레이션](#) 항목을 참조하세요.

하나 이상의 컨테이너 이미지를 사용하여 시뮬레이션 및 로봇 애플리케이션을 개발하고 실행할 수 있습니다. 이미지에 대한 자세한 내용은 [Amazon ECS용 Docker 기본 사항](#)을 참조하세요. 사용하는 이미지는 [AWS RoboMaker 호환 컨테이너에 대한 요구 사항](#)에 나열된 요구 사항을 충족해야 합니다.

당사에서 지원하는 개발 환경 중 하나를 사용하는 경우 자체 이미지를 AWS RoboMaker와 함께 사용할 수 있습니다.

컨테이너 이미지를 사용하여 애플리케이션을 개발하는 방법은 여러 가지가 있습니다. 애플리케이션 개발 방법의 예를 보려면 [hello world 샘플 애플리케이션을 실행하기 위한 이미지 생성](#) 항목을 참조하세요.

이미지를 사용하여 애플리케이션을 개발한 후 테스트할 수 있습니다. 애플리케이션이 작동하는지 테스트하기 위해 로컬 Linux 시스템에서 애플리케이션을 시각화할 수 있습니다.

시뮬레이션이 작동하는지 테스트한 후, 이미지를 Amazon ECR로 푸시하고 시뮬레이션 작업을 실행하여 로봇이 가상 환경에서 어떻게 상호 작용하는지 확인할 수 있습니다.

### 주제

- [ROS 애플리케이션을 컨테이너로 마이그레이션](#)
- [ROS 컨테이너 FAQ](#)
- [AWS RoboMaker 호환 컨테이너에 대한 요구 사항](#)

- [GPU 애플리케이션을 실행하기 위한 이미지 생성](#)
- [hello world 샘플 애플리케이션을 실행하기 위한 이미지 생성](#)

## ROS 애플리케이션을 컨테이너로 마이그레이션

2021년 10월부터 모든 로봇 및 시뮬레이션 소프트웨어 세트를 사용할 수 있도록 AWS RoboMaker의 지원이 확대되었습니다. 이전에는 로봇 운영 체제(ROS)와 Gazebo만 AWS RoboMaker에서 실행할 수 있도록 유일하게 지원되는 로봇 및 시뮬레이션 소프트웨어 구성이었습니다. 이번 변경으로 이제 AWS RoboMaker에서 시뮬레이션을 실행하는 동안 원하는 로봇 및 시뮬레이션 소프트웨어를 구성할 수 있습니다.

이는 ROS와 Gazebo를 계속 사용하려는 고객에게 어떤 의미가 있습니까?

즉, AWS RoboMaker에서 사용할 자체 애플리케이션 컨테이너를 빌드하려면 Docker 기반 워크플로로 전환해야 합니다. Docker는 개발자가 애플리케이션의 종속성을 번들로 제공하고 소프트웨어를 번들 패키지 (컨테이너)로 제공할 수 있는 업계 표준 도구입니다. 자세한 내용은 [Amazon ECS용 Docker 기본 사항](#)을 참조하세요. 사용하는 이미지는 [AWS RoboMaker 호환 컨테이너에 대한 요구 사항](#)에 나열된 요구 사항을 충족해야 합니다.

이미 ROS 기반 컨테이너를 사용하고 있다면 어떻게 됩니까?

그럼 거의 다 됐습니다! AWS 콘솔 또는 CLI를 통해 [로봇](#) 및 [시뮬레이션](#) 애플리케이션의 소프트웨어 제품군을 ROS 관련 소프트웨어 제품군에서 일반 및 시뮬레이션 런타임 소프트웨어 제품군으로 업데이트해야 합니다. [시뮬레이션 실행](#)에 대해 다음 단계를 따르십시오.

Docker 기반 워크플로로 마이그레이션하는 방법

1. 원하는 ROS 버전에 따라 다음 튜토리얼 중 하나를 선택하고 해당 튜토리얼 내의 단계를 따르세요.
  - [ROS Melodic과 Gazebo 9를 사용하여 샘플 애플리케이션 실행](#)
  - [ROS 2, Foxy, Gazebo 11을 사용하여 샘플 애플리케이션 실행](#)
2. 컨테이너를 생성한 후 시뮬레이션 작업 제출을 진행할 수 있습니다.
  - [시뮬레이션 실행](#)

## ROS 컨테이너 FAQ

이 페이지에는 ROS 기반 로봇 및 시뮬레이션 애플리케이션을 AWS RoboMaker에서 실행하기에 적합한 Docker 컨테이너로 마이그레이션하는 것과 관련된 일반적인 질문과 답변이 나열되어 있습니다.

워크플로는 **colcon** 번들 로봇 및 시뮬레이션 애플리케이션을 사용하여 시뮬레이션 작업을 제출합니다. 마이그레이션이 필요한가요?

네, 마이그레이션해야 합니다. 마이그레이션 단계는 [ROS 애플리케이션을 컨테이너로 마이그레이션에](#) 있습니다.

로봇 및 시뮬레이션 애플리케이션을 마이그레이션해야 하는지 잘 모르겠습니다. 어떻게 알 수 있습니까?

AWS 콘솔이나 AWS CLI를 통해 확인할 수 있습니다. 지침을 보려면 다음 해당 탭을 선택하세요.

### Using the console

1. [AWS RoboMaker 콘솔](#)에 로그인합니다.
2. 왼쪽 탐색 창에서 개발을 선택한 다음 시뮬레이션 애플리케이션을 선택합니다.
3. 세부 사항을 보려는 시뮬레이션 애플리케이션의 이름을 선택합니다.

일반 및 시뮬레이션 런타임이 표시되면 마이그레이션이 필요하지 않습니다. ROS 또는 Gazebo 전용 값이 표시되면 마이그레이션이 필요합니다.

### Using the AWS CLI

#### Example

다음은 콘솔 기반 단계에 해당하는 작업을 수행하는 AWS CLI 명령의 예입니다.

```
aws robomaker describe-simulation-application --application YOUR-SIM-APP-ARN
```

이 명령은 `simulationSoftwareSuite`, `robotSoftwareSuite` 해당하는 경우) 및 `environment URI`를 나타내는 출력을 반환합니다. 시뮬레이션 런타임이 `simulationSoftwareSuite`로 표시되고 일반이 `robotSoftwareSuite`로 표시되고, `environment URI`가 설정된 경우 시뮬레이션 애플리케이션을 마이그레이션할 필요가 없습니다.

로봇과 시뮬레이션 애플리케이션 컨테이너는 서로 어떻게 통신합니까?

이는 일반적으로 ROS 기반 애플리케이션이 ROS 미들웨어를 사용하여 서로 통신하는 방식과 다르지 않습니다. 하지만 시뮬레이션 작업 요청의 시작 구성 객체 내에 일부 ROS별 환경 변수를 설정해야 합니다.

다음은 로봇 애플리케이션 launchConfig에 사용해야 하는 설정의 일부 예시입니다.

```
"robotApplications": [
  {
    "application": "YOUR-ROBOT-APP-ARN",
    "applicationVersion": "$LATEST",
    "launchConfig": {
      "environmentVariables": {
        "ROS_IP": "ROBOMAKER_ROBOT_APP_IP",
        "ROS_MASTER_URI": "http://ROBOMAKER_ROBOT_APP_IP:11311",
        "GAZEBO_MASTER_URI": "http://ROBOMAKER_SIM_APP_IP:11345"
      },
      ... # Removed extra data for clarity
    }
  }
]
```

다음은 시뮬레이션 애플리케이션 launchConfig에 사용해야 하는 설정의 일부 예시입니다.

```
"simulationApplications": [
  {
    "application": "YOUR-SIM-APP-ARN",
    "applicationVersion": "$LATEST",
    "launchConfig": {
      "environmentVariables": {
        "ROS_IP": "ROBOMAKER_SIM_APP_IP",
        "ROS_MASTER_URI": "http://ROBOMAKER_ROBOT_APP_IP:11311",
        "GAZEBO_MASTER_URI": "http://ROBOMAKER_SIM_APP_IP:11345"
      },
      ... # Removed extra data for clarity
    }
  }
]
```

제공된 ROBOMAKER\_\* 문자열과 포트 번호를 사용하여 ROS\_IP, ROS\_MASTER\_URI, GAZEBO\_MASTER\_URI를 설정하는 경우 컨테이너는 예상대로 서로 통신합니다.

자세한 내용은 [시뮬레이션 실행](#) 섹션을 참조하세요.

실시간 요소(RTF) 지표는 어디로 이동했습니까? 어떻게 복원할 수 있습니까?

AWS RoboMaker는 더 이상 이 지표를 자동으로 게시하지 않습니다. 이 지표를 CloudWatch에 게시하려면 [AWS RoboMaker CloudWatch Publisher](#) 패키지를 시뮬레이션 애플리케이션으로 가져온 다음 [README.md](#) 파일에 제공된 지침에 따라 시뮬레이션 시작 파일을 수정해야 합니다.

시뮬레이션 작업을 취소하고 태그를 지정하려면 어떻게 해야 하나요?

VPC 구성을 사용하면 일반 AWS API를 사용하여 AWS RoboMaker 시뮬레이션 작업을 자체 태그 지정하거나 자체 취소할 수 있습니다. 다음 접근 방식을 사용하려면 [NAT](#) 또는 [IGW](#)를 통해 AWS API로 연결되는 공개 경로가 있는 VPC에서 컨테이너를 실행해야 합니다. 가장 간단한 방법은 [기본 VPC](#)의 퍼블릭 서브넷을 사용하여 AWS API에 연결하는 것입니다. 프라이빗 서브넷에서 시뮬레이션을 실행하려는 경우 NAT를 설정하거나 인터페이스 VPC 엔드포인트를 설정할 수도 있습니다. 자세한 내용은 [AWS RoboMaker 및 인터페이스 VPC 엔드포인트\(AWS PrivateLink\)](#) 섹션을 참조하세요.

### Note

IGW를 사용하는 경우 아래 설명서에 설명된 대로 `assignPublicIp=True`를 설정해야 합니다. 퍼블릭 IP를 사용하는 경우 보안 그룹이 충분히 잠겨 있는지 확인하세요.

요청 파라미터에 다음 블록을 추가해야 합니다.

```
vpcConfig={
  'subnets': [
    'string',
  ],
  'securityGroups': [
    'string',
  ],
  'assignPublicIp': True|False
},
```

또한 AWS RoboMaker 시뮬레이션 작업에는 시뮬레이션 작업에 태그를 지정하고 취소할 수 있는 권한이 있는 IAM 역할이 있어야 합니다.

시뮬레이션 작업에서 AWS CLI 또는 boto3 Python 라이브러리를 사용하여 퍼블릭 AWS RoboMaker API를 호출할 수 있습니다. AWS CLI 및 boto3 라이브러리가 컨테이너에 사전 설치되어 있어야 AWS RoboMaker 시뮬레이션 작업에 사용할 수 있습니다. 다음 Python 샘플 코드는 시뮬레이션 작업을 취소하는 방법을 보여줍니다.

```
class RoboMakerUtils:
```

```

def __init__(self):
    self.job_arn = os.getenv('AWS_ROBOMAKER_SIMULATION_JOB_ARN')
    self.client = boto3.client('robomaker',
region_name=os.getenv('AWS_ROBOMAKER_REGION', 'us-east-1'))

def tag_robomaker_sim_job(self, key, value):
    self.client.tag_resource(
        resourceArn=self.job_arn,
        tags={
            key: str(value)
        }
    )

def cancel_robomaker_sim_job(self):
    self.tag_robomaker_sim_job("END_TIME", time.time())
    response = self.client.cancel_simulation_job(
        job=self.job_arn
    )

```

Simulation WorldForge 월드를 시뮬레이션 작업으로 가져오려면 어떻게 해야 합니까?

Simulation WorldForge 자산을 시뮬레이션 작업으로 가져와야 하는 경우, [DataSource API](#)를 사용합니다. 이렇게 하면 월드 내보내기 작업의 Amazon S3 출력 디렉터리에서 시뮬레이션 작업 컨테이너 내에서 선택한 대상으로 월드 자산을 가져올 수 있습니다.

자세한 내용은 [시뮬레이션에서 내보낸 월드 사용](#) 섹션을 참조하세요.

애플리케이션의 로그 파일이 생성되지 않았습니다. 어떻게 된 일입니까?

Dockerfile에서 관련 아티팩트를 디버깅하는 데 사용하는 모든 출력 디렉터리를 생성했는지 확인하세요. 예를 들어 Dockerfile에 다음 줄을 추가할 수 있습니다.

```
RUN mkdir -p $YOUR_LOG_DIR
```

자세한 내용은 [사용자 지정 업로드 구성 추가](#) 섹션을 참조하세요.

시뮬레이션 애플리케이션이 'run\_id on parameter server does not match declared run\_id'이라는 메시지와 함께 실패했습니다. 어떻게 해야 합니까?

로봇 애플리케이션과 시뮬레이션 애플리케이션을 모두 사용하여 ROS 시뮬레이션 작업을 시작하는 경우 --wait를 roslaunch 명령에 추가해야 합니다.

## AWS RoboMaker 호환 컨테이너에 대한 요구 사항

AWS RoboMaker 호환 가능한 컨테이너(컨테이너 이미지)를 실행하고 시뮬레이션을 성공적으로 시작하려면 일련의 요구 사항을 충족해야 합니다. 이러한 요구 사항을 충족했는데도 여전히 시뮬레이션 실행에 문제가 있는 경우, [시뮬레이션 작업](#) 및 [Simulation WorldForge](#)를 참조하세요.

### 시뮬레이션 런타임 요구 사항

컨테이너 이미지는 Dockerfile에서 VOLUME를 사용할 수 없습니다. VOLUME가 Dockerfile에 있는 경우 시뮬레이션은 4XX 오류 코드와 함께 실패하게 됩니다.

컨테이너 이미지는 Dockerfile에서 EXPOSE를 사용할 수 없습니다. EXPOSE가 Dockerfile에 있는 경우, AWS RoboMaker는 4XX 오류 코드와 함께 시뮬레이션에 실패하게 됩니다.

컨테이너 이미지는 반드시 압축된 크기가 20GB 이하여야 합니다. 컨테이너 이미지가 20GB 이상 압축된 경우, AWS RoboMaker는 4XX 오류 코드와 함께 시뮬레이션에 실패하게 됩니다.

Dockerfile에서는 CMD를 지정할 수 없습니다. 그럴 경우 AWS RoboMaker가 이를 패키지 이름과 시작 파일로 덮어씁니다. 대신 [CreateSimulationJob](#) 요청 내에서 각 시뮬레이션 애플리케이션 또는 로봇 애플리케이션의 launchConfig에 있는 command 파라미터를 사용하여 시작 명령 목록을 제공할 수 있습니다. 이는 시뮬레이션 작업에서와 CMD로 설정됩니다. command의 예: ["/bin/bash", "-c", "sleep 365d"]

시뮬레이션 작업에 도구를 추가하려면 반드시 컨테이너 이미지에 bash를 설치해야 합니다. 도구는 ["/bin/bash", "-c", "<command>"]와 함께 실행됩니다.

컨테이너에서 ROS를 실행하고 있고 로봇 애플리케이션과 시뮬레이션 애플리케이션 간에 통신이 필요한 경우, 다음과 같은 로봇 프레임워크를 설정해야 합니다.

- ROS Master
- Gazebo Master
- ROS IP

컨테이너에 있는 /etc/resolv.conf 파일을 사용자 지정할 수 없습니다. AWS RoboMaker가 파일을 자체 파일로 덮어씁니다.

AWS에서 Dockerfile을 실행 중인 경우 이미지를 마운트할 수 없습니다. Dockerfile에서 Mount를 지정하는 경우 AWS RoboMaker는 4XX 오류 코드와 함께 시뮬레이션에 실패하게 됩니다.

컨테이너 이미지는 기본 Docker `seccomp` 프로파일로 차단된 시스템 호출을 사용할 수 없습니다. 차단된 시스템 호출에 대한 자세한 내용은 [Seccomp 보안 프로파일](#)을 참조하세요.

이미지를 실행하는 사용자를 지정하려면 Dockerfile에서 `USER` 키워드를 지정할 수 있습니다. 사용자를 지정하지 않는 경우 AWS RoboMaker는 컨테이너의 루트 사용자를 사용합니다.

컨테이너 이미지에서 `USER`를 이름 또는 `UID:GID`로 지정할 수 있습니다. 컨테이너 이미지에 `UID`가 없는 경우 기본값은 `1000`입니다.

컨테이너 이미지는 `/opt/amazon/robomaker` 또는 해당 하위 폴더에 데이터를 저장할 수 없습니다. AWS RoboMaker만 해당 디렉터리만 사용할 수 있습니다. 해당 디렉터리를 사용하면 시뮬레이션이 제대로 작동하지 않을 수 있습니다.

다음 런타임 구성은 지원되지 않습니다.

	Docker 실행 인수	설명
1	<code>--add-host</code>	사용자 지정 host-to-IP 매핑 (host:ip) 추가
2	<code>--attach</code> , <code>-a</code>	STDIN, STDOUT 또는 STDERR에 연결
3	<code>--blkio-weight</code>	10에서 1000 사이의 블록 IO(상대적 가중치), 비활성화하려면 0(기본값 0)
4	<code>--blkio-weight-device</code>	블록 IO 가중치(상대적 디바이스 가중치)
5	<code>--cap-add</code>	Linux 기능 추가
6	<code>--cap-drop</code>	Linux 기능 중지
7	<code>--cgroup-parent</code>	컨테이너의 선택적 상위 cgroup
8	<code>--cgroupns</code>	API 1.41+ < <a href="https://docs.docker.com/engine/api/v1.41/">https://docs.docker.com/engine/api/v1.41/</a> > <code>__Cgroup</code> 네임스페이스 사용

	Docker 실행 인수	설명
		(host private) 'host': Docker 호스트의 cgroup 네임스페이스 'private'에서 컨테이너를 실행합니다. 자체 프라이빗 cgroup 네임스페이스에서 컨테이너를 실행합니다 ": 대몬(daemon)의 default-cgroupns-mode 옵션으로 구성된 대로 cgroup 네임스페이스를 사용합니다(기본값)
9	<code>-\-cidfile</code>	컨테이너 ID를 파일에 기록
10	<code>-\-cpu-count</code>	CPU 개수(Windows만 해당)
11	<code>-\-cpu-percent</code>	CPU 비율(Windows만 해당)
12	<code>-\-cpu-period</code>	CPU CFS(Completely Fair Scheduler) 기간 제한
13	<code>-\-cpu-quota</code>	CPU CFS (Completely Fair Scheduler) 할당량 제한
14	<code>-\-cpu-rt-period</code>	API 1.25+ < <a href="https://docs.docker.com/engine/api/v1.25/">https://docs.docker.com/engine/api/v1.25/</a> > <code>__CPU</code> 실시간 기간(마이크로초 단위) 제한
15	<code>-\-cpu-rt-runtime</code>	API 1.25+ < <a href="https://docs.docker.com/engine/api/v1.25/">https://docs.docker.com/engine/api/v1.25/</a> > <code>__CPU</code> 실시간 런타임 제한 (마이크로초 단위)
16	<code>-\-cpu-shares , -c</code>	CPU 점유율(상대적 가중치)
17	<code>-\-cpus</code>	API 1.25+ < <a href="https://docs.docker.com/engine/api/v1.25/">https://docs.docker.com/engine/api/v1.25/</a> > <code>__CPU</code> 개수

	Docker 실행 인수	설명
18	<code>--cpuset-cpus</code>	실행이 가능한 CPU(0-3, 0,1)
19	<code>--cpuset-mems</code>	실행을 허용하는 MEM(0-3, 0,1)
20	<code>--detach</code> , <code>-d</code>	컨테이너를 백그라운드에서 실행하고 컨테이너 ID 인쇄
21	<code>--detach-keys</code>	컨테이너 분리를 위한 키 시퀀스 재정의
22	<code>--device</code>	컨테이너에 호스트 디바이스 추가
23	<code>--device-cgroup-rule</code>	cgroup 허용 디바이스 목록에 규칙 추가
24	<code>--device-read-bps</code>	디바이스의 읽기 속도(초당 바이트) 제한
25	<code>--device-read-iops</code>	디바이스의 읽기 속도(초당 IO) 제한
26	<code>--device-write-bps</code>	디바이스 쓰기 속도(초당 바이트) 제한
27	<code>--device-write-iops</code>	디바이스 쓰기 속도(초당 IO) 제한
28	<code>--disable-content-trust</code>	이미지 인증 건너뛰기
29	<code>--dns</code>	사용자 지정 DNS 서버 설정
30	<code>--dns-opt</code>	DNS 옵션 설정
31	<code>--dns-option</code>	DNS 옵션 설정

	Docker 실행 인수	설명
32	<code>--dns-search</code>	사용자 지정 DNS 검색 도메인 설정
33	<code>--domainname</code>	컨테이너 NIS 도메인 이름
34	<code>--gpus</code>	API 1.40+ < <a href="https://docs.docker.com/engine/api/v1.40/">https://docs.docker.com/engine/api/v1.40/</a> >__컨테이너에 추가할 GPU 디바이스('all'은 모든 GPU 통과)
35	<code>--group-add</code>	가입할 추가 그룹 추가
36	<code>--health-cmd</code>	상태 확인을 위해 실행할 명령
37	<code>--health-interval</code>	검사 실행 사이의 시간(ms h) (기본값 0초)
38	<code>--health-retries</code>	비정상 보고를 위해 필요한 연속 실패 수
39	<code>--health-start-period</code>	API 1.29+ < <a href="https://docs.docker.com/engine/api/v1.29/">https://docs.docker.com/engine/api/v1.29/</a> >__상태 재시도 카운트다운을 시작하기 전에 컨테이너가 초기화되는 시작 기간(ms h) (기본값 0초)
40	<code>--health-timeout</code>	한 번의 검사를 실행할 수 있는 최대 시간(ms h) (기본값 0초)
41	<code>--help</code>	인쇄 사용량
42	<code>--hostname</code> , <code>-h</code>	컨테이너 호스트 이름

	Docker 실행 인수	설명
43	<code>--init</code>	API 1.25+ < <a href="https://docs.docker.com/engine/api/v1.25/">https://docs.docker.com/engine/api/v1.25/</a> > _신호를 전달하고 프로세스를 회수하는 컨테이너 내부에서 초기화 실행
44	<code>--interactive , -i</code>	연결되지 않은 경우에도 STDIN을 연 상태로 유지
45	<code>--io-maxbandwidth</code>	시스템 드라이브의 최대 IO 대역폭 제한(Windows만 해당)
46	<code>--io-maxiops</code>	시스템 드라이브의 최대 IOPS 제한(Windows만 해당)
47	<code>--ip</code>	IPv4 주소(예: 172.30.100.104)
48	<code>--ip6</code>	IPv6 주소(예: 2001:db8: :33)
49	<code>--ipc</code>	사용할 IPC 모드
50	<code>--isolation</code>	컨테이너 격리 기술
51	<code>--kernel-memory</code>	커널 메모리 제한
52	<code>--label , -l</code>	컨테이너에 메타데이터 설정
53	<code>--label-file</code>	줄로 구분된 레이블 파일 읽기
54	<code>--link</code>	다른 컨테이너에 링크 추가
55	<code>--link-local-ip</code>	컨테이너 IPv4/IPv6 link-local 주소
56	<code>--log-driver</code>	컨테이너용 로깅 드라이버
57	<code>--log-opt</code>	로그 드라이버 옵션

	Docker 실행 인수	설명
58	<code>--mac-address</code>	컨테이너 MAC 주소(예: 92:d0:c6:0a:29:33)
59	<code>--memory</code> , <code>-m</code>	메모리 제한
60	<code>--memory-reservation</code>	메모리 소프트 제한
61	<code>--memory-swap</code>	스왑 제한은 메모리에 스왑을 더한 값과 같음:'-1'인 경우 무제한 스왑 활성화
62	<code>--memory-swappiness</code>	컨테이너 메모리 스왑 조정 (0~100)
63	<code>--name</code>	컨테이너에 이름 지정
64	<code>--net</code>	컨테이너를 네트워크에 연결
65	<code>--net-alias</code>	컨테이너에 네트워크 범위 별칭 추가
66	<code>--network</code>	컨테이너를 네트워크에 연결
67	<code>--network-alias</code>	컨테이너에 네트워크 범위 별칭 추가
68	<code>--no-healthcheck</code>	모든 컨테이너 지정 HEALTHCHECK 비활성화
69	<code>--oom-kill-disable</code>	OOM Killer 비활성화
70	<code>--oom-score-adj</code>	호스트의 OOM 기본 설정 조정 (-1000~1000)
71	<code>--pid</code>	사용할 PID 네임스페이스

	Docker 실행 인수	설명
72	<code>--pids-limit</code>	컨테이너 pid 제한 조정(무제한의 경우 -1 설정)
73	<code>--platform</code>	API 1.32+ < <a href="https://docs.docker.com/engine/api/v1.32/">https://docs.docker.com/engine/api/v1.32/</a> >_서버가 멀티플랫폼을 지원하는 경우 플랫폼 설정
74	<code>--privileged</code>	이 컨테이너에 확장된 권한 부여
75	<code>--publish, -p</code>	컨테이너의 포트를 호스트에 게시
76	<code>--publish-all, -P</code>	노출된 모든 포트를 임의 포트에 게시
77	<code>--pull</code>	실행하기 전에 이미지 가져오기("always" "never")
78	<code>--read-only</code>	컨테이너의 루트 파일 시스템을 읽기 전용으로 마운트
79	<code>--restart</code>	컨테이너 종료 시 적용할 재시작 정책
80	<code>--rm</code>	컨테이너 종료 시 자동으로 제거
81	<code>--runtime</code>	이 컨테이너에 사용할 런타임
82	<code>--security-opt</code>	보안 옵션
83	<code>--shm-size</code>	/dev/shm 크기
84	<code>--sig-proxy</code>	프로세스에 대한 프록시 수신 신호

	Docker 실행 인수	설명
85	<code>-\-stop-timeout</code>	API 1.25+ < <a href="https://docs.docker.com/engine/api/v1.25/">https://docs.docker.com/engine/api/v1.25/</a> >_컨테이너 중지 시 소요되는 제한 시간(초)
86	<code>-\-storage-opt</code>	컨테이너의 스토리지 드라이버 옵션
87	<code>-\-sysctl</code>	Sysctl 옵션
88	<code>-\-tmpfs</code>	tmpfs 디렉터리 마운트
89	<code>-\-tty , -t</code>	pseudo-TTY 할당
90	<code>-\-ulimit</code>	무제한 옵션
91	<code>-\-usersns</code>	사용할 사용자 네임스페이스
92	<code>-\-uts</code>	사용할 UTS 네임스페이스
93	<code>-\-volume , -v</code>	볼륨 바인드 마운트
94	<code>-\-volume-driver</code>	컨테이너용 옵션 볼륨 드라이버
95	<code>-\-volumes-from</code>	지정된 컨테이너에서 볼륨 마운트

이전 런타임 구성으로 시뮬레이션 작업을 실행하면 AWS RoboMaker는 4XX 오류 코드와 함께 시뮬레이션에 실패하게 됩니다.

## 메타데이터 요구 사항

### 컨테이너 이미지:

- 반드시 [Open Container Initiative\(OCI\)](#) 볼만 사항이어야 합니다.
- 반드시 X86\_64 아키텍처용으로 빌드되어야 합니다. 다른 아키텍처용으로 빌드된 경우, AWS RoboMaker는 4XX 오류 코드와 함께 시뮬레이션에 실패하게 됩니다.

- 반드시 압축하지 않은 크기가 40GB 이하여야 합니다. 압축되지 않은 컨테이너 이미지가 40GB 이상인 경우 AWS RoboMaker는 4XX 오류 코드와 함께 시뮬레이션에 실패하게 됩니다.
- 반드시 V2 이미지 매니페스트가 있어야 하며, 스키마 버전 2와 호환되어야 합니다.
- 반드시 Linux 기반 기본 이미지를 사용해야 합니다. Linux 기반 기본 이미지를 사용하지 않는 경우, AWS RoboMaker는 4XX 오류 코드와 함께 시뮬레이션에 실패하게 됩니다.
- 반드시 서로 호환되는 개발 환경과 운영 체제를 사용해야 합니다. 다음은 개발 환경과 운영 체제의 호환 가능한 조합의 예입니다.
  - 로봇 운영 체제(ROS) Melodic – ubuntu:bionic
  - 로봇 운영 체제(ROS) 2 Foxy – ubuntu:focal

호환 가능한 로봇 프레임워크와 운영 체제의 조합을 사용하지 않으면 시뮬레이션에서 예상치 못한 동작이 나타날 수 있습니다.

## 바이너리 요구 사항

컨테이너 이미지의 바이너리 요구 사항은 다음과 같습니다.

GUI 스트리밍을 지원하려면 다음 바이너리를 설치하고 소싱하는 것이 좋습니다.

- `devilspie`

컨테이너 이미지는 실행 파일에 절대 경로를 사용하는 것이 좋습니다. 또한 컨테이너 내부의 실행 파일을 올바르게 실행하는 것이 좋습니다. 실행 파일 경로를 찾지 못하면 시뮬레이션이 실패하게 됩니다.

## GPU 요구 사항

컨테이너 이미지:

- 애플리케이션에서 OpenGL을 사용하는 경우 반드시 `glvnd`가 설치되어 있어야 합니다.
- 애플리케이션에서 CUDA를 사용하는 경우 반드시 NVIDIA CUDA 11.2 이하가 설치되어 있어야 합니다.
- 애플리케이션에서 OpenGL을 사용하는 경우 반드시 OpenGL 버전 4.6 이하가 있어야 합니다.
- 애플리케이션에서 Vulkan API를 사용하는 경우 반드시 Vulkan 버전 1.2 이하가 있어야 합니다.
- 애플리케이션에서 OpenCL을 사용하는 경우 반드시 OpenCL 버전 1.2 이하가 있어야 합니다.

### 참고

AWS RoboMaker는 오프스크린 렌더링에 대해서만 Vulkan을 지원하여, GUI 디스플레이에서는 작동하지 않습니다. 따라서 Vulkan을 사용하는 경우 streamUI를 false로 설정해야 합니다.

GPU 이미지를 생성하는 방법에 대한 자세한 지침은 [GPU 애플리케이션을 실행하기 위한 이미지 생성 항목](#)을 참조하세요.

### Dockerfile 및 환경 변수 요구 사항

컨테이너 이미지는 반드시 소싱을 위한 엔트리 포인트 스크립트를 제공해야 합니다. 엔트리 포인트 스크립트의 마지막 줄에 반드시 `exec "${@:1}"`가 있어야 AWS RoboMaker가 엔트리 포인트 스크립트를 실행할 수 있습니다. 엔트리 포인트 스크립트를 실행하면 `roslaunch package-name` 명령을 사용할 수 있습니다. `launch-file` 명령을 사용하여 컨테이너를 실행할 수 있습니다.

컨테이너 이미지는 Dockerfile에서 VOLUME를 사용할 수 없습니다. VOLUME가 Dockerfile에 있는 경우 시뮬레이션은 4XX 오류 코드와 함께 실패하게 됩니다.

Dockerfile의 EXPOSE 키워드는 AWS RoboMaker에서 무시됩니다. EXPOSE 키워드로 노출된 모든 포트는 시스템에서 자동으로 노출하지 않습니다. 시뮬레이션에서 포트를 노출하려는 경우 AWS RoboMaker [포트 포워딩 구성](#)을 사용할 수 있습니다.

AWS RoboMaker은 다음 환경 변수를 사용합니다. AWS에서 시뮬레이션을 실행하면 AWS RoboMaker가 다음과 같은 환경 변수에 지정한 모든 값을 덮어씁니다.

- ROBOMAKER\*
- DCV\_VIRTUAL\_SESSION
- XDG\_SESSION\_ID
- DCV\_SESSION\_ID
- XDG\_SESSION\_TYPE
- XDG\_RUNTIME\_DIR
- SHLVL
- XAUTHORITY

Dockerfile에서는 CMD를 지정할 수 없습니다. 그렇게 하는 경우 AWS RoboMaker가 시뮬레이션 launchConfig에서 해당 명령으로 덮어씁니다.

## 네트워크, 마운트, 보안 및 사용자 요구 사항

컨테이너에서 ROS를 실행하고 있고 로봇 애플리케이션과 시뮬레이션 애플리케이션 간에 통신이 필요한 경우, 다음과 같은 로봇 프레임워크를 설정해야 합니다.

- ROS Master
- Gazebo Master
- ROS IP

컨테이너에 있는 `/etc/resolv.conf` 파일을 사용자 지정할 수 없습니다. AWS RoboMaker가 파일을 자체 파일로 덮어씁니다.

AWS에서 Dockerfile을 실행 중인 경우 이미지를 마운트할 수 없습니다. Dockerfile에서 Mount를 지정하는 경우 AWS RoboMaker는 4XX 오류 코드와 함께 시뮬레이션에 실패하게 됩니다.

컨테이너 이미지는 기본 Docker seccomp 프로파일로 차단된 시스템 호출을 사용할 수 없습니다. 차단된 시스템 호출에 대한 자세한 내용은 [Seccomp 보안 프로파일](#)을 참조하세요.

이미지를 실행하는 사용자를 지정하려면 Dockerfile에서 USER 키워드를 지정할 수 있습니다. 사용자를 지정하지 않는 경우 AWS RoboMaker는 컨테이너의 루트 사용자를 사용합니다.

컨테이너 이미지에서 USER를 이름 또는 UID:GID로 지정할 수 있습니다. 컨테이너 이미지에 UID가 없는 경우 기본값은 1000입니다.

## 기타 요구 사항

컨테이너 이미지는 `/opt/amazon/robomaker` 또는 해당 하위 폴더에 데이터를 저장할 수 없습니다. AWS RoboMaker만 해당 디렉터리만 사용할 수 있습니다. 해당 디렉터리를 사용하면 시뮬레이션이 제대로 작동하지 않을 수 있습니다.

다음 런타임 구성은 지원되지 않습니다.

	Docker 실행 인수	설명
1	<code>--add-host</code>	사용자 지정 host-to-IP 매핑 (host:ip) 추가
2	<code>--attach</code> , <code>-a</code>	STDIN, STDOUT 또는 STDERR에 연결

	Docker 실행 인수	설명
3	<code>--blkio-weight</code>	10에서 1000 사이의 블록 IO(상대적 가중치), 비활성화하려면 0(기본값 0)
4	<code>--blkio-weight-device</code>	블록 IO 가중치(상대적 디바이스 가중치)
5	<code>--cap-add</code>	Linux 기능 추가
6	<code>--cap-drop</code>	Linux 기능 중지
7	<code>--cgroup-parent</code>	컨테이너의 선택적 상위 cgroup
8	<code>--cgroupns</code>	API 1.41+ < <a href="https://docs.docker.com/engine/api/v1.41/">https://docs.docker.com/engine/api/v1.41/</a> > __Cgroup 네임스페이스 사용 (host private) 'host': Docker 호스트의 cgroup 네임스페이스 'private'에서 컨테이너를 실행합니다: 자체 프라이빗 cgroup 네임스페이스에서 컨테이너를 실행합니다 ": 대몬(daemon)의 default-cgroupns-mode 옵션으로 구성된 대로 cgroup 네임스페이스를 사용합니다(기본값)
9	<code>--cidfile</code>	컨테이너 ID를 파일에 기록
10	<code>--cpu-count</code>	CPU 개수(Windows만 해당)
11	<code>--cpu-percent</code>	CPU 비율(Windows만 해당)
12	<code>--cpu-period</code>	CPU CFS(Completely Fair Scheduler) 기간 제한

	Docker 실행 인수	설명
13	<code>--cpu-quota</code>	CPU CFS (Completely Fair Scheduler) 할당량 제한
14	<code>--cpu-rt-period</code>	API 1.25+ < <a href="https://docs.docker.com/engine/api/v1.25/">https://docs.docker.com/engine/api/v1.25/</a> > <code>_CPU</code> 실시간 기간(마이크로초 단위) 제한
15	<code>--cpu-rt-runtime</code>	API 1.25+ < <a href="https://docs.docker.com/engine/api/v1.25/">https://docs.docker.com/engine/api/v1.25/</a> > <code>_CPU</code> 실시간 런타임 제한 (마이크로초 단위)
16	<code>--cpu-shares</code> , <code>-c</code>	CPU 점유율(상대적 가중치)
17	<code>--cpus</code>	API 1.25+ < <a href="https://docs.docker.com/engine/api/v1.25/">https://docs.docker.com/engine/api/v1.25/</a> > <code>_CPU</code> 개수
18	<code>--cpuset-cpus</code>	실행이 가능한 CPU(0-3, 0,1)
19	<code>--cpuset-mems</code>	실행을 허용하는 MEM(0-3, 0,1)
20	<code>--detach</code> , <code>-d</code>	컨테이너를 백그라운드에서 실행하고 컨테이너 ID 인쇄
21	<code>--detach-keys</code>	컨테이너 분리를 위한 키 시퀀스 재정의
22	<code>--device</code>	컨테이너에 호스트 디바이스 추가
23	<code>--device-cgroup-rule</code>	cgroup 허용 디바이스 목록에 규칙 추가

	Docker 실행 인수	설명
24	<code>--device-read-bps</code>	디바이스의 읽기 속도(초당 바이트) 제한
25	<code>--device-read-iops</code>	디바이스의 읽기 속도(초당 IO) 제한
26	<code>--device-write-bps</code>	디바이스 쓰기 속도(초당 바이트) 제한
27	<code>--device-write-iops</code>	디바이스 쓰기 속도(초당 IO) 제한
28	<code>--disable-content-trust</code>	이미지 인증 건너뛰기
29	<code>--dns</code>	사용자 지정 DNS 서버 설정
30	<code>--dns-opt</code>	DNS 옵션 설정
31	<code>--dns-option</code>	DNS 옵션 설정
32	<code>--dns-search</code>	사용자 지정 DNS 검색 도메인 설정
33	<code>--domainname</code>	컨테이너 NIS 도메인 이름
34	<code>--gpus</code>	API 1.40+ < <a href="https://docs.docker.com/engine/api/v1.40/">https://docs.docker.com/engine/api/v1.40/</a> > _컨테이너에 추가할 GPU 디바이스('all'은 모든 GPU 통과)
35	<code>--group-add</code>	가입할 추가 그룹 추가
36	<code>--health-cmd</code>	실행하여 상태 확인
37	<code>--health-interval</code>	검사 실행 사이의 시간(msm h) (기본값 0초)

	Docker 실행 인수	설명
38	<code>--health-retries</code>	비정상 보고를 위해 필요한 연속 실패 수
39	<code>--health-start-period</code>	API 1.29+ < <a href="https://docs.docker.com/engine/api/v1.29/">https://docs.docker.com/engine/api/v1.29/</a> > _상태 재시도 카운트다운을 시작하기 전에 컨테이너가 초기화되는 시작 기간(msm h) (기본값 0초)
40	<code>--health-timeout</code>	한 번의 검사를 실행할 수 있는 최대 시간(msm h) (기본값 0초)
41	<code>--help</code>	인쇄 사용량
42	<code>--hostname , -h</code>	컨테이너 호스트 이름
43	<code>--init</code>	API 1.25+ < <a href="https://docs.docker.com/engine/api/v1.25/">https://docs.docker.com/engine/api/v1.25/</a> > _신호를 전달하고 프로세스를 회수하는 컨테이너 내부에서 초기화 실행
44	<code>--interactive , -i</code>	연결되지 않은 경우에도 STDIN을 연 상태로 유지
45	<code>--io-maxbandwidth</code>	시스템 드라이브의 최대 IO 대역폭 제한(Windows만 해당)
46	<code>--io-maxiops</code>	시스템 드라이브의 최대 IOPS 제한(Windows만 해당)
47	<code>--ip</code>	IPv4 주소(예: 172.30.100.104)
48	<code>--ip6</code>	IPv6 주소(예: 2001:db8: :33)
49	<code>--ipc</code>	사용할 IPC 모드

	Docker 실행 인수	설명
50	<code>--isolation</code>	컨테이너 격리 기술
51	<code>--kernel-memory</code>	커널 메모리 제한
52	<code>--label , -l</code>	컨테이너에 메타데이터 설정
53	<code>--label-file</code>	줄로 구분된 레이블 파일 읽기
54	<code>--link</code>	다른 컨테이너에 링크 추가
55	<code>--link-local-ip</code>	컨테이너 IPv4/IPv6 link-local 주소
56	<code>--log-driver</code>	컨테이너용 로깅 드라이버
57	<code>--log-opt</code>	로그 드라이버 옵션
58	<code>--mac-address</code>	컨테이너 MAC 주소(예: 92:d0:c 6:0 a: 29:33)
59	<code>--memory , -m</code>	메모리 제한
60	<code>--memory-reservation</code>	메모리 소프트 제한
61	<code>--memory-swap</code>	스왑 제한은 메모리에 스왑을 더한 값과 같음:'-1'인 경우 무제한 스왑 활성화
62	<code>--memory-swappiness</code>	컨테이너 메모리 스왑 조정 (0~100)
63	<code>--name</code>	컨테이너에 이름 지정
64	<code>--net</code>	컨테이너를 네트워크에 연결
65	<code>--net-alias</code>	컨테이너에 네트워크 범위 별칭 추가
66	<code>--network</code>	컨테이너를 네트워크에 연결

	Docker 실행 인수	설명
67	<code>--network-alias</code>	컨테이너에 네트워크 범위 별칭 추가
68	<code>--no-healthcheck</code>	모든 컨테이너 지정 HEALTHCHECK 비활성화
69	<code>--oom-kill-disable</code>	OOM Killer 비활성화
70	<code>--oom-score-adj</code>	호스트의 OOM 기본 설정 조정 (-1000~1000)
71	<code>--pid</code>	사용할 PID 네임스페이스
72	<code>--pids-limit</code>	컨테이너 pid 제한 조정(무제한의 경우 -1 설정)
73	<code>--platform</code>	API 1.32+ < <a href="https://docs.docker.com/engine/api/v1.32/">https://docs.docker.com/engine/api/v1.32/</a> >_서버가 멀티플랫폼을 지원하는 경우 플랫폼 설정
74	<code>--privileged</code>	이 컨테이너에 확장된 권한 부여
75	<code>--publish , -p</code>	컨테이너의 포트를 호스트에 게시
76	<code>--publish-all , -P</code>	노출된 모든 포트를 임의 포트에 게시
77	<code>--pull</code>	실행하기 전에 이미지 가져오기("always" " never")
78	<code>--read-only</code>	컨테이너의 루트 파일 시스템을 읽기 전용으로 마운트
79	<code>--restart</code>	컨테이너 종료 시 적용할 재시작 정책

	Docker 실행 인수	설명
80	<code>--rm</code>	컨테이너 종료 시 자동으로 제거
81	<code>--runtime</code>	이 컨테이너에 사용할 런타임
82	<code>--security-opt</code>	보안 옵션
83	<code>--shm-size</code>	/dev/shm 크기
84	<code>--sig-proxy</code>	프로세스에 대한 프록시 수신 신호
85	<code>--stop-timeout</code>	API 1.25+ < <a href="https://docs.docker.com/engine/api/v1.25/">https://docs.docker.com/engine/api/v1.25/</a> >__컨테이너 종지에 소요되는 제한 시간(초)
86	<code>--storage-opt</code>	컨테이너의 스토리지 드라이버 옵션
87	<code>--sysctl</code>	Sysctl 옵션
88	<code>--tmpfs</code>	tmpfs 디렉터리 마운트
89	<code>--tty , -t</code>	pseudo-TTY 할당
90	<code>--ulimit</code>	무제한 옵션
91	<code>--usersns</code>	사용할 사용자 네임스페이스
92	<code>--uts</code>	사용할 UTS 네임스페이스
93	<code>--volume , -v</code>	볼륨 바인드 마운트
94	<code>--volume-driver</code>	컨테이너용 옵션 볼륨 드라이버
95	<code>--volumes-from</code>	지정된 컨테이너에서 볼륨 마운트

이전 런타임 구성으로 시뮬레이션 작업을 실행하면 AWS RoboMaker는 4XX 오류 코드와 함께 시뮬레이션에 실패하게 됩니다.

## GPU 애플리케이션을 실행하기 위한 이미지 생성

AWS RoboMaker GPU 시뮬레이션 작업은 CUDA, OpenGL, OpenCL, Vulkan API 액세스를 지원합니다. 따라서 이러한 API를 사용하는 애플리케이션의 이미지에 해당 드라이버가 설치되어 있어야 합니다.

### Note

OpenGL API를 얻으려면 Nvidia 기본 이미지를 사용하는 것이 좋습니다. 튜토리얼에 사용된 예제 Dockerfile은 OpenGL 지원을 제공하는 `nvidia/opengl:1.0-glvnd-runtime-ubuntu20.04`만 다룹니다. CUDA, Vulkan, OpenCL을 지원하는 컨테이너 이미지를 찾으려면 Nvidia 설명서를 참조하세요.

DCV 디스플레이를 GPU 렌더링과 함께 사용하려면 `nice-dcv-g1`을 설치해야 합니다. 참고로 X0은 GPU와 통신하는 시스템의 Xorg 프로세스입니다. X1과 X2는 대신 XDCV 프로세스입니다. X1 또는 X2에서 OpenGL 애플리케이션을 시작하면 호출을 리디렉션하고 `nice-dcv-g1`가 GPU를 사용할 수 있는 X0에서 렌더링을 수행합니다.

`nice-dcv-g1`을 설치하려면 아카이브를 다운로드하고 압축을 풀고 DCV 공개 설명서에 따라 `nice-dcv-g1` 패키지를 설치하세요. [Install the NICE DCV Server on Linux](#)를 참조하세요.

다음 예제는 `ubuntu18.04` 기본 이미지에 `nice-dcv-g1_2021.2`를 설치하는 Dockerfile을 보여줍니다.

```
FROM nvidia/opengl:1.0-glvnd-runtime-ubuntu20.04

ENV DEBIAN_FRONTEND="noninteractive"

RUN apt-get update && apt-get install -y --no-install-recommends \
    ca-certificates \
    gnupg2 \
    wget

RUN wget https://d1uj6qtbmh3dt5.cloudfront.net/NICE-GPG-KEY && gpg --import NICE-GPG-KEY && \
    wget https://d1uj6qtbmh3dt5.cloudfront.net/2021.2/Servers/nice-dcv-2021.2-11048-ubuntu1804-x86_64.tgz && \
    tar xvzf nice-dcv-2021.2-11048-ubuntu1804-x86_64.tgz && \
```

```
cd nice-dcv-2021.2-11048-ubuntu1804-x86_64 && \
apt install -y ./nice-dcv-gl_2021.2.944-1_amd64.ubuntu1804.deb
```

GPU 애플리케이션 빌드에 대한 자세한 지침은 [ROS2 Foxy와 Gazebo 11를 사용하여 GPU 샘플 애플리케이션 실행](#) 항목을 참조하세요.

## hello world 샘플 애플리케이션을 실행하기 위한 이미지 생성

제공되는 Hello World 샘플 애플리케이션을 사용하여 시뮬레이션 및 로봇 애플리케이션을 생성하고 실행하는 방법을 이해할 수 있습니다. 다음 섹션에서는 다음과 같은 개발 환경을 위한 이미지를 만들고 실행하는 방법을 보여줍니다.

- ROS Melodic 및 Gazebo 9
- ROS 2 Foxy 및 Gazebo 11

ROS는 로봇 애플리케이션에 사용되는 로봇 운영 체제입니다. Gazebo는 시뮬레이션 애플리케이션을 위한 운영 체제입니다. AWS RoboMaker는 두 소프트웨어 제품군을 모두 사용하여 컨테이너 이미지를 사용하고 유효성 검사를 제공합니다.

튜토리얼은 AWS RoboMaker 컨테이너 이미지를 사용하여 Hello World 로봇 및 시뮬레이션 애플리케이션을 설정하는 방법을 안내합니다. Hello World 애플리케이션은 AWS RoboMaker 사용 방법을 이해하는 데 도움을 주는 예제 애플리케이션입니다.

각 튜토리얼에서 로봇 및 시뮬레이션 애플리케이션 모두에 사용할 이미지를 만들게 됩니다. 이미지를 로컬에서 실행하여 작동 방식을 테스트할 수 있습니다. 시뮬레이션이 제대로 작동하면 Amazon ECR로 푸시하고 클라우드에서 시뮬레이션 작업을 실행할 수 있습니다. 시뮬레이션 작업에 대한 자세한 내용은 [AWS RoboMaker를 이용한 시뮬레이션](#) 섹션을 참조하십시오.

## ROS 2, Foxy, Gazebo 11을 사용하여 샘플 애플리케이션 실행

다음 튜토리얼에서는 Hello World 로봇 애플리케이션 및 시뮬레이션 애플리케이션을 만들고 실행하여 컨테이너 이미지를 사용해 ROS 2 Foxy 및 Gazebo 11로 개발하는 방법을 보여줍니다. 이 문서에 설명된 명령을 실행하여 샘플 애플리케이션이 작동하도록 할 수 있습니다.

이 튜토리얼에서는 세 개의 컨테이너 이미지를 만들고 사용합니다. 다음은 이 예제 애플리케이션에 사용하는 디렉터리 구조를 보여줍니다.

```
### HelloWorldSampleAppROS2FoxyGazebo11 // Base Image
#   ### Dockerfile
### HelloWorldSampleAppROS2FoxyGazebo11RobotApp // Image for Robot App
```

```
#   ### Dockerfile
#   ### robot-entrypoint.sh
### HelloWorldSampleAppROS2FoxyGazebo11SimApp // Image for Simulation App
#   ### Dockerfile
#   ### simulation-entrypoint.sh
```

각 Dockerfile에는 각 이미지를 빌드하는 데 필요한 지침이 있습니다.

- 기본 이미지의 Dockerfile에는 ROS 및 Gazebo를 설정하는 명령이 있습니다.
- 로봇 애플리케이션용 Dockerfile에는 Hello World 로봇 애플리케이션을 설정하는 명령이 있습니다.
- 시뮬레이션 애플리케이션용 Dockerfile에는 Hello World 시뮬레이션 애플리케이션을 설정하는 명령이 있습니다.

로봇 애플리케이션과 시뮬레이션 애플리케이션 모두 엔트리 포인트 스크립트가 있습니다. 이 스크립트는 해당 애플리케이션의 환경을 소싱합니다. 로봇 및 시뮬레이션 애플리케이션을 시작하는 명령을 실행할 수 있는 경로를 대신 설정합니다.

## 기본 이미지 생성

기본 이미지를 만들려면 환경을 만드는 명령을 Dockerfile에 저장합니다. 그런 다음 Dockerfile을 빌드합니다.

- 다음 명령을 Dockerfile에 저장합니다.

```
# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: MIT-0
FROM ros:foxy

ENV DEBIAN_FRONTEND noninteractive

RUN apt-get clean
RUN apt-get update && apt-get install -y \
    lsb \
    unzip \
    wget \
    curl \
    sudo \
    python3-vcstool \
    python3-rosinstall \
    python3-colcon-common-extensions \
    ros-foxy-rviz2 \
```

```

ros-foxy-rqt \
ros-foxy-rqt-common-plugins \
devilspie \
xfce4-terminal

RUN wget https://packages.osrfoundation.org/gazebo.key -O - | sudo apt-key add -; \
  sh -c 'echo "deb http://packages.osrfoundation.org/gazebo/ubuntu-stable
`lsb_release -cs` main" > /etc/apt/sources.list.d/gazebo-stable.list'
RUN apt-get update && apt-get install -y gazebo11

ENV QT_X11_NO_MITSHM=1

ARG USERNAME=robomaker
RUN groupadd $USERNAME
RUN useradd -ms /bin/bash -g $USERNAME $USERNAME
RUN sh -c 'echo "$USERNAME ALL=(root) NOPASSWD:ALL" >> /etc/sudoers'
USER $USERNAME

RUN sh -c 'cd /home/$USERNAME'

# Download and build our Robot and Simulation application
RUN sh -c 'mkdir -p /home/robomaker/workspace'
RUN sh -c 'cd /home/robomaker/workspace && wget https://github.com/aws-
robotics/aws-robomaker-sample-application-helloworld/archive/3527834.zip
&& unzip 3527834.zip && mv aws-robomaker-sample-application-
helloworld-3527834771373beff0ed3630c13479567db4149e aws-robomaker-sample-
application-helloworld-ros2'
RUN sh -c 'cd /home/robomaker/workspace/aws-robomaker-sample-application-
helloworld-ros2'

RUN sudo rosdep fix-permissions
RUN rosdep update

```

Dockerfile을 생성한 후 터미널에서 다음 명령을 사용하여 빌드합니다.

```

cd ../HelloWorldSampleAppROS2FoxyGazebo11
docker build -t helloworldsampleappros2foxygazebo11:latest .

```

기본 이미지를 빌드하면 ROS 2 Foxy와 Gazebo 11이 설치됩니다. 애플리케이션을 성공적으로 실행하려면 두 라이브러리를 모두 설치해야 합니다.

## 로봇 애플리케이션용 이미지 생성

기본 이미지를 만든 후, 로봇 애플리케이션용 이미지를 만들 수 있습니다. 다음 스크립트를 Dockerfile에 저장하고 빌드합니다. 이 스크립트는 Hello World 로봇 애플리케이션을 다운로드하고 설정합니다.

```
# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: MIT-0
FROM helloworldsampleappros2foxygazebo11:latest

# Build the Robot application
RUN cd /home/robomaker/workspace/aws-robomaker-sample-application-helloworld-ros2/robot_ws && \
  /bin/bash -c "source /opt/ros/foxy/setup.bash && vcs import < .rosinstall && rosdep install --rosdistro foxy --from-paths src --ignore-src -r -y && colcon build"

COPY robot-entrypoint.sh /home/robomaker/robot-entrypoint.sh
RUN sh -c 'sudo chmod +x /home/robomaker/robot-entrypoint.sh'
RUN sh -c 'sudo chown robomaker:robomaker /home/robomaker/robot-entrypoint.sh'

CMD ros2 launch hello_world_robot rotate.launch.py
ENTRYPOINT [ "/home/robomaker/robot-entrypoint.sh" ]
```

다음 명령은 Dockerfile에서 로봇 애플리케이션용 이미지를 생성합니다.

```
cd HelloWorldSampleAppROS2FoxyGazebo11RobotApp/
HelloWorldSampleAppROS2FoxyGazebo11RobotApp
docker build -t helloworldsampleappros2foxygazebo11robotapp:latest .
```

다음은 robot-entrypoint.sh로 저장할 수 있는 스크립트의 콘텐츠입니다. 이 스크립트는 로봇 애플리케이션에 대한 환경을 소싱합니다.

```
#!/bin/bash

if [ ! -z $GAZEBO_MASTER_URI ]; then
    tmp_GAZEBO_MASTER_URI=$GAZEBO_MASTER_URI
fi

cd /home/robomaker/workspace/aws-robomaker-sample-application-helloworld-ros2/robot_ws
source /opt/ros/foxy/setup.bash
source /usr/share/gazebo-11/setup.sh
```

```
source ./install/setup.sh

if [ ! -z $tmp_GAZEBO_MASTER_URI ]; then
    export GAZEBO_MASTER_URI=$tmp_GAZEBO_MASTER_URI
    unset tmp_GAZEBO_MASTER_URI
fi

printenv

exec "${@:1}"
```

## 시뮬레이션 애플리케이션에 대한 이미지 생성

기본 이미지와 로봇 애플리케이션의 이미지를 만든 후 시뮬레이션 애플리케이션용 이미지를 만들 수 있습니다. 다음 스크립트를 Dockerfile에 저장하고 빌드합니다. 이 스크립트는 Hello World 로봇 애플리케이션을 다운로드하고 설정합니다.

```
# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: MIT-0
FROM helloworldsampleappros2foxygazebo11:latest

# Build the Simulation application
RUN cd /home/robomaker/workspace/aws-robomaker-sample-application-helloworld-ros2/
simulation_ws && \
/bin/bash -c "source /opt/ros/foxy/setup.bash && vcs import < .rosinstall && rosdep
install --rosdistro foxy --from-paths src --ignore-src -r -y && colcon build"

COPY simulation-entrypoint.sh /home/robomaker/simulation-entrypoint.sh

RUN sh -c 'sudo chmod +x /home/robomaker/simulation-entrypoint.sh'
RUN sh -c 'sudo chown robomaker:robomaker /home/robomaker/simulation-entrypoint.sh'

CMD ros2 launch hello_world_simulation empty_world.launch.py
ENTRYPOINT [ "/home/robomaker/simulation-entrypoint.sh" ]
```

다음 명령은 이미지를 생성합니다.

```
cd HelloWorldSampleAppROS2FoxyGazebo11SimApp/HelloWorldSampleAppROS2FoxyGazebo11SimApp
docker build -t helloworldsampleappros2foxygazebo11simapp:latest .
```

다음은 `simulation-entripoint.sh`로 저장할 수 있는 스크립트의 콘텐츠입니다. 이 스크립트는 시뮬레이션 애플리케이션의 환경을 소싱합니다.

```
#!/bin/bash

if [ ! -z $GAZEBO_MASTER_URI ]; then
    tmp_GAZEBO_MASTER_URI=$GAZEBO_MASTER_URI
fi

cd /home/robomaker/workspace/aws-robomaker-sample-application-helloworld-ros2/
simulation_ws
source /opt/ros/foxy/setup.bash
source /usr/share/gazebo-11/setup.sh
source ./install/setup.sh

if [ ! -z $tmp_GAZEBO_MASTER_URI ]; then
    export GAZEBO_MASTER_URI=$tmp_GAZEBO_MASTER_URI
    unset tmp_GAZEBO_MASTER_URI
fi

printenv

exec "${@:1}"
```

애플리케이션을 실행하고 Amazon ECR로 푸시하기

이미지를 생성한 후에는 로컬 Linux 환경에서 이미지가 제대로 실행되는지 확인하세요. 이미지가 실행되는지 확인한 후, 도커 이미지를 Amazon ECR로 푸시하고 시뮬레이션 작업을 생성할 수 있습니다.

다음 명령을 사용하면 로컬 Linux 환경에서 Hello World 애플리케이션을 실행할 수 있습니다.

```
docker run -it -e DISPLAY -v /tmp/.X11-unix/:/tmp/.X11-unix/ --name robot_app \
-u robomaker -e ROBOMAKER_GAZEBO_MASTER_URI=http://localhost:5555 \
-e ROBOMAKER_ROS_MASTER_URI=http://localhost:11311 \
helloworldsampleappros2foxygazebo11robotapp:latest
```

```
docker run -it -e DISPLAY -v /tmp/.X11-unix/:/tmp/.X11-unix/ --name sim_app \
-u robomaker -e ROBOMAKER_GAZEBO_MASTER_URI=http://localhost:5555 \
-e ROBOMAKER_ROS_MASTER_URI=http://localhost:11311 \
helloworldsampleappros2foxygazebo11simapp:latest
```

로봇 애플리케이션과 시뮬레이션 애플리케이션 컨테이너를 실행하면 Gazebo GUI 도구를 사용하여 시뮬레이션을 시각화할 수 있습니다. 다음 명령을 사용하십시오.

1. 시뮬레이션 애플리케이션을 실행하는 컨테이너에 연결합니다.
2. Gazebo 그래픽 사용자 인터페이스(GUI)를 실행하여 애플리케이션을 시각화하세요.

```
# Enable access to X server to launch Gazebo from docker container
$ xhost +

# Check that the robot_app and sim_app containers are running. The command should list
  both containers
$ docker container ls

# Connect to the sim app container
$ docker exec -it sim_app bash

# Launch Gazebo from within the container
$ /home/robomaker/simulation-entrypoint.sh ros2 launch gazebo_ros gzclient.launch.py
```

이미지에 태그를 추가할 수 있습니다. 다음 명령을 사용하면 이미지에 태그를 지정할 수 있습니다.

```
docker tag helloworldsampleappros2foxygazebo11robotapp:latest accountID.dkr.ecr.us-
west-2.amazonaws.com/helloworldsampleappros2foxygazebo11robotapp:latest
```

```
docker tag helloworldsampleappros2foxygazebo11simapp:latest accountID.dkr.ecr.us-
west-2.amazonaws.com/helloworldsampleappros2foxygazebo11simapp:latest
```

애플리케이션이 제대로 작동하는지 확인한 후 다음 명령을 사용하여 Amazon ECR로 푸시할 수 있습니다.

```
aws ecr get-login-password --region us-west-2 | docker login --username AWS --password-
stdin accountID.dkr.ecr.us-west-2.amazonaws.com
docker push accountID.dkr.ecr.us-west-2.amazonaws.com/
helloworldsampleappros2foxygazebo11robotapp:latest
docker push accountID.dkr.ecr.us-west-2.amazonaws.com/
helloworldsampleappros2foxygazebo11simapp:latest
```

그런 다음 이미지에서 시뮬레이션 작업을 실행할 수 있습니다. 시뮬레이션 작업에 대한 자세한 내용은 [AWS RoboMaker를 이용한 시뮬레이션](#) 섹션을 참조하십시오.

## ROS Melodic과 Gazebo 9를 사용하여 샘플 애플리케이션 실행

다음 튜토리얼에서는 Hello World 로봇 애플리케이션 및 시뮬레이션 애플리케이션을 만들고 실행하여 컨테이너 이미지를 사용해 ROS 및 Gazebo 9로 개발하는 방법을 보여줍니다. 이 문서에 설명된 명령을 실행하여 샘플 애플리케이션이 작동하도록 할 수 있습니다.

이 튜토리얼에서는 세 개의 컨테이너 이미지를 만들고 사용합니다. 다음은 이 예제 애플리케이션에 사용하는 디렉터리 구조를 보여줍니다.

```
### HelloWorldSampleAppROSMelodicGazebo9 // Base Image
# ### Dockerfile
### HelloWorldSampleAppROSMelodicGazebo9RobotApp // Image for Robot App
# ### Dockerfile
# ### robot-entrypoint.sh
### HelloWorldSampleAppROSMelodicGazebo9SimApp // Image for Simulation App
# ### Dockerfile
# ### simulation-entrypoint.sh
```

각 Dockerfile에는 각 이미지를 빌드하는 데 필요한 지침이 있습니다.

- 기본 이미지의 Dockerfile에는 ROS 및 Gazebo를 설정하는 명령이 있습니다.
- 로봇 애플리케이션용 Dockerfile에는 Hello World 로봇 애플리케이션을 설정하는 명령이 있습니다.
- 시뮬레이션 애플리케이션용 Dockerfile에는 Hello World 시뮬레이션 애플리케이션을 설정하는 명령이 있습니다.

로봇 애플리케이션과 시뮬레이션 애플리케이션 모두 엔트리 포인트 스크립트가 있습니다. 이 스크립트는 해당 애플리케이션의 환경을 소싱합니다. 이는 로봇 및 시뮬레이션 애플리케이션을 실행할 수 있는 명령을 실행할 수 있는 경로를 설정합니다.

### 기본 이미지 생성

기본 이미지를 생성하려면 예제의 명령을 저장하여 Dockerfile에 환경을 생성합니다. 그런 다음 Dockerfile을 빌드합니다.

1. 다음 명령을 Dockerfile에 저장합니다.

```
# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
```

```
# SPDX-License-Identifier: MIT-0
FROM ros:melodic

ENV DEBIAN_FRONTEND noninteractive

RUN apt-get clean
RUN apt-get update && apt-get install -y \
  lsb \
  unzip \
  wget \
  curl \
  sudo \
  python-vcstool \
  python-rosinstall \
  python3-colcon-common-extensions \
  ros-melodic-rviz \
  ros-melodic-rqt \
  ros-melodic-rqt-common-plugins \
  devilspie \
  xfce4-terminal \
  ros-melodic-gazebo-ros-pkgs \
  ros-melodic-gazebo-ros-control \
  ros-melodic-turtlebot3

ENV QT_X11_NO_MITSHM=1

ARG USERNAME=robomaker
RUN groupadd $USERNAME
RUN useradd -ms /bin/bash -g $USERNAME $USERNAME
RUN sh -c 'echo "$USERNAME ALL=(root) NOPASSWD:ALL" >> /etc/sudoers'
USER $USERNAME

RUN sh -c 'cd /home/$USERNAME'

# Download and build our Robot and Simulation application
RUN sh -c 'mkdir -p /home/robomaker/workspace'
RUN sh -c 'cd /home/robomaker/workspace && wget https://github.com/aws-robotics/
aws-robomaker-sample-application-helloworld/archive/ros1.zip && unzip ros1.zip'
RUN sh -c 'cd /home/robomaker/workspace/aws-robomaker-sample-application-
helloworld-ros1'

RUN sudo rosdep fix-permissions
RUN rosdep update
```

2. Dockerfile을 생성한 후 터미널에서 다음 명령을 사용하여 빌드합니다.

```
cd ../HelloWorldSampleAppROSMelodicGazebo9
docker build -t helloworldsampleapprosmelodicgazebo9:latest .
```

기본 이미지를 빌드하면 ROS Melodic과 Gazebo 9가 설치됩니다. 애플리케이션을 성공적으로 실행하려면 두 라이브러리를 모두 설치해야 합니다.

### 로봇 애플리케이션용 이미지 생성

기본 이미지를 만든 후 로봇 애플리케이션용 이미지를 만듭니다.

1. 다음 스크립트를 Dockerfile에 저장하고 빌드합니다. 이 스크립트는 Hello World 로봇 애플리케이션을 다운로드하고 설정합니다.

```
# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: MIT-0
FROM helloworldsampleapprosmelodicgazebo9:latest

# Build the Robot application
RUN cd /home/robomaker/workspace/aws-robomaker-sample-application-helloworld-ros1/robot_ws && \
/bin/bash -c "source /opt/ros/melodic/setup.bash && vcs import < .rosinstall && rosdep install --rosdistro melodic --from-paths src --ignore-src -r -y && colcon build"

COPY robot-entrypoint.sh /home/robomaker/robot-entrypoint.sh

RUN sh -c 'sudo chmod +x /home/robomaker/robot-entrypoint.sh'
RUN sh -c 'sudo chown robomaker:robomaker /home/robomaker/robot-entrypoint.sh'

CMD roslaunch hello_world_robot rotate.launch
ENTRYPOINT [ "/home/robomaker/robot-entrypoint.sh" ]
```

2. 다음 명령을 사용하여 Dockerfile에서 로봇 애플리케이션용 이미지를 생성합니다.

```
cd HelloWorldSampleAppROSMelodicGazebo9RobotApp/
HelloWorldSampleAppROSMelodicGazebo9RobotApp
docker build -t helloworldsampleapprosmelodicgazebo9robotapp:latest image/.
```

- 다음은 `robot-entrypoint.sh`로 저장할 수 있는 스크립트의 콘텐츠입니다. 이 스크립트는 로봇 애플리케이션에 대한 환경을 소싱합니다.

```
#!/bin/bash

if [ ! -z $GAZEBO_MASTER_URI ]; then
    tmp_GAZEBO_MASTER_URI=$GAZEBO_MASTER_URI
fi

cd /home/robomaker/workspace/aws-robomaker-sample-application-helloworld-ros1/robot_ws
source /opt/ros/melodic/setup.bash
source /usr/share/gazebo-9/setup.sh
source ./install/setup.sh

if [ ! -z $tmp_GAZEBO_MASTER_URI ]; then
    export GAZEBO_MASTER_URI=$tmp_GAZEBO_MASTER_URI
    unset tmp_GAZEBO_MASTER_URI
fi

printenv

exec "${@:1}"
```

### 시뮬레이션 애플리케이션에 대한 이미지 생성

기본 이미지와 로봇 애플리케이션의 이미지를 만든 후 시뮬레이션 애플리케이션용 이미지를 만들 수 있습니다.

- 다음 스크립트를 `Dockerfile`에 저장하고 빌드합니다. 이 스크립트는 Hello World 로봇 애플리케이션을 다운로드하고 설정합니다.

```
# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: MIT-0
FROM helloworldsampleappmelodicgazebo9:latest

# Build the Simulation application
RUN cd /home/robomaker/workspace/aws-robomaker-sample-application-helloworld-ros1/simulation_ws && \
```

```
/bin/bash -c "source /opt/ros/melodic/setup.bash && vcs import < .rosinstall &&
rosdep install --rosdistro melodic --from-paths src --ignore-src -r -y && colcon
build"
```

```
COPY simulation-entrypoint.sh /home/robomaker/simulation-entrypoint.sh
```

```
RUN sh -c 'sudo chmod +x /home/robomaker/simulation-entrypoint.sh'
```

```
RUN sh -c 'sudo chown robomaker:robomaker /home/robomaker/simulation-entrypoint.sh'
```

```
CMD roslaunch hello_world_simulation empty_world.launch
```

```
ENTRYPOINT [ "/home/robomaker/simulation-entrypoint.sh" ]
```

2. 다음 `simulation-entrypoint.sh` 스크립트를 저장하십시오. 이 스크립트는 시뮬레이션 애플리케이션의 환경을 소싱합니다.

```
#!/bin/bash

if [ ! -z $GAZEBO_MASTER_URI ]; then
    tmp_GAZEBO_MASTER_URI=$GAZEBO_MASTER_URI
fi

cd /home/robomaker/workspace/aws-robomaker-sample-application-helloworld-ros1/
simulation_ws
source /opt/ros/melodic/setup.bash
source /usr/share/gazebo-9/setup.sh
source ./install/setup.sh

if [ ! -z $tmp_GAZEBO_MASTER_URI ]; then
    export GAZEBO_MASTER_URI=$tmp_GAZEBO_MASTER_URI
    unset tmp_GAZEBO_MASTER_URI
fi

printenv

exec "${@:1}"
```

## 애플리케이션을 실행하고 ECR로 푸시하기

이미지를 생성한 후에는 로컬 Linux 환경에서 이미지가 제대로 실행되는지 확인하세요. 도커 이미지가 실행되는지 확인한 후 Amazon ECR로 푸시하여 시뮬레이션 작업을 생성할 수 있습니다.

1. 다음 명령을 사용하여 로컬 Linux 환경에서 Hello World 애플리케이션을 실행합니다.

```
docker run -it -e DISPLAY -v /tmp/.X11-unix/:/tmp/.X11-unix/ \
-u robomaker -e ROBOMAKER_GAZEBO_MASTER_URI=http://localhost:5555 \
-e ROBOMAKER_ROS_MASTER_URI=http://localhost:11311 \
helloworldsampleapprosmelodicgazebo9robotapp:latest
```

```
docker run -it -e DISPLAY -v /tmp/.X11-unix/:/tmp/.X11-unix/ \
-u robomaker -e ROBOMAKER_GAZEBO_MASTER_URI=http://localhost:5555 \
-e ROBOMAKER_ROS_MASTER_URI=http://localhost:11311 \
helloworldsampleapprosmelodicgazebo9simapp:latest
```

2. 로봇 애플리케이션과 시뮬레이션 애플리케이션 컨테이너를 실행하여 Gazebo GUI 도구를 사용하여 시뮬레이션을 시각화합니다. 다음 명령을 사용하십시오.

1. 시뮬레이션 애플리케이션을 실행하는 컨테이너에 연결합니다.
2. Gazebo 그래픽 사용자 인터페이스(GUI)를 실행하여 애플리케이션을 시각화하세요.

```
# Enable access to X server to launch Gazebo from docker container
$ xhost +

# Check that the robot_app and sim_app containers are running. The command should
list both containers
$ docker container ls

# Connect to the sim app container
$ docker exec -it sim_app bash

# Launch Gazebo from within the container
$ rosrun gazebo_ros gzclient
```

3. 이미지에 태그를 추가하여 정리된 상태를 유지할 수 있습니다. 다음 명령을 사용하여 이미지에 태그를 지정합니다.

```
docker tag
helloworldsampleapprosmelodicgazebo9robotapp:latest accountID.dkr.ecr.us-
west-2.amazonaws.com/helloworldsampleapprosmelodicgazebo9robotapp:latest
```

```
docker tag helloworldsampleapprosmelodicgazebo9simapp:latest accountID.dkr.ecr.us-
west-2.amazonaws.com/helloworldsampleapprosmelodicgazebo9simapp:latest
```

4. 애플리케이션이 제대로 작동하는지 확인한 후 다음 명령을 사용하여 Amazon ECR로 푸시할 수 있습니다.

```
aws ecr get-login-password --region us-west-2 | docker login --username AWS --
password-stdin accountID.dkr.ecr.us-west-2.amazonaws.com
docker push accountID.dkr.ecr.us-west-2.amazonaws.com/
helloworldsampleapprosmelodicgazebo9robotapp:latest
docker push accountID.dkr.ecr.us-west-2.amazonaws.com/
helloworldsampleapprosmelodicgazebo9simapp:latest
```

그런 다음 이미지에서 시뮬레이션 작업을 실행할 수 있습니다. 시뮬레이션 작업에 대한 자세한 내용은 [AWS RoboMaker를 이용한 시뮬레이션](#) 섹션을 참조하십시오.

## ROS2 Foxy와 Gazebo 11를 사용하여 GPU 샘플 애플리케이션 실행

이 튜토리얼에서는 다음 예제에 설명된 세 개의 컨테이너 이미지를 사용하여 Hello World 로봇 애플리케이션과 시뮬레이션 애플리케이션을 만들고 실행함으로써 컨테이너 이미지 내에서 GPU 드라이버를 사용하여 ROS 2 Foxy 및 Gazebo 11로 개발하는 방법을 설명합니다.

```
### SampleGPUBaseApp // Base Image
#   ### Dockerfile
### SampleGPURobotApp // Image for Robot App
#   ### Dockerfile
#   ### robot-entrypoint.sh
### SampleGPUSimulationApp // Image for Simulation App
#   ### Dockerfile
#   ### simulation-entrypoint.sh
```

각 Dockerfile에는 각 이미지를 빌드하는 데 필요한 지침이 있습니다.

- 기본 이미지의 Dockerfile에는 ROS, Gazebo, GPU 드라이버를 설정하는 명령이 포함되어 있습니다.
- 로봇 애플리케이션용 Dockerfile에는 Hello World 로봇 애플리케이션을 설정하는 명령이 포함되어 있습니다.
- 시뮬레이션 애플리케이션용 Dockerfile에는 Hello World 시뮬레이션 애플리케이션을 설정하는 명령이 포함되어 있습니다.

로봇 애플리케이션과 시뮬레이션 애플리케이션 모두 엔트리 포인트 스크립트가 있습니다. 이 스크립트는 해당 애플리케이션의 환경을 소싱하고 로봇 및 시뮬레이션 애플리케이션을 시작하기 위한 명령을 실행할 경로를 설정합니다.

## 기본 GPU 이미지 생성

다음 Dockerfile에는 NVIDIA OpenGL에서 기본 이미지를 생성하고 DCV를 설치하는 명령이 포함되어 있습니다.

- SampleGPUBaseApp 디렉터리의 Dockerfile에 다음 명령을 저장합니다.

```
# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: MIT-0
FROM nvidia/opengl:1.0-glvnd-runtime-ubuntu20.04

ENV DEBIAN_FRONTEND="noninteractive"
ENV QT_X11_NO_MITSHM=1

RUN apt-get clean
RUN apt-get update && apt-get install -y --no-install-recommends \
    ca-certificates \
    devilspie \
    gnupg2 \
    mesa-utils \
    sudo \
    unzip \
    wget \
    xfce4-terminal

RUN wget https://d1uj6qtbmh3dt5.cloudfront.net/NICE-GPG-KEY && gpg --import NICE-GPG-KEY && \
    wget https://d1uj6qtbmh3dt5.cloudfront.net/2021.2/Servers/nice-dcv-2021.2-11048-ubuntu1804-x86_64.tgz && \
    tar xvzf nice-dcv-2021.2-11048-ubuntu1804-x86_64.tgz && \
    cd nice-dcv-2021.2-11048-ubuntu1804-x86_64 && \
    apt install -y ./nice-dcv-gl_2021.2.944-1_amd64.ubuntu1804.deb

RUN apt update && apt -y install locales && \
    locale-gen en_US en_US.UTF-8 && \
    update-locale LC_ALL=en_US.UTF-8 LANG=en_US.UTF-8

ENV LANG=en_US.UTF-8
```

```

RUN apt-get update && apt-get install -y --no-install-recommends curl lsb-release

RUN curl -sSL https://raw.githubusercontent.com/ros/rosdistro/master/ros.key -o /usr/
share/keyrings/ros-archive-keyring.gpg && \
    curl -s https://raw.githubusercontent.com/ros/rosdistro/master/ros.asc | apt-
key add - && \
    echo "deb [arch=$(dpkg --print-architecture) signed-by=/usr/share/keyrings/ros-
archive-keyring.gpg] http://packages.ros.org/ros2/ubuntu $(lsb_release -cs) main" |
tee /etc/apt/sources.list.d/ros2.list > /dev/null && \
    apt update && \
    apt install -y ros-foxy-desktop && \
    /bin/bash -c "source /opt/ros/foxy/setup.bash"

RUN apt -y install ros-foxy-gazebo-ros-pkgs

RUN apt-key adv --fetch-keys 'http://packages.osrfoundation.org/gazebo.key' && \
    apt update && \
    apt install -y python3-rosdep git

RUN if [ ! -f "/etc/ros/rosdep/sources.list.d/20-default.list" ]; then \
    rosdep init; \
fi

RUN rosdep update

RUN apt-get install -y python3-apt python3-pip python3-vcstool python3-testresources

RUN pip3 install -U pytest setuptools colcon-ros-bundle

RUN useradd --create-home robomaker && \
    sh -c 'echo "robomaker ALL=(root) NOPASSWD:ALL" >> /etc/sudoers'

RUN sh -c 'mkdir -p /home/robomaker/workspace' && \
    sh -c 'cd /home/robomaker/workspace && wget https://github.com/aws-robotics/
aws-robomaker-sample-application-helloworld/archive/ros2.zip && unzip ros2.zip'

```

Dockerfile을 생성한 후 터미널에서 다음 명령을 사용하여 빌드합니다.

```

cd SampleGPUBaseApp
docker build -t samplegpubaseapp:latest .

```

기본 이미지를 빌드하려면 ROS 2 Foxy, Gazebo 11, NVIDIA OpenGL, NICE-DCV를 설치합니다.

## 로봇 애플리케이션용 이미지 생성

기본 이미지를 만든 후, 로봇 애플리케이션용 이미지를 만들 수 있습니다. 다음 스크립트를 SampleGPURobotApp 디렉터리의 Dockerfile에 저장하고 빌드합니다. 이 스크립트는 Hello World 로봇 애플리케이션을 다운로드하고 설정합니다.

```
# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: MIT-0
FROM samplegpubaseapp:latest

# Build the Robot application
RUN cd /home/robomaker/workspace/aws-robomaker-sample-application-helloworld-ros2/robot_ws && \
  /bin/bash -c "source /opt/ros/foxy/setup.bash && vcs import < .rosinstall && rosdep install --rosdistro foxy --from-paths src --ignore-src -r -y && colcon build"

COPY robot-entrypoint.sh /home/robomaker/robot-entrypoint.sh
RUN sh -c 'sudo chmod +x /home/robomaker/robot-entrypoint.sh'
RUN sh -c 'sudo chown robomaker:robomaker /home/robomaker/robot-entrypoint.sh'

CMD ros2 launch hello_world_robot rotate.launch.py
ENTRYPOINT [ "/home/robomaker/robot-entrypoint.sh" ]
```

robot-entrypoint.sh로 저장한 스크립트의 콘텐츠는 다음과 같습니다. 이 스크립트는 로봇 애플리케이션에 대한 환경을 소싱합니다.

```
#!/bin/bash
cd /home/robomaker/workspace/aws-robomaker-sample-application-helloworld-ros2/robot_ws
source /opt/ros/foxy/setup.bash
source /usr/share/gazebo-11/setup.sh
source ./install/setup.sh
printenv

exec "${@:1}"
```

다음 명령은 Dockerfile에서 로봇 애플리케이션용 이미지를 생성합니다.

```
cd SampleGPURobotApp
docker build -t samplegpubaseapp:latest .
```

## 시뮬레이션 애플리케이션에 대한 이미지 생성

### 시뮬레이션 애플리케이션에 대한 이미지 생성

기본 이미지와 로봇 애플리케이션의 이미지를 만든 후 시뮬레이션 애플리케이션용 이미지를 만들 수 있습니다. 다음 스크립트를 SampleGPUSimulationApp 디렉터리의 Dockerfile에 저장하고 빌드합니다. 이 스크립트는 Hello World 시뮬레이션 애플리케이션을 다운로드하고 설정합니다.

```
# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: MIT-0
FROM samplepubbaseapp:latest

# Build the Simulation application
RUN cd /home/robomaker/workspace/aws-robomaker-sample-application-helloworld-ros2/
simulation_ws && \
/bin/bash -c "source /opt/ros/foxy/setup.bash && vcs import < .rosinstall && rosdep
install --rosdistro foxy --from-paths src --ignore-src -r -y && colcon build"

COPY simulation-entrypoint.sh /home/robomaker/simulation-entrypoint.sh

RUN sh -c 'sudo chmod +x /home/robomaker/simulation-entrypoint.sh'
RUN sh -c 'sudo chown robomaker:robomaker /home/robomaker/simulation-entrypoint.sh'

CMD ros2 launch hello_world_simulation empty_world.launch.py
ENTRYPOINT [ "/home/robomaker/simulation-entrypoint.sh" ]
```

simulation-entrypoint.sh로 저장한 스크립트의 콘텐츠는 다음과 같습니다. 이 스크립트는 시뮬레이션 애플리케이션의 환경을 소싱합니다.

```
#!/bin/bash
if [ ! -z $GAZEBO_MASTER_URI ]; then
    tmp_GAZEBO_MASTER_URI=$GAZEBO_MASTER_URI
fi

cd /home/robomaker/workspace/aws-robomaker-sample-application-helloworld-ros2/
simulation_ws
source /opt/ros/foxy/setup.bash
source /usr/share/gazebo-11/setup.sh

if [ ! -z $tmp_GAZEBO_MASTER_URI ]; then
    export GAZEBO_MASTER_URI=$tmp_GAZEBO_MASTER_URI
    unset tmp_GAZEBO_MASTER_URI
fi
```

```
source ./install/setup.sh
printenv

exec "${@:1}"
```

다음 명령은 이미지를 생성합니다.

```
cd SampleGPUSimulationApp
docker build -t samplegpusimulationapp:latest .
```

애플리케이션을 실행하고 Amazon ECR로 푸시하기

이미지를 생성한 후에는 로컬 Linux 환경에서 제대로 실행되는지 확인하세요. 이미지가 실행되는지 확인한 후, 도커 이미지를 Amazon ECR로 푸시하고 시뮬레이션 작업을 생성할 수 있습니다.

다음 명령을 사용하면 로컬 Linux 환경에서 Hello World 애플리케이션을 실행할 수 있습니다.

```
docker run -it -e DISPLAY -v /tmp/.X11-unix:/tmp/.X11-unix/ --name gpu_robot_app \
-u robomaker -e ROBOMAKER_GAZEBO_MASTER_URI=http://localhost:5555 \
-e ROBOMAKER_ROS_MASTER_URI=http://localhost:11311 \
samplegrobotapp:latest

docker run -it -e DISPLAY -v /tmp/.X11-unix:/tmp/.X11-unix/ --name gpu_sim_app \
-u robomaker -e ROBOMAKER_GAZEBO_MASTER_URI=http://localhost:5555 \
-e ROBOMAKER_ROS_MASTER_URI=http://localhost:11311 \
samplegpusimulationapp:latest
```

로봇 애플리케이션과 시뮬레이션 애플리케이션 컨테이너를 실행하면 Gazebo GUI 도구를 사용하여 시뮬레이션을 시각화할 수 있습니다. 다음 명령을 사용하십시오.

- 시뮬레이션 애플리케이션을 실행하는 컨테이너에 연결합니다.
- Gazebo 그래픽 사용자 인터페이스(GUI)를 실행하여 애플리케이션을 시각화하세요.

```
# Enable access to X server to launch Gazebo from docker container
$ xhost +

# Check that the robot_app and sim_app containers are running. The command should list
both containers
```

```
$ docker container ls

# Connect to the sim app container
$ docker exec -it gpu_sim_app bash

# Launch Gazebo from within the container
$ /home/robomaker/simulation-entrypoint.sh ros2 launch gazebo_ros gzclient.launch.py
```

이미지에 태그를 추가할 수 있습니다. 다음 명령을 사용하면 이미지에 태그를 지정할 수 있습니다.

```
docker tag samplegpurobotapp:latest accountID.dkr.ecr.us-west-2.amazonaws.com/
samplegpurobotapp:latest

docker tag samplepusimulationapp:latest accountID.dkr.ecr.us-west-2.amazonaws.com/
samplepusimulationapp:latest
```

애플리케이션이 제대로 작동하는지 확인한 후 다음 명령을 사용하여 Amazon ECR로 푸시할 수 있습니다.

```
aws ecr get-login-password --region us-west-2 | docker login --username AWS --password-
stdin accountID.dkr.ecr.us-west-2.amazonaws.com
docker push accountID.dkr.ecr.us-west-2.amazonaws.com/samplegpurobotapp:latest
docker push accountID.dkr.ecr.us-west-2.amazonaws.com/samplepusimulationapp:latest
```

이제 이러한 이미지를 사용하여 GPU 컴퓨팅으로 시뮬레이션 작업을 실행할 수 있습니다. 시뮬레이션 작업에 대한 자세한 내용은 [AWS RoboMaker를 이용한 시뮬레이션](#) 섹션을 참조하십시오.

# AWS RoboMaker를 이용한 시뮬레이션

AWS RoboMaker 시뮬레이션 작업은 클라우드에서 실행되는 로봇 애플리케이션과 시뮬레이션 애플리케이션을 페어링한 것입니다. 시뮬레이션 작업이 실행 중인 경우 그래픽 도구와 터미널로 상호 작용해 센서 데이터를 시각화하거나 로봇의 구성요소를 제어할 수 있습니다. 다음 주제에서는 AWS RoboMaker 시뮬레이션 작업을 실행, 구성, 관리, 기록 및 배치하는 방법을 설명합니다.

## 주제

- [시뮬레이션 실행](#)
- [시뮬레이션 구성](#)
- [시뮬레이션 관리](#)
- [시뮬레이션 로깅](#)
- [배치 시뮬레이션](#)

## 시뮬레이션 실행

시뮬레이션 실행을 시작하려면 다음 AWS CLI 명령을 사용하여 애플리케이션을 설명합니다. 이러한 명령은 시뮬레이션 작업 생성을 진행할 준비가 되었는지 확인하기 위해 검사할 수 있는 출력을 제공합니다.

다음 명령은 로봇 애플리케이션과 관련된 데이터를 검색합니다.

```
aws robomaker describe-robot-application --application YOUR-ROBOT-APP-ARN
```

describe-robot-application의 출력에는 다음 데이터가 포함됩니다.

```
{
  "arn": "YOUR-ROBOT-APP-ARN",
  "name": "YOUR-ROBOT-APP-NAME",
  ... # Removed extra data for clarity

  "robotSoftwareSuite": {
    "name": "General"
  },
}
```

```

... # Removed extra data for clarity

"environment": {
  "uri": "YOUR-ROBOT-APP-ECR-URI"
}
}

```

다음 명령은 시뮬레이션 애플리케이션과 관련된 데이터를 검색합니다.

```
aws robomaker describe-simulation-application --application YOUR-SIM-APP-ARN
```

describe-simulation-application의 출력에는 다음 데이터가 포함됩니다.

```

{
  "arn": "YOUR-SIM-APP-ARN",
  "name": "YOUR-SIM-APP-NAME",

  ... # Removed extra data for clarity

  "simulationSoftwareSuite": {
    "name": "SimulationRuntime"
  },
  "robotSoftwareSuite": {
    "name": "General"
  },

  ... # Removed extra data for clarity

  "environment": {
    "uri": "YOUR-SIM-APP-ECR-URI"
  }
}

```

YOUR-ROBOT-APP-ARN 및 YOUR-SIM-APP-ARN의 반환된 값을 저장합니다. 시뮬레이션 작업을 제출하려면 둘 다 필요합니다. WorldForge 에셋을 시뮬레이션 작업으로 가져와야 하는 경우 [DataSource](#) API를 사용합니다. 이렇게 하면 월드 내보내기 작업의 Amazon S3 출력 디렉터리에서 시뮬레이션 작업 컨테이너 내에서 선택한 대상으로 월드 자산을 가져올 수 있습니다. 자세한 내용은 [시뮬레이션에서 내보낸 월드 사용](#) 섹션을 참조하세요.

시뮬레이션 작업을 제출하려면 create\_simulation\_job.json 작업 디렉토리에 JSON 파일을 생성합니다. 빨간색 기울임꼴 텍스트로 식별되는 문자열을 복사, 붙여넣기 및 편집하여 YOUR-IAM-

ROLE-ARN, YOUR-ROBOT-APP-ARN, 및 YOUR-SIM-APP-ARN을 포함합니다. 아래 `roslaunch` 명령, TurtleBot 환경 변수 및 도구 설정은 [Hello World 샘플 애플리케이션](#)에만 해당됩니다. 시뮬레이션 작업의 필요에 따라 이러한 설정을 사용자 지정 값으로 업데이트해야 합니다. 자세한 정보는 [CreateSimulationJob](#) API를 참조하세요.

```
{
  "maxJobDurationInSeconds": 3600,
  "iamRole": "IAM-ROLE-ARN",
  "robotApplications": [
    {
      "application": "YOUR-ROBOT-APP-ARN",
      "applicationVersion": "$LATEST",
      "launchConfig": {
        "environmentVariables": {
          "ROS_IP": "ROBOMAKER_ROBOT_APP_IP",
          "ROS_MASTER_URI": "http://ROBOMAKER_ROBOT_APP_IP:11311",
          "GAZEBO_MASTER_URI": "http://ROBOMAKER_SIM_APP_IP:11345"
        },
        "streamUI": false,
        "command": [
          "/bin/bash", "-c", "roslaunch hello_world_robot rotate.launch"
        ]
      },
      "tools": [
        {
          "streamUI": true,
          "name": "robot-terminal",
          "command": "/entrypoint.sh && xfce4-terminal",
          "streamOutputToCloudWatch": true,
          "exitBehavior": "RESTART"
        }
      ]
    }
  ],
  "simulationApplications": [
    {
      "application": "YOUR-SIM-APP-ARN",
      "launchConfig": {
        "environmentVariables": {
          "ROS_IP": "ROBOMAKER_SIM_APP_IP",
          "ROS_MASTER_URI": "http://ROBOMAKER_ROBOT_APP_IP:11311",
          "GAZEBO_MASTER_URI": "http://ROBOMAKER_SIM_APP_IP:11345",
          "TURTLEBOT3_MODEL": "waffle_pi"
        }
      }
    }
  ]
}
```



- [Amazon VPC 액세스를 위한 시뮬레이션 작업 구성](#)
- [시뮬레이션 작업에 대한 인터넷 액세스](#)
- [SimulationJob 컴퓨팅 구성](#)
- [사용자 지정 시뮬레이션 도구 구성](#)
- [루트 액세스 및 시스템 기능](#)

## Amazon VPC 액세스를 위한 시뮬레이션 작업 구성

Amazon Virtual Private Cloud(VPC)에서 리소스를 생성한 경우 공용 인터넷을 통해 리소스를 읽을 수 없습니다. 리소스의 예로는 Amazon Redshift 데이터 웨어하우스 또는 Amazon ElastiCache 클러스터가 있습니다. Amazon Elastic Compute Cloud 인스턴스에 대한 서비스도 해당될 수 있습니다. 기본적으로 Amazon VPC의 리소스는 AWS RoboMaker 시뮬레이션 작업에서 액세스할 수 없습니다.

### Note

AWS RoboMaker는 외부 연결이 없는 격리된 네트워크에서 시뮬레이션 작업을 실행합니다. 하지만 작업에서 Amazon VPC의 리소스에 액세스하도록 허용하려면 Amazon VPC 서브넷 ID 및 보안 그룹 ID가 포함된 VPC 관련 데이터를 제공해야 합니다. AWS RoboMaker에서는 이 데이터를 사용하여 [ENI\(탄력적 네트워크 인터페이스\)](#)를 설정합니다. ENI를 사용하면 작업을 비공개 Amazon VPC의 다른 리소스에 안전하게 연결할 수 있습니다.

AWS RoboMaker는 전용 테넌시 VPC 내 리소스에 연결하지 않습니다. 자세한 내용은 [전용 VPC](#)를 참조하십시오.

작업을 생성할 때 VpcConfig 파라미터를 사용하여 AWS RoboMaker 시뮬레이션 작업에 Amazon VPC 데이터를 추가합니다([CreateSimulationJob](#) 참조). 다음은 퍼블릭 IP가 할당된 AWS CLI 예시입니다.

```
aws robomaker create-simulation-job \
--output-location s3Bucket=my-bucket,s3Prefix=my-output-folder \
--max-job-duration-in-seconds 3600 \
--iam-role my-role-arn \
--failure-behavior Continue \
--robot-applications application='my-robot-application-
arn,launchConfig={command=["roslaunch", "hello_world_robot", "rotate.launch"]} \
```

```
--simulation-applications application='my-simulation-application-arn,launchConfig={command=["roslaunch", "hello_world_simulation", "empty_world.launch"]}' \
--vpc-config assignPublicIp=true,subnets=comma-separated-vpc-subnet-ids,securityGroups=comma-separated-security-group-ids
```

### Note

시뮬레이션 작업이 VPC 내에서 실행되도록 구성되면 ENI 페널티가 발생합니다. 네트워크 리소스에 연결하려고 하면 주소 확인이 지연될 수 있습니다.

## 시뮬레이션 작업에 대한 인터넷 액세스

AWS RoboMaker는 사용자가 제공한 VPC 데이터를 사용하여 ENI를 설정합니다. ENI를 사용하면 작업 중에 VPC 리소스에 액세스할 수 있습니다. 각 ENI에는 지정한 서브넷 내의 범위에서 프라이빗 IP 주소가 할당됩니다. 퍼블릭 IP 주소는 기본적으로 ENI에 할당되지 않습니다.

작업 중에 인터넷에 액세스해야 하는 경우(VPC 엔드포인트가 없는 AWS 서비스를 찾기 위해) 비공개 서브넷을 사용 하는 경우 VPC 내부에 NAT를 설정할 수 있습니다. Amazon VPC NAT 게이트웨이를 사용하여 AWS RoboMaker에 퍼블릭 IP를 할당하도록 요청할 수 있습니다. 자세한 정보는 Amazon VPC 사용 설명서의 [NAT 게이트웨이](#) 섹션을 참조하세요.

### Note

인터넷 접속은 퍼블릭 IP 주소를 필요로 하므로 VPC에 직접 연결된 인터넷 게이트웨이는 사용할 수 없습니다. 기본적으로 ENI에는 프라이빗 IP 주소가 있습니다.

퍼블릭 서브넷을 사용하는 경우 인터넷 액세스를 구성하려면 ENI에 퍼블릭 IP를 할당하도록 `assignPublicIp=true`를 설정합니다.

시뮬레이션 작업에 공개 AWS API에 대한 액세스만 필요하고 개인 정보 보호를 강화하려면 [AWS RoboMaker 및 인터페이스 VPC 엔드포인트\(AWS PrivateLink\)](#) 항목을 참조하세요. 이 정보를 사용하여 인터페이스 VPC 엔드포인트를 생성하고 [CreateSimulationJob](#) API를 사용하여 VPC를 추가할 수 있습니다.

## SimulationJob 컴퓨팅 구성

SimulationJobs에서 GPU를 사용하기 위해 GPU 컴퓨팅을 사용하도록 SimulationJob의 ComputeType을 구성할 수 있습니다. AWS RoboMaker에서 그래픽 처리 장치(GPU) 기반 시뮬레이션 작업을 사용하면 다음과 같은 이점을 얻을 수 있습니다.

- GPU 기반 시뮬레이션 작업을 통해 GPU 지원 센서 플러그인과 OpenGL, CUDA, OpenCL 및 Vulkan을 사용한 높은 충실도의 렌더링 및 성능이 필요한 애플리케이션을 실행할 수 있습니다.
- GPU 기반 시뮬레이션 작업은 AWS RoboMaker GUI 도구의 고품질 HD 해상도를 지원하므로 물체를 더 자세히 볼 수 있습니다. GPU가 더 높은 초당 프레임 속도를 보장하므로 GUI 도구 경험이 이상적입니다.
- GPU 기반 시뮬레이션은 시뮬레이션 작업 완료 시간을 단축합니다. GPU를 사용하면 실시간 요소 및 초당 프레임 수에 대한 성능 저하 없이 복잡한 시뮬레이션 장면을 실행할 수 있습니다.
- GPU 기반 시뮬레이션 작업은 강화 학습 모델의 훈련을 개선합니다.

### 컴퓨팅

CreateSimulationJob 요청의 Compute 파라미터를 사용하여 SimulationJob에 필요한 컴퓨팅 종류를 구성할 수 있습니다.

### ComputeType

ComputeType은 작업에 필요한 컴퓨팅 유형을 지정합니다. 유효한 값은 CPU 및 GPU\_AND\_CPU입니다. 기본값은 CPU입니다. GPU\_AND\_CPU이 지정되어 있으면, 생성된 작업에서 CPU와 함께 GPU를 사용할 수 있습니다.

### GpuUnitLimit

GpuUnitLimit 파라미터를 사용하여 작업에 할당해야 하는 GPU 유닛 수를 지정할 수 있습니다. GPU\_AND\_CPU ComputeType의 경우 1이어야 합니다. CPU ComputeType의 경우 0이어야 합니다.

GPU를 활용하기 위한 컨테이너 구축에 대한 자세한 내용은 [GPU 애플리케이션을 실행하기 위한 이미지 생성](#) 항목을 참조하세요.

## 사용자 지정 시뮬레이션 도구 구성

AWS RoboMaker를 사용하여 시뮬레이션 작업의 애플리케이션을 위한 사용자 지정 도구를 구성할 수 있습니다. 사용자 지정 도구를 사용하여 시뮬레이션과 상호 작용하거나, 진단 유틸리티로 사용하거나

다른 용도로 사용할 수 있습니다. AWS RoboMaker에서 제공하는 `rqt` 또는 `rviz` 같은 기본 도구를 구성할 수도 있습니다. 시뮬레이션 작업이 자동화된 파이프라인에 포함된 경우, 기본 도구를 비활성화하고 리소스를 더 적게 사용할 수 있습니다.

최대 10개의 사용자 지정 도구를 구성할 수 있습니다. 사용자 지정 도구는 기본 ROS 프로세스가 시작된 후에 시작됩니다.

사용자 지정 도구 구성에는 다음 요소가 포함됩니다.

- 도구 이름 — 도구의 이름입니다.
- 명령 — `bash` 셸에서 도구를 호출하는 명령입니다. 도구 실행 파일 이름을 포함해야 합니다. 인수에 사용자 지정 변수를 비롯한 환경 변수를 사용할 수 있습니다. 예를 들어, 현재 시뮬레이션 작업 ID를 사용하기 위해 `AWS_ROBOMAKER_SIMULATION_JOB_ID`를 참조할 수 있습니다.
- 종료 동작 — 사용자 지정 도구가 종료될 경우 수행할 작업을 결정합니다. `fail`을 지정하면 시뮬레이션 작업이 실패합니다. `restart`을 지정하면 도구가 다시 시작됩니다. 기본값은 `restart`입니다.
- UI 스트리밍 - 스트리밍 세션이 도구에 구성되어 있는지 여부를 지정합니다. `True`인 경우 AWS RoboMaker가 연결을 구성하여 시뮬레이션에서 실행 중인 도구와 상호 작용할 수 있습니다. 그래픽 사용자 인터페이스가 있어야 합니다. 기본값은 `false`입니다.
- 로그 동작 - 도구 `stdout` 및 `stderr`를 CloudWatch Logs로 스트리밍할지 여부를 지정합니다. 기본값은 `false`입니다.

## 루트 액세스 및 시스템 기능

AWS RoboMaker에서는 시뮬레이션 작업에서 실행되는 애플리케이션에 대해 제한된 루트(`sudo`) 액세스를 제공합니다. 다음 목록에는 차단된 중요한 시스템 호출(전체가 아닌 일부 항목)이 포함되어 있습니다.

- `acct`
- `add_key`
- `bpf`
- `clock_adjtime`
- `clock_settime`
- 복제
- `create_module`
- `delete_module`
- `finit_module`
- `get_kernel_syms`

- get\_mempolicy
- init\_module
- IOPERM
- iopl
- kcmp
- kexec\_file\_load
- kexec\_load
- keyctl
- lookup\_dcookie
- mbind
- 마운트에 의해
- move\_pages
- name\_to\_handle\_at
- nfsservctl
- open\_by\_handle\_at
- perf\_event\_open
- personality
- pivot\_root
- process\_vm\_readv
- process\_vm\_writev
- ptrace
- query\_module
- quotactl
- reboot
- request\_key
- set\_mempolicy
- setns
- settimeofday
- stime
- swapon
- swapoff
- sysfs
- \_sysctl

- umount
- umount2
- unshare
- uselib
- userfaultfd
- ustat
- vm86
- vm86old

## 시뮬레이션 관리

다음 섹션에서는 시뮬레이션 작업을 생성하고 확인하고 취소하고 복제하고 다시 시작하는 방법에 대해 설명합니다.

### Sections

- [시뮬레이션 작업 생성](#)
- [시뮬레이션 작업 보기](#)
- [시뮬레이션 작업 취소](#)
- [시뮬레이션 작업 복제](#)
- [시뮬레이션 작업 다시 시작](#)

## 시뮬레이션 작업 생성

가상 세계에서 로봇 애플리케이션을 실행하려는 경우 시뮬레이션 플랫폼을 사용하여 시뮬레이션 작업을 생성합니다. 시뮬레이션 애플리케이션을 지정할 때 소프트웨어 제품군 이름을 선택하세요. 현재 General 및 SimulationRuntime 소프트웨어 제품군을 지원합니다.

시뮬레이션 작업을 생성하려면

다음 탭 중 하나에 있는 단계를 따릅니다.

### Using the console

1. AWS RoboMaker 콘솔(<https://console.aws.amazon.com/robomaker/>)에 로그인합니다.

2. 왼쪽 탐색 창에서 시뮬레이션 실행을 선택하고 시뮬레이션 작업을 선택합니다.
3. 시뮬레이션 작업 생성을 선택합니다.
4. Simulation configuration(시뮬레이션 구성) 페이지에서 시뮬레이션 작업 기간을 선택합니다. 5분~14일 범위에서 값을 선택합니다.

**⚠ Important**

AWS RoboMaker에 대한 요금 청구 방식에 대해 자세히 알아보려면 [AWS RoboMaker 요금](#)을 참조하세요.

5. 오류 시 동작을 선택합니다. 실패를 선택하면 시뮬레이션 작업에 실패한 경우 호스트 인스턴스를 종료합니다. 계속을 선택하면 호스트 인스턴스가 유지되어 연결 및 조사할 수 있습니다.

다음 단계에서 선택 사항인 S3 폴더를 지정하면 시뮬레이션 데이터가 포함됩니다. 이 폴더는 선택한 오류 시 동작과 상관 없이 사용할 수 있습니다.

6. IAM 역할에 대해 역할을 선택하거나 새 역할 생성을 선택하여 새로 생성합니다. AWS RoboMaker에서는 이 역할을 사용하여 사용자를 대신해 리소스에 액세스합니다. 이 역할은 애플리케이션에서 Amazon Rekognition 또는 Amazon Lex 등 AWS 리소스에 액세스하는 데에도 사용됩니다.
7. 선택 사항: 컴퓨팅에서 시뮬레이션 단위 제한을 선택합니다. 시뮬레이션에 제공된 시뮬레이션 단위 제한에 비례하여 CPU와 메모리가 할당됩니다. 시뮬레이션 단위는 1 vcpu 및 2GB의 메모리입니다. 기본값은 15입니다.
8. 선택 사항: 출력 대상에서 시뮬레이션 작업 출력을 저장할 Amazon S3 폴더 이름을 입력합니다. 경우에 따라 새 S3 폴더 생성을 선택해 새 Amazon S3 폴더를 생성합니다.
9. 선택 사항: 네트워킹에서 로봇 애플리케이션 또는 시뮬레이션 애플리케이션이 Amazon VPC의 리소스에 액세스하는 경우 VPC, 서브넷 및 보안 그룹을 선택합니다. 사용 가능한 모든 서브넷을 선택하여 모든 리소스 제한을 사용할 수 있는지 확인합니다. 자세한 내용은 [VPC 및 서브넷](#)을 참조하세요.

VPC 외부에서 시뮬레이션 작업에 액세스하려면 퍼블릭 IP 할당을 선택합니다.

10. 필요한 경우 태그에서, 시뮬레이션 작업을 위한 태그를 한 개 또는 여러 개 지정합니다. 태그는 AWS 리소스를 식별 및 구성하기 위한 메타데이터로 작동하는 단어나 구문입니다. 각 태그는 키와 값으로 구성됩니다. 시뮬레이션 작업 세부 정보 페이지에서 시뮬레이션 작업의 태그를 관리할 수 있습니다.

태그에 대한 자세한 내용은 AWS 과금 정보 및 비용 관리 사용 설명서의 [비용 할당 태그 사용](#)을 참조하세요.

11. 다음을 선택합니다.
12. Specify robot application(로봇 애플리케이션 지정) 페이지의 로봇 애플리케이션에서 새 애플리케이션 생성을 선택합니다. 경우에 따라 기존 애플리케이션 선택을 선택하여 이미 생성한 로봇 애플리케이션을 사용할 수 있습니다.
13. 로봇 애플리케이션의 이름을 입력합니다.
14. 컨테이너 이미지에서 로봇 애플리케이션 컨테이너의 Amazon ECR 리포지토리 위치를 지정합니다. 자세한 내용은 [AWS RoboMaker 호환 컨테이너에 대한 요구 사항](#) 섹션을 참조하세요.

 Note

\$LATEST를 사용한다고 해서 Amazon ECR이 변경되는 것을 방지할 수는 없습니다. AWS RoboMaker이 리포지토리에 액세스할 때 리포지토리가 읽기 전용으로 설정됩니다.

버전 관리에 대한 자세한 내용은 [애플리케이션 버전 지정](#) 섹션을 참조하십시오.

15. 로봇 애플리케이션 구성에서 로봇 애플리케이션에 대한 시작 명령을 입력합니다.
16. 선택 사항: 로봇 애플리케이션 도구를 구성하려면 로봇 애플리케이션 도구를 확장합니다. 사전 구성된 도구를 사용하려면 기본 도구 사용을 선택합니다. 도구 사용자 지정을 선택하여 애플리케이션에 사용할 사용자 지정 도구를 추가, 제거 또는 편집합니다.

새 사용자 지정 도구를 추가하려면:

- a. 도구 추가를 선택합니다.
  - b. 애플리케이션 도구 추가에서 도구 이름을 지정합니다.
  - c. 도구의 명령줄 인수를 지정합니다. 도구 실행 파일 이름을 포함해야 합니다.
  - d. 동작 종료를 선택합니다. 실패를 선택하는 경우 도구가 종료되면 시뮬레이션 작업이 실패합니다. 도구를 다시 시작하려면 다시 시작을 선택합니다. 기본값은 다시 시작입니다.
  - e. UI 스트리밍 활성화 또는 비활성화를 선택합니다. UI 스트리밍은 기본적으로 비활성화되어 있습니다.
  - f. CloudWatch로 출력 보내기를 선택하여 도구에 대한 로그를 기록합니다. 로그는 CloudWatch에서 확인할 수 있습니다. 기본적으로 출력은 CloudWatch로 전송되지 않습니다. 사용자 지정 도구는 기본 ROS 시작 프로세스가 시작된 후에만 시작됩니다.
17. 선택 사항: 애플리케이션에 그래픽 사용자 인터페이스가 포함되어 있는 경우 스트리밍 세션으로 실행을 선택합니다. AWS RoboMaker은 시뮬레이션에서 실행되는 동안 애플리케이션과 상

호 작동할 수 있도록 연결을 구성합니다. 시뮬레이션 작업 세부 정보 페이지의 시뮬레이션 도구에서 로봇 애플리케이션을 선택하여 연결할 수 있습니다.

18. 선택 사항: 로봇 애플리케이션에서 환경 변수를 사용하는 경우 이름 및 값 페어를 지정합니다. 환경 변수 이름은 A-Z 또는 밑줄로 시작해야 하며 A-Z, 0-9 및 밑줄로 구성되어야 합니다. AWS 로 시작하는 이름은 예약되어 있습니다.

환경 변수 추가를 선택하여 변수를 추가합니다.

roslaunch [substituion args](#)를 사용하여 launch 파일에서 환경 변수를 읽을 수 있습니다.

19. 선택 사항: 트래픽 전달을 시뮬레이션 작업 포트에서 애플리케이션 포트로 구성합니다. 로봇 및 시뮬레이션 애플리케이션을 위한 포트 매핑을 지정하려면 시뮬레이션 작업 네트워킹을 구성해야 합니다.
20. 선택 사항: 하나 이상의 로봇 애플리케이션 업로드 구성을 지정합니다. 업로드 구성을 지정하려면 시뮬레이션 작업 출력 대상을 구성해야 합니다. 각 구성은 업로드 동작, Unix 글로브 파일 일치 규칙, 일치하는 파일을 배치할 위치를 지정합니다. 사용자 지정 업로드에 대한 자세한 내용은 [사용자 지정 업로드 구성 추가](#) 섹션을 참조하세요.
21. 다음을 선택합니다.
22. Specify simulation application(시뮬레이션 애플리케이션 지정) 페이지에서 새 애플리케이션 생성을 선택합니다. 경우에 따라 기존 애플리케이션 선택을 선택하여 이미 생성한 시뮬레이션 애플리케이션을 사용할 수 있습니다.
23. 시뮬레이션 애플리케이션의 이름을 입력합니다.
24. 컨테이너 이미지에서 로봇 애플리케이션 컨테이너의 Amazon ECR 리포지토리 위치를 지정합니다. 자세한 내용은 [???](#) 섹션을 참조하세요. \$LATEST를 사용한다고 해서 Amazon ECR이 변경되는 것을 방지할 수는 없습니다. AWS RoboMaker이 리포지토리에 액세스할 때 리포지토리가 읽기 전용으로 설정됩니다.

버전 관리에 대한 자세한 내용은 [애플리케이션 버전 지정](#) 섹션을 참조하십시오.

25. 시뮬레이션 애플리케이션 구성에서 로봇 애플리케이션에 대한 시작 명령을 입력합니다.
26. 선택 사항: 로봇 애플리케이션 도구를 구성하려면 시뮬레이션 애플리케이션 도구를 확장합니다. 사전 구성된 도구를 사용하려면 기본 도구 사용을 선택합니다. 도구 사용자 지정을 선택하여 애플리케이션에 사용할 사용자 지정 도구를 추가, 제거 또는 편집합니다.

새 사용자 지정 도구를 추가하려면:

- a. 도구 추가를 선택합니다.
- b. 애플리케이션 도구 추가에서 도구 이름을 지정합니다.

- c. 도구의 명령줄 인수를 지정합니다. 도구 실행 파일 이름을 포함해야 합니다.
- d. 동작 종료를 선택합니다. 실패를 선택하는 경우 도구가 종료되면 시뮬레이션 작업이 실패합니다. 도구를 다시 시작하려면 다시 시작을 선택합니다. 기본값은 다시 시작입니다.
- e. UI 스트리밍 활성화 또는 비활성화를 선택합니다. UI 스트리밍은 기본적으로 비활성화되어 있습니다.
- f. CloudWatch로 출력 보내기를 선택하여 도구에 대한 로그를 기록합니다. 로그는 CloudWatch에서 확인할 수 있습니다. 기본적으로 출력은 CloudWatch로 전송되지 않습니다.

사용자 지정 도구는 기본 시작 프로세스가 시작된 후에만 시작됩니다.

- 27. 선택 사항: 애플리케이션에 그래픽 사용자 인터페이스가 포함되어 있는 경우 스트리밍 세션으로 실행을 선택합니다. AWS RoboMaker은 시뮬레이션에서 실행되는 동안 애플리케이션과 상호 작용할 수 있도록 연결을 구성합니다. 시뮬레이션 작업 세부 정보 페이지의 시뮬레이션 도구에서 시뮬레이션 애플리케이션을 선택하여 연결할 수 있습니다.
- 28. 선택 사항: 시뮬레이션 애플리케이션에서 환경 변수를 사용하는 경우 이름 및 값 페어를 지정합니다. 환경 변수 추가를 선택하여 변수를 추가합니다.
- 29. 선택 사항: 트래픽 전달을 시뮬레이션 작업 포트에서 애플리케이션 포트로 구성합니다. 로봇 및 시뮬레이션 애플리케이션을 위한 포트 매핑을 지정하려면 시뮬레이션 작업 네트워킹을 구성해야 합니다.
- 30. 선택 사항: 하나 이상의 시뮬레이션 애플리케이션 업로드 구성을 지정합니다. 업로드 구성을 지정하려면 시뮬레이션 작업 출력 대상을 구성해야 합니다. 각 구성은 업로드 동작, Unix 글로벌 파일 일치 규칙, 일치하는 파일을 배치할 위치를 지정합니다.

기본 업로드 구성은 이전 시뮬레이션 작업 출력 구성과의 호환성을 유지합니다. 기본 구성은 생성한 추가 업로드 구성에 추가됩니다. 사용자 지정 업로드에 대한 자세한 내용은 [사용자 지정 업로드 구성 추가](#) 섹션을 참조하세요.

- 31. 다음을 선택합니다.
- 32. 생성을 선택하여 시뮬레이션 작업을 생성합니다.

## Using the AWS CLI

## Example

다음은 다른 쪽 탭의 콘솔 기반 시뮬레이션 작업 생성과 동일한 역할을 수행하는 AWS CLI 명령의 예입니다.

```
aws robomaker create-simulation-job --max-job-duration-in-seconds 3600
  --iam-role arn:aws:iam::111111111111:role/MyRole --robot-applications
  application=arn:aws:robomaker:us-west-2:111111111111:robot-application/
  MyRobotApplication/1551203485821,launchConfig="{command=["roslaunch",
  "hello_world_robot", "rotate.launch"]}" --simulation-applications
  application=arn:aws:robomaker:us-west-2:111111111111:simulation-application/
  MySimulationApplication/1551203427605,launchConfig="{command=["roslaunch",
  "hello_world_simulation", "empty_world.launch"]}" --tags Region=North
```

## 시뮬레이션 작업 보기

시뮬레이션 작업에 대한 정보를 볼 수 있고 작업이 실행 중인 경우에는 그래픽 도구 및 터미널을 실행하여 시뮬레이션과 상호 작용할 수 있습니다. 시뮬레이션 작업에 대한 세부 정보를 보고 태그를 관리할 수도 있습니다.

시뮬레이션 작업을 보려면

다음 탭 중 하나에 있는 단계를 따릅니다.

Using the console

1. AWS RoboMaker 콘솔(<https://console.aws.amazon.com/robomaker/>)에 로그인합니다.
2. 왼쪽 탐색 창에서 시뮬레이션을 선택하고 시뮬레이션 작업을 선택합니다.
3. 시뮬레이션 작업의 Id를 선택하여 생성된 시간, 로봇 애플리케이션 및 시뮬레이션 애플리케이션의 시작 명령을 포함한 세부 정보를 봅니다.

Using the AWS CLI

## Example

다음은 다른 쪽 탭의 콘솔 기반 시뮬레이션 작업 보기와 동일한 역할을 수행하는 AWS CLI 명령의 예입니다.

```
aws robomaker list-simulation-jobs
aws robomaker describe-simulation-job --job my-simulation-job-arn
```

## 시뮬레이션 작업 취소

실행 중인데 더 이상 필요하지 않은 경우 시뮬레이션 작업을 취소할 수 있습니다.

시뮬레이션 작업을 취소하려면

다음 탭 중 하나에 있는 단계를 따릅니다.

### Using the console

1. AWS RoboMaker 콘솔(<https://console.aws.amazon.com/robomaker/>)에 로그인합니다.
2. 왼쪽 탐색 창에서 시뮬레이션을 선택하고 시뮬레이션 작업을 선택합니다.
3. 취소하려는 시뮬레이션 작업의 Id를 선택합니다.
4. 시뮬레이션 작업 세부 정보 페이지의 작업에서 취소를 선택합니다.
5. 시뮬레이션 작업 취소 페이지에서 예, 취소를 선택합니다.

### Using the AWS CLI

#### Example

다음은 다른 쪽 탭의 콘솔 기반 시뮬레이션 작업 취소와 동일한 역할을 수행하는 AWS CLI 명령의 예입니다.

```
aws robomaker list-simulation-jobs
aws robomaker cancel-simulation-job --job my-simulation-job-arn
```

시뮬레이션 작업 내에서 취소하는 방법에 대한 자세한 내용은 [ROS 컨테이너 FAQ](#)를 참조하세요.

## 시뮬레이션 작업 복제

AWS Management Console의 시뮬레이션 작업 세부 정보 페이지에서 기존 시뮬레이션 작업을 복제하여 기존 시뮬레이션 작업에서 새 시뮬레이션 작업을 생성할 수 있습니다.

**Note**

ROS 및 Gazebo 소프트웨어 제품군을 사용한 시뮬레이션 작업은 복제할 수 없습니다. 자세한 내용은 [지원 정책](#) 섹션을 참조하세요.

1. AWS RoboMaker 콘솔(<https://console.aws.amazon.com/robomaker/>)에 로그인합니다.
2. 왼쪽 탐색 창에서 시뮬레이션을 선택하고 시뮬레이션 작업을 선택합니다.
3. 다시 시작하려는 실행 중인 시뮬레이션 작업의 Id를 선택합니다.
4. 시뮬레이션 작업 세부 정보 페이지의 작업에서 복제를 선택합니다.
5. 시뮬레이션 작업 검토 및 생성에서 편집을 선택하여 변경합니다.
6. 생성을 선택하여 시뮬레이션 작업을 생성합니다.

## 시뮬레이션 작업 다시 시작

실행 중인 시뮬레이션 작업을 다시 시작할 수 있습니다. 다시 시작하면 시뮬레이션 작업은 Amazon S3 위치에 있는 로봇 애플리케이션 및 시뮬레이션 애플리케이션 소스 파일을 사용하고, 시뮬레이션 작업을 생성했을 때 지정한 기타 구성 설정을 모두 사용합니다.

시뮬레이션 작업을 다시 시작하려면

다음 탭 중 하나에 있는 단계를 따릅니다.

### Using the console

1. AWS RoboMaker 콘솔(<https://console.aws.amazon.com/robomaker/>)에 로그인합니다.
2. 왼쪽 탐색 창에서 시뮬레이션을 선택하고 시뮬레이션 작업을 선택합니다.
3. 다시 시작하려는 실행 중인 시뮬레이션 작업의 Id를 선택합니다.
4. 시뮬레이션 작업 세부 정보 페이지의 작업에서 다시 시작을 선택합니다.
5. Restart simulation job(시뮬레이션 작업 다시 시작) 페이지에서 Yes, restart(예, 재시작)를 선택합니다.

### Using the AWS CLI

## Example

다음은 다른 쪽 탭의 콘솔 기반 시뮬레이션 작업 다시 시작과 동일한 역할을 수행하는 AWS CLI 명령의 예입니다. 시뮬레이션 작업이 실행 중이어야 합니다.

```
aws robomaker restart-simulation-job --job my-simulation-job-arn
```

## 시뮬레이션 로깅

시뮬레이션 작업에서 출력 파일 및 기타 아티팩트를 캡처하기 위해 사용자 지정 업로드를 구성할 수 있습니다. 로봇 애플리케이션과 시뮬레이션 애플리케이션에 대한 사용자 지정 업로드를 구성할 수 있습니다. 사용자 지정 업로드를 구성하면 지정한 파일이 시뮬레이션 작업에서 사용자가 제공한 Amazon S3 시뮬레이션 출력 위치로 업로드됩니다. 이는 시뮬레이션 실행 중에 생성된 애플리케이션 출력을 검토 또는 분석하거나 아티팩트를 재사용하려는 경우에 유용할 수 있습니다.

사용자 지정 업로드를 구성하려면 먼저 시뮬레이션 작업을 위한 Amazon S3 출력 대상을 제공해야 합니다. AWS RoboMaker은 사용자가 지정한 이름을 사용하여 일치하는 파일을 폴더에 업로드합니다. 일치하는 파일은 모든 시뮬레이션 작업 도구가 종료될 때 업로드되거나 생성된 후 제거되면 업로드될 수 있습니다.

기본 업로드 구성은 비활성화되지 않는 한 사용자 지정 업로드 구성에 자동으로 추가됩니다. 기본 업로드 구성은 ROS 및 Gazebo 기본 로깅 출력을 업로드합니다. 이렇게 하면 ROS 및 Gazebo 기본 로깅 출력을 업로드한 이전 시뮬레이션 작업 출력 구성과의 호환성이 유지됩니다. 콘솔에서 시뮬레이션 작업을 구성할 때 기본 업로드 구성을 비활성화할 수 있습니다. [CreateSimulationJob](#) API에서 `useDefaultUploadConfigurations`를 `false`로 설정하여 비활성화할 수도 있습니다.

시뮬레이션 애플리케이션은 단일 128GB 파티션에 추가되며 사용자는 파티션에 대한 쓰기 권한을 가집니다.

### 섹션

- [사용자 지정 업로드 구성 추가](#)
- [AWS RoboMaker에 의해 생성된 환경 변수](#)

## 사용자 지정 업로드 구성 추가

사용자 지정 업로드 구성을 생성하려면 Amazon S3에서 파일이 업로드되는 위치를 지정하는 이름 접두사, 업로드할 파일을 지정하는 Unix 글로브 경로, 파일 업로드 시기를 지정하는 업로드 동작을 지정해야 합니다.

### 이름

이름은 Amazon S3에 파일이 업로드되는 방식을 지정하는 접두사입니다. 최종 경로를 결정하기 위해 시뮬레이션 출력 위치에 추가됩니다.

예를 들어 시뮬레이션 출력 위치가 `s3://my-bucket`이고 업로드 구성 이름이 `robot-test`인 경우 파일이 `s3://my-bucket/<simid>/<runid>/robot-test`에 업로드됩니다.

### 경로

경로는 업로드할 파일을 지정합니다. 표준 Unix 글로브 일치 규칙은 다음 조건에 따라 허용됩니다.

- 경로는 `/home/robomaker/` 또는 `/var/log`로 시작해야 합니다.
- 경로에는 역방향 경로 표현식(`../`)이 포함되어서는 안 됩니다.
- 심볼릭 링크는 따르지 않습니다.
- 경로에서 `**`을 슈퍼 별표로 사용할 수 있습니다. 예를 들어 `/var/log/**/*.log`를 지정하면 `/var/log` 디렉터리 트리의 모든 `.log` 파일이 수집됩니다.

표준 별표를 표준 와일드카드도 사용할 수도 있습니다. 예를 들어, `/var/log/system.log*`는 `/var/log`에서 `system.log_1111`, `system.log_2222` 등의 파일과 일치합니다.

### 업로드 동작

다음 업로드 동작 중 한 가지를 선택할 수 있습니다.

- 종료 시 업로드(`UPLOAD_ON_TERMINATE`)는 시뮬레이션 작업이 종료 상태로 전환되면 경로와 일치하는 모든 파일을 업로드합니다. AWS RoboMaker은 최대 60분 동안 로그 업로드를 시도합니다.

AWS RoboMaker은 시뮬레이션에서 실행 중인 모든 도구가 중지될 때까지 파일 업로드를 시작하지 않습니다.

- 자동 제거를 통한 업로드 롤링(`UPLOAD_ROLLING_AUTO_REMOVE`)은 생성된 경로와 일치하는 모든 파일을 업로드합니다. 경로는 5초마다 확인됩니다. 파일이 업로드되면 소스 파일이 삭제됩니다. 파일이 삭제된 후 같은 이름의 새 파일이 생성되면 이전에 업로드한 파일을 대체합니다. AWS

RoboMaker은 시뮬레이션에서 실행 중인 모든 애플리케이션이 중지되면 파일에 대한 최종 확인을 수행합니다.

자동 제거를 통한 업로드 롤링은 롤링 로그를 업로드하는 데 유용합니다. 경로 글로브에 포함되지 않는 “활성” 파일에 출력을 쓰거나 스트리밍합니다. 활성 파일에 기록이 끝나면 파일을 경로 글로브에 포함된 위치로 롤링하여 업로드 및 제거합니다.

이 설정은 시뮬레이션 작업의 스페이스를 절약하는 데 도움이 될 수 있습니다. 시뮬레이션 작업을 종료하기 전에 파일에 액세스하는 데에도 도움이 될 수 있습니다.

시뮬레이션 작업 파티션 크기는 128gb입니다. 어떤 이유로든 시뮬레이션 작업이 종료되면 AWS RoboMaker은 사용자 지정 업로드 구성에 지정된 모든 파일의 업로드를 시도합니다.

## AWS RoboMaker에 의해 생성된 환경 변수

AWS RoboMaker은 다음 시뮬레이션 작업 환경 변수를 정의합니다.

- AWS\_ROBOMAKER\_SIMULATION\_JOB\_ID
- AWS\_ROBOMAKER\_SIMULATION\_JOB\_ARN
- AWS\_ROBOMAKER\_SIMULATION\_RUN\_ID

애플리케이션 또는 명령줄에서 이러한 변수를 가져올 수 있습니다. 예를 들어 Python에서 현재 시뮬레이션 작업 Amazon 리소스 이름(ARN)을 얻으려면 `os.environ.get("AWS_ROBOMAKER_SIMULATION_JOB_ARN")`을 사용하세요.

시뮬레이션 작업에 대한 Amazon Simple Storage Service 출력 버킷을 지정한 경우 환경 변수를 사용하여 출력 경로를 찾을 수 있습니다. AWS RoboMaker은 에 출력을 `s3://bucket-name/AWS_ROBOMAKER_SIMULATION_JOB_ID/AWS_ROBOMAKER_SIMULATION_RUN_ID`에 씁니다. 이를 사용하여 코드 또는 명령줄에서 Amazon S3의 객체를 관리합니다.

또한 AWS RoboMaker은 로봇 및 시뮬레이션 애플리케이션 컨테이너가 서로 통신할 수 있도록 `CreateSimulationJobRequest`에 설정된 특정 환경 변수를 처리합니다. 자세한 내용은 [ROS 컨테이너 FAQ](#) 섹션을 참조하세요.

## 배치 시뮬레이션

이 단원에서는 시뮬레이션 작업 배치를 시작하고 관리하는 방법에 대해 설명합니다. 시뮬레이션 작업 배치를 사용하면 단일 API 호출을 사용하여 많은 시뮬레이션을 시작하고 실행하여 회귀 테스트, 파라미터 최적화, 기계 학습 모델 교육 및 합성 데이터 생성을 수행할 수 있습니다.

### Note

시뮬레이션 작업 배치는 AWS RoboMaker SDK 또는 AWS CLI를 사용해서만 시작할 수 있습니다. AWS RoboMaker 콘솔을 사용하여 시뮬레이션 배치를 확인, 복제 및 취소할 수 있습니다.

### 섹션

- [시뮬레이션 작업 배치 시작](#)
- [시뮬레이션 작업 배치 보기](#)
- [시뮬레이션 작업 배치 취소](#)
- [시뮬레이션 작업 배치 복제](#)

## 시뮬레이션 작업 배치 시작

시뮬레이션 작업 배치는 AWS SDK 또는 AWS CLI에서 시작할 수 있습니다. 시뮬레이션 작업 배치에는 하나 이상의 시뮬레이션 작업 요청이 포함됩니다. 각 시뮬레이션 작업 요청은 각 시뮬레이션에서 사용할 애플리케이션, 작업의 최대 기간 및 기타 정보를 식별합니다. 시뮬레이션 작업 배치 및 각 시뮬레이션 작업 요청에 태그를 적용할 수 있습니다.

시뮬레이션 작업 배치를 시작하려면 다음을 수행해야 합니다.

1. AWS Command Line Interface를 설치합니다. AWS CLI 설치에 대한 자세한 내용은 [AWS CLI 설치](#)를 참조하십시오.
2. 다음 JSON을 `startsimjobbatch.json`이라는 파일에 복사합니다. 원하는 구성과 일치하도록 파일을 수정한 다음 저장합니다.

```
{
  "batchPolicy": {
    "timeoutInSeconds": 400,
    "maxConcurrency": 2
  }
}
```

```

    },
    "createSimulationJobRequests": [
      {
        "maxJobDurationInSeconds": 300,
        "iamRole": "arn:aws:iam::111111111111:role/MyRole",
        "failureBehavior": "Fail",
        "robotApplications": [
          {
            "application": "arn:aws:robomaker:us-east-1:111111111111:robot-
application/MyRobotApplicationArn",
            "launchConfig": {
              "packageName": "hello_world_robot",
              "launchFile": "rotate.launch"
            }
          }
        ],
        "simulationApplications": [
          {
            "application": "arn:aws:robomaker:us-
east-1:111111111111:simulation-applicationMySimulationApplicationArn",
            "launchConfig": {
              "command": [
                "roslaunch", "hello_world_robot", "rotate.launch"
              ]
            }
          }
        ],
        "tags": {
          "myRequestTagKey" : "myRequestTagValue"
        }
      },
      {
        "maxJobDurationInSeconds": 300,
        "iamRole": "arn:aws:iam::111111111111:role/MyRole",
        "failureBehavior": "Fail",
        "simulationApplications": [
          {
            "application": "arn:aws:robomaker:us-
east-1:111111111111:simulation-applicationMySimulationApplicationArn",
            "launchConfig": {
              "command": [
                "roslaunch", "hello_world_simulation",
                "empty_world.launch"
              ]
            }
          }
        ]
      }
    ]
  }
}

```

```

    }
  }
],
"tags": {
  "myBatchTagKey" : "myBatchTagValue"
}
}

```

3. 명령 프롬프트를 열고 다음 AWS CLI 명령을 실행합니다.

```

$ aws robomaker start-simulation-job-batch --cli-input-json
file://startsimjobbatch.json

```

시뮬레이션 작업 배치를 보려면 [시뮬레이션 작업 배치 보기](#) 단원을 참조하십시오.

## 시뮬레이션 작업 배치 보기

배치의 시뮬레이션 작업 요청에 대한 세부 정보를 포함하여 시뮬레이션 작업 배치에 대한 정보를 확인할 수 있습니다.

시뮬레이션 작업 배치의 세부 정보를 확인하려면

다음 탭 중 하나에 있는 단계를 따릅니다.

### Using the console

1. AWS RoboMaker 콘솔(<https://console.aws.amazon.com/robomaker/>)에 로그인합니다.
2. 왼쪽 탐색 창에서 시뮬레이션을 선택하고 시뮬레이션 작업 배치를 선택합니다.
3. 시뮬레이션 작업 배치의 Id를 선택하여 세부 정보를 확인합니다.

### Using the AWS CLI

#### Example

다음은 다른 쪽 탭의 콘솔 기반 시뮬레이션 작업 보기와 동일한 역할을 수행하는 AWS CLI 명령의 예입니다.

```
aws robomaker list-simulation-job-batches
aws robomaker describe-simulation-job-batch --job my-simulation-job-batch-arn
```

## 시뮬레이션 작업 배치 취소

실행 중인데 더 이상 필요하지 않은 경우 시뮬레이션 작업을 취소할 수 있습니다.

시뮬레이션 작업을 취소하려면

다음 탭 중 하나에 있는 단계를 따릅니다.

### Using the console

1. AWS RoboMaker 콘솔(<https://console.aws.amazon.com/robomaker/>)에 로그인합니다.
2. 왼쪽 탐색 창에서 시뮬레이션을 선택하고 시뮬레이션 작업 배치를 선택합니다.
3. 취소하려는 시뮬레이션 작업 배치의 Id를 선택합니다.
4. 시뮬레이션 작업 배치 세부 정보 페이지의 배치 작업에서 배치 취소를 선택합니다.
5. 시뮬레이션 작업 배치 취소 페이지에서 취소를 선택합니다.

### Using the AWS CLI

#### Example

다음은 다른 쪽 탭의 콘솔 기반 시뮬레이션 작업 취소와 동일한 역할을 수행하는 AWS CLI 명령의 예입니다.

```
$ aws robomaker list-simulation-job-batches
$ aws robomaker cancel-simulation-job-batch --job my-simulation-job-batch-arn
```

## 시뮬레이션 작업 배치 복제

기존 배치를 복제하여 새 시뮬레이션 작업 배치를 시작할 수 있습니다. 복제할 때 시뮬레이션 작업 요청을 모두 포함하거나 요청의 하위 세트를 선택할 수 있습니다.

**Note**

ROS 및 Gazebo 소프트웨어 제품군을 사용한 시뮬레이션 작업 배치는 복제에 사용할 수 없습니다. 자세한 내용은 [지원 정책](#) 섹션을 참조하세요.

## 시뮬레이션 작업 배치를 복제하려면

1. AWS RoboMaker 콘솔(<https://console.aws.amazon.com/robomaker/>)에 로그인합니다.
2. 왼쪽 탐색 창에서 시뮬레이션을 선택하고 시뮬레이션 작업 배치를 선택합니다.
3. 복제하려는 시뮬레이션 작업 배치의 Id를 선택합니다.
4. 전체 배치를 복제하려면 시뮬레이션 작업 배치 세부 정보 페이지의 배치 작업에서 배치 복제를 선택합니다.

배치에서 특정 시뮬레이션 작업 요청을 복제하려면, 시뮬레이션 작업 요청에서 복제할 시뮬레이션 작업 요청을 선택한 다음 작업 요청을 선택하고 요청 복제를 선택합니다.

5. 시뮬레이션 작업 배치 복제 페이지에서 제출을 선택합니다.

## Simulation WorldForge로 월드 생성

Simulation WorldForge를 사용하면 사용자가 정의한 시뮬레이션 월드 템플릿에서 월드가 생성됩니다. 시뮬레이션 월드 템플릿은 월드 레이아웃, 방 크기, 가구, 방 연결 방식 및 기타 세부 정보를 지정합니다. 벽과 바닥 및 기타 방 특성에는 자재 특성이 있을 수 있습니다. 방은 방 유형별로 자동으로 가구가 채워질 수 있고, 원하는 가구를 선택할 수도 있습니다. 생성된 월드를 시뮬레이션 작업에 사용하고 내보내기하여 개발자 컴퓨터에서 사용할 수 있습니다.

Simulation WorldForge는 도메인 무작위화를 통해 많은 시뮬레이션 월드를 필요로 하는 시뮬레이션 워크로드를 관리하는 데 도움이 될 수 있습니다. 일반적인 Simulation WorldForge 시나리오의 예는 다음과 같습니다.

- 회귀 테스트 — 수백 개의 월드에서 로봇 애플리케이션을 테스트하여 올바른 동작을 검증합니다.
- 합성 이미지 데이터 생성 — 생성된 월드에서 이미지를 캡처하여 다른 로봇 애플리케이션에 사용할 수 있습니다. 예를 들어 가구 배치와 자재 구성이 다른 방의 이미지를 캡처할 수 있습니다.
- 강화 학습 — 로봇 애플리케이션이 탐색할 수 있도록 인테리어 구조가 있는 수백 개의 고유한 월드를 만들 수 있습니다. 월드의 구성을 직접 제어할 수 있습니다.
- 알고리즘 개발 — 로봇 탐색 엔지니어는 가구 배치가 서로 다른 알려진 레이아웃에서 탐색 알고리즘이 제대로 작동하는지 확인할 수 있습니다. 로봇 현지화 엔지니어는 레이아웃 알고리즘이 다양한 평면도에서 다양한 구조 요소를 감지하도록 할 수 있습니다.

월드 생성 알고리즘이나 인프라 생성 및 관리 방법을 몰라도 됩니다. Simulation WorldForge와 AWS RoboMaker는 완전 관리형 서비스입니다.

## Simulation WorldForge 개념

Simulation WorldForge는 파라미터 모음(시뮬레이션 월드 템플릿)을 사용하여 새로운 월드를 생성하는 방법을 결정합니다. 하나의 시뮬레이션 월드 템플릿으로 수백 개의 월드를 생성할 수 있습니다. 각 월드에는 건물이 있습니다. 건물은 한 층으로 되어 있습니다. 층에는 방의 크기와 모양을 설명하는 평면도 템플릿이 있습니다. 또한 방이 어떻게 연결될 수 있는지 제안합니다. 바닥에는 평면도의 벽 및 바닥과 같은 구조 요소의 마감 방법을 지정하는 내부 템플릿도 있습니다. 또한 내부 템플릿에는 테이블과 소파와 같은 가구와 의류, 주방용품과 같은 액세서리로 각 방을 채우는 방법을 설명하는 매개변수가 있습니다.

샘플 템플릿으로 시뮬레이션 월드 템플릿을 만들거나, 기존 템플릿을 복제하거나, <https://console.aws.amazon.com/robomaker/>에서 콘솔을 사용하여 처음부터 시작할 수 있습니다. 예를 들어

침실 1개가 포함된 월드를 생성하려는 경우 침실 1개의 아파트 샘플 템플릿으로 시작할 수 있습니다. 주방과 거실이 있는 침실 1개, 욕실 1개 개방형 평면도입니다. 각 방 유형에 적합한 전형적인 재료와 가구 및 액세서리를 사용합니다. 저장한 후에는 월드 생성기 작업을 시작하여 월드를 생성할 수 있습니다. 월드 생성 작업 하나에서 최대 50개의 월드를 생성할 수 있습니다.

SDK 또는 AWS Command Line Interface를 사용하여 시뮬레이션 월드 템플릿을 생성할 수도 있습니다. 예를 들어, AWS CLI에서 템플릿을 만들려면 먼저 템플릿 본문을 사용하여 월드 템플릿 JSON 문서를 생성해야 합니다. 이는 건물, 평면도, 내부, 기타 세부 사항에 대한 파라미터를 지정합니다. 저장한 후에는 `create-world-template`를 호출하고 JSON 파일을 지정하여 시뮬레이션 월드 템플릿을 만들 수 있습니다.

```
aws robomaker create-world-template --name "my-template" --templateBody file://
my_template_body.json
```

시뮬레이션 월드 템플릿을 구성하고 저장한 후 월드 생성 작업을 생성하고 월드를 생성할 수 있습니다. 하나의 시뮬레이션 월드 템플릿에서 수백 개의 월드를 생성할 수 있습니다. 하나의 월드 생성 작업으로 최대 100개의 월드를 생성할 수 있습니다. AWS RoboMaker에서 월드를 시뮬레이션과 함께 사용할 수 있습니다. 월드를 내보내기하여 자체 ROS 환경에서 수정하고 사용할 수도 있습니다.

## 시뮬레이션 월드 템플릿 알아보기

이 섹션에서는 시뮬레이션 월드 템플릿의 구성 요소에 대해 설명합니다. 구성 요소에는 평면도, 인테리어 재료 및 가구에 대한 선호도가 포함됩니다. Simulation WorldForge는 재료, 가구 선택, 방 연결을 포함한 많은 구성 요소에 대한 기본값을 제공합니다. 원하는 대로 기본값을 재정의할 수 있습니다. Simulation WorldForge는 월드를 생성할 때 사용자의 선호도를 따르기 위해 최선을 다합니다.

### 평면도

평면도는 단층 주거용 건물의 실내 평면도를 지정합니다. 여기에는 월드 치수, 방의 수와 유형, 방의 연결 방식에 영향을 미치는 파라미터가 포함됩니다.

모든 월드에는 로봇의 기본 시작 위치인  $(0, 0, 0)$  좌표를 중심으로 투명한 1미터 원통이 있어야 합니다. Simulation WorldForge는 방을 결정합니다.

### 월드 차원

건물의 가로 세로 비율과 천장 높이를 구성할 수 있습니다. 유효한 가로 세로 비율은 1:4에서 4:1 사이입니다. 유효한 천장 높이는 2.4~4.0미터입니다. 모든 측정값은 미터와 제곱미터 단위입니다. 콘솔은 미국 시스템과 미터법 간의 변환을 지원합니다.

## 방

방 개수, 방 유형, 방 이름, 원하는 면적, 원하는 가로 세로 비율, 인테리어 특징 등을 지정할 수 있습니다. 다음 데이터 유형이 지원됩니다.

- 침실
- 욕실
- Living
- 다이닝룸
- 주방
- 복도
- 옷장

가구, 벽재, 바닥재는 방 유형에 적합한 유형에서 선택됩니다. 예를 들어, 욕실에는 타일 벽과 리놀륨 바닥이 지정되고, 화장실과 샤워 시설이 있을 수 있습니다.

## 연결

Simulation WorldForge는 기본적으로 모든 방을 자동으로 연결합니다. 개구부 또는 출입구로 방을 연결할 수 있습니다. 방이 개구부로 연결된 경우 방은 개방형 평면도로 표시됩니다. 벽은 없습니다. 출입구로 연결된 방에는 문이 없는 좁은 개구부가 있습니다. 출입구 개구부는 인접한 벽을 따라 무작위로 배치됩니다.

기본 연결을 원하는 연결로 재정의할 수 있습니다. 예를 들어 주방, 다이닝룸, 침실이 있는 경우 주방과 침실 사이의 문 연결을 요청할 수 있습니다. Simulation WorldForge는 이러한 연결을 위해 최선을 다하지만 보장하지는 않습니다.

## 인테리어

다양한 인테리어 소재와 가구 유형 중에서 선택할 수 있습니다. Simulation WorldForge는 방 유형별로 바닥재, 벽, 가구를 무작위로 방에 할당합니다. 예를 들어 주방에 오븐과 식탁 및 의자가 배정될 수 있습니다.

바닥재 및 벽재의 유형을 사용자 지정 세트에서 선택할 수 있습니다. 사용자 지정 세트를 만들 때 방 유형 또는 방 이름별로 사용자 지정 할당을 적용할 수 있습니다. 사용자 지정 세트를 여러 개 만들 수 있습니다. 충돌이 있는 경우, 방에 대한 사용자 지정 할당이 항상 방 유형에 대한 사용자 지정 할당보다 우선합니다.

예를 들어 모든 침실에 “모던 바닥재” 사용자 지정 세트를 할당하고 “마스터 침실” 방에 “세련된 바닥재” 사용자 지정 세트를 할당했다고 가정해 보겠습니다. Simulation WorldForge에서 바닥재를 할당할 때 “마스터 침실”에는 “세련된 바닥재” 세트의 바닥재가 할당됩니다. 다른 침실에는 “모던 바닥재” 세트에서 선택한 바닥재가 사용됩니다.

이 규칙은 맞춤형 가구 세트에도 적용됩니다.

## 바닥재 유형

지원되는 바닥 형식은 다음과 같습니다.

- 카펫
- 콘크리트
- 마루판
- 리놀륨
- 쪽모이 세공
- 타일

바닥재는 선택한 모든 바닥재 유형 중에서 무작위로 선택됩니다. 예를 들어 Carpet, Concrete, linoleum, parquetry를 지정하면 방 바닥이 콘크리트가 될 수 있습니다.

## 벽재 유형

지원되는 벽재의 유형은 다음과 같습니다.

- 벽돌
- 콘크리트
- 돌
- 타일
- 목재 패널
- 벽 페인트
- 벽지

벽재는 선택한 모든 벽재 유형 중에서 무작위로 선택됩니다. 예를 들어 Brick, Tiles, Wallpaper를 지정하면 방에 타일과 벽지를 사용하는 벽이 있을 수 있습니다. Simulation WorldForge는 선택한 모든 벽재 유형에서 벽재를 할당하지 않을 수 있습니다.

## 가구 유형

Simulation WorldForge는 다음과 같은 가구 유형을 지원합니다.

- 욕조
- 바 캐비닛
- 침대
- 책장
- 커피 테이블
- 콘솔 테이블
- 코너 캐비닛
- 책상 의자
- 책상
- 식탁 의자
- 식탁
- 식기 세척기
- 옷장
- 엔드 테이블 및 사이드 테이블
- 플로어 램프
- 냉장고
- 거실 의자
- 키친 아일랜드와 카트
- 미디어 스토리지
- 침대 협탁
- 오토만
- 오븐
- 서빙 카트
- 샤워기
- 사이드보드 및 식기장
- 소파

- 스토리지
- 수납 벤치
- 변기
- 화장대 카운터
- 세탁기 및 건조기

가구는 선택한 모든 가구 유형 중에서 무작위로 선택됩니다. 예를 들어 Sideboards and buffets, Sofas, Console tables를 지정하면 방에 하나의 소파와 두 개의 콘솔 테이블이 있지만 사이드보드나 식기장은 없을 수 있습니다. Simulation WorldForge는 선택한 모든 가구 유형의 재료 유형을 할당하지 않을 수 있습니다.

## 일반적인 작업

이 섹션에는 시뮬레이션 월드 템플릿을 만드는 일반적인 작업이 포함되어 있습니다. 대부분의 작업은 원하는 연결 또는 원하는 모양을 지정합니다. Simulation WorldForge는 시뮬레이션 월드 템플릿 파라미터에 따라 월드를 생성하기 위해 최선을 다하고 있습니다. 생성된 월드가 항상 원하는 모든 속성을 포함하지는 않을 수 있습니다.

### 주제

- [한 층의 방 목록 지정](#)
- [긴 복도 요청](#)
- [방 사이 출입구 요청](#)
- [모든 방에 구성 적용](#)
- [출입구에 문 요청](#)
- [출입구에 문 없음 요청](#)
- [넓은 평면도 공간 요청](#)
- [사용자 지정 천장 높이 요청](#)
- [여러 방의 바닥에 동일한 자재 유형 지정](#)
- [같은 유형의 방 사이의 바닥에 서로 다른 자재 유형 지정](#)
- [방에 가구를 더 많이, 적게 지정](#)
- [모든 침실 및 단일 공용 거실/식당에 특정 가구 유형 추가](#)
- [가구가 없는 방 지정](#)

## 한 층의 방 목록 지정

방 유형은 어떤 방이 인접하는지에 기여하여 평면도에 영향을 미칩니다. 또한 방 유형은 바닥재 및 벽재 유형과 기본적으로 무작위로 배치할 가구 유형을 결정하는 데에도 사용됩니다. 기본 바닥재 및 벽재 유형과 가구 유형을 방 유형이나 방 이름으로 재정의할 수 있습니다.

침실, 욕실, 거실, 식당, 주방, 복도, 옷장 등의 방 유형 중에서 선택할 수 있습니다.

다음 예에서는 침실이 세 개인 집을 지정합니다. 방의 크기와 모양은 기본적으로 결정됩니다.

### Using the console

1. 시뮬레이션 월드 템플릿 편집 화면의 평면도에서 방을 선택합니다.
2. 방 창에서 방 추가를 클릭합니다.
3. 방의 세부 정보를 추가합니다. 방 이름, 방 유형, 원하는 면적, 원하는 종횡비를 지정할 수 있습니다.
4. 저장을 선택하여 새 방을 저장합니다. 원하는 방이 나올 때까지 반복합니다. 너무 많이 추가한 경우, 방 창에서 삭제할 수 있습니다.

### Using the AWS CLI

#### Example

`create-world-template` 호출의 일부로 `templateBody`에서 다음 JSON을 사용할 수 있습니다.

```
"Rooms": [  
  {  
    "Type": "Bedroom",  
    "Name": "My Master Bedroom",  
  },  
  {  
    "Type": "Bathroom",  
    "Name": "My Ensuite",  
  },  
  {  
    "Type": "Kitchen",  
    "Name": "My Kitchen",  
  }  
]
```

## 긴 복도 요청

DesiredShape 속성을 통해 원하는 방 모양을 요청할 수 있습니다. Type은 모양에 영향을 주지 않습니다. 다음 예에서는 Hallway 종횡비가 낮습니다. 충분히 큰 Area와 결합 시 길고 좁은 복도를 원한다는 뜻입니다. Simulation WorldForge는 원하는 모양과 비슷한 방을 생성하려고 합니다.

### Using the console

1. 시뮬레이션 월드 템플릿 편집 화면의 평면도에서 방을 선택합니다.
2. 방 창에서 방 추가를 클릭합니다.
3. 방 이름을 지정한 다음 방 유형의 복도를 선택합니다.
4. 원하는 면적으로 20, 원하는 종횡비로 4:1을 지정합니다.
5. 저장을 선택하여 복도를 저장합니다.

### Using the AWS CLI

#### Example

create-world-template 호출의 일부로 templateBody에서 다음 JSON을 사용할 수 있습니다.

```
"Rooms": [
  {
    "Type": "Hallway",
    "Name": "My Hallway",
    "DesiredShape": {
      "Area": 20.0,
      "AspectRatio": {
        "x": 4, "y": 1
      }
    }
  }
]
```

유효한 방 면적 범위는 10미터~300미터입니다. 유효한 방 종횡비 범위는 1:4~4:1입니다.

## 방 사이 출입구 요청

방이 두 개이고 벽이 한 개 이상인 경우 두 방 사이의 DesiredConnections를 요청할 수 있습니다. Simulation WorldForge는 방을 인접하게 배치하고 ConnectionType에 따라 인접한 벽을 따라 임의의 위치에 Doorway를 배치하거나 인접한 벽을 완전히 제거하여 Opening을 만듭니다.

다음 예시에서는 거실과 주방의 개방형 연결을 요청합니다. 또한 침실과 욕실을 위한 별도의 출입구 연결도 요청합니다.

### Using the console

1. 시뮬레이션 월드 템플릿 편집 화면의 평면도에서 연결을 선택합니다.
2. 연결 창에서 연결 추가를 선택합니다.
3. 원하는 연결 창에서 연결 유형으로 개방을 선택한 다음 “내 거실” 및 “내 주방” 등의 위치 1 및 위치 2에 대한 방을 선택합니다.
4. 저장을 선택하여 원하는 연결을 저장합니다.
5. 반복하여 “내 침실” 및 “내 욕실” 등의 다른 두 위치 간에 원하는 연결로 문을 추가합니다.

### Using the AWS CLI

#### Example

create-world-template 호출의 일부로 templateBody에서 다음 JSON을 사용할 수 있습니다.

```
"DesiredConnections": [
  {
    "Location": [ "My Living Room", "My Kitchen" ],
    "ConnectionType": "Opening"
  },
  {
    "Location": [ "My Bedroom", "My Bathroom" ],
    "ConnectionType": "Doorway"
  }
]
```

방별로 유효한 연결 수는 4회이며 방 한 쌍당 최대 1회의 개방 연결이 가능합니다.

## 모든 방에 구성 적용

### Note

버전 2 이상의 템플릿을 사용해야만 모든 방에 구성을 적용할 수 있습니다. 자세한 내용은 [모든 방에 구성 적용](#) 섹션을 참조하세요.

Target.All 키워드를 사용하여 모든 방에 구성을 적용할 수 있습니다.

다음 예시에서는 모든 문의 상태를 변경합니다.

### Using the console

다음 절차를 통해 월드의 모든 문에 구성을 적용할 수 있습니다. 단일 구성을 모든 바닥, 자재 세트, 벽 및 가구에 적용할 수도 있습니다.

1. 시뮬레이션 월드 템플릿 편집 화면의 인테리어에서 문을 선택합니다.
2. 문 창에서 사용자 지정 문 추가를 선택합니다.
3. 세트 이름에서 사용자 지정 문 세트의 이름을 지정합니다.
4. 영향을 받는 방의 경우 모든 방을 지정합니다.
5. 문 상태에서 문 열림 상태를 선택합니다.
6. 저장을 선택하여 문 구성을 저장합니다.

### Using the AWS CLI

#### Example

create-world-template 호출의 일부로 templateBody에서 다음 JSON을 사용할 수 있습니다. 다음 예시는 출입구 세트 내의 모든 문을 대상으로 합니다.

```
"Interior": {
  "Doorways": {
    "DoorwaySets": [
      {
        "Name": "your-doorway-set",
        "TargetSet": "Target.All",
```

```

    "Door": {
      "InitialState": {
        "OpenPosition": {
          "Percent": "percentage-that-you-specify"
        }
      }
    }
  ]
}

```

## 출입구에 문 요청

### Note

버전 2 이상의 월드 템플릿을 사용한 경우에만 출입구에 문이 포함되도록 구성할 수 있습니다.

템플릿을 사용하여 AWS RoboMaker Simulation WorldForge 월드에서 출입구의 문을 지정할 수 있습니다.

다음 유형의 문을 지정할 수 있습니다.

- 여닫이 문

이러한 문이 열려 있는 비율을 구성할 수 있습니다. 예를 들어 다음과 같은 열림 상태를 지정할 수 있습니다.

- 0% 열림 — 닫힘
- 50% 열림 — 반쯤 열림
- 70% 열림 — 대부분 열림
- 100% 열림 — 완전 열림

AWS RoboMaker가 각 문에 무작위로 개방 비율을 할당하도록 선택할 수도 있습니다.

다음 절차를 사용하여 출입구에 문을 추가할 수 있습니다.

## Using the console

1. 시뮬레이션 월드 템플릿 편집 화면의 인테리어에서 문을 선택합니다.
2. 문 창에서 사용자 지정 문 추가를 선택합니다.
3. 세트 이름에서 사용자 지정 문 세트의 이름을 지정합니다.
4. 위치의 영향을 받는 방에서 문을 설치할 방을 선택합니다.
5. 사용자 지정의 문 유형에서 추가하려는 문 유형을 선택합니다.
6. 문 상태에서 문이 열려 있는지, 닫혀 있는지, 부분적으로 열려 있는지 또는 임의 상태인지를 선택합니다.
7. 저장을 선택하여 구성을 저장합니다.

## Using the AWS CLI

### Example

`create-world-template` 호출의 일부로 `templateBody`에서 다음 JSON을 사용할 수 있습니다.

```
"Interior": {
  "Doorways": {
    "DoorwaySets": [
      {
        "Name": "your-doorway-set",
        "TargetSet": "the-doorways-that-you-want-to-target",
        "Door": {
          "InitialState": {
            "OpenPosition": {
              "Percent": "the-open-percentage-that-you-specify-for-the-doors-that-you're-targeting"
            }
          }
        }
      }
    ]
  }
}
```

## 출입구에 문 없음 요청

### Note

버전 2 이상의 월드 템플릿을 사용한 경우에만 출입구에 문이 포함되지 않도록 분명하게 구성할 수 있습니다.

템플릿을 사용하여 AWS RoboMaker Simulation WorldForge 월드의 출입구에는 문이 없도록 분명하게 지정할 수 있습니다.

다음 예시에서는 방 사이의 출입구에 문이 없도록 요청합니다.

### Using the console

1. 시뮬레이션 월드 템플릿 편집 화면의 인테리어에서 문을 선택합니다.
2. 문 창에서 사용자 지정 문 추가를 선택합니다.
3. 위치의 영향을 받는 방 창에서 모든 방을 선택합니다.
4. 사용자 지정의 문 유형에서 출입구에 문 없음을 선택합니다.
5. 저장을 선택합니다.

### Using the AWS CLI

#### Example

`create-world-template` 호출의 일부로 `templateBody`에서 다음 JSON을 사용할 수 있습니다.

```
"Interior": {
  "Doorways": {
    "DoorwaySets": [
      {
        "Name": "doorway-set-name",
        "TargetSet": "Target.All",
        "Door": null
      }
    ]
  }
}
```

## 넓은 평면도 공간 요청

모든 방에 영향을 미치는 더 길거나 더 넓은 평면도 배치를 원하는 경우 Footprint에 대한 DesiredAspectRatio를 요청할 수 있습니다. Simulation WorldForge는 평면도가 요청된 면적의 종횡비에 더 잘 맞도록 이 기본 설정을 사용하여 방의 전체 모양과 위치에 영향을 미칩니다. 원하는 종횡비는 선택 사항이며 기본값은 정사각형입니다.

다음 예시에서는 기본 정사각형 비율(1:1)을 재정의하여 모든 방을 늘리고 배치하여 정사각형이 아닌 공간을 만들 가능성이 높은 더 넓은 레이아웃을 취합니다.

### Using the console

1. 시뮬레이션 월드 템플릿 편집 화면의 평면도에서 월드 치수를 선택합니다.
2. 월드 치수 창의 원하는 종횡비에서 너비로 1, 길이로 4를 지정합니다.
3. 저장을 선택하여 새 방을 저장합니다.

### Using the AWS CLI

#### Example

create-world-template 호출의 일부로 templateBody에서 다음 JSON을 사용할 수 있습니다.

```
"Footprint": {
  "DesiredAspectRatio": {
    "x": 1, "y": 4
  }
}
```

DesiredAspectRatio의 유효 범위는 1:4~4:1입니다.

## 사용자 지정 천장 높이 요청

평면도 천장 높이에 따라 모든 방의 벽 높이가 결정됩니다. 기본 천장 높이는 2.4미터입니다. 이 예시에서는 기본값을 3.2미터로 재정의합니다.

### Using the console

1. 시뮬레이션 월드 템플릿 편집 화면의 평면도에서 월드 치수를 선택합니다.

2. 월드 치수 창에서 3.2의 천장 높이를 지정합니다.
3. 저장을 선택하여 새 방을 저장합니다.

## Using the AWS CLI

### Example

`create-world-template` 호출의 일부로 `templateBody`에서 다음 JSON을 사용할 수 있습니다.

```
"Ceiling": {
  "Height": 3.2
}
```

## 여러 방의 바닥에 동일한 자재 유형 지정

방 유형이나 방 이름을 사용하고 인테리어 바닥재 섹션에 여러 개의 방을 나열합니다. 다음 예시에서는 모든 침실, 거실 및 식당에 임의의 바닥재가 지정되어 있습니다.

## Using the console

1. 시뮬레이션 월드 템플릿 편집 화면의 인테리어에서 바닥재를 선택합니다.
2. 바닥재 창에서 바닥재 추가를 선택합니다.
3. 사용자 지정 바닥재 창에서 바닥재 세트 이름을 지정합니다(예: "바닥재 세트 1").
4. 필터 유형에서 방 유형별을 선택합니다.
5. 방 유형에서 침실, 거실, 식당을 선택합니다.
6. 사용자 지정 바닥재에서 자재 추가를 선택한 다음 마루판을 선택합니다.
7. 저장을 선택하여 바닥재 세트를 저장합니다.

## Using the AWS CLI

### Example

`create-world-template` 호출의 일부로 `templateBody`에서 다음 JSON을 사용할 수 있습니다.

```
"Flooring": {
```

```

"MaterialSets": [
  {
    "Name": "Flooring Material Set 1",
    "TargetSet": {
      "RoomTypes": [ "Bedroom", "Living", "Dining" ]
    },
    "SampleSet": {
      "MaterialTypes": [ "Floorboards" ]
    }
  }
]
}

```

## 같은 유형의 방 사이의 바닥에 서로 다른 자재 유형 지정

다음 예시에서는 Bedroom 3을 제외한 모든 침실, 거실 및 식당에 임의의 바닥재가 지정되어 있습니다. 무작위로 카펫 자재가 지정됩니다.

### Using the console

1. 시뮬레이션 월드 템플릿 편집 화면의 인테리어에서 바닥재를 선택합니다.
2. 바닥재 창에서 바닥재 추가를 선택합니다.
3. 사용자 지정 바닥재 창에서 바닥재 세트 이름을 지정합니다(예: Flooring Material Set 1).
4. 필터 유형에서 방 유형별을 선택합니다.
5. 방 유형에서 침실, 거실, 식당을 선택합니다.
6. 사용자 지정 바닥재에서 자재 추가를 선택한 다음 마루판을 선택합니다.
7. 저장을 선택하여 바닥재 세트를 저장합니다.
8. 바닥재 창에서 바닥재 추가를 선택합니다.
9. 사용자 지정 바닥재 창에서 바닥재 세트 이름을 지정합니다(예: Flooring Material Set for Bedroom 3).
10. 필터 유형에서 방 이름별을 선택합니다.
11. 방 이름에서 방 이름을 선택합니다(예: Bedroom 3).
12. 사용자 지정 바닥재에서 자재 추가를 선택한 다음 카펫을 선택합니다.
13. 저장을 선택하여 바닥재 세트를 저장합니다.

## Using the AWS CLI

### Example

`create-world-template` 호출의 일부로 `templateBody`에서 다음 JSON을 사용할 수 있습니다.

```
"Flooring": {
  "MaterialSets": [
    {
      "Name": "Flooring Material Set 1",
      "TargetSet": {
        "RoomTypes": [ "Bedroom", "Living", "Dining" ]
      },
      "SampleSet": {
        "MaterialTypes": [ "Floorboards" ]
      }
    },
    {
      "Name": "Flooring Material Set for Bedroom 3",
      "TargetSet": {
        "RoomNames": [ "Bedroom 3" ]
      },
      "SampleSet": {
        "MaterialTypes": [ "Carpet" ]
      }
    }
  ]
}
```

## 방에 가구를 더 많이, 적게 지정

방 이름 또는 방 유형별로 가구 간격을 지정할 수 있습니다. 기본적으로 방의 가구는 적당한 간격으로 무작위로 배치됩니다. 다음 예시에서는 모든 침실의 가구가 무작위로 촘촘한 간격으로 배치되어 있습니다. 거실과 식당에는 가구가 거의 없습니다. 다른 모든 방에는 기본적으로 가구가 비치되어 있습니다.

### Using the console

1. 시뮬레이션 월드 템플릿 편집 화면의 인테리어에서 가구를 선택합니다.
2. 가구 창에서 사용자 지정 가구 추가를 선택합니다.

3. 사용자 지정 가구 창에서 사용자 지정 가구 세트 이름을 지정합니다(예: Dense Furniture Arrangement).
4. 필터 유형에서 방 유형별을 선택합니다.
5. 방 유형에서 침실을 선택합니다.
6. 가구 재정의를 토글하여 기본 가구를 사용합니다.
7. 가구 밀도에서 고밀도를 선택합니다.
8. 저장을 선택하여 가구 세트를 저장합니다.
9. 가구 창에서 사용자 지정 가구 추가를 선택합니다.
10. 사용자 지정 가구 창에서 사용자 지정 가구 세트 이름을 지정합니다(예: Sparse Furniture Arrangement).
11. 필터 유형에서 방 이름별을 선택합니다.
12. 방 이름에서 가구 밀도를 낮추고 싶은 방을 선택합니다(예: My Living Room 및 My Dining Room).
13. 가구 재정의를 토글하여 기본 가구를 사용합니다.
14. 가구 밀도에서 저밀도를 선택합니다.
15. 저장을 선택하여 가구 세트를 저장합니다.

## Using the AWS CLI

### Example

create-world-template 호출의 일부로 templateBody에서 다음 JSON을 사용할 수 있습니다.

```
"Furniture": {
  "FurnitureArrangements": [
    {
      "Name": "Dense Furniture Arrangement",
      "TargetSet": {
        "RoomTypes": [ "Bedroom" ]
      },
      "DesiredSpatialDensity": "Dense"
    },
    {
      "Name": "Sparse Furniture Arrangement",
      "TargetSet": {
```

```

    "RoomNames": [ "My Living Room", "My Dining Room" ]
  },
  "DesiredSpatialDensity": "Sparse"
}
]
}

```

## 모든 침실 및 단일 공용 거실/식당에 특정 가구 유형 추가

방 이름 또는 방 유형별로 방의 가구 유형을 지정할 수 있습니다. 다음 예시에서는 모든 침실에 침대, 책상, 옷장, 플로어 램프가 적당히 배치되어 있습니다. “내 거실/내 식당”에는 무작위로 식탁, 식탁 의자, 플로어 램프, 소파, 커피 테이블이 조밀하게 배치되어 있습니다. 다른 모든 방에는 기본적으로 가구가 배치되어 있습니다.

### Using the console

1. 시뮬레이션 월드 템플릿 편집 화면의 인테리어에서 가구를 선택합니다.
2. 가구 창에서 사용자 지정 가구 추가를 선택합니다.
3. 사용자 지정 가구 창에서 사용자 지정 가구 세트 이름을 지정합니다(예: Bedroom Furniture).
4. 필터 유형에서 방 유형별을 선택합니다.
5. 방 유형에서 침실을 선택합니다.
6. 가구 재정의가 선택되었는지 확인합니다. 선택되지 않은 경우 Simulation WorldForge는 기본 가구를 사용합니다.
7. 가구 유형에서 가구 추가를 선택한 다음 침대, 책상, 옷장 및 플로어 램프를 선택합니다.
8. 저장을 선택하여 가구 세트를 저장합니다.
9. 가구 창에서 사용자 지정 가구 추가를 선택합니다.
10. 사용자 지정 가구 창에서 사용자 지정 가구 세트 이름을 지정합니다(예: Living and Dining Furniture).
11. 필터 유형에서 방 이름별을 선택합니다.
12. 방 이름에서 방을 선택합니다(예: My living and dining room).
13. 가구 재정의가 선택되었는지 확인합니다. 선택되지 않은 경우 Simulation WorldForge는 기본 가구를 사용합니다.
14. 가구 유형에서 식탁, 식탁 의자, 플로어 램프, 소파 및 커피 테이블을 선택합니다.

15. 가구 밀도에서 고밀도를 선택합니다.
16. 저장을 선택하여 가구 세트를 저장합니다.

## Using the AWS CLI

### Example

create-world-template 호출의 일부로 templateBody에서 다음 JSON을 사용할 수 있습니다.

```
"Furniture": {
  "FurnitureArrangements": [
    {
      "Name": "Bedroom Furniture",
      "TargetSet": {
        "RoomTypes": [ "Bedroom" ]
      },
      "SampleSet": {
        "ModelTypes": [
          "Beds",
          "Desks",
          "Dressers",
          "FloorLamps"
        ]
      }
    }
  ]
}
{
  "Name": "Living and Dining Furniture",
  "TargetSet": {
    "RoomNames": [ "My living and dining room" ]
  },
  "SampleSet": {
    "ModelTypes": [
      "DiningTables",
      "DiningChairs",
      "FloorLamps",
      "Sofas",
      "CoffeeTables"
    ],
    "DesiredSpatialDensity": "Dense"
  }
}
}
```

```

]
}

```

## 가구가 없는 방 지정

가구 배치를 위한 모델 세트에 빈 목록을 지정합니다. 다른 모든 방에는 기본적으로 가구가 비치되어 있습니다.

### Using the console

1. 시뮬레이션 월드 템플릿 편집 화면의 인테리어에서 가구를 선택합니다.
2. 가구 창에서 사용자 지정 가구 추가를 선택합니다.
3. 사용자 지정 가구 창에서 사용자 지정 가구 세트 이름을 지정합니다(예: No furniture).
4. 필터 유형에서 방 이름별을 선택합니다.
5. 방 이름에서 가구를 없애고 싶은 방을 선택합니다(예: My Spare Room).
6. 가구 재정의가 선택되었는지 확인합니다. 선택되지 않은 경우 Simulation WorldForge는 기본 가구를 사용합니다.
7. 가구 유형에서 선택된 유형이 없도록 확인합니다.
8. 저장을 선택하여 가구 세트를 저장합니다.

### Using the AWS CLI

#### Example

create-world-template 호출의 일부로 templateBody에서 다음 JSON을 사용할 수 있습니다.

```

"Furniture": {
  "FurnitureArrangements": [
    {
      "Name": "No Furniture",
      "TargetSet": {
        "RoomNames": [ "My Spare Room" ]
      },
      "SampleSet": {
        "ModelTypes": []
      }
    }
  ]
}

```

```
]
}
```

## 시뮬레이션 월드 템플릿 본문을 위한 JSON 스키마

templateBody(시뮬레이션 월드 템플릿 본문)는 [CreateWorldTemplate](#) 작업의 입력 파라미터입니다. 이 파라미터는 JSON 형식의 문자열입니다. JSON은 시뮬레이션 월드 템플릿을 지정하며 Simulation WorldForge가 월드를 생성하는 데 사용하는 파라미터를 포함합니다.

다음은 다양한 버전의 월드 템플릿에 대한 스키마를 보여줍니다.

### 버전 2

다음은 버전 2 스키마의 템플릿입니다.

```
{
  "title": "WorldTemplate",
  "description": "The top-level template for parameterizing a randomly generated world. By default, a single\nresidential building with one floor and one room is generated.",
  "type": "object",
  "properties": {
    "Version": {
      "title": "Version",
      "type": "string"
    },
    "Buildings": {
      "title": "Buildings",
      "default": [
        {
          "Floors": [
            {
              "Floorplan": {
                "Footprint": {
                  "DesiredAspectRatio": {
                    "x": 1.0,
                    "y": 1.0
                  }
                }
              },
              "Ceiling": {
                "Height": 3.0
              },
              "Rooms": [
```

```

        {
            "Type": "Living",
            "Name": "My_Living_Room",
            "OriginalName": "My Living Room",
            "DesiredShape": {
                "Area": 20.0,
                "AspectRatio": {
                    "x": 1.0,
                    "y": 1.0
                }
            }
        }
    ],
    "DesiredConnections": [],
},
"Interior": {
    "Doorways": {
        "DoorwaySets": []
    },
    "Flooring": {
        "MaterialSets": []
    },
    "Walls": {
        "MaterialSets": []
    },
    "Furniture": {
        "FurnitureArrangements": []
    }
}
}
]
}
],
"type": "array",
"items": {
    "$ref": "#/definitions/BuildingTemplate"
},
"minItems": 1,
"maxItems": 1
}
},
"required": [
    "Version"
],

```

```
"additionalProperties": false,
"definitions": {
  "AspectRatio": {
    "title": "AspectRatio",
    "type": "object",
    "properties": {
      "x": {
        "title": "X",
        "default": 1,
        "minimum": 1,
        "maximum": 4,
        "type": "number"
      },
      "y": {
        "title": "Y",
        "default": 1,
        "minimum": 1,
        "maximum": 4,
        "type": "number"
      }
    },
    "additionalProperties": false
  },
  "FloorplanFootprint": {
    "title": "FloorplanFootprint",
    "description": "The desired footprint of this floorplan.",
    "type": "object",
    "properties": {
      "DesiredAspectRatio": {
        "title": "Desiredaspectratio",
        "default": {
          "x": 1.0,
          "y": 1.0
        },
        "allOf": [
          {
            "$ref": "#/definitions/AspectRatio"
          }
        ]
      },
      "additionalProperties": false
    },
    "FloorplanCeiling": {
```

```
"title": "FloorplanCeiling",
"description": "The height of the ceiling for this floorplan in metres.",
"type": "object",
"properties": {
  "Height": {
    "title": "Height",
    "default": 3.0,
    "type": "number",
    "minimum": 2.4,
    "maximum": 4.0
  }
},
"additionalProperties": false
},
"Rectangle": {
  "title": "Rectangle",
  "description": "A rectangle defined by area in square metres and aspect ratio.",
  "type": "object",
  "properties": {
    "Area": {
      "title": "Area",
      "type": "number"
    },
    "AspectRatio": {
      "$ref": "#/definitions/AspectRatio"
    }
  },
  "required": [
    "Area",
    "AspectRatio"
  ],
  "additionalProperties": false
},
"FloorplanRoom": {
  "title": "FloorplanRoom",
  "description": "A description for single room for this floorplan.",
  "type": "object",
  "properties": {
    "Type": {
      "title": "Type",
      "enum": [
        "Bedroom",
        "Bathroom",
        "Living",

```

```

        "Dining",
        "Kitchen",
        "Hallway",
        "Closet"
    ],
    "type": "string"
},
"Name": {
    "title": "Name",
    "maxLength": 255,
    "minLength": 1,
    "pattern": "^[a-zA-Z0-9_\\- ]*$",
    "type": "string"
},
"OriginalName": {
    "title": "Originalname",
    "type": "string"
},
"DesiredShape": {
    "title": "Desiredshape",
    "default": {
        "Area": 20.0,
        "AspectRatio": {
            "x": 1.0,
            "y": 1.0
        }
    }
},
"allOf": [
    {
        "$ref": "#/definitions/Rectangle"
    }
]
}
},
"required": [
    "Type",
    "Name"
],
"additionalProperties": false
},
"FloorplanConnection": {
    "title": "FloorplanConnection",
    "description": "Describes the desired layout of the rooms and their adjacent rooms. A connection can be either a doorway or\nan open space without any walls. Two

```

```
rooms cannot both share an interior doorway and an opening.\n
The same two rooms can have multiple doorways, up to a limit.",
  "type": "object",
  "properties": {
    "Location": {
      "title": "Location",
      "type": "array",
      "items": {
        "type": "string"
      },
      "minItems": 2,
      "maxItems": 2
    },
    "ConnectionType": {
      "title": "Connectiontype",
      "enum": [
        "Doorway",
        "Opening"
      ],
      "type": "string"
    }
  },
  "required": [
    "Location",
    "ConnectionType"
  ],
  "additionalProperties": false
},
"FloorplanTemplate": {
  "title": "FloorplanTemplate",
  "description": "The top-level floorplan template that parameterizes the randomly
generated\narchitectural layout. By default, a residential floorplan with bedroom
and\nliving room are generated with a random doorway or opening connection.\n\n
The footprint contributes to the overall shape of the floor layout along\nwith
rooms. The footprint shape is desired as it is a preference and not\n
guaranteed.\n\nThe ceiling determines the height of the walls. There are minimum
and maximum ceiling heights. The ceiling height is guaranteed.\n\n
Rooms are required. Each room has a desired shape. Together, the room\n
shapes and footprint determine floor layout. The room types contribute to\n
the layout and are used when randomly selecting furniture and materials for\n
the walls and floors.\n\nDesiredConnections are optional. Two rooms are
connected if they share a\nwall and doorway or adjacent without any wall aka
\n\"opening \". All rooms are\n
guaranteed to be connected randomly if they are not specified in the\n
connections list. Connections that are specified are _not_ guaranteed but\n
will be attempted as best-effort.",
```

```
"type": "object",
"properties": {
  "Footprint": {
    "title": "Footprint",
    "default": {
      "DesiredAspectRatio": {
        "x": 1.0,
        "y": 1.0
      }
    },
    "allOf": [
      {
        "$ref": "#/definitions/FloorplanFootprint"
      }
    ]
  },
  "Ceiling": {
    "title": "Ceiling",
    "default": {
      "Height": 3.0
    },
    "allOf": [
      {
        "$ref": "#/definitions/FloorplanCeiling"
      }
    ]
  },
  "Rooms": {
    "title": "Rooms",
    "default": [
      {
        "Type": "Living",
        "Name": "My_Living_Room",
        "OriginalName": "My Living Room",
        "DesiredShape": {
          "Area": 20.0,
          "AspectRatio": {
            "x": 1.0,
            "y": 1.0
          }
        }
      }
    ]
  },
  "type": "array",
```

```
    "items": {
      "$ref": "#/definitions/FloorplanRoom"
    },
    "minItems": 1,
    "maxItems": 6
  },
  "DesiredConnections": {
    "title": "Desiredconnections",
    "default": [],
    "type": "array",
    "items": {
      "$ref": "#/definitions/FloorplanConnection"
    },
    "minItems": 0,
    "maxItems": 12
  }
},
"additionalProperties": false
},
"RoomNameList": {
  "title": "RoomNameList",
  "description": "The set of all rooms matching any of the listed room names.",
  "type": "object",
  "properties": {
    "RoomNames": {
      "title": "Roomnames",
      "type": "array",
      "items": {
        "type": "string"
      },
      "minItems": 1,
      "maxItems": 6
    }
  },
  "required": [
    "RoomNames"
  ],
  "additionalProperties": false
},
"RoomTypeList": {
  "title": "RoomTypeList",
  "description": "The set of all rooms matching any of the listed room types.",
  "type": "object",
  "properties": {
```

```

    "RoomTypes": {
      "title": "Roomtypes",
      "type": "array",
      "items": {
        "enum": [
          "Bedroom",
          "Bathroom",
          "Living",
          "Dining",
          "Kitchen",
          "Hallway",
          "Closet"
        ],
        "type": "string"
      },
      "minItems": 1,
      "maxItems": 7
    }
  },
  "required": [
    "RoomTypes"
  ],
  "additionalProperties": false
},
"RoomPairTargetFilter": {
  "title": "RoomPairTargetFilter",
  "description": "Defines a target set as a pair of rooms. The pairs are defined as the cross product of two lists\nFrom and To.",
  "type": "object",
  "properties": {
    "From": {
      "title": "From",
      "anyOf": [
        {
          "$ref": "#/definitions/RoomNameList"
        },
        {
          "$ref": "#/definitions/RoomTypeList"
        }
      ]
    },
    "To": {
      "title": "To",
      "anyOf": [

```

```

        {
            "$ref": "#/definitions/RoomNameList"
        },
        {
            "$ref": "#/definitions/RoomTypeList"
        }
    ]
}
},
"required": [
    "From",
    "To"
],
"additionalProperties": false
},
"DoorOpenPosition": {
    "title": "DoorOpenPosition",
    "description": "Defines the amount of openness of an InteriorDoor.\n\nThe range
for Percent is [0., 100.]",
    "type": "object",
    "properties": {
        "Percent": {
            "title": "Percent",
            "default": 100.0,
            "anyOf": [
                {
                    "type": "number",
                    "minimum": 0.0,
                    "maximum": 100.0
                },
                {
                    "const": "Random",
                    "type": "string"
                }
            ]
        }
    }
},
"additionalProperties": false
},
"DoorInitialState": {
    "title": "DoorInitialState",
    "description": "Defines the initial state for an InteriorDoor object\n
\nOpenPosition specifies how much the door should be open.",
    "type": "object",

```

```

    "properties": {
      "OpenPosition": {
        "title": "Openposition",
        "default": {
          "Percent": 100.0
        },
        "allOf": [
          {
            "$ref": "#/definitions/DoorOpenPosition"
          }
        ]
      },
      "additionalProperties": false
    },
    "InteriorDoor": {
      "title": "InteriorDoor",
      "description": "Custom configuration for each Doorway Set.\n\nInitial State of doors includes the ability to configure how much the door should be open in\npercent [0., 100.]",
      "type": "object",
      "properties": {
        "InitialState": {
          "title": "Initialstate",
          "default": {
            "OpenPosition": {
              "Percent": 100.0
            }
          },
          "allOf": [
            {
              "$ref": "#/definitions/DoorInitialState"
            }
          ]
        },
        "additionalProperties": false
      },
      "InteriorDoorwaySet": {
        "title": "InteriorDoorwaySet",
        "description": "A set of doors to randomly assign to a set of interior target elements.\n\nThe target set determines *what room pairs* are receive the doors as specified in `Door`.\nRooms may be targeted by room type or room name.\n\nThe Door customizes the configuration for doors added in the specified target set.",

```

```

"type": "object",
"properties": {
  "Name": {
    "title": "Name",
    "maxLength": 255,
    "minLength": 1,
    "pattern": "^[a-zA-Z0-9_\\- ]*$",
    "type": "string"
  },
  "TargetSet": {
    "title": "Targetset",
    "anyOf": [
      {
        "const": "Target.All",
        "type": "string"
      },
      {
        "$ref": "#/definitions/RoomPairTargetFilter"
      }
    ]
  },
  "Door": {
    "title": "Door",
    "anyOf": [
      {
        "$ref": "#/definitions/InteriorDoor"
      },
      {
        "const": null
      }
    ]
  }
},
"required": [
  "Name",
  "TargetSet"
],
"additionalProperties": false
},
"InteriorDoorways": {
  "title": "InteriorDoorways",
  "description": "Describes the interior template parameters for all doorways for this floorplan.\nAll doorways not explicitly targeted will have a random door assigned fully opened.",

```

```

    "type": "object",
    "properties": {
      "DoorwaySets": {
        "title": "Doorwaysets",
        "default": [],
        "type": "array",
        "items": {
          "$ref": "#/definitions/InteriorDoorwaySet"
        },
        "minItems": 0,
        "maxItems": 13
      }
    },
    "additionalProperties": false
  },
  "MaterialSetByMaterialType": {
    "title": "MaterialSetByMaterialType",
    "description": "The set of materials that match any of the material types listed. An empty\nset is invalid since all targets require materials.",
    "type": "object",
    "properties": {
      "MaterialTypes": {
        "title": "Materialtypes",
        "type": "array",
        "items": {
          "type": "string"
        },
        "minItems": 1
      }
    },
    "required": [
      "MaterialTypes"
    ],
    "additionalProperties": false
  },
  "InteriorMaterialSet": {
    "title": "InteriorMaterialSet",
    "description": "A set of sample materials to randomly assign to a set of interior target elements.\n\nThe target set determines *what rooms* receive the materials in the sample\nset. The targets in a room are the walls and flooring. Rooms may be targeted\nby room type or room name.\n\nThe sample set determines *what materials* to randomly select for the\ntarget rooms' walls and floors.\n\nThe sample set is optional and when not specified (null) materials are\nrandomly selected according to the room"
  }
}

```

type for each room in the target\nset.\n\nA sample set with an empty material set is invalid since all wall\nand flooring targets require materials.",

```

"type": "object",
"properties": {
  "Name": {
    "title": "Name",
    "maxLength": 255,
    "minLength": 1,
    "pattern": "^[a-zA-Z0-9_\\- ]*$",
    "type": "string"
  },
  "TargetSet": {
    "title": "Targetset",
    "anyOf": [
      {
        "const": "Target.All",
        "type": "string"
      },
      {
        "anyOf": [
          {
            "$ref": "#/definitions/RoomNameList"
          },
          {
            "$ref": "#/definitions/RoomTypeList"
          }
        ]
      }
    ]
  },
  "SampleSet": {
    "$ref": "#/definitions/MaterialSetByMaterialType"
  }
},
"required": [
  "Name",
  "TargetSet"
],
"additionalProperties": false
},
"InteriorFlooring": {
  "title": "InteriorFlooring",

```

```
    "description": "Describes the interior template parameters for all floors for
this floorplan.\nAll floors not explicitly targeted will have a random floor material
assigned by room type.",
    "type": "object",
    "properties": {
      "MaterialSets": {
        "title": "Materialsets",
        "default": [],
        "type": "array",
        "items": {
          "$ref": "#/definitions/InteriorMaterialSet"
        },
        "minItems": 0,
        "maxItems": 6
      }
    },
    "additionalProperties": false
  },
  "InteriorWalls": {
    "title": "InteriorWalls",
    "description": "Describes the interior template parameters for all walls for
this floorplan.\nAll walls not explicitly targeted will have a random wall material
assigned by room type.",
    "type": "object",
    "properties": {
      "MaterialSets": {
        "title": "Materialsets",
        "default": [],
        "type": "array",
        "items": {
          "$ref": "#/definitions/InteriorMaterialSet"
        },
        "minItems": 0,
        "maxItems": 6
      }
    },
    "additionalProperties": false
  },
  "ModelTypeList": {
    "title": "ModelTypeList",
    "description": "The set of all models matching any of the listed model types.\nAn
empty set means zero models to sample/select.",
    "type": "object",
    "properties": {
```

```
"ModelTypes": {
  "title": "Modeltypes",
  "type": "array",
  "items": {
    "enum": [
      "Baths",
      "BarCabinets",
      "Beds",
      "Bookcases",
      "CoffeeTables",
      "ConsoleTables",
      "CornerCabinets",
      "DeskChairs",
      "Desks",
      "DiningChairs",
      "DiningTables",
      "DishWashers",
      "Dressers",
      "EndAndSideTables",
      "FloorLamps",
      "Fridges",
      "LivingRoomChairs",
      "KitchenIslandsAndCarts",
      "MediaStorage",
      "Nightstands",
      "Ottomans",
      "Ovens",
      "ServingCarts",
      "Showers",
      "SideboardsAndBuffets",
      "Sofas",
      "Storage",
      "StorageBenches",
      "Toilets",
      "VanityCounters",
      "WashingMachinesAndDryers"
    ],
    "type": "string"
  },
  "minItems": 0
},
"required": [
  "ModelTypes"
```

```

    ],
    "additionalProperties": false
  },
  "FurnitureArrangementSet": {
    "title": "FurnitureArrangementSet",
    "description": "Describes the interior template for placing furniture in one
or more rooms.\n\n- TargetSet is the set of rooms to furnish, filter by room name or
room\n type.\n- SampleSet is a set of all furnishing models to randomly choose and\n
place.\n- DesiredSpatialDensity is the desired level of free space after placing\n
furniture.",
    "type": "object",
    "properties": {
      "Name": {
        "title": "Name",
        "maxLength": 255,
        "minLength": 1,
        "pattern": "^[a-zA-Z0-9_\\- ]*$",
        "type": "string"
      },
      "TargetSet": {
        "title": "Targetset",
        "anyOf": [
          {
            "const": "Target.All",
            "type": "string"
          },
          {
            "anyOf": [
              {
                "$ref": "#/definitions/RoomNameList"
              },
              {
                "$ref": "#/definitions/RoomTypeList"
              }
            ]
          }
        ]
      },
      "SampleSet": {
        "$ref": "#/definitions/ModelTypeList"
      },
      "DesiredSpatialDensity": {
        "title": "Desiredspatialdensity",
        "default": "Moderate",

```

```

        "enum": [
            "Sparse",
            "Moderate",
            "Dense"
        ],
        "type": "string"
    }
},
"required": [
    "Name",
    "TargetSet"
],
"additionalProperties": false
},
"InteriorFurnishings": {
    "title": "InteriorFurnishings",
    "description": "Describes the types of furniture models for randomly placing into
each room\nin the world. Rooms are targeted by room type or room name. Rooms that
are\nnot targeted are furnished at random by their room type with moderate density.
\ndensity. For an empty room, specify an empty sample set.",
    "type": "object",
    "properties": {
        "FurnitureArrangements": {
            "title": "Furniturearrangements",
            "default": [],
            "type": "array",
            "items": {
                "$ref": "#/definitions/FurnitureArrangementSet"
            },
            "minItems": 0,
            "maxItems": 6
        }
    },
    "additionalProperties": false
},
"InteriorTemplate": {
    "title": "InteriorTemplate",
    "description": "Top-level template for parameterizing the interior finishes and
furnishings for\nthis floorplan.",
    "type": "object",
    "properties": {
        "Doorways": {
            "title": "Doorways",
            "default": {

```

```
    "DoorwaySets": []
  },
  "allOf": [
    {
      "$ref": "#/definitions/InteriorDoorways"
    }
  ]
},
"Flooring": {
  "title": "Flooring",
  "default": {
    "MaterialSets": []
  },
  "allOf": [
    {
      "$ref": "#/definitions/InteriorFlooring"
    }
  ]
},
"Walls": {
  "title": "Walls",
  "default": {
    "MaterialSets": []
  },
  "allOf": [
    {
      "$ref": "#/definitions/InteriorWalls"
    }
  ]
},
"Furniture": {
  "title": "Furniture",
  "default": {
    "FurnitureArrangements": []
  },
  "allOf": [
    {
      "$ref": "#/definitions/InteriorFurnishings"
    }
  ]
}
},
"additionalProperties": false
},
```

```
"FloorTemplate": {
  "title": "FloorTemplate",
  "description": "Describes a single floor within a building. Defaults to a
single residential room\nof a random type and size, and the interior is randomly
furnished.",
  "type": "object",
  "properties": {
    "Floorplan": {
      "title": "Floorplan",
      "default": {
        "Footprint": {
          "DesiredAspectRatio": {
            "x": 1.0,
            "y": 1.0
          }
        },
        "Ceiling": {
          "Height": 3.0
        },
        "Rooms": [
          {
            "Type": "Living",
            "Name": "My_Living_Room",
            "OriginalName": "My Living Room",
            "DesiredShape": {
              "Area": 20.0,
              "AspectRatio": {
                "x": 1.0,
                "y": 1.0
              }
            }
          }
        ],
        "DesiredConnections": []
      },
      "allOf": [
        {
          "$ref": "#/definitions/FloorplanTemplate"
        }
      ]
    },
    "Interior": {
      "title": "Interior",
      "default": {
```

```

    "Doorways": {
      "DoorwaySets": []
    },
    "Flooring": {
      "MaterialSets": []
    },
    "Walls": {
      "MaterialSets": []
    },
    "Furniture": {
      "FurnitureArrangements": []
    }
  },
  "allOf": [
    {
      "$ref": "#/definitions/InteriorTemplate"
    }
  ]
},
"additionalProperties": false
},
"BuildingTemplate": {
  "title": "BuildingTemplate",
  "description": "Describes a building to be randomly generated. Defaults to one residential floor.",
  "type": "object",
  "properties": {
    "Floors": {
      "title": "Floors",
      "default": [
        {
          "Floorplan": {
            "Footprint": {
              "DesiredAspectRatio": {
                "x": 1.0,
                "y": 1.0
              }
            },
            "Ceiling": {
              "Height": 3.0
            }
          },
          "Rooms": [
            {

```

```
        "Type": "Living",
        "Name": "My_Living_Room",
        "OriginalName": "My Living Room",
        "DesiredShape": {
            "Area": 20.0,
            "AspectRatio": {
                "x": 1.0,
                "y": 1.0
            }
        }
    ],
    "DesiredConnections": [],
},
"Interior": {
    "Doorways": {
        "DoorwaySets": []
    },
    "Flooring": {
        "MaterialSets": []
    },
    "Walls": {
        "MaterialSets": []
    },
    "Furniture": {
        "FurnitureArrangements": []
    }
}
},
],
"type": "array",
"items": {
    "$ref": "#/definitions/FloorTemplate"
},
"minItems": 1,
"maxItems": 1
}
},
"additionalProperties": false
}
}
```

## 버전 1

다음은 버전 1 스키마의 템플릿입니다.

```
{
  "title": "WorldTemplate",
  "description": "The top-level template for parameterizing a randomly generated world.
  By default, a single\nresidential building with one floor and one room is generated.",
  "type": "object",
  "properties": {
    "Version": {
      "title": "Version",
      "default": "1",
      "type": "string"
    },
    "Buildings": {
      "title": "Buildings",
      "default": [
        {
          "Floors": [
            {
              "Floorplan": {
                "Footprint": {
                  "DesiredAspectRatio": {
                    "x": 1.0,
                    "y": 1.0
                  }
                }
              },
              "Ceiling": {
                "Height": 3.0
              },
              "Rooms": [
                {
                  "Type": "Living",
                  "Name": "My Living Room",
                  "DesiredShape": {
                    "Area": 20.0,
                    "AspectRatio": {
                      "x": 1.0,
                      "y": 1.0
                    }
                  }
                }
              ]
            }
          ]
        }
      ]
    }
  }
}
```

```

        "Type": "Bedroom",
        "Name": "My Bedroom",
        "DesiredShape": {
            "Area": 20.0,
            "AspectRatio": {
                "x": 1.0,
                "y": 1.0
            }
        }
    ],
    "DesiredConnections": [],
},
"Interior": {
    "Flooring": {
        "MaterialSets": []
    },
    "Walls": {
        "MaterialSets": []
    },
    "Furniture": {
        "FurnitureArrangements": []
    }
}
]
}
],
"type": "array",
"items": {
    "$ref": "#/definitions/BuildingTemplate"
},
"minItems": 1,
"maxItems": 1
}
},
"additionalProperties": false,
"definitions": {
    "AspectRatio": {
        "title": "AspectRatio",
        "type": "object",
        "properties": {
            "x": {
                "title": "X",

```

```
        "default": 1,
        "minimum": 1,
        "maximum": 4,
        "type": "number"
    },
    "y": {
        "title": "Y",
        "default": 1,
        "minimum": 1,
        "maximum": 4,
        "type": "number"
    }
},
"additionalProperties": false
},
"FloorplanFootprint": {
    "title": "FloorplanFootprint",
    "description": "The desired footprint of this floorplan.",
    "type": "object",
    "properties": {
        "DesiredAspectRatio": {
            "title": "Desiredaspectratio",
            "default": {
                "x": 1.0,
                "y": 1.0
            },
            "allOf": [
                {
                    "$ref": "#/definitions/AspectRatio"
                }
            ]
        }
    }
},
"additionalProperties": false
},
"FloorplanCeiling": {
    "title": "FloorplanCeiling",
    "description": "The height of the ceiling for this floorplan in metres.",
    "type": "object",
    "properties": {
        "Height": {
            "title": "Height",
            "default": 3.0,
            "type": "number",
```

```
        "minimum": 2.4,
        "maximum": 4.0
    }
},
"additionalProperties": false
},
"Rectangle": {
    "title": "Rectangle",
    "description": "A rectangle defined by area in square metres and aspect ratio.",
    "type": "object",
    "properties": {
        "Area": {
            "title": "Area",
            "type": "number"
        },
        "AspectRatio": {
            "$ref": "#/definitions/AspectRatio"
        }
    },
    "required": [
        "Area",
        "AspectRatio"
    ],
    "additionalProperties": false
},
"FloorplanRoom": {
    "title": "FloorplanRoom",
    "description": "A description for single room for this floorplan.",
    "type": "object",
    "properties": {
        "Type": {
            "title": "Type",
            "enum": [
                "Bedroom",
                "Bathroom",
                "Living",
                "Dining",
                "Kitchen",
                "Hallway",
                "Closet"
            ],
            "type": "string"
        },
        "Name": {
```

```

    "title": "Name",
    "type": "string"
  },
  "DesiredShape": {
    "title": "Desiredshape",
    "default": {
      "Area": 20.0,
      "AspectRatio": {
        "x": 1.0,
        "y": 1.0
      }
    },
  },
  "allOf": [
    {
      "$ref": "#/definitions/Rectangle"
    }
  ]
}
},
"required": [
  "Type",
  "Name"
],
"additionalProperties": false
},
"FloorplanConnection": {
  "title": "FloorplanConnection",
  "description": "Describes the desired layout of the rooms and their adjacent rooms. A connection can be either a doorway or \nan open space without any walls. Two rooms cannot both share an interior doorway and an opening. \nThe same two rooms can have multiple doorways, up to a limit.",
  "type": "object",
  "properties": {
    "Location": {
      "title": "Location",
      "type": "array",
      "items": {
        "type": "string"
      },
      "minItems": 2,
      "maxItems": 2
    },
  },
  "ConnectionType": {
    "title": "Connectiontype",

```

```

    "enum": [
      "Doorway",
      "Opening"
    ],
    "type": "string"
  }
},
"required": [
  "Location",
  "ConnectionType"
],
"additionalProperties": false
},
"FloorplanTemplate": {
  "title": "FloorplanTemplate",
  "description": "The top-level floorplan template that parameterizes the randomly
generated \narchitectural layout. By default, a residential floorplan with bedroom
and \nliving room are generated with a random doorway or opening connection. \n\nThe
footprint contributes to the overall shape of the floor layout along\nwith rooms. The
footprint shape is desired as it is a preference and not\nnguaranteed.\n\nThe ceiling
determines the height of the walls. There are minimum and\nmaximum ceiling heights.
The ceiling height is guaranteed.\n\nRooms are required. Each room has a desired
shape. Together, the room\nshapes and footprint determine floor layout. The room
types contribute to\nthe layout and are used when randomly selecting furniture and
materials for\nthe walls and floors.\n\nDesiredConnections are optional. Two rooms are
connected if they share a\nwall and doorway or adjacent without any wall aka \"opening
\". All rooms are\nnguaranteed to be connected randomly if they are not specified in the
\nconnections list. Connections that are specified are _not_ guaranteed but\nwill be
attempted as best-effort.",
  "type": "object",
  "properties": {
    "Footprint": {
      "title": "Footprint",
      "default": {
        "DesiredAspectRatio": {
          "x": 1.0,
          "y": 1.0
        }
      }
    },
    "allOf": [
      {
        "$ref": "#/definitions/FloorplanFootprint"
      }
    ]
  }
}

```

```
    },
    "Ceiling": {
      "title": "Ceiling",
      "default": {
        "Height": 3.0
      },
      "allof": [
        {
          "$ref": "#/definitions/FloorplanCeiling"
        }
      ]
    },
  ],
  "Rooms": {
    "title": "Rooms",
    "default": [
      {
        "Type": "Living",
        "Name": "My Living Room",
        "DesiredShape": {
          "Area": 20.0,
          "AspectRatio": {
            "x": 1.0,
            "y": 1.0
          }
        }
      },
      {
        "Type": "Bedroom",
        "Name": "My Bedroom",
        "DesiredShape": {
          "Area": 20.0,
          "AspectRatio": {
            "x": 1.0,
            "y": 1.0
          }
        }
      }
    ],
    "type": "array",
    "items": {
      "$ref": "#/definitions/FloorplanRoom"
    },
    "minItems": 1,
    "maxItems": 6
  }
}
```

```
    },
    "DesiredConnections": {
      "title": "Desiredconnections",
      "default": [],
      "type": "array",
      "items": {
        "$ref": "#/definitions/FloorplanConnection"
      },
      "minItems": 0,
      "maxItems": 12
    }
  },
  "additionalProperties": false
},
"RoomNameList": {
  "title": "RoomNameList",
  "description": "The set of all rooms matching any of the listed room names.",
  "type": "object",
  "properties": {
    "RoomNames": {
      "title": "Roomnames",
      "type": "array",
      "items": {
        "type": "string"
      }
    }
  },
  "required": [
    "RoomNames"
  ],
  "additionalProperties": false
},
"RoomTypeList": {
  "title": "RoomTypeList",
  "description": "The set of all rooms matching any of the listed room types.",
  "type": "object",
  "properties": {
    "RoomTypes": {
      "title": "Roomtypes",
      "type": "array",
      "items": {
        "enum": [
          "Bedroom",
          "Bathroom",

```

```

        "Living",
        "Dining",
        "Kitchen",
        "Hallway",
        "Closet"
    ],
    "type": "string"
  }
}
},
"required": [
  "RoomTypes"
],
"additionalProperties": false
},
"MaterialSetByMaterialType": {
  "title": "MaterialSetByMaterialType",
  "description": "The set of materials that match any of the material types listed.
An empty\nset is invalid since all targets require materials.",
  "type": "object",
  "properties": {
    "MaterialTypes": {
      "title": "Materialtypes",
      "type": "array",
      "items": {
        "type": "string"
      },
      "minItems": 1
    }
  },
  "required": [
    "MaterialTypes"
  ],
  "additionalProperties": false
},
"InteriorMaterialSet": {
  "title": "InteriorMaterialSet",
  "description": "A set of sample materials to randomly assign to a set of interior
target elements.\n\nThe target set determines *what rooms* receive the materials
in the sample\nset. The targets in a room are the walls and flooring. Rooms may be
targeted \nby room type or room name. \n\nThe sample set determines *what materials*
to randomly select for the\ntarget rooms' walls and floors. \n\nThe sample set is
optional and when not specified (null) materials are\nrandomly selected according to

```

```

the room type for each room in the target\nset.\n\nA sample set with an empty material
set is invalid since all wall \nand flooring targets require materials.",
  "type": "object",
  "properties": {
    "Name": {
      "title": "Name",
      "type": "string"
    },
    "TargetSet": {
      "title": "Targetset",
      "anyOf": [
        {
          "$ref": "#/definitions/RoomNameList"
        },
        {
          "$ref": "#/definitions/RoomTypeList"
        }
      ]
    },
    "SampleSet": {
      "$ref": "#/definitions/MaterialSetByMaterialType"
    }
  },
  "required": [
    "Name",
    "TargetSet"
  ],
  "additionalProperties": false
},
"InteriorFlooring": {
  "title": "InteriorFlooring",
  "description": "Describes the interior template parameters for all floors for
this floorplan.\nAll floors not explicitly targeted will have a random floor material
assigned by room type.",
  "type": "object",
  "properties": {
    "MaterialSets": {
      "title": "Materialsets",
      "default": [],
      "type": "array",
      "items": {
        "$ref": "#/definitions/InteriorMaterialSet"
      }
    },
    "minItems": 0,

```

```

        "maxItems": 6
    }
},
"additionalProperties": false
},
"InteriorWalls": {
    "title": "InteriorWalls",
    "description": "Describes the interior template parameters for all walls for
this floorplan.\nAll walls not explicitly targeted will have a random wall material
assigned by room type.",
    "type": "object",
    "properties": {
        "MaterialSets": {
            "title": "Materialsets",
            "default": [],
            "type": "array",
            "items": {
                "$ref": "#/definitions/InteriorMaterialSet"
            },
            "minItems": 0,
            "maxItems": 6
        }
    },
    "additionalProperties": false
},
"ModelTypeList": {
    "title": "ModelTypeList",
    "description": "The set of all models matching any of the listed model types.\nAn
empty set means zero models to sample/select.",
    "type": "object",
    "properties": {
        "ModelTypes": {
            "title": "Modeltypes",
            "type": "array",
            "items": {
                "type": "string"
            },
            "minItems": 0
        }
    },
    "required": [
        "ModelTypes"
    ],
    "additionalProperties": false
}

```

```
  },
  "FurnitureArrangementSet": {
    "title": "FurnitureArrangementSet",
    "description": "Describes the interior template for placing furniture in one
or more rooms.\n\n- TargetSet is the set of rooms to furnish, filter by room name or
room\n type.\n- SampleSet is a set of all furnishing models to randomly choose and\n
place. \n- DesiredSpatialDensity is the desired level of free space after placing\n
furniture.",
    "type": "object",
    "properties": {
      "Name": {
        "title": "Name",
        "type": "string"
      },
    },
    "TargetSet": {
      "title": "Targetset",
      "anyOf": [
        {
          "$ref": "#/definitions/RoomNameList"
        },
        {
          "$ref": "#/definitions/RoomTypeList"
        }
      ]
    },
    "SampleSet": {
      "$ref": "#/definitions/ModelTypeList"
    },
    "DesiredSpatialDensity": {
      "title": "Desiredspatialdensity",
      "default": "Moderate",
      "enum": [
        "Sparse",
        "Moderate",
        "Dense"
      ],
      "type": "string"
    }
  },
  "required": [
    "Name",
    "TargetSet"
  ],
  "additionalProperties": false
}
```

```
  },
  "InteriorFurnishings": {
    "title": "InteriorFurnishings",
    "description": "Describes the types of furniture models for randomly placing into
each room\nin the world. Rooms are targeted by room type or room name. Rooms that
are\nnot targeted are furnished at random by their room type with moderate density.
\nndensity. For an empty room, specify an empty sample set.",
    "type": "object",
    "properties": {
      "FurnitureArrangements": {
        "title": "Furniturearrangements",
        "default": [],
        "type": "array",
        "items": {
          "$ref": "#/definitions/FurnitureArrangementSet"
        },
        "minItems": 0,
        "maxItems": 6
      }
    },
    "additionalProperties": false
  },
  "InteriorTemplate": {
    "title": "InteriorTemplate",
    "description": "Top-level template for parameterizing the interior finishes and
furnishings for\nthis floorplan.",
    "type": "object",
    "properties": {
      "Flooring": {
        "title": "Flooring",
        "default": {
          "MaterialSets": []
        },
        "allOf": [
          {
            "$ref": "#/definitions/InteriorFlooring"
          }
        ]
      },
      "Walls": {
        "title": "Walls",
        "default": {
          "MaterialSets": []
        }
      }
    }
  }
}
```

```

    "allOf": [
      {
        "$ref": "#/definitions/InteriorWalls"
      }
    ]
  },
  "Furniture": {
    "title": "Furniture",
    "default": {
      "FurnitureArrangements": []
    },
    "allOf": [
      {
        "$ref": "#/definitions/InteriorFurnishings"
      }
    ]
  },
  "additionalProperties": false
},
"FloorTemplate": {
  "title": "FloorTemplate",
  "description": "Describes a single floor within a building. Defaults to a single residential room\nof a random type and size, and the interior is randomly furnished.",
  "type": "object",
  "properties": {
    "Floorplan": {
      "title": "Floorplan",
      "default": {
        "Footprint": {
          "DesiredAspectRatio": {
            "x": 1.0,
            "y": 1.0
          }
        }
      },
      "Ceiling": {
        "Height": 3.0
      },
      "Rooms": [
        {
          "Type": "Living",
          "Name": "My Living Room",
          "DesiredShape": {

```

```

        "Area": 20.0,
        "AspectRatio": {
            "x": 1.0,
            "y": 1.0
        }
    },
    {
        "Type": "Bedroom",
        "Name": "My Bedroom",
        "DesiredShape": {
            "Area": 20.0,
            "AspectRatio": {
                "x": 1.0,
                "y": 1.0
            }
        }
    }
],
"DesiredConnections": [],
},
"allOf": [
    {
        "$ref": "#/definitions/FloorplanTemplate"
    }
]
},
"Interior": {
    "title": "Interior",
    "default": {
        "Flooring": {
            "MaterialSets": []
        },
        "Walls": {
            "MaterialSets": []
        },
        "Furniture": {
            "FurnitureArrangements": []
        }
    }
},
"allOf": [
    {
        "$ref": "#/definitions/InteriorTemplate"
    }
]

```

```
    ]
  }
},
"additionalProperties": false
},
"BuildingTemplate": {
  "title": "BuildingTemplate",
  "description": "Describes a building to be randomly generated. Defaults to one residential floor.",
  "type": "object",
  "properties": {
    "Floors": {
      "title": "Floors",
      "default": [
        {
          "Floorplan": {
            "Footprint": {
              "DesiredAspectRatio": {
                "x": 1.0,
                "y": 1.0
              }
            },
            "Ceiling": {
              "Height": 3.0
            },
            "Rooms": [
              {
                "Type": "Living",
                "Name": "My Living Room",
                "DesiredShape": {
                  "Area": 20.0,
                  "AspectRatio": {
                    "x": 1.0,
                    "y": 1.0
                  }
                }
              },
              {
                "Type": "Bedroom",
                "Name": "My Bedroom",
                "DesiredShape": {
                  "Area": 20.0,
                  "AspectRatio": {
                    "x": 1.0,
```

```

        "y": 1.0
      }
    }
  ],
  "DesiredConnections": [],
},
"Interior": {
  "Flooring": {
    "MaterialSets": []
  },
  "Walls": {
    "MaterialSets": []
  },
  "Furniture": {
    "FurnitureArrangements": []
  }
}
],
"type": "array",
"items": {
  "$ref": "#/definitions/FloorTemplate"
},
"minItems": 1,
"maxItems": 1
}
},
"additionalProperties": false
}
}
}

```

## JSON의 샘플 월드 템플릿

templateBody(시뮬레이션 월드 템플릿 본문)는 [CreateWorldTemplate](#) API의 입력 파라미터입니다. 이 파라미터는 JSON 형식의 문자열입니다. JSON은 시뮬레이션 월드 템플릿을 지정하며 Simulation WorldForge가 월드를 생성하는 데 사용하는 파라미터를 포함합니다.

이 섹션에는 샘플 시뮬레이션 월드 템플릿 본문이 포함되어 있습니다.

주제

- [침실이 하나인 집](#)
- [방 한 개](#)
- [방 두 개](#)

## 침실이 하나인 집

다음 예시에서는 침실이 하나인 집을 지정합니다. 인테리어 자재와 가구를 지정합니다.

```
{
  "name": "OneBedroomHouse",
  "templateBody": {
    "Version": "2",
    "Buildings": [
      {
        "Floors": [
          {
            "Floorplan": {
              "Footprint": {
                "DesiredAspectRatio": {
                  "x": 1,
                  "y": 1
                }
              },
              "Ceiling": {
                "Height": 3
              },
              "Rooms": [
                {
                  "Type": "Bedroom",
                  "Name": "Bedroom",
                  "DesiredShape": {
                    "Area": 25,
                    "AspectRatio": {
                      "x": 1,
                      "y": 1.2
                    }
                  }
                },
                {
                  "Type": "Living",
                  "Name": "Living room",
                  "DesiredShape": {
```

```
        "Area": 30,
        "AspectRatio": {
            "x": 1,
            "y": 1.5
        }
    },
    {
        "Type": "Bathroom",
        "Name": "Bathroom",
        "DesiredShape": {
            "Area": 10,
            "AspectRatio": {
                "x": 1,
                "y": 1.5
            }
        }
    },
    {
        "Type": "Kitchen",
        "Name": "Kitchen",
        "DesiredShape": {
            "Area": 15,
            "AspectRatio": {
                "x": 1.5,
                "y": 1
            }
        }
    }
],
"DesiredConnections": [
    {
        "Location": [
            "Bathroom",
            "Living room"
        ],
        "ConnectionType": "Doorway"
    },
    {
        "Location": [
            "Living room",
            "Kitchen"
        ],
        "ConnectionType": "Opening"
    }
]
```

```
    },
    {
      "Location": [
        "Bedroom",
        "Living room"
      ],
      "ConnectionType": "Doorway"
    }
  ]
},
"Interior": {
  "Flooring": {
    "MaterialSets": [
      {
        "Name": "Floorboard room types",
        "TargetSet": {
          "RoomTypes": [
            "Kitchen"
          ]
        },
        "SampleSet": {
          "MaterialTypes": [
            "Floorboards"
          ]
        }
      },
      {
        "Name": "Carpet room types",
        "TargetSet": {
          "RoomTypes": [
            "Living",
            "Bedroom"
          ]
        },
        "SampleSet": {
          "MaterialTypes": [
            "Carpet"
          ]
        }
      }
    ],
    "Bathroom": {
      "Name": "Bathroom",
      "TargetSet": {
        "RoomNames": [
```

```

        "Bathroom"
      ]
    },
    "SampleSet": {
      "MaterialTypes": [
        "Parquetry"
      ]
    }
  ]
},
"Walls": {
  "MaterialSets": [
    {
      "Name": "Brick room types",
      "TargetSet": {
        "RoomTypes": [
          "Living"
        ]
      },
      "SampleSet": {
        "MaterialTypes": [
          "Brick"
        ]
      }
    },
    {
      "Name": "Tiles room types",
      "TargetSet": {
        "RoomTypes": [
          "Bathroom"
        ]
      },
      "SampleSet": {
        "MaterialTypes": [
          "Tiles"
        ]
      }
    }
  ]
},
"Furniture": {
  "FurnitureArrangements": [
    {

```



```
        "Type": "Bedroom",
        "Name": "Bedroom",
        "DesiredShape": {
            "Area": 40,
            "AspectRatio": {
                "x": 1,
                "y": 1.61
            }
        }
    ],
    "DesiredConnections": [],
},
"Interior": {
    "Furniture": {
        "FurnitureArrangements": [
            {
                "Name": "Bedroom furniture",
                "TargetSet": {
                    "RoomNames": [
                        "Bedroom"
                    ]
                },
                "DesiredSpatialDensity": "Dense"
            }
        ]
    }
}
]
```

## 방 두 개

다음 예시에서는 침실이 하나인 집을 지정합니다. Simulation WorldForge는 바닥재, 벽재, 가구 배치 및 연결을 포함한 세부 사항을 결정합니다.

```
{
  "name": "TwoRooms",
  "templateBody": {
    "Version": "2",
```

```
"Buildings": [  
  {  
    "Floors": [  
      {  
        "Floorplan": {  
          "Footprint": {  
            "DesiredAspectRatio": {  
              "x": 1,  
              "y": 1  
            }  
          },  
          "Ceiling": {  
            "Height": 3  
          },  
          "Rooms": [  
            {  
              "Type": "Living",  
              "Name": "Living room",  
              "DesiredShape": {  
                "Area": 30,  
                "AspectRatio": {  
                  "x": 1,  
                  "y": 1.5  
                }  
              }  
            },  
            {  
              "Type": "Dining",  
              "Name": "Dining room",  
              "DesiredShape": {  
                "Area": 30,  
                "AspectRatio": {  
                  "x": 1,  
                  "y": 1.5  
                }  
              }  
            }  
          ],  
          "DesiredConnections": []  
        },  
        "Interior": {}  
      }  
    ]  
  }  
]
```

```

    ]
  }
}

```

## 시뮬레이션 월드 템플릿 관리

이 섹션에서는 시뮬레이션 월드 템플릿을 생성하고 관리하는 방법에 대해 설명합니다. 시뮬레이션 월드 템플릿을 사용하여 Simulation WorldForge에서 월드를 생성하는 방법을 지정합니다. 방 수, 방 연결 방식, 가구, 인테리어 요소에 사용되는 자재 유형 등을 지정할 수 있습니다.

시뮬레이션 월드 템플릿에 대해 자세히 알아보려면 [시뮬레이션 월드 템플릿 알아보기](#) 섹션부터 시작하세요. 시뮬레이션 월드 템플릿을 설명하는 JSON `templateBody`을 검토할 수도 있습니다. 자세한 내용은 [시뮬레이션 월드 템플릿 본문을 위한 JSON 스키마](#) 섹션을 참조하세요.

### 주제

- [시뮬레이션 월드 템플릿 생성](#)
- [시뮬레이션 월드 템플릿 보기](#)
- [시뮬레이션 월드 템플릿 수정](#)
- [시뮬레이션 월드 템플릿 삭제](#)
- [시뮬레이션 월드 템플릿 버전, 기능 및 변경 사항](#)

## 시뮬레이션 월드 템플릿 생성

시뮬레이션 월드 템플릿을 생성하여 Simulation WorldForge에서 월드를 생성하는 방법을 지정합니다. 시뮬레이션 월드 템플릿이 완성되면 월드 생성 작업을 만들어 방 및 인테리어 구성이 서로 다른 월드를 생성합니다.

샘플 템플릿, 저장된 템플릿을 사용하거나 처음부터 시뮬레이션 월드 템플릿을 만들 수 있습니다. 템플릿을 생성한 후 평면도, 인테리어 및 기타 세부 사항을 수정할 수 있습니다. 시뮬레이션 월드 템플릿 수정에 대한 자세한 정보는 [시뮬레이션 월드 템플릿 수정](#) 섹션을 참조하세요.

### 시뮬레이션 월드 템플릿을 만들려면

다음 탭 중 하나에 있는 단계를 따릅니다.

#### Using the console

## 시뮬레이션 월드 템플릿을 만들려면

1. <https://console.aws.amazon.com/robomaker/>에서 AWS RoboMaker 콘솔에 로그인합니다.
2. AWS RoboMaker 콘솔에서 왼쪽의 Simulation WorldForge를 펼친 다음 월드 템플릿을 선택합니다.
3. 월드 템플릿 페이지에서 템플릿 생성을 선택합니다.
4. 월드 템플릿 생성 페이지에서 템플릿 중 하나를 선택합니다. 사전 구성된 샘플 템플릿 중 하나를 선택하거나, 저장된 템플릿을 복제 및 수정하거나, 기본 월드로 처음부터 시작할 수 있습니다.
5. 템플릿 세부 정보 페이지의 왼쪽 상단에서 이름 바꾸기를 선택한 다음 템플릿 이름을 지정합니다.
6. (선택 사항) 평면도 및 인테리어 세부 정보를 사용자 지정합니다. 자세한 내용은 [시뮬레이션 월드 템플릿 알아보기](#) 섹션을 참조하세요.
7. 템플릿 세부 정보 페이지에서 저장 후 종료를 선택합니다.

## Using the AWS CLI

### Example

AWS CLI를 사용하여 시뮬레이션 월드 템플릿을 업데이트할 수 있습니다. 먼저 Simulation WorldForge가 생성하는 월드를 지정하는 JSON 문서를 생성합니다. 다음으로, `create-world-template`을 사용하여 시뮬레이션 월드 템플릿을 생성합니다.

예를 들어 다음 JSON 문서는 침실이 하나인 집을 지정합니다.

```
{
  "title": "WorldTemplate",
  "description": "The top-level template for parameterizing a randomly generated world. By default, a single\nresidential building with one floor and one room is generated.",
  "type": "object",
  "properties": {
    "Version": {
      "title": "Version",
      "default": "1",
      "type": "string"
    },
    "Buildings": {
```

```
"title": "Buildings",
"default": [
  {
    "Floors": [
      {
        "Floorplan": {
          "Footprint": {
            "DesiredAspectRatio": {
              "x": 1.0,
              "y": 1.0
            }
          },
          "Ceiling": {
            "Height": 3.0
          },
          "Rooms": [
            {
              "Type": "Living",
              "Name": "My Living Room",
              "DesiredShape": {
                "Area": 20.0,
                "AspectRatio": {
                  "x": 1.0,
                  "y": 1.0
                }
              }
            }
          ],
          "DesiredConnections": []
        },
        "Interior": {
          "Flooring": {
            "MaterialSets": []
          },
          "Walls": {
            "MaterialSets": []
          },
          "Furniture": {
            "FurnitureArrangements": []
          }
        }
      }
    ]
  }
]
```

```
    ],
    "type": "array",
    "items": {
      "$ref": "#/definitions/BuildingTemplate"
    },
    "minItems": 1,
    "maxItems": 1
  }
},
"additionalProperties": false,
"definitions": {
  "AspectRatio": {
    "title": "AspectRatio",
    "type": "object",
    "properties": {
      "x": {
        "title": "X",
        "default": 1,
        "minimum": 1,
        "maximum": 4,
        "type": "number"
      },
      "y": {
        "title": "Y",
        "default": 1,
        "minimum": 1,
        "maximum": 4,
        "type": "number"
      }
    }
  },
  "FloorplanFootprint": {
    "title": "FloorplanFootprint",
    "description": "The desired footprint of this floorplan.",
    "type": "object",
    "properties": {
      "DesiredAspectRatio": {
        "title": "Desiredaspectratio",
        "default": {
          "x": 1.0,
          "y": 1.0
        },
        "allOf": [
```

```
        {
          "$ref": "#/definitions/AspectRatio"
        }
      ]
    }
  },
  "additionalProperties": false
},
"FloorplanCeiling": {
  "title": "FloorplanCeiling",
  "description": "The height of the ceiling for this floorplan in metres.",
  "type": "object",
  "properties": {
    "Height": {
      "title": "Height",
      "default": 3.0,
      "type": "number",
      "minimum": 2.4,
      "maximum": 4.0
    }
  },
  "additionalProperties": false
},
"Rectangle": {
  "title": "Rectangle",
  "description": "A rectangle defined by area in square metres and aspect
ratio.",
  "type": "object",
  "properties": {
    "Area": {
      "title": "Area",
      "type": "number"
    },
    "AspectRatio": {
      "$ref": "#/definitions/AspectRatio"
    }
  },
  "required": [
    "Area",
    "AspectRatio"
  ],
  "additionalProperties": false
},
"FloorplanRoom": {
```

```
"title": "FloorplanRoom",
"description": "A description for single room for this floorplan.",
"type": "object",
"properties": {
  "Type": {
    "title": "Type",
    "enum": [
      "Bedroom",
      "Bathroom",
      "Living",
      "Dining",
      "Kitchen",
      "Hallway",
      "Closet"
    ],
    "type": "string"
  },
  "Name": {
    "title": "Name",
    "maxLength": 255,
    "minLength": 1,
    "pattern": "^[a-zA-Z0-9_\\- ]*$",
    "type": "string"
  },
  "DesiredShape": {
    "title": "Desiredshape",
    "default": {
      "Area": 20.0,
      "AspectRatio": {
        "x": 1.0,
        "y": 1.0
      }
    },
    "allOf": [
      {
        "$ref": "#/definitions/Rectangle"
      }
    ]
  },
  "required": [
    "Type",
    "Name"
  ],
}
```

```

    "additionalProperties": false
  },
  "FloorplanConnection": {
    "title": "FloorplanConnection",
    "description": "Describes the desired layout of the rooms and their adjacent rooms. A connection can be either a doorway or \nan open space without any walls. Two rooms cannot both share an interior doorway and an opening. \nThe same two rooms can have multiple doorways, up to a limit.",
    "type": "object",
    "properties": {
      "Location": {
        "title": "Location",
        "type": "array",
        "items": {
          "type": "string"
        },
        "minItems": 2,
        "maxItems": 2
      },
      "ConnectionType": {
        "title": "Connectiontype",
        "enum": [
          "Doorway",
          "Opening"
        ],
        "type": "string"
      }
    },
    "required": [
      "Location",
      "ConnectionType"
    ],
    "additionalProperties": false
  },
  "FloorplanTemplate": {
    "title": "FloorplanTemplate",
    "description": "The top-level floorplan template that parameterizes the randomly generated \narchitectural layout. By default, a residential floorplan with bedroom and \nliving room are generated with a random doorway or opening connection. \n\nThe footprint contributes to the overall shape of the floor layout along\nwith rooms. The footprint shape is desired as it is a preference and not \nguaranteed.\n\nThe ceiling determines the height of the walls. There are minimum and\nmaximum ceiling heights. The ceiling height is guaranteed.\n\nRooms are required. Each room has a desired shape. Together, the room\nshapes and footprint

```

determine floor layout. The room types contribute to the layout and are used when randomly selecting furniture and materials for the walls and floors. DesiredConnections are optional. Two rooms are connected if they share a wall and doorway or adjacent without any wall aka "opening". All rooms are guaranteed to be connected randomly if they are not specified in the connections list. Connections that are specified are not guaranteed but will be attempted as best-effort.",

```

"type": "object",
"properties": {
  "Footprint": {
    "title": "Footprint",
    "default": {
      "DesiredAspectRatio": {
        "x": 1.0,
        "y": 1.0
      }
    },
    "allOf": [
      {
        "$ref": "#/definitions/FloorplanFootprint"
      }
    ]
  },
  "Ceiling": {
    "title": "Ceiling",
    "default": {
      "Height": 3.0
    },
    "allOf": [
      {
        "$ref": "#/definitions/FloorplanCeiling"
      }
    ]
  },
  "Rooms": {
    "title": "Rooms",
    "default": [
      {
        "Type": "Living",
        "Name": "My Living Room",
        "DesiredShape": {
          "Area": 20.0,
          "AspectRatio": {
            "x": 1.0,

```

```
        "y": 1.0
      }
    }
  ],
  "type": "array",
  "items": {
    "$ref": "#/definitions/FloorplanRoom"
  },
  "minItems": 1,
  "maxItems": 6
},
"DesiredConnections": {
  "title": "Desiredconnections",
  "default": [],
  "type": "array",
  "items": {
    "$ref": "#/definitions/FloorplanConnection"
  },
  "minItems": 0,
  "maxItems": 12
}
},
"additionalProperties": false
},
"RoomNameList": {
  "title": "RoomNameList",
  "description": "The set of all rooms matching any of the listed room names.",
  "type": "object",
  "properties": {
    "RoomNames": {
      "title": "Roomnames",
      "type": "array",
      "items": {
        "type": "string"
      },
      "minItems": 1,
      "maxItems": 6
    }
  },
  "required": [
    "RoomNames"
  ],
  "additionalProperties": false
}
```

```
},
"RoomTypeList": {
  "title": "RoomTypeList",
  "description": "The set of all rooms matching any of the listed room types.",
  "type": "object",
  "properties": {
    "RoomTypes": {
      "title": "Roomtypes",
      "type": "array",
      "items": {
        "enum": [
          "Bedroom",
          "Bathroom",
          "Living",
          "Dining",
          "Kitchen",
          "Hallway",
          "Closet"
        ],
        "type": "string"
      },
      "minItems": 1,
      "maxItems": 7
    }
  },
  "required": [
    "RoomTypes"
  ],
  "additionalProperties": false
},
"MaterialSetByMaterialType": {
  "title": "MaterialSetByMaterialType",
  "description": "The set of materials that match any of the material types listed. An empty\nset is invalid since all targets require materials.",
  "type": "object",
  "properties": {
    "MaterialTypes": {
      "title": "Materialtypes",
      "type": "array",
      "items": {
        "type": "string"
      },
      "minItems": 1
    }
  }
}
```

```

    },
    "required": [
      "MaterialTypes"
    ],
    "additionalProperties": false
  },
  "InteriorMaterialSet": {
    "title": "InteriorMaterialSet",
    "description": "A set of sample materials to randomly assign to a set of interior target elements.\n\nThe target set determines *what rooms* receive the materials in the sample\nset. The targets in a room are the walls and flooring. Rooms may be targeted\nby room type or room name.\n\nThe sample set determines *what materials* to randomly select for the\ntarget rooms' walls and floors.\n\nThe sample set is optional and when not specified (null) materials are\nrandomly selected according to the room type for each room in the target\nset.\n\nA sample set with an empty material set is invalid since all wall\nand flooring targets require materials.",
    "type": "object",
    "properties": {
      "Name": {
        "title": "Name",
        "maxLength": 255,
        "minLength": 1,
        "pattern": "^[a-zA-Z0-9_\\- ]*$",
        "type": "string"
      },
      "TargetSet": {
        "title": "Targetset",
        "anyOf": [
          {
            "$ref": "#/definitions/RoomNameList"
          },
          {
            "$ref": "#/definitions/RoomTypeList"
          }
        ]
      },
      "SampleSet": {
        "$ref": "#/definitions/MaterialSetByMaterialType"
      }
    }
  },
  "required": [
    "Name",
    "TargetSet"
  ]

```

```
    ],
    "additionalProperties": false
  },
  "InteriorFlooring": {
    "title": "InteriorFlooring",
    "description": "Describes the interior template parameters for all floors
for this floorplan.\nAll floors not explicitly targeted will have a random floor
material assigned by room type.",
    "type": "object",
    "properties": {
      "MaterialSets": {
        "title": "Materialsets",
        "default": [],
        "type": "array",
        "items": {
          "$ref": "#/definitions/InteriorMaterialSet"
        },
        "minItems": 0,
        "maxItems": 6
      }
    },
    "additionalProperties": false
  },
  "InteriorWalls": {
    "title": "InteriorWalls",
    "description": "Describes the interior template parameters for all walls for
this floorplan.\nAll walls not explicitly targeted will have a random wall material
assigned by room type.",
    "type": "object",
    "properties": {
      "MaterialSets": {
        "title": "Materialsets",
        "default": [],
        "type": "array",
        "items": {
          "$ref": "#/definitions/InteriorMaterialSet"
        },
        "minItems": 0,
        "maxItems": 6
      }
    },
    "additionalProperties": false
  },
  "ModelTypeList": {
```

```
"title": "ModelTypeList",
"description": "The set of all models matching any of the listed model types.
\nAn empty set means zero models to sample/select.",
"type": "object",
"properties": {
  "ModelTypes": {
    "title": "Modeltypes",
    "type": "array",
    "items": {
      "enum": [
        "Baths",
        "BarCabinets",
        "Beds",
        "Bookcases",
        "CoffeeTables",
        "ConsoleTables",
        "CornerCabinets",
        "DeskChairs",
        "Desks",
        "DiningChairs",
        "DiningTables",
        "DishWashers",
        "Dressers",
        "EndAndSideTables",
        "FloorLamps",
        "Fridges",
        "LivingRoomChairs",
        "KitchenIslandsAndCarts",
        "MediaStorage",
        "Nightstands",
        "Ottomans",
        "Ovens",
        "ServingCarts",
        "Showers",
        "SideboardsAndBuffets",
        "Sofas",
        "Storage",
        "StorageBenches",
        "Toilets",
        "VanityCounters",
        "WashingMachinesAndDryers"
      ],
      "type": "string"
    }
  },
}
```

```

        "minItems": 0
      }
    },
    "required": [
      "ModelTypes"
    ],
    "additionalProperties": false
  },
  "FurnitureArrangementSet": {
    "title": "FurnitureArrangementSet",
    "description": "Describes the interior template for placing furniture in one or more rooms.\n\n- TargetSet is the set of rooms to furnish, filter by room name or room\n type.\n- SampleSet is a set of all furnishing models to randomly choose and\n place.\n- DesiredSpatialDensity is the desired level of free space after placing\n furniture.",
    "type": "object",
    "properties": {
      "Name": {
        "title": "Name",
        "maxLength": 255,
        "minLength": 1,
        "pattern": "^[a-zA-Z0-9_\\- ]*$",
        "type": "string"
      },
      "TargetSet": {
        "title": "Targetset",
        "anyOf": [
          {
            "$ref": "#/definitions/RoomNameList"
          },
          {
            "$ref": "#/definitions/RoomTypeList"
          }
        ]
      },
      "SampleSet": {
        "$ref": "#/definitions/ModelTypeList"
      },
      "DesiredSpatialDensity": {
        "title": "Desiredspatialdensity",
        "default": "Moderate",
        "enum": [
          "Sparse",
          "Moderate",

```

```

        "Dense"
      ],
      "type": "string"
    }
  },
  "required": [
    "Name",
    "TargetSet"
  ],
  "additionalProperties": false
},
"InteriorFurnishings": {
  "title": "InteriorFurnishings",
  "description": "Describes the types of furniture models for randomly placing
into each room\n\nin the world. Rooms are targeted by room type or room name. Rooms
that are\n\nnot targeted are furnished at random by their room type with moderate
density.\n\ndensity. For an empty room, specify an empty sample set.",
  "type": "object",
  "properties": {
    "FurnitureArrangements": {
      "title": "Furniturearrangements",
      "default": [],
      "type": "array",
      "items": {
        "$ref": "#/definitions/FurnitureArrangementSet"
      },
      "minItems": 0,
      "maxItems": 6
    }
  },
  "additionalProperties": false
},
"InteriorTemplate": {
  "title": "InteriorTemplate",
  "description": "Top-level template for parameterizing the interior finishes
and furnishings for\n\nthis floorplan.",
  "type": "object",
  "properties": {
    "Flooring": {
      "title": "Flooring",
      "default": {
        "MaterialSets": []
      },
      "allOf": [

```



```
    }
  },
  "Ceiling": {
    "Height": 3.0
  },
  "Rooms": [
    {
      "Type": "Living",
      "Name": "My Living Room",
      "DesiredShape": {
        "Area": 20.0,
        "AspectRatio": {
          "x": 1.0,
          "y": 1.0
        }
      }
    }
  ],
  "DesiredConnections": [],
},
"allOf": [
  {
    "$ref": "#/definitions/FloorplanTemplate"
  }
]
},
"Interior": {
  "title": "Interior",
  "default": {
    "Flooring": {
      "MaterialSets": []
    },
    "Walls": {
      "MaterialSets": []
    },
    "Furniture": {
      "FurnitureArrangements": []
    }
  }
},
"allOf": [
  {
    "$ref": "#/definitions/InteriorTemplate"
  }
]
]
```

```
    }
  },
  "additionalProperties": false
},
"BuildingTemplate": {
  "title": "BuildingTemplate",
  "description": "Describes a building to be randomly generated. Defaults to one residential floor.",
  "type": "object",
  "properties": {
    "Floors": {
      "title": "Floors",
      "default": [
        {
          "Floorplan": {
            "Footprint": {
              "DesiredAspectRatio": {
                "x": 1.0,
                "y": 1.0
              }
            },
            "Ceiling": {
              "Height": 3.0
            },
            "Rooms": [
              {
                "Type": "Living",
                "Name": "My Living Room",
                "DesiredShape": {
                  "Area": 20.0,
                  "AspectRatio": {
                    "x": 1.0,
                    "y": 1.0
                  }
                }
              }
            ],
            "DesiredConnections": []
          },
          "Interior": {
            "Flooring": {
              "MaterialSets": []
            },
            "Walls": {
```

```

        "MaterialSets": []
      },
      "Furniture": {
        "FurnitureArrangements": []
      }
    }
  ],
  "type": "array",
  "items": {
    "$ref": "#/definitions/FloorTemplate"
  },
  "minItems": 1,
  "maxItems": 1
}
},
"additionalProperties": false
}
}
}

```

JSON을 one-bedroom-house.json 파일에 저장하면 AWS CLI와 함께 사용하여 시뮬레이션 월드 템플릿을 만들 수 있습니다.

```
$ aws robomaker create-world-template --template my-simulation-world-template-arn --template-body file://one-bedroom-house.json
```

## 시뮬레이션 월드 템플릿 보기

시뮬레이션 월드 템플릿에 대한 세부 정보를 봅니다.

시뮬레이션 월드 템플릿의 세부 정보를 보려면

다음 탭 중 하나에 있는 단계를 따릅니다.

Using the console

1. <https://console.aws.amazon.com/robomaker/>에서 AWS RoboMaker 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 Simulation WorldForge를 선택한 다음 월드 템플릿을 선택합니다.

3. 시뮬레이션 월드 템플릿의 ID를 선택하면 평면도 및 인테리어를 포함한 세부 정보를 볼 수 있습니다. 세부 보기에서 월드를 생성할 수도 있습니다.

## Using the AWS CLI

### Example

다음 AWS CLI 예시에서는 `list-world-templates`을 사용하여 기존 템플릿을 나열한 다음 `describe-world-template` 및 `get-world-template-body`를 사용하여 시뮬레이션 월드 템플릿의 세부 정보를 확인합니다.

```
$ aws robomaker list-world-templates
$ aws robomaker describe-world-template --template my-simulation-world-template-arn
$ aws robomaker get-world-template-body --template my-simulation-world-template-arn
```

## 시뮬레이션 월드 템플릿 수정

평면도를 선택하여 평면도에서 방의 수와 유형, 방 상이의 연결을 사용자 지정할 수 있습니다. 인테리어를 선택하여 바닥, 벽, 가구를 사용자 지정할 수 있습니다.

시뮬레이션 월드 템플릿을 수정하려면

다음 탭 중 하나에 있는 단계를 따릅니다.

### Using the console

시뮬레이션 월드 템플릿을 수정하려면

1. <https://console.aws.amazon.com/robomaker/>에서 AWS RoboMaker 콘솔에 로그인합니다.
2. AWS RoboMaker 콘솔의 왼쪽 탐색 창에서 Simulation WorldForge를 펼친 다음 월드 템플릿을 선택합니다.
3. 월드 템플릿 페이지에서 수정하려는 시뮬레이션 월드 템플릿을 선택합니다.
4. 수정하려는 각 요소 옆에 있는 편집 또는 재정의의를 선택합니다. 시뮬레이션 월드 템플릿 구성 요소에 대한 자세한 내용은 [시뮬레이션 월드 템플릿 알아보기](#) 섹션을 참조하세요.

## Using the AWS CLI

### Example

다음 AWS CLI 예시에서는 `list-world-templates`을 사용하여 기존 템플릿을 나열한 다음 `describe-world-template`을 사용하여 시뮬레이션 월드 템플릿의 세부 정보를 확인하고 `get-world-template-body`를 사용하여 템플릿 본문 JSON을 검색하고 파일에 기록합니다.

```
$ aws robomaker list-world-templates
$ aws robomaker describe-world-template --template my-simulation-world-template-arn
$ aws robomaker get-world-template-body --template my-simulation-world-template-arn
  --output json > myTemplateBody.json
$ aws robomaker update-world-template-body --template my-simulation-world-template-arn
  --template-body file://myTemplateBody.json
```

## 시뮬레이션 월드 템플릿 삭제

시뮬레이션 월드 템플릿을 더 이상 사용할 필요가 없는 경우 삭제할 수 있습니다.

### Using the console

1. <https://console.aws.amazon.com/robomaker/>에서 AWS RoboMaker 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 Simulation WorldForge를 선택한 다음 월드 템플릿을 선택합니다.
3. 시뮬레이션 월드 템플릿의 ID를 선택하고, 템플릿 작업을 선택하고, 삭제를 선택한 다음, 대화 상자에서 삭제를 선택하여 삭제를 확인합니다.

### Using the AWS CLI

#### Example

다음 AWS CLI 예시에서는 `list-world-templates`을 사용하여 기존 템플릿을 나열한 다음 `delete-world-template`을 사용하여 시뮬레이션 월드 템플릿을 삭제합니다.

```
$ aws robomaker list-world-templates
$ aws robomaker delete-world-template --template my-simulation-world-template-arn
```

## 시뮬레이션 월드 템플릿 버전, 기능 및 변경 사항

AWS RoboMaker Simulation WorldForge는 새로운 버전의 월드 템플릿을 릴리스합니다. 이러한 템플릿의 새로운 기능과 개선 사항을 사용하여 사용 사례에 더 적합한 월드를 만들 수 있습니다.

월드 템플릿의 모든 기능을 사용하려면 월드 템플릿을 최신 버전으로 업그레이드하세요. 월드 템플릿의 최신 버전에는 이전 버전의 모든 기능이 포함되어 있습니다.

AWS RoboMaker 콘솔 또는 AWS CLI를 사용하여 월드 템플릿을 업데이트할 수 있습니다. AWS RoboMaker 콘솔을 사용하는 경우 템플릿을 업그레이드하는 데 사용할 수 있는 프롬프트가 표시됩니다.

API를 사용하여 월드 템플릿을 최신 버전으로 업그레이드하려면 월드 템플릿을 정의하는 JSON Version 필드를 최신 버전의 숫자 값으로 설정합니다. 예를 들어 버전 2가 최신 버전인 경우 월드 템플릿의 본문에 "Version": "2"를 지정할 수 있습니다. 최신 스키마를 보려면 [시뮬레이션 월드 템플릿 본문을 위한 JSON 스키마](#) 섹션을 참조하세요.

다음 설명은 월드 템플릿의 기능 및 업데이트에 대한 정보를 제공합니다. 최신 버전의 업데이트가 먼저 표시됩니다.

### 시뮬레이션 월드 템플릿 버전 2 릴리스

버전 2의 업데이트에는 다음이 포함됩니다.

- 월드에 여닫이 문을 추가할 수 있습니다.
- 모든 방에 구성을 적용할 수 있습니다.
- 월드를 설명하는 새로운 필드가 포함됩니다.
- 바닥 마찰 값이 변경됩니다.
- 버전에 구애받지 않는 업데이트입니다.

### 문

AWS RoboMaker Simulation WorldForge 템플릿 버전 2를 사용하여 여닫이 문이 있는 월드를 만들 수 있습니다.

이러한 문이 열려 있는 비율을 구성할 수 있습니다. 예를 들어 다음과 같은 열림 상태를 지정할 수 있습니다.

- 0% 열림 — 닫힘

- 50% 열림 — 반쯤 열림
- 70% 열림 — 대부분 열림
- 100% 열림 — 완전 열림

또한 Simulation WorldForge에서 열림 비율을 임의 상태로 설정하여 문 개방도를 무작위로 지정하도록 지정할 수 있습니다.

월드 템플릿의 Interior 섹션에서 월드에 표시하고 싶은 문을 구성할 수 있습니다. 월드 템플릿을 사용하여 문이 있는 방을 만드는 방법을 알아보려면 [출입구에 문 요청](#) 섹션을 참조하세요.

### 모든 방에 구성 적용

월드 템플릿의 Target.All 키워드를 사용하여 모든 방에 구성 변경을 적용할 수 있습니다. 모든 방에서 다음과 같은 사항을 변경할 수 있습니다.

- 바닥재
- 벽재
- 출입구
- 가구 배치

예를 들어 월드 템플릿에서 모든 문이 닫혀 있도록 지정하려면 문이 0%로 열리도록 지정하고 Target.All 키워드를 사용하여 해당 조건을 모든 문에 적용할 수 있습니다. 자세한 내용은 [모든 방에 구성 적용](#) 섹션을 참조하세요.

### 월드를 설명하는 새로운 필드

버전 2 템플릿으로 만든 월드는 world\_description.json 파일을 가집니다. 이 파일은 Gazebo WorldForge .world 파일과 동일한 디렉토리에 있습니다.

world\_description.json 파일에는 Simulation WorldForge 월드의 모든 문이 나열됩니다. [DescribeWorld](#) 작업을 사용하여 월드에 대한 설명을 볼 수 있습니다. 설명은 worldDescriptionBody 필드의 값입니다. 버전 1 템플릿으로 월드를 생성한 경우 필드 값은 비어 있습니다.

### 버전 2의 바닥 마찰 값 변경

버전 2에서는 바닥의 마찰 값이 Gazebo 지면과 동일합니다. 버전 1의 바닥 마찰 값은 변경되지 않습니다.

버전에 구매받지 않는 업데이트

모든 월드 템플릿의 경우 방 이름의 공백은 Gazebo 모델 이름에서 밑줄로 바뀝니다. 이 변경을 통해 모든 Simulation WorldForge Gazebo 모델에 ROS 토픽을 사용할 수 있게 되었습니다. ROS 토픽을 사용하여 모델에 대한 정보를 얻거나 모델을 변경할 수 있습니다.

## 월드 생성 작업 관리

월드 생성 작업을 사용하여 시뮬레이션 월드 템플릿에서 월드를 생성합니다. 월드 생성 작업을 생성할 때는 다양한 평면도와 인테리어 구성의 수를 지정합니다. 월드 생성 작업당 최대 50개의 월드를 생성할 수 있습니다.

주제

- [월드 생성 작업 만들기](#)
- [월드 생성 작업 보기](#)
- [월드 생성 작업 취소하기](#)

## 월드 생성 작업 만들기

월드 생성 작업을 만들어 다양한 방과 내부 구성으로 구성된 월드를 생성합니다. 각 월드 생성 작업은 최대 50개의 월드를 생성할 수 있습니다.

월드 생성 작업을 만들려면

다음 탭 중 하나에 있는 단계를 따릅니다.

Using the console

시뮬레이션 월드 템플릿을 만들려면

1. <https://console.aws.amazon.com/robomaker/>에서 AWS RoboMaker 콘솔에 로그인합니다.
2. AWS RoboMaker 콘솔에서 왼쪽의 Simulation WorldForge를 펼친 다음 월드 템플릿을 선택합니다.
3. 월드 템플릿 페이지에서 월드 생성에 사용할 시뮬레이션 월드 템플릿을 선택한 다음 월드 생성을 선택합니다.
4. 월드 생성 페이지에서 평면도 수를 지정합니다. 평면도 수에 평면도별 내부 변형 수를 곱한 값은 50을 초과할 수 없습니다.

5. 평면도별 내부 변형 수를 지정합니다. 평면도 수에 평면도별 내부 변형 수를 곱한 값은 50을 초과할 수 없습니다.
6. 선택 사항: 생성하는 모든 월드에 할당되는 월드 태그를 추가합니다.
7. 선택 사항: 생성 작업에 할당된 생성 작업 태그를 추가합니다. 이러한 태그는 생성한 월드에는 적용되지 않습니다.
8. 생성을 선택합니다.

월드 생성 세부 정보 페이지에서 월드 생성 작업의 진행 상황을 추적할 수 있습니다. 월드를 생성하는 데 필요한 시간은 시뮬레이션 월드 템플릿의 복잡성과 생성하는 월드 수에 따라 달라집니다.

## Using the AWS CLI

### Example

AWS CLI를 사용하여 시뮬레이션 월드 템플릿에서 월드를 생성할 수 있습니다. `create-world-generation-job`를 사용하여 월드 생성 작업을 생성합니다.

다음 AWS CLI 예제는 2개의 서로 다른 내부 평면도가 있는 2개의 평면도를 사용하여 4개의 월드를 생성하는 방법을 보여줍니다.

```
$ aws robomaker list-world-templates
$ aws robomaker create-world-generation-job --template my-simulation-world-template-arn --worldCount floorplanCount=2,interiorCountPerFloorplan=2
$ aws robomaker list-world-generation-jobs
$ aws robomaker describe-world-generation-job --job my-world-generation-job-arn
```

## 월드 생성 작업 보기

월드 세대 생성 진행 상황, 요약 정보 및 월드 생성 작업에 대한 기타 세부 정보를 볼 수 있습니다.

월드 생성 작업 세부 정보를 보려면

다음 탭 중 하나에 있는 단계를 따릅니다.

### Using the console

1. <https://console.aws.amazon.com/robomaker/>에서 AWS RoboMaker 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 Simulation WorldForge를 선택한 다음 월드 템플릿을 선택합니다.
3. 세부 정보를 보려면 월드 생성 작업의 Id를 선택합니다. 검색 창을 사용하여 생성 작업을 찾을 수 있습니다.

## Using the AWS CLI

### Example

다음 AWS CLI 예제에서는 `list-world-generation-jobs`를 사용하여 기존 월드 생성 작업을 나열한 다음 `describe-world-generation-job`를 사용하여 특정 월드 생성 작업의 세부 정보를 확인합니다.

```
$ aws robomaker list-world-generation-jobs
$ aws robomaker describe-world-generation-job --job my-world-generation-job-arn
```

## 월드 생성 작업 취소하기

진행 중인 월드 생성 작업을 취소할 수 있습니다.

월드 생성 작업을 취소하려면

다음 탭 중 하나에 있는 단계를 따릅니다.

### Using the console

1. <https://console.aws.amazon.com/robomaker/>에서 AWS RoboMaker 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 Simulation WorldForge를 선택한 다음 작업 만들기를 선택합니다.
3. 생성 작업 페이지에서 취소하려는 월드 생성 작업을 선택합니다.
4. 취소를 선택합니다. 생성 작업 취소 페이지에서 작업 취소를 선택하여 작업을 취소합니다.

### Using the AWS CLI

## Example

다음 AWS CLI 예제에서는 `list-world-generation-jobs`를 사용하여 기존 월드 생성 작업을 나열한 다음 `cancel-world-generation-job`를 사용하여 특정 월드 생성 작업을 취소합니다.

```
$ aws robomaker list-world-generation-jobs
$ aws robomaker cancel-world-generation-job --job my-world-generation-job-arn
```

## 월드 내보내기 작업 관리

Simulation WorldForge로 생성된 월드를 내보낼 수 있어 자신의 환경에서 사용할 수 있습니다. 월드는 Amazon S3 버킷으로 내보내집니다. .zip 파일에는 Gazebo 자산과 월드를 위한 ROS 작업 영역이 포함되어 있습니다.

### 주제

- [월드 내보내기 작업 생성](#)
- [월드 내보내기 작업 보기](#)

## 월드 내보내기 작업 생성

Amazon S3 버킷으로 내보낼 월드를 선택할 수 있습니다. 내보내기 작업을 위해 선택한 모든 월드는 하나의 .zip 파일에 있습니다.

### 월드 내보내기 작업 생성 방법

다음 탭 중 하나에 있는 단계를 따릅니다.

#### Using the console

내보내기 작업당 하나의 월드를 내보낼 수 있습니다.

시뮬레이션 월드 템플릿을 만들려면

1. <https://console.aws.amazon.com/robomaker/>에서 AWS RoboMaker 콘솔에 로그인합니다.
2. AWS RoboMaker 콘솔의 왼쪽 탐색 창에서 Simulation WorldForge를 확장한 다음 월드를 선택합니다.
3. 월드 페이지에서 내보내기 작업 생성을 선택합니다.

4. 내보내기 작업 생성 페이지에서 내보낼 월드를 선택합니다.
5. Amazon S3 버킷에 대한 PutObject, GetObject, AbortMultipartUpload 권한이 있는 IAM 역할을 선택합니다. 적절한 권한을 가진 역할을 생성하려면 생성을 선택합니다.
6. 월드 출력을 위한 S3 대상을 선택합니다. 페이지 하단에서 새 S3 버킷 생성을 선택하여 새 Amazon S3 버킷을 생성할 수도 있습니다.
7. 선택 사항: 내보내기 작업 생성 페이지에서 내보낸 월드에 할당되는 태그를 추가합니다.
8. 생성을 선택하여 월드 내보내기 작업을 생성합니다.

월드 내보내기 작업 세부 정보 페이지에서 내보내기 작업의 진행 상황을 추적할 수 있습니다. 작업을 생성하면 자동으로 해당 페이지로 이동합니다.

## Using the AWS CLI

### Example

AWS CLI를 사용하여 월드를 내보낼 수 있습니다. `create-world-export-job`를 사용하여 월드 내보내기 작업을 생성합니다. 내보내기 작업당 하나의 월드를 내보낼 수 있습니다.

다음 AWS CLI 예시에서는 월드를 내보내는 방법을 보여줍니다. 먼저 `list-worlds`를 사용하여 월드를 나열한 다음, `create-world-export-job`를 호출하여 월드 Amazon 리소스 이름(ARN)을 지정할 수 있습니다. `list-world-export-jobs`과 `describe-world-export-job`을 호출하여 상태를 확인할 수 있습니다.

```
aws robomaker list-worlds
aws robomaker create-world-export-job --worlds my-simulation-world-arn --iam-role
my-iam-role-arn --outputLocation s3Bucket=my-bucket,s3prefix=prefix
aws robomaker list-world-export-jobs
aws robomaker describe-world-export-job --job my-world-export-job-arn
```

## 월드 내보내기 작업 보기

월드 내보내기 작업의 상태와 기타 세부 정보를 확인합니다.

월드 내보내기 작업의 세부 정보를 보려면

다음 탭 중 하나에 있는 단계를 따릅니다.

## Using the console

1. <https://console.aws.amazon.com/robomaker/>에서 AWS RoboMaker 콘솔에 로그인합니다.
2. 왼쪽 탐색 창에서 Simulation WorldForge를 선택한 다음 작업 내보내기를 선택합니다.
3. 월드 내보내기 작업 ID를 선택하여 세부 정보를 확인합니다. 월드 내보내기 작업을 검색하고 취소할 수도 있습니다.

## Using the AWS CLI

### Example

다음 AWS CLI 예제에서는 `list-world-export-jobs`을 사용하여 기존 월드 수출 작업을 나열한 다음 `describe-world-export-job`을 사용하여 특정 월드 내보내기 작업의 세부 정보를 확인합니다.

```
aws robomaker list-world-export-jobs
aws robomaker describe-world-export-job --job my-world-export-job-arn
```

## 시뮬레이션에서 내보낸 월드 사용

Simulation WorldForge를 사용하여 AWS RoboMaker와 함께 사용할 수 있는 월드를 생성할 수 있습니다. 월드를 생성한 후에는 이를 내보내기하여 시뮬레이션에 사용해야 합니다. 시뮬레이션에 사용할 월드를 업로드할 수도 있습니다.

월드를 내보내기하면 다음을 사용할 수 있습니다.

- [기본 SDF 물리](#)와 다른 물리
- 특화된 조명
- 사용자 지정 모델

다음 섹션에서는 시뮬레이션에서 생성된 월드를 사용하는 방법에 대한 추가 정보를 제공합니다.

**⚠ Important**

AWS RoboMaker 요금이 청구되는 방식에 대해 자세히 알아보려면 [AWS RoboMaker 요금](#)을 참조하세요.

**섹션**

- [내보낸 월드를 데이터 소스로 사용](#)
- [ROS와 Gazebo에서 내보낸 월드 사용](#)
- [사용자 지정 물리, 조명, 모델이 포함된 내보낸 월드 사용](#)

## 내보낸 월드를 데이터 소스로 사용

Simulation WorldForge를 사용하면 ROS 환경에서 사용할 수 있는 월드를 내보낼 수 있습니다. 내보내려는 월드는 Amazon S3 버킷의 단일 .zip 파일로 복사됩니다. 이 섹션에서는 시작 파일을 조정한 다음 명령줄 인터페이스를 통해 AWS Management Console 또는 명령줄 인터페이스를 사용하여 시뮬레이션 작업을 생성하는 방법에 대한 지침을 제공하여 시뮬레이션 작업 시 Amazon S3 버킷에서 내보낸 월드를 사용하는 방법을 설명합니다.

AWS Management Console 또는 AWS CLI를 사용하여 데이터 소스를 추가하기 전에 먼저 시뮬레이션 애플리케이션 시작 파일을 업데이트해야 합니다.

시뮬레이션 시작 파일을 업데이트하려면:

1. 다음 시작 명령을 실행합니다.

```
<launch>
  <!-- Always set GUI to false for AWS RoboMaker Simulation
  Use gui:=true on roslaunch command-line to run with gzclient.
  -->
  <arg name="gui" default="false"/>

  <include file="$(find aws_robomaker_worldforge_worlds)/launch/
  launch_world.launch">
    <arg name="gui" value="$(arg gui)"/>
  </include>

  <!-- Your other launch commands go here. -->
</launch>
```

(0, 0, 0)에서 로봇을 생성할 수 있습니다. Simulation WorldForge가 생성하는 월드는 (0, 0, 0)에서 1미터 실린더의 투명도를 보장합니다.

2. 이미지를 다시 구축하고 평소처럼 푸시합니다. 자세한 내용은 [이미지를 사용하여 AWS RoboMaker 애플리케이션 개발](#) 섹션을 참조하세요.

데이터 소스를 추가하려면

다음 탭 중 하나에 있는 단계를 따릅니다.

Using the console

1. [월드 내보내기 작업 생성](#)에 나와 있는 절차를 따라 월드를 내보냅니다.
2. 시뮬레이션 작업을 생성하는 동안 새 데이터 소스를 추가합니다. 데이터 소스에 의미 있는 이름(예: 예시에 사용된 이름인 WorldForge)을 지정합니다.
3. 선택적으로 월드를 배치할 대상 디렉토리를 식별합니다.

 Note

AWS RoboMaker는 대상을 dataSource 파일로 덮어쓰므로 작업 영역 디렉토리를 대상으로 사용하지 마세요. 대신 작업 영역 아래에 대상으로 별도의 디렉토리를 지정할 수 있습니다(예: your\_workspace/src/aws\_exported\_world).

4. 유형으로 아카이브를 선택합니다. AWS RoboMaker는 대상 디렉토리에 월드의 압축을 풉니다.
5. Browse S3를 선택하고 월드에서 생성된 올바른 내보내기를 찾습니다.
6. 평소처럼 시뮬레이션 작업 생성을 계속합니다.

Using the AWS CLI

Example

[월드 내보내기 작업 생성](#)에 나와 있는 절차를 따라 월드를 내보냅니다.

작업 영역이 /home/simulation\_ws의 컨테이너에 있다고 가정하고, 다음 명령은 월드를 기본 대상 디렉토리로 추출합니다.

```
aws robomaker create-simulation-job \
  --max-job-duration-in-seconds <time> \
  --iam-role <IAM role ARN> \
  --data-sources '[{
    "name": "WorldForge",
    "type": "Archive",
    "destination": "/home/simulation_ws/src/aws_exported_world",
    "s3Bucket": "worldforge-test",
    "s3Keys": ["aws-robomaker-worldforge-export-q376mqk4z7gm.zip"]} ]' \
  --robot-applications <config> \
  --simulation-applications <config>
```

## ROS와 Gazebo에서 내보낸 월드 사용

이전 섹션에서 설명했듯이 Simulation WorldForge는 선택한 월드를 단일 .zip 파일로 내보냅니다. .zip 파일에는 ROS 및 Gazebo를 사용하여 월드를 수정하고 시각화하는 데 필요한 모든 에셋이 포함되어 있습니다. 여기에는 다음과 같은 중요한 폴더가 포함됩니다.

- workspace\_src 루트 폴더는 ROS 작업 영역입니다. 여기에는 공유 모델, 월드 데이터 및 기타 월드 정보가 포함됩니다. ROS 1 및 ROS 2와 호환됩니다.
- 공유 모델이 workspace\_src/src/aws\_robomaker\_worldforge\_shared\_models/models에 복사됩니다. 예를 들어, 같은 의자가 두 개 이상의 월드에서 사용되는 경우 공유 모델 폴더에 배치됩니다.
- 월드 데이터가 workspace\_src/src/aws\_robomaker\_worldforge\_worlds/worlds/에 복사됩니다.

시뮬레이션 시작 파일을 업데이트하려면:

1. [월드 내보내기 작업 생성](#)에 나와 있는 절차를 따라 월드를 내보냅니다.
2. ROS 작업 영역에 월드의 Unzip.

```
cd MyApplication/simulation_ws
unzip MyExportedWorld.zip
```

3. 월드를 구축합니다.

```
rosdep install --from-paths src --ignore-src -r -y$ colcon build
```

#### 4. 월드를 시작합니다.

```
source install/setup.sh
roslaunch aws_robomaker_worldforge_worlds launch_world.launch gui:=true
```

시뮬레이션 작업으로 월드를 구축하고 시작하려면:

1. [월드 내보내기 작업 생성](#)에 나와 있는 절차를 따라 월드를 내보냅니다.
2. dataSource를 사용하여 내보낸 월드를 작업 영역 소스 패키지 디렉토리 /home/simulation\_ws/src/aws\_exported\_world에 가져옵니다.
3. 시뮬레이션 애플리케이션의 LaunchConfig를 수정합니다.

```
"launchConfig": {
  "environmentVariables": {
    "ROS_IP": "ROBOMAKER_SIM_APP_IP",
    "ROS_MASTER_URI": "http://ROBOMAKER_ROBOT_APP_IP:11311",
    "GAZEBO_MASTER_URI": "http://ROBOMAKER_SIM_APP_IP:11345",
    "GAZEBO_MODEL_PATH": "@GAZEBO_MODEL_PATH:/home/
simulation_ws/src/aws_exported_world/aws_robomaker_worldforge_pkgs/
aws_robomaker_worldforge_shared_models/models"
  },
  "streamUI": true,
  "command": [
    "/bin/bash", "-c", "cd /home/simulation_ws && colcon build && source
install/setup.sh && roslaunch hello_world_simulation worldforge_world.launch"
  ]
},
```

## 사용자 지정 물리, 조명, 모델이 포함된 내보낸 월드 사용

시뮬레이션 시나리오에 사용자 지정이 필요한 경우 월드를 내보내고 수정할 수 있습니다. 예를 들어 사용자 지정 물리, 다양한 조명 효과를 적용하거나 사용자 지정 모델을 추가하거나 기타 수정을 수행할 수 있습니다.

월드 내보낸 후에는 내보낸 월드 모델을 포함하도록 .world 파일을 수정해야 합니다. .world 파일은 SDF를 사용합니다. SDF에 대한 자세한 내용은 [SDFFormat](#)을 참조하세요.

내보낸 월드 모델을 포함하도록 **.world** 파일을 수정하려면:

1. [월드 내보내기 작업 생성](#)에 나와 있는 절차를 따라 월드를 내보냅니다.
2. 다음 코드를 **.world** 파일에 복사합니다. 월드 이름이 내보낸 모델 이름과 일치하는지 확인합니다.

```
<sdf version="1.6">
  <world name="generation_82856b0yq33y_world_16">
    <model name="WorldForge World">
      <include>
        <uri>model://generation_82856b0yq33y_world_16</uri>
      </include>
    </model>
    <!-- Your other <world> elements go here -->
  </world>
</sdf>
```

3. 시작 파일에 수정된 **.world** 파일이 포함되어 있는지 확인합니다. 업데이트된 시작 파일을 사용하여 시뮬레이션을 시작합니다.

# 보안

이 섹션에서는 AWS RoboMaker의 다양한 측면을 보호하기 위한 지침을 제공합니다.

## 주제

- [AWS RoboMaker의 데이터 보호](#)
- [AWS RoboMaker에 대한 인증 및 액세스 제어](#)
- [AWS RoboMaker의 로깅 및 모니터링](#)
- [AWS RoboMaker 리소스 태깅](#)
- [보안 규정 준수](#)
- [AWS RoboMaker 복원성](#)
- [AWS RoboMaker의 인프라 보안](#)
- [AWS RoboMaker 및 인터페이스 VPC 엔드포인트\(AWS PrivateLink\)](#)

## AWS RoboMaker의 데이터 보호

AWS [공동 책임 모델](#)은 AWS RoboMaker의 데이터 보호에 적용됩니다. 이 모델이 설명하는 것처럼 AWS는 모든 AWS 클라우드를 실행하는 글로벌 인프라를 보호할 책임이 있습니다. 사용자는 인프라에서 호스팅되는 콘텐츠를 관리해야 합니다. 사용하는 AWS 서비스의 보안 구성과 관리 작업에 대한 책임도 사용자에게 있습니다. 데이터 프라이버시에 대한 자세한 내용은 [Data Privacy FAQ](#)(데이터 프라이버시 FAQ)를 참조하세요. 유럽의 데이터 보호에 대한 자세한 내용은 AWS 보안 블로그의 [AWS Shared Responsibility Model and GDPR](#) 블로그 게시물을 참조하세요.

데이터를 보호하려면 AWS 계정보안 인증 정보를 보호하고 AWS IAM Identity Center 또는 AWS Identity and Access Management(IAM)를 통해 개별 사용자 계정을 설정하는 것이 좋습니다. 이렇게 하면 개별 사용자에게 자신의 직무를 충실히 이행하는 데 필요한 권한만 부여됩니다. 또한 다음과 같은 방법으로 데이터를 보호하는 것이 좋습니다.

- 각 계정에 멀티 팩터 인증 설정(MFA)을 사용하세요.
- SSL/TLS를 사용하여 AWS 리소스와 통신하세요. TLS 1.2는 필수이며 TLS 1.3를 권장합니다.
- AWS CloudTrail로 API 및 사용자 활동 로깅을 설정하세요.
- AWS 암호화 솔루션을 AWS 서비스 내의 모든 기본 보안 컨트롤과 함께 사용합니다.
- Amazon S3에 저장된 민감한 데이터를 검색하고 보호하는 데 도움이 되는 Amazon Macie와 같은 고급 관리형 보안 서비스를 사용하세요.

- 명령줄 인터페이스 또는 API를 통해 AWS에 액세스할 때 FIPS 140-2 검증된 암호화 모듈이 필요한 경우 FIPS 엔드포인트를 사용합니다. 사용 가능한 FIPS 엔드포인트에 대한 자세한 내용은 [FIPS\(Federal Information Processing Standard\) 140-2](#)를 참조하세요.

고객의 이메일 주소와 같은 기밀 정보나 중요한 정보는 태그나 이름 필드와 같은 자유 양식 텍스트 필드에 입력하지 않는 것이 좋습니다. 여기에는 AWS RoboMaker 또는 기타 AWS 서비스에서 콘솔, API, AWS CLI 또는 AWS SDK를 사용하여 작업하는 경우가 포함됩니다. 이름에 사용되는 태그 또는 자유 형식 텍스트 필드에 입력하는 모든 데이터는 청구 또는 진단 로그에 사용될 수 있습니다. 외부 서버로 URL을 제공할 때 해당 서버에 대한 요청을 검증하기 위해 자격 증명 정보를 URL에 포함해서는 안 됩니다.

## AWS RoboMaker에 대한 인증 및 액세스 제어

AWS Identity and Access Management(IAM)은 관리자가 AWS RoboMaker 리소스에 대한 액세스를 안전하게 제어할 수 있도록 지원하는 AWS 서비스입니다. 관리자는 IAM을 사용하여 AWS RoboMaker 리소스를 사용할 수 있는 인증(로그인) 및 권한(권한 있음)을 가진 사용자를 제어합니다. IAM은 추가 비용 없이 AWS 계정에 제공되는 기능입니다.

### Important

빠르게 시작하려면 이 페이지에 나와 있는 기본 정보를 검토한 다음 [IAM 시작하기](#) 및 [정책이란 무엇입니까?](#) 섹션을 살펴보세요.

### 주제

- [권한 부여 및 액세스 제어 소개](#)
- [필요한 권한](#)
- [AWS RoboMaker에서 IAM을 사용하는 방식 이해](#)
- [인증 및 액세스 제어 문제 해결](#)

## 권한 부여 및 액세스 제어 소개

AWS RoboMaker은 AWS Identity and Access Management(IAM)과 통합되어 다음과 같은 다양한 기능을 제공합니다.

- AWS 계정에서 사용자 및 그룹을 생성합니다.

- AWS 계정의 사용자 간에 AWS 리소스를 쉽게 공유합니다.
- 각 사용자에게 고유한 보안 자격 증명을 할당합니다.
- 서비스 및 리소스에 대한 각 사용자의 액세스 권한을 제어합니다.
- AWS 계정에서 모든 사용자에 대해 단일 청구서를 받습니다.

IAM에 대한 자세한 내용은 다음을 참조하세요.

- [AWS Identity and Access Management\(IAM\)](#)
- [시작하기](#)
- [IAM 사용 설명서](#)

## 필요한 권한

AWS RoboMaker를 사용하거나 자신 또는 다른 사람에 대한 권한 부여 및 액세스 제어를 관리하려면 올바른 권한이 있어야 합니다.

### AWS RoboMaker 콘솔 사용에 필요한 권한

AWS RoboMaker 콘솔에 액세스하려면 AWS 계정의 AWS RoboMaker 리소스에 대한 세부 정보를 나열하고 볼 수 있는 최소 권한이 있어야 합니다. 최소 필수 권한보다 더 제한적인 자격 증명 기반 권한 정책을 만들면 콘솔이 해당 정책에 연결된 엔터티에 대해 의도대로 작동하지 않습니다.

AWS RoboMaker 콘솔에 대한 모든 읽기 전용 액세스 권한을 부여하려면 AWSRoboMakerReadOnlyAccess 정책을 사용합니다.

IAM 사용자가 시뮬레이션 작업을 생성하려는 경우 해당 사용자에게 iam:PassRole 권한을 부여해야 합니다. 역할 전달에 대한 자세한 내용은 [사용자에게 AWS 서비스에 역할을 전달할 수 있는 권한 부여](#)를 참조하세요.

예를 들어 다음 정책을 사용자에게 연결할 수 있습니다. 이 정책은 시뮬레이션 작업을 생성할 수 있는 권한을 제공합니다.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "iam:PassRole",
```

```

    "Resource": "arn:aws:iam::123456789012:role/S3AndCloudWatchAccess"
  }
]
}

```

AWS CLI 또는 AWS API만 호출하는 사용자에게 최소 콘솔 권한을 허용할 필요가 없습니다. 대신 수행하려는 API 작업을 일치시키는 권한만 있으면 됩니다.

## 콘솔에서 AWS RoboMaker의 월드를 보는 데 필요한 권한

사용자에게 다음 정책을 연결하여 AWS RoboMaker 콘솔에서 AWS RoboMaker 월드를 보는 데 필요한 권한을 부여할 수 있습니다.

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "robomaker: DescribeWorld"
      ],
      "Resource": "*",
      "Effect": "Allow"
    }
  ]
}

```

## AWS RoboMaker 시뮬레이션 도구 사용에 필요한 권한

시뮬레이션을 만드는 데 사용되는 IAM 사용자 또는 역할은 시뮬레이션 도구에 액세스할 수 있는 권한을 자동으로 갖게 됩니다. 다른 사용자 또는 역할인 경우, `robomaker:CreateSimulationJob` 권한이 있어야 합니다.

## 인증 관리에 필요한 권한

자신의 자격 증명(예: 암호, 액세스 키 및 멀티 팩터 인증(MFA) 디바이스)을 관리하려면 관리자가 필요한 권한을 부여해야 합니다. 이러한 권한이 포함된 정책을 보려면 [사용자가 자신의 자격 증명을 스스로 관리하도록 허용](#) 단원을 참조하세요.

IAM에서 사용자, 그룹, 역할 및 정책을 생성해 관리할 수 있도록 AWS 관리자에게는 IAM에 대한 모든 액세스 권한이 필요합니다. 모든 AWS에 대한 전체 액세스 권한이 포함된 [AdministratorAccess](#) AWS 관리형 정책을 사용해야 합니다. 이 정책은 AWS Billing and Cost Management 콘솔에 대한 액세스 권

한을 제공하지 않거나 루트 사용자 보안 인증을 필요로 하는 작업은 허용하지 않습니다. 자세한 내용은 AWS 일반 참조의 [AWS 계정 루트 사용자 보안 인증이 필요한 AWS 작업을 참조하세요](#).

### ⚠ Warning

관리자 사용자만 AWS에 대한 모든 액세스 권한을 가져야 합니다. 이 정책이 적용된 사용자는 누구나 AWS의 모든 리소스를 수정할 수 있는 권한 이외에 인증 및 액세스 제어를 완전히 관리할 수 있는 권한을 가지고 있습니다. 이 사용자를 생성하는 방법에 대한 자세한 내용은 [IAM Admin 사용자 생성](#) 단원을 참조하세요.

## 액세스 제어에 필요한 권한

관리자가 IAM 사용자 보안 인증을 제공한 경우 여러분이 액세스할 수 있는 리소스를 제어할 수 있도록 IAM 사용자에게 정책을 연결합니다. AWS Management Console에서 사용자에게 연결된 정책을 보려면 다음 권한이 있어야 합니다.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "ViewOwnUserInfo",
      "Effect": "Allow",
      "Action": [
        "iam:GetUserPolicy",
        "iam:ListGroupsForUser",
        "iam:ListAttachedUserPolicies",
        "iam:ListUserPolicies",
        "iam:GetUser"
      ],
      "Resource": [
        "arn:aws:iam::*:user/${aws:username}"
      ]
    },
    {
      "Sid": "ListUsersViewGroupsAndPolicies",
      "Effect": "Allow",
      "Action": [
        "iam:GetGroupPolicy",
        "iam:GetPolicyVersion",
        "iam:GetPolicy",

```

```

        "iam:ListAttachedGroupPolicies",
        "iam:ListGroupPolicies",
        "iam:ListPolicyVersions",
        "iam:ListPolicies",
        "iam:ListUsers"
    ],
    "Resource": "*"
}
]
}

```

추가 권한이 필요한 경우 관리자에게 필요한 작업에 액세스할 수 있도록 정책을 업데이트해 달라고 요청하세요.

## 시뮬레이션 작업에 필요한 권한

시뮬레이션 작업을 생성할 때는 아래 권한을 가진 IAM 역할을 부여해야 합니다.

- my-input-bucket을 로봇 및 시뮬레이션 애플리케이션 번들이 포함된 버킷의 이름으로 바꿉니다.
- my-output-bucket을 AWS RoboMaker에서 출력 파일을 쓰는 버킷으로 가리키도록 바꿉니다.
- account#를 계정 번호로 바꿉니다.

퍼블릭 ECR 작업에는 `ecr-public:GetAuthorizationToken`, `sts:GetServiceBearerToken` 및 최종 구현에 필요한 기타 권한과 같은 별도의 권한이 필요합니다. 자세한 내용을 알아보려면 Amazon ECR 사용 설명서의 [퍼블릭 리포지토리 정책](#)을 참조하세요.

## Jobs with Private ECR images

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": "s3:ListBucket",
      "Resource": [
        "arn:aws:s3:::my-input-bucket"
      ],
      "Effect": "Allow"
    },
    {
      "Action": [

```

```

        "s3:Get*",
        "s3:List*"
    ],
    "Resource": [
        "arn:aws:s3:::my-input-bucket/*"
    ],
    "Effect": "Allow"
},
{
    "Action": "s3:Put*",
    "Resource": [
        "arn:aws:s3:::my-output-bucket/*"
    ],
    "Effect": "Allow"
},
{
    "Action": [
        "logs:CreateLogGroup",
        "logs:CreateLogStream",
        "logs:PutLogEvents",
        "logs:DescribeLogStreams"
    ],
    "Resource": [
        "arn:aws:logs:*:account#:log-group:/aws/robomaker/SimulationJobs*"
    ],
    "Effect": "Allow"
},
{
    "Action": [
        "ecr:BatchGetImage",
        "ecr:GetAuthorizationToken",
        "ecr:BatchCheckLayerAvailability",
        "ecr:GetDownloadUrlForLayer"
    ],
    "Resource":
    "arn:partition:ecr:region:account#:repository/repository_name",
    "Effect": "Allow"
}
]
}

```

## Jobs with Public ECR images

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": "s3:ListBucket",
      "Resource": [
        "arn:aws:s3::my-input-bucket"
      ],
      "Effect": "Allow"
    },
    {
      "Action": [
        "s3:Get*",
        "s3:List*"
      ],
      "Resource": [
        "arn:aws:s3::my-input-bucket/*"
      ],
      "Effect": "Allow"
    },
    {
      "Action": "s3:Put*",
      "Resource": [
        "arn:aws:s3::my-output-bucket/*"
      ],
      "Effect": "Allow"
    },
    {
      "Action": [
        "logs:CreateLogGroup",
        "logs:CreateLogStream",
        "logs:PutLogEvents",
        "logs:DescribeLogStreams"
      ],
      "Resource": [
        "arn:aws:logs:*:account#:log-group:/aws/robomaker/SimulationJobs*"
      ],
      "Effect": "Allow"
    },
    {
      "Action": [

```

```

        "ecr-public:GetAuthorizationToken",
        "sts:GetServiceBearerToken"
    ],
    "Resource": "*",
    "Effect": "Allow"
}
]
}

```

정책은 다음 신뢰 정책을 통해 역할에 연결되어야 합니다.

```

{
  "Version": "2012-10-17",
  "Statement": {
    "Effect": "Allow",
    "Principal": { "Service": "robomaker.amazonaws.com" },
    "Action": "sts:AssumeRole",
    "Condition": {
      "StringEquals": {
        "aws:SourceAccount": "account#" // Account where the simulation job
resource is created
      },
      "StringEquals": {
        "aws:SourceArn": "arn:aws:robomaker:region:account#:simulation-job/*"
      }
    }
  }
}

```

조건 키를 사용하여 AWS 서비스가 서비스간 트랜잭션 중에 [혼동된 대리자](#)로 사용되는 것을 방지합니다. 조건 키에 대한 추가 정보는 [SourceAccount](#) 및 [SourceARN](#)을 참조하세요.

## ROS 애플리케이션 또는 ROS 명령줄에서 태그를 사용하는 데 필요한 권한

ROS 명령줄 또는 ROS 애플리케이션이 실행되는 동안 시뮬레이션 작업의 태그에 태그를 지정하고 태그를 해제하고 나열할 수 있습니다. 아래 권한이 있는 IAM 역할이 있어야 합니다. `account#`를 계정 번호로 바꿉니다.

```

{
  "Version": "2012-10-17",

```

```

"Statement": [
  {
    "Action": [
      "robomaker:TagResource",
      "robomaker:UntagResource",
      "robomaker:ListTagsForResource",
    ],
    "Resource": [
      "arn:aws:robomaker:*:account:simulation-job*"
    ],
    "Effect": "Allow"
  }
]
}

```

정책은 다음 신뢰 정책을 통해 역할에 연결되어야 합니다.

```

{
  "Version": "2012-10-17",
  "Statement": {
    "Effect": "Allow",
    "Principal": { "Service": "robomaker.amazonaws.com" },
    "Action": "sts:AssumeRole",
    "Condition": {
      "StringEquals": {
        "aws:SourceAccount": "account" // Account where the simulation job
resource is created
      },
      "StringEquals": {
        "aws:SourceArn": "arn:aws:robomaker:region:account:simulation-job/*"
      }
    }
  }
}

```

조건 키를 사용하여 AWS 서비스가 서비스간 트랜잭션 중에 [혼동된 대리자](#)로 사용되는 것을 방지합니다. 조건 키에 대한 추가 정보는 [SourceAccount](#) 및 [SourceARN](#)을 참조하세요.

## AWS RoboMaker에서 IAM을 사용하는 방식 이해

서비스는 다음과 같은 여러 가지 방식으로 IAM을 사용할 수 있습니다.

- 작업 – AWS RoboMaker은 정책에서 작업 사용을 지원합니다. 따라서 관리자가 AWS RoboMaker에서 엔터티가 작업을 완료할 수 있는지 제어할 수 있습니다. 예를 들어, GetPolicy AWS API 작업을 수행해 엔터티가 정책을 보도록 하려면 관리자는 iam:GetPolicy 작업을 허용하는 정책을 연결해야 합니다.
- 리소스 수준 권한 – AWS RoboMaker은 리소스 수준 권한을 지원하지 않습니다. 리소스 수준 권한이 있으면 [ARN](#)을 사용하여 정책에서 개별 리소스를 지정할 수 있습니다. AWS RoboMaker에서 이 기능을 지원하지 않으므로 [정책 시각적 편집기](#)에서 모든 리소스를 선택해야 합니다. JSON 정책 문서의 \* 요소에서 Resource를 사용해야 합니다.
- 태그 기반 권한 부여 – AWS RoboMaker에서는 태그 기반 권한 부여를 지원합니다. 이 기능을 사용하면 정책 조건에서 [리소스 태그](#)를 사용할 수 있습니다.
- 임시 보안 인증 – AWS RoboMaker에서는 임시 보안 인증을 지원합니다. 이 기능을 사용하면 페더레이션을 사용해 로그인하거나, IAM 역할을 맡거나, 교차 계정 역할을 맡을 수 있습니다. [AssumeRole](#) 또는 [GetFederationToken](#) 같은 AWS STS API 태스크를 호출하여 임시 보안 자격 증명을 가져옵니다.
- 서비스 연결 역할 – AWS RoboMaker에서는 서비스 역할을 지원합니다. 이 기능을 사용하면 서비스가 사용자를 대신하여 [서비스 연결 역할](#)을 맡을 수 있습니다. 이 역할을 사용하면 서비스가 다른 서비스의 리소스에 액세스해 사용자를 대신해 태스크를 완료할 수 있습니다. 서비스 연결 역할은 IAM 계정에 표시되고, 서비스가 소유합니다. IAM 관리자는 서비스 연결 역할의 권한을 볼 수 있지만 편집은 할 수 없습니다.
- 서비스 역할 – AWS RoboMaker에서는 서비스 역할을 지원합니다. 이 기능을 사용하면 서비스가 사용자를 대신하여 [서비스 역할](#)을 수입할 수 있습니다. 이 역할을 사용하면 서비스가 다른 서비스의 리소스에 액세스해 사용자를 대신해 태스크를 완료할 수 있습니다. 서비스 역할은 IAM 계정에 나타나고, 해당 계정이 소유합니다. 즉, IAM 관리자가 이 역할에 대한 권한을 변경할 수 있습니다. 그러나 권한을 변경하면 서비스의 기능이 손상될 수 있습니다.

## 인증 및 액세스 제어 문제 해결

다음 정보를 사용하여 IAM에서 발생할 수 있는 공통적인 문제를 진단하고 수정할 수 있습니다.

### 주제

- [AWS RoboMaker에서 작업을 수행할 권한이 없음](#)
- [관리자인데, 다른 사용자가 AWS RoboMaker에 액세스할 수 있게 허용하려고 합니다](#)

## AWS RoboMaker에서 작업을 수행할 권한이 없음

AWS Management Console에서 작업을 수행할 권한이 없다는 오류가 표시되면 사용자 이름 및 암호를 제공한 관리자에게 연락해야 합니다.

다음 오류 예시는 my-user-name이라는 IAM 사용자가 콘솔을 사용해 CreateRobotApplication 작업을 수행하려고 하는데 권한이 없음을 보여줍니다.

```
User: arn:aws:iam::123456789012:user/my-user-name is not authorized to perform: aws-robomaker:CreateRobotApplication on resource: my-example-robot-application
```

이 예시에서는 aws-robomaker:CreateRobotApplication 작업을 사용해 my-example-robot-application 리소스에 액세스할 수 있도록 정책을 업데이트 해달라고 관리자에게 요청합니다.

관리자인데, 다른 사용자가 AWS RoboMaker에 액세스할 수 있게 허용하려고 합니다

다른 사용자가 AWS RoboMaker에 액세스하도록 허용하려면 액세스 권한이 필요한 사용자나 애플리케이션에 대한 IAM 엔터티(사용자 또는 역할)를 생성해야 합니다. 다른 사용자들은 해당 엔터티에 대한 자격 증명을 사용해 AWS에 액세스합니다. 그런 다음 AWS RoboMaker에 대한 올바른 권한을 부여하는 정책을 엔터티에 연결해야 합니다.

바로 시작하려면 [IAM 시작하기](#) 단원을 참조하세요.

## 정책이란 무엇입니까?

정책을 생성하고 IAM 자격 증명 또는 AWS 리소스에 연결하여 AWS에서 액세스를 제어합니다.

### Note

빠르게 시작하려면 [AWS RoboMaker에 대한 인증 및 액세스 제어](#)에 나와 있는 기본 정보를 검토한 다음 [IAM 시작하기](#) 섹션을 참조하세요.

정책은 개체 또는 리소스에 연결될 때 해당 권한을 정의하는 AWS의 객체입니다. AWS는 사용자와 같은 보안 주체가 요청할 때 이러한 정책을 평가합니다. 정책에서 권한은 요청이 허용되거나 거부되는지를 결정합니다. 대부분의 정책은 AWS에 JSON 문서로서 저장됩니다.

IAM 정책은 태스크를 수행하기 위해 사용하는 방법과 상관없이 작업에 대한 권한을 정의합니다. 예를 들어, 정책이 [GetUser](#) 작업을 허용한다면 이 정책이 있는 사용자는 AWS Management Console, AWS

CLI, 또는 AWS API에서 사용자 정보를 얻을 수 있습니다. IAM 사용자를 생성할 경우, 사용자가 콘솔 또는 프로그래밍 방식 액세스를 허용하도록 설정할 수 있습니다. IAM 사용자는 사용자 이름 및 암호를 사용하는 콘솔에 로그인할 수 있습니다. 또는 CLI 또는 API를 사용해 액세스 키를 사용할 수 있습니다.

액세스 권한을 제공하려면 사용자, 그룹 또는 역할에 권한을 추가합니다.

- AWS IAM Identity Center의 사용자 및 그룹:

권한 세트를 생성합니다. AWS IAM Identity Center 사용 설명서의 [권한 세트 생성](#)의 지침을 따르세요.

- 자격 증명 공급자를 통해 IAM에서 관리되는 사용자:

아이덴티티 페더레이션을 위한 역할을 생성합니다. IAM 사용 설명서의 [서드 파티 자격 증명 공급자의 역할 만들기\(연합\)](#)의 지침을 따르세요.

- IAM 사용자:

- 사용자가 맡을 수 있는 역할을 생성합니다. IAM 사용 설명서에서 [IAM 사용자의 역할 생성](#)의 지침을 따르세요.
- (권장되지 않음) 정책을 사용자에게 직접 연결하거나 사용자를 사용자 그룹에 추가합니다. IAM 사용 설명서에서 [사용자\(콘솔\)에 권한 추가](#)의 지침을 따르세요.

**i** AWS RoboMaker에서 지원되지 않는 정책

AWS RoboMaker에서 리소스 기반 정책 및 액세스 제어 목록(ACL)은 지원되지 않습니다. 자세한 내용은 IAM 사용 설명서의 [정책 유형](#)을 참조하세요.

## 주제

- [자격 증명 기반 정책](#)
- [정책 액세스 레벨 분류](#)

## 자격 증명 기반 정책

정책을 IAM 자격 증명에 연결할 수 있습니다. 예를 들면, 다음을 수행할 수 있습니다.

- 계정 내 사용자 또는 그룹에 권한 정책 연결 – 특정 사용자 또는 그 사용자가 속한 그룹에 권한 정책을 연결하여 AWS RoboMaker 리소스(예: 로봇 애플리케이션) 생성 권한을 해당 사용자에게 부여할 수 있습니다.

- 역할에 권한 정책 연결(교차 계정 권한 부여) - 자격 증명 기반 권한 정책을 IAM 역할에 연결하여 교차 계정 권한을 부여할 수 있습니다. 예를 들어, 계정 A의 관리자는 다음과 같이 다른 AWS 계정(예: 계정 B) 또는 AWS 서비스에 교차 계정 권한을 부여할 역할을 생성할 수 있습니다.
  1. 계정 A 관리자가 IAM 역할을 생성하고 계정 A의 리소스에 대한 권한을 부여하는 권한 정책을 해당 역할에 연결합니다.
  2. 계정 A 관리자는 계정 B를 역할을 수임할 보안 주체로 식별하는 역할에 신뢰 정책을 연결합니다.
  3. 계정 B 관리자는 계정 B의 사용자에게 역할을 수임할 권한을 위임할 수 있습니다. 그러면 계정 B의 사용자가 계정 A에서 리소스를 생성하거나 액세스할 수 있습니다. AWS 서비스에 역할 수임 권한을 부여할 경우 신뢰 정책의 보안 주체는 AWS 서비스 보안 주체일 수도 있습니다.

IAM을 사용하여 권한을 위임하는 방법에 대한 자세한 내용은 IAM User Guide의 [Access Management](#)를 참조하세요.

사용자, 그룹, 역할 및 권한에 대한 자세한 내용은 IAM 사용 설명서의 [자격 증명\(사용자, 그룹 및 역할\)](#)을 참조하세요.

## 정책 액세스 레벨 분류

IAM 콘솔에서는 다음 액세스 레벨 분류를 사용하여 작업이 그룹화됩니다.

- 나열 - 서비스 내의 리소스를 나열하여 객체가 존재하는지 판단할 수 있는 권한을 제공합니다. 이 액세스 레벨의 작업은 객체를 나열할 수 있으나 리소스의 내용을 확인할 수 없습니다. 액세스 레벨이 나열인 대부분의 작업은 특정 리소스에 대해 수행할 수 없습니다. 이러한 작업에 관련한 정책 설명을 생성할 때 모든 리소스("\*")를 지정해야 합니다.
- 읽기 - 서비스에서 리소스 내용과 속성을 읽을 수 있으나 편집할 수 없는 권한을 제공합니다. 예를 들어 Amazon S3 작업 GetObject 및 GetBucketLocation의 액세스 수준은 Read입니다.
- 쓰기 - 서비스에서 리소스를 생성, 삭제하거나 수정할 수 있는 권한을 제공합니다. 예를 들어 Amazon S3 작업 CreateBucket, DeleteBucket 및 PutObject의 액세스 레벨은 쓰기입니다.
- 권한 관리 - 서비스에서 리소스 권한을 부여하거나 수정할 수 있는 권한을 제공합니다. 예를 들어, 대부분의 IAM 및 AWS Organizations 정책 작업의 액세스 레벨은 권한 관리입니다.

### 도움말

AWS 계정의 보안을 개선하려면 권한 관리 액세스 레벨 분류를 포함하는 정책을 제한하거나 정기적으로 모니터링합니다.

- 태깅 – 서비스에서 리소스에 연결된 태그를 생성, 삭제하거나 수정할 수 있는 권한을 제공합니다. 예를 들어 Amazon EC2 CreateTags 및 DeleteTags 작업의 액세스 레벨은 태깅입니다.

## AWS RoboMaker의 AWS 관리형 정책

사용자, 그룹 또는 역할에 권한을 추가할 때 정책을 직접 작성하는 것보다 AWS 관리형 정책을 사용하는 것이 더욱 편리합니다. 팀에 필요한 권한만 제공하는 [IAM 고객 관리형 정책을 생성](#)하기 위해서는 시간과 전문 지식이 필요합니다. 신속하게 시작하려면 AWS 관리 정책을 사용할 수 있습니다. 이 정책은 일반적인 사용 사례를 다루며 사용자의 AWS 계정에서 사용할 수 있습니다. AWS 관리형 정책에 대한 자세한 내용은 [IAM 사용 설명서](#)에서 AWS 관리형 정책을 참조하세요.

AWS 서비스 유지 관리 및 AWS 관리형 정책 업데이트입니다. AWS 관리형 정책에서 권한을 변경할 수 없습니다. 서비스는 때때로 추가 권한을 AWS 관리형 정책에 추가하여 새로운 기능을 지원합니다. 이 유형의 업데이트는 정책이 연결된 모든 자격 증명(사용자, 그룹 및 역할)에 적용됩니다. 서비스는 새로운 기능이 시작되거나 새 태스크를 사용할 수 있을 때 AWS 관리형 정책에 업데이트됩니다. 서비스는 AWS 관리형 정책에서 권한을 제거하지 않기 때문에 정책 업데이트로 인해 기존 권한이 손상되지 않습니다.

또한 AWS는 여러 서비스의 직무에 대한 관리형 정책을 지원합니다. 예를 들어 ReadOnlyAccess라는 이름의 AWS 관리형 정책은 모든 AWS 서비스 및 리소스에 대한 읽기 전용 액세스 권한을 제공합니다. 서비스에서 새 기능을 시작하면 AWS가 새 작업 및 리소스에 대한 읽기 전용 권한을 추가합니다. 직무 정책의 목록과 설명은 IAM 사용 설명서의 [직무에 관한 AWS 관리형 정책](#)을 참조하세요.

### AWS 관리형 정책: AWSRoboMaker\_FullAccess

이 정책은 기여자에게 AWS RoboMaker가 애플리케이션을 생성하는 데 사용할 수 있는 이미지 또는 번들을 읽을 수 있는 권한을 부여합니다. 또한 이 정책을 통해 모든 AWS RoboMaker 리소스 및 작업에 액세스할 수 있습니다. 또한 계정의 Amazon EC2 리소스를 관리하는 IAM 역할을 계정에 생성합니다.

#### 권한 세부 정보

이 정책에는 다음 권한이 포함되어 있습니다.

- `s3:GetObject` — 로봇 또는 시뮬레이션 애플리케이션용 번들을 사용하는 경우 AWS RoboMaker가 Amazon S3 버킷에서 zip 파일을 가져올 수 있습니다.
- `ecr:BatchGetImage` — 로봇 또는 시뮬레이션 애플리케이션용 이미지를 사용하는 경우 AWS RoboMaker는 Amazon ECR 리포지토리에서 이미지를 가져올 수 있습니다.
- `ecr-public:DescribeImages` — 로봇 또는 시뮬레이션 애플리케이션에 공개적으로 사용 가능한 이미지를 사용하는 경우 AWS RoboMaker는 Amazon ECR 리포지토리에서 해당 이미지에 대한 정보를 가져올 수 있습니다.
- `iam:CreateServiceLinkedRole` — AWS RoboMaker가 성공적으로 작동하는 데 필요한 Amazon EC2 리소스에 액세스할 수 있도록 합니다. 자세한 정보는 [AWS RoboMaker에 서비스 연결 역할 사용](#) 섹션을 참조하세요.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "robomaker:*",
      "Resource": "*"
    },
    {
      "Effect": "Allow",
      "Action": "s3:GetObject",
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "aws:CalledViaFirst": "robomaker.amazonaws.com"
        }
      }
    },
    {
      "Effect": "Allow",
      "Action": "ecr:BatchGetImage",
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "aws:CalledViaFirst": "robomaker.amazonaws.com"
        }
      }
    },
    {
```

```

    "Effect": "Allow",
    "Action": "ecr-public:DescribeImages",
    "Resource": "*",
    "Condition": {
      "StringEquals": {
        "aws:CalledViaFirst": "robomaker.amazonaws.com"
      }
    }
  },
  {
    "Effect": "Allow",
    "Action": "iam:CreateServiceLinkedRole",
    "Resource": "*",
    "Condition": {
      "StringEquals": {
        "iam:AWSServiceName": "robomaker.amazonaws.com"
      }
    }
  }
]
}

```

## AWS 관리형 정책: AWSRoboMakerReadOnlyAccess

이 관리형 정책 예시는 AWS Management Console 및 SDK를 통해 AWS RoboMaker에 대한 읽기 전용 액세스를 제공합니다.

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "VisualEditor0",
      "Effect": "Allow",
      "Action": [
        "robomaker:List*",
        "robomaker:BatchDescribe*",
        "robomaker:Describe*",
        "robomaker:Get*"
      ],
      "Resource": "*"
    }
  ]
}

```

```

]
}

```

## AWS 관리형 정책으로 AWS RoboMaker 업데이트

이 서비스가 이러한 변경 내용을 추적하기 시작한 이후부터 AWS RoboMaker의 AWS 관리형 정책 업데이트에 대한 세부 정보를 봅니다. 이 페이지의 변경 사항에 대한 자동 알림을 받으려면 AWS RoboMaker 문서 기록 페이지에서 RSS 피드를 구독하세요.

변경 사항	설명	날짜
<a href="#">AWSRoboMaker_FullAccess</a> – 새 정책	<p>AWS RoboMaker는 성공적으로 실행하는 데 필요한 리소스에 대한 액세스를 허용하기 위한 새로운 정책을 추가했습니다.</p> <p>이 정책을 통해 AWS RoboMaker는 Amazon S3에 저장한 Amazon ECR 이미지 또는 zip 파일에 액세스하여 로봇 및 시뮬레이션 애플리케이션을 생성할 수 있습니다. 또한 AWS RoboMaker는 이를 통해 성공적으로 실행하는 데 필요한 Amazon EC2에 액세스할 수 있습니다.</p>	2021년 7월 27일
<a href="#">AWSRoboMakerReadOnlyAccess</a> – 새 정책	AWS RoboMaker는 AWS RoboMaker 리소스에 대한 읽기 전용 액세스를 활성화하는 새로운 정책을 추가했습니다.	2022년 1월 11일
AWS RoboMaker 변경 사항 추적 시작	AWS RoboMaker가 AWS 관리형 정책에 대한 변경 내용 추적을 시작했습니다.	2021년 7월 27일

## AWS RoboMaker에 서비스 연결 역할 사용

AWS RoboMaker은 AWS Identity and Access Management(IAM) [서비스 연결 역할](#)을 사용합니다. 서비스 연결 역할은 AWS RoboMaker에 직접 연결된 고유한 유형의 IAM 역할입니다. 서비스 연결 역할은 AWS RoboMaker에서 사전 정의하며 서비스에서 다른 AWS 서비스를 자동으로 호출하기 위해 필요한 모든 권한을 포함합니다.

서비스 연결 역할을 통해 AWS RoboMaker 설정이 쉬워지는데 필요한 권한을 수동으로 추가할 필요가 없기 때문입니다. AWS RoboMaker에서 서비스 연결 역할 권한을 정의하므로, 달리 정의되지 않은 한 AWS RoboMaker에서만 해당 역할을 맡을 수 있습니다. 정의된 권한에는 신뢰 정책과 권한 정책이 포함되며, 이 권한 정책은 다른 IAM 엔터티에 연결할 수 없습니다.

먼저 관련 리소스를 삭제한 후에만 서비스 연결 역할을 삭제할 수 있습니다. 이렇게 하면 리소스에 대한 액세스 권한을 부주의로 삭제할 수 없기 때문에 AWS RoboMaker 리소스가 보호됩니다.

서비스 연결 역할을 지원하는 기타 서비스에 대한 자세한 내용은 [IAM으로 작업하는 AWS 서비스](#)를 참조하고 [서비스 연결 역할(Service-Linked Role)] 열에 [예(Yes)]가 있는 서비스를 찾습니다. 해당 서비스에 대한 서비스 연결 역할 설명서를 보려면 예(Yes) 링크를 선택합니다.

### AWS RoboMaker에 대한 서비스 연결 역할 권한

AWSServiceRoleForRoboMaker라는 이름의 서비스 연결 역할을 사용하는 AWS RoboMaker – RoboMaker가 사용자 대신 EC2 및 Lambda 리소스에 액세스할 수 있도록 합니다.

AWSServiceRoleForRoboMaker 서비스 연결 역할은 역할을 수임하기 위해 다음 서비스를 신뢰합니다.

- `robomaker.amazonaws.com`

역할 권한 정책은 AWS RoboMaker가 지정된 리소스에서 다음 작업을 완료하도록 허용합니다.

- 시뮬레이션 작업 배치의 일부로 생성된 시뮬레이션 작업 생성 및 취소
- Amazon EC2 네트워킹 리소스 관리
- AWS Lambda 함수 생성 및 가져오기

IAM 엔터티(사용자, 그룹, 역할 등)가 서비스 연결 역할을 작성하고 편집하거나 삭제할 수 있도록 권한을 구성할 수 있습니다. 자세한 내용은 IAM 사용 설명서의 [서비스 연결 역할 권한](#) 섹션을 참조하세요.

## 서비스 연결 역할 만들기

서비스 연결 역할은 수동으로 생성할 필요가 없습니다. AWS Management Console, AWS CLI 또는 AWS API에서 SimulationJob 또는 DeploymentJob을 실행하면, AWS RoboMaker에서는 자동으로 서비스 링크 역할을 생성합니다.

이 서비스 연결 역할을 삭제한 다음 다시 생성해야 하는 경우 동일한 프로세스를 사용하여 계정에서 역할을 다시 생성할 수 있습니다. SimulationJob, SimulationJobBatch 또는 DeploymentJob을 실행하면, AWS RoboMaker에서는 서비스 링크 역할을 다시 생성합니다.

또한 IAM 콘솔을 사용해 RoboMaker 사용 사례로 서비스 연결 역할을 생성할 수도 있습니다. AWS CLI 또는 AWS API에서 `robomaker.amazonaws.com` 서비스 이름의 서비스 연결 역할을 생성합니다. 자세한 내용은 IAM 사용 설명서의 [서비스 연결 역할 생성](#) 섹션을 참조하세요. 이 서비스 연결 역할을 삭제한 후에는 동일한 프로세스를 사용하여 역할을 다시 생성할 수 있습니다.

## 서비스 연결 역할 편집

AWS RoboMaker은 AWSServiceRoleForRoboMaker 서비스 연결 역할을 편집하도록 허용하지 않습니다. 서비스 연결 역할을 생성한 후에는 다양한 개체가 역할을 참조할 수 있기 때문에 역할 이름을 변경할 수 없습니다. 하지만 IAM을 사용하여 역할의 설명을 편집할 수 있습니다. 자세한 내용은 IAM 사용 설명서의 [서비스 연결 역할 편집](#)을 참조하세요.

## 서비스 연결 역할 삭제

서비스 연결 역할이 필요한 기능 또는 서비스가 더 이상 필요 없는 경우에는 해당 역할을 삭제할 것을 권장합니다. 이렇게 하면 적극적으로 모니터링하거나 유지 관리하지 않은 미사용 엔터티가 없습니다. 단, 서비스 연결 역할에 대한 리소스를 먼저 정리해야 수동으로 삭제할 수 있습니다.

### Note

리소스를 삭제하려 할 때 AWS RoboMaker 서비스가 역할을 사용 중이면 삭제에 실패할 수 있습니다. 이 문제가 발생하면 몇 분 기다렸다가 작업을 다시 시도하세요.

## IAM을 사용하여 수동으로 서비스 연결 역할 삭제

IAM 콘솔, AWS CLI 또는 AWS API를 사용하여 AWSServiceRoleForRoboMaker 서비스 연결 역할을 삭제합니다. 자세한 내용은 [IAM 사용 설명서](#)의 서비스 연결 역할 삭제 섹션을 참조하세요.

## AWS RoboMaker 서비스 연결 역할이 지원되는 리전

AWS RoboMaker는 서비스가 제공되는 모든 리전에서 서비스 연결 역할을 사용하도록 지원합니다. 자세한 내용은 [AWS 리전 및 엔드포인트](#) 섹션을 참조하세요.

AWS RoboMaker에서는 서비스가 제공되는 모든 리전에서 서비스 연결 역할을 사용하도록 지원하지 않습니다. 다음 리전에서 AWSServiceRoleForRoboMaker 역할을 사용할 수 있습니다.

리전 이름	리전 자격 증명	AWS RoboMaker의 지원
미국 동부(버지니아 북부)	us-east-1	예
미국 동부(오하이오)	us-east-2	예
미국 서부(캘리포니아 북부)	us-west-1	예
미국 서부(오레곤)	us-west-2	예
아시아 태평양(뭄바이)	ap-south-1	예
Asia Pacific (Osaka)	ap-northeast-3	예
아시아 태평양(서울)	ap-northeast-2	예
아시아 태평양(싱가포르)	ap-southeast-1	예
아시아 태평양(시드니)	ap-southeast-2	예
아시아 태평양(도쿄)	ap-northeast-1	예
Canada (Central)	ca-central-1	예
유럽(프랑크푸르트)	eu-central-1	예
유럽(아일랜드)	eu-west-1	예
Europe (London)	eu-west-2	예
Europe (Paris)	eu-west-3	예
남아메리카(상파울루)	sa-east-1	예

리전 이름	리전 자격 증명	AWS RoboMaker의 지원
AWS GovCloud (US)	us-gov-west-1	아니요

## IAM 시작하기

AWS Identity and Access Management(IAM)은 서비스와 리소스에 대한 액세스를 안전하게 관리할 수 있는 AWS 서비스입니다. IAM은 추가 비용 없이 AWS 계정에 제공되는 기능입니다.

### Note

IAM을 시작하기 전에 [AWS RoboMaker에 대한 인증 및 액세스 제어](#)에 대한 기본 정보를 살펴보세요.

AWS 계정을 생성할 때는 해당 계정의 모든 AWS 서비스 및 리소스에 대한 완전한 액세스 권한이 있는 단일 로그인 ID로 시작합니다. 이 ID는 AWS 계정 루트 사용자라고 하며, 계정을 생성할 때 사용한 이메일 주소와 암호로 로그인하여 액세스합니다. 일상적인 태스크에 루트 사용자를 사용하지 않을 것을 강력히 권장합니다. 루트 사용자 보안 인증 정보를 보호하고 루트 사용자만 수행할 수 있는 태스크를 수행하는 데 사용합니다. 루트 사용자로 로그인해야 하는 전체 작업 목록은 IAM 사용 설명서의 [Tasks that require root user credentials](#)를 참조하세요.

## IAM Admin 사용자 생성

관리자 사용자를 생성하려면 다음 옵션 중 하나를 선택합니다.

관리자를 관리하는 한 가지 방법 선택	To	By	다른 방법
IAM Identity Center에서 (권장)	단기 보안 인증 정보를 사용하여 AWS에 액세스합니다.  이는 보안 모범 사례에 부합됩니다. 모범 사례	AWS IAM Identity Center 사용 설명서의 <a href="#">시작하기</a> 에 나온 지침을 따릅니다.	AWS Command Line Interface 사용 설명서에 나온 <a href="#">AWS IAM Identity Center을 사용하도록 AWS CLI 구성</a> 단계를 수행하

관리자를 관리하는 한 가지 방법 선택	To	By	다른 방법
	에 대한 자세한 내용은 IAM 사용 설명서의 <a href="#">IAM의 보안 모범 사례</a> 를 참조하세요.		여 프로그래밍 방식 액세스를 구성합니다.
IAM에서 (권장되지 않음)	장기 보안 인증 정보를 사용하여 AWS에 액세스합니다.	IAM 사용 설명서의 <a href="#">첫 번째 IAM 관리자 및 사용자 그룹 만들기</a> 에 나온 지침을 따릅니다.	IAM 사용 설명서에 나온 <a href="#">IAM 사용자의 액세스 키 관리</a> 단계를 수행하여 프로그래밍 방식 액세스를 구성합니다.

## AWS RoboMaker에 대한 위임 사용자 생성

AWS 계정에서 여러 사용자를 지원하려면 다른 사용자가 허용된 작업만 수행할 수 있도록 권한을 위임해야 합니다. 이렇게 하려면 해당 사용자에게 필요한 권한이 있는 IAM 그룹을 생성한 다음 필요에 따라 IAM 사용자를 필요한 그룹에 추가합니다. 이 프로세스를 사용하여 전체 AWS 계정에 대한 그룹, 사용자 및 권한을 설정할 수 있습니다. 이 솔루션은 AWS 관리자가 수동으로 사용자 및 그룹을 관리할 수 있는 중소 규모 조직에서 가장 적합합니다. 대규모 조직에서는 [사용자 지정 IAM 역할](#), [페더레이션](#) 또는 [Single Sign-On](#)을 사용할 수 있습니다.

위임된 사용자에 대한 예시 및 자세한 정보는 [IAM 사용 설명서](#)의 IAM 사용자에 대한 권한을 위임할 역할 생성을 참조하세요.

## 사용자가 자신의 자격 증명을 스스로 관리하도록 허용

MFA를 구성하려면 사용자의 가상 MFA 디바이스가 호스팅되는 하드웨어에 대한 물리적 액세스가 필요합니다. 예를 들어, 스마트폰에서 가상 MFA 디바이스를 실행하는 사용자에게 MFA를 구성할 수 있습니다. 이 경우 마법사를 완료하기 위해 스마트폰을 사용할 수 있어야 합니다. 이러한 이유로 사용자가 자신의 가상 MFA 디바이스를 직접 구성 및 관리할 수 있도록 허용하는 것이 좋습니다. 이 경우에는 사용자에게 필요한 IAM 작업을 수행할 권한을 부여해야 합니다.

필요한 권한을 부여하는 정책의 예시는 IAM 사용 설명서의 [IAM: IAM 사용자가 MFA 디바이스를 직접 관리할 수 있도록 허용](#)을 참조하세요.

## IAM 사용자에게 대해 MFA 활성화

보안 강화를 위해 모든 IAM 사용자는 멀티 팩터 인증(MFA)을 구성하여 AWS RoboMaker 리소스를 보호하는 것이 좋습니다. MFA는 사용자의 정규 로그인 자격 증명 외에도 AWS가 지원되는 MFA 디바이스의 고유 인증을 제출하라고 요청함으로써 보안을 더욱 강화합니다. 설정 지침 및 MFA 선택 사항에 대한 자세한 정보는 IAM 사용 설명서의 [AWS에서 사용자를 위한 MFA 디바이스 활성화](#)를 참조하세요.

### Note

IAM 사용자를 위해 MFA를 구성하려면 사용자의 가상 MFA 디바이스가 호스팅되는 모바일 디바이스에 대한 물리적 액세스가 필요합니다.

## AWS RoboMaker의 로깅 및 모니터링

모니터링은 AWS RoboMaker와 사용자 AWS 솔루션의 신뢰성, 가용성 및 성능을 유지하는 중요한 역할을 합니다. 발생하는 다중 지점 실패를 보다 쉽게 디버깅할 수 있도록 AWS 솔루션의 모든 부분으로부터 모니터링 데이터를 수집해야 합니다.

### 주제

- [Amazon CloudWatch를 사용한 AWS RoboMaker 모니터링](#)
- [AWS CloudTrail을 사용한 호출 로깅](#)

## Amazon CloudWatch를 사용한 AWS RoboMaker 모니터링

AWS RoboMaker는 지표를 Amazon CloudWatch로 전송합니다. AWS Management Console, AWS CLI 또는 API를 사용하여 AWS RoboMaker가 CloudWatch로 전송하는 지표를 나열할 수 있습니다.

지표는 생성된 리전에만 존재합니다. 지표는 삭제가 불가능하지만, 지표에 새 데이터가 게시되지 않을 경우 15개월 후에 자동으로 만료됩니다.

Amazon CloudWatch에 대한 자세한 내용은 [Amazon CloudWatch 사용 설명서](#)를 참조하세요.

### 주제

- [AWS RoboMaker 시뮬레이션 지표](#)
- [AWS RoboMaker 사용량 지표](#)

## AWS RoboMaker 시뮬레이션 지표

Amazon CloudWatch를 이용하여 시뮬레이션 작업을 모니터링함으로써 AWS RoboMaker 시뮬레이션 작업에 대한 정보를 수집하고 읽기 가능한 실시간에 가까운 지표를 만들 수 있습니다. 정보는 1분 주기로 제공됩니다.

다음 지표는 SimulationJobId 차원에서 사용할 수 있습니다.

지표	설명
RealTimeFactor	<p>시뮬레이션된 시간 대 일반 시계 시간의 비. 예를 들어, 30분 시뮬레이션하기 위해 한 시간이 소요된 경우 이 계수는 0.5입니다.</p> <p>보다 복잡한 시뮬레이션의 경우 실제 계수는 더 낮습니다.</p>
vCPU*	<p>시뮬레이션 작업에서 사용되는 가상 CPU 코어 수</p> <p>단위: 수</p>
Memory*	<p>시뮬레이션 작업에서 사용되는 메모리 양(GB)</p> <p>단위: GB</p>
SimulationUnit*	<p>SimulationUnit 은 시뮬레이션 작업의 vCPU 및 메모리 소비를 기반으로 계산됨</p> <p>단위: 수</p>

### Important

\*로 표시된 지표는 추정을 위한 것입니다. AWS RoboMaker는 시뮬레이션 작업 실행을 준비하는 동안 지표를 내보냅니다. 시뮬레이션 작업이 Running 상태가 될 때까지 요금은 발생하지 않습니다.

## AWS RoboMaker 사용량 지표

CloudWatch 사용량 지표를 사용하여 계정의 리소스 사용량을 확인할 수 있습니다. 이러한 지표를 사용하여 CloudWatch 그래프 및 대시보드에서 현재 서비스 사용량을 시각화합니다.

AWS RoboMaker 사용량 지표는 AWS Service Quotas에 해당합니다. 사용량이 서비스 할당량에 가까워지면 경고하는 경보를 구성할 수 있습니다. CloudWatch와 서비스 할당량의 통합에 대한 자세한 내용은 [Service Quotas 통합 및 사용량 지표](#)를 참조하세요.

다음 지표는 AWS/Usage 차원에서 사용할 수 있습니다.

지표	설명
ResourceCount	계정에서 실행 중인 지정된 리소스의 수입니다. 리소스는 지표와 연결된 차원에 의해 정의됩니다.  이 지표에 대한 가장 유용한 통계는 1분 동안 사용되는 최대 리소스 수를 나타내는 MAXIMUM입니다.

다음 차원은 AWS RoboMaker에 의해 게시되는 사용량 지표를 구체화하는 데 사용됩니다.

차원	설명
Service	리소스가 포함된 AWS 서비스의 이름 AWS RoboMaker 사용량 지표의 경우 해당 차원 값은 RoboMaker 입니다.
Type	보고되는 엔터티의 유형입니다. 현재 AWS RoboMaker 사용량 지표의 유효한 값은 Resource뿐입니다.
Resource	실행 중인 리소스의 유형입니다. 현재 AWS RoboMaker 사용량 지표의 유효한 값은 RobotApplication , SimulationApplication , ActiveSimulationJob

차원	설명
	b 및 ActiveSimulationJobBatch 입니다.
Class	추적 중인 리소스의 클래스. 리소스 차원의 값이 ActiveSimulationJob 인 AWS RoboMaker 사용량 지표의 경우 유효한 값은 CPU   GPU_AND_CPU 입니다. 이 차원의 값은 해당 지표에서 보고한 시뮬레이션 작업에 사용되는 컴퓨팅 리소스의 종류를 정의합니다. 다른 리소스의 경우 클래스 값은 None입니다.

이러한 지표는 매분 내보내집니다. 이러한 지표를 사용하여 사용량을 모니터링한 다음 필요한 경우 해당 한도 증가를 요청하십시오. 사용량 모니터링에 대한 자세한 내용은 [Service Quotas 시각화 및 경보 설정](#)을 참조하십시오.

## AWS CloudTrail을 사용한 호출 로깅

AWS RoboMaker는 사용자, 역할 또는 AWS RoboMaker의 AWS 서비스가 수행한 작업에 대한 레코드를 제공하는 서비스인 AWS CloudTrail과 통합됩니다. CloudTrail은 AWS RoboMaker에 대한 모든 API 호출을 이벤트로 캡처합니다. 캡처되는 호출에는 AWS RoboMaker 콘솔로부터의 호출과 AWS API 작업에 대한 코드 호출이 포함됩니다. 추적을 생성하면 AWS RoboMaker 이벤트를 포함한 CloudTrail 이벤트를 지속적으로 Amazon S3 버킷에 배포할 수 있습니다. 추적을 구성하지 않은 경우에도 CloudTrail 콘솔의 이벤트 기록에서 최신 이벤트를 볼 수 있습니다. CloudTrail에서 수집한 정보를 사용하여 AWS RoboMaker 플랜에 수행된 요청, 요청이 수행된 IP 주소, 요청을 수행한 사람, 요청이 수행된 시간 및 추가 세부 정보를 확인할 수 있습니다.

CloudTrail에 대한 자세한 내용은 [AWS CloudTrail 사용 설명서](#)를 참조하세요.

## CloudTrail의 AWS RoboMaker 정보

CloudTrail은 계정 생성 시 AWS 계정에서 사용되도록 설정됩니다. AWS RoboMaker에서 활동이 발생하면 해당 활동이 이벤트 기록의 다른 AWS 서비스 이벤트와 함께 CloudTrail 이벤트에 기록됩니다. AWS 계정에서 최신 이벤트를 확인, 검색 및 다운로드할 수 있습니다. 자세한 내용은 [CloudTrail 이벤트 기록을 사용하여 이벤트 보기](#)를 참조하세요.

AWS RoboMaker에 대한 이벤트를 포함하여 AWS 계정에 이벤트를 지속적으로 기록하려면 추적을 생성합니다. CloudTrail은 추적을 사용하여 Amazon S3 버킷으로 로그 파일을 전송할 수 있습니다.

콘솔에서 추적을 생성하면 기본적으로 모든 AWS 리전에 추적이 적용됩니다. 추적은 AWS 파티션에 있는 모든 리전의 이벤트를 로깅하고 지정된 Amazon S3 버킷으로 로그 파일을 전송합니다. 또는 CloudTrail 로그에서 수집된 이벤트 데이터를 추가 분석 및 처리하도록 다른 AWS 서비스를 구성할 수 있습니다. 자세한 내용은 다음 자료를 참조하세요.

- [추적 생성 개요](#)
- [CloudTrail 지원 서비스 및 통합](#)
- [CloudTrail에 대한 Amazon SNS 알림 구성](#)
- [여러 리전에서 CloudTrail 로그 파일 받기](#) 및 [여러 계정에서 CloudTrail 로그 파일 받기](#)

모든 AWS RoboMaker 작업은 CloudTrail에서 로깅되고 [AWS API 참조](#)에 기록됩니다. 예를 들어 CreateSimulationJob, RegisterRobot, UpdateRobotApplication 작업을 호출하면 CloudTrail 로그 파일에 항목이 생성됩니다.

모든 이벤트 및 로그 항목에는 요청을 생성한 사용자에 대한 정보가 들어 있습니다. 자격 증명 정보를 이용하면 다음을 쉽게 판단할 수 있습니다.

- 요청을 루트로 했는지 아니면 AWS Identity and Access Management(IAM) 사용자 자격 증명으로 했는지.
- 역할 또는 페더레이션 사용자에게 대한 임시 보안 자격 증명을 사용하여 요청이 생성되었는지 여부.
- 다른 AWS 서비스에서 요청했는지 여부.

자세한 내용은 [CloudTrail userIdentity 요소](#)를 참조하세요.

## AWS RoboMaker 로그 파일 항목 이해

추적이란 지정한 Amazon S3 버킷에 이벤트를 로그 파일로 입력할 수 있게 하는 구성입니다.

CloudTrail 로그 파일에는 하나 이상의 로그 항목이 포함될 수 있습니다. 이벤트는 모든 소스의 단일 요청을 나타내며 요청된 작업, 작업 날짜와 시간, 요청 파라미터 등에 대한 정보를 포함합니다. CloudTrail 로그 파일은 퍼블릭 API 호출의 주문 스택 트레이스가 아니므로 특정 순서로 표시되지 않습니다.

다음은 DescribeRobot 작업을 보여주는 CloudTrail 로그 항목이 나타낸 예제입니다.

```
{
  "eventVersion": "1.05",
  "userIdentity": {
    "type": "IAMUser",
    "principalId": "my-principal-id",
```

```

    "arn": "my-arn",
    "accountId": "my-account-id",
    "accessKeyId": "my-access-key",
    "userName": "my-user-name"
  },
  "eventTime": "2018-12-07T00:28:03Z",
  "eventSource": "robomaker.amazonaws.com",
  "eventName": "DescribeRobot",
  "awsRegion": "us-west-2",
  "sourceIPAddress": "my-ip-address",
  "userAgent": "aws-internal/3 aws-sdk-java/1.11.455
Linux/4.4.83-0.1.fm.327.54.326.metal1.x86_64 OpenJDK_64-Bit_Server_VM/25.192-b12
java/1.8.0_192,",
  "requestParameters": {
    "robot": "my-robot-arn"
  },
  "responseElements": null,
  "requestID": "f54cdf8b-f9b6-11e8-8883-c3f04579eca3",
  "eventID": "affb0303-ff48-4f65-af8e-d7d19710bac3",
  "readOnly": true,
  "eventType": "AwsApiCall",
  "recipientAccountId": "my-recipient-account-id"
}

```

## AWS RoboMaker 리소스 태깅

플릿, 로봇, 로봇 애플리케이션, 시뮬레이션 애플리케이션 및 시뮬레이션 작업 등을 쉽게 관리하고 체계화하기 위해 자체 메타데이터를 태그 형태로 특정 리소스에 할당할 수 있습니다. 이 단원에서는 태그를 설명하고 태그를 생성하는 방법을 보여 줍니다.

### 태그 기본 사항

태그를 사용하면 용도, 소유자 또는 환경을 기준으로 하는 등 AWS RoboMaker 리소스를 다양한 방식으로 분류할 수 있습니다. 이 기능은 지정한 태그에 따라 특정 리소스를 빠르게 식별할 수 있으므로 동일한 유형의 리소스가 많을 때 유용합니다. 각 태그는 사용자가 정의하는 키와 선택적 값으로 구성됩니다. 예를 들어, 로봇에 대한 태그 세트를 정의하여 디바이스를 기능별로 추적할 수 있습니다. 각 리소스 유형에 대한 요건을 충족하는 태그 키 세트를 생성하는 것이 좋습니다. 일관된 태그 키 세트를 사용하면 리소스를 보다 쉽게 관리할 수 있습니다.

추가하거나 적용한 태그를 기준으로 리소스를 검색하고 필터링할 수 있습니다. [IAM 정책에 태그 사용](#)의 설명처럼 리소스에 대한 액세스 제어에 태그를 사용할 수도 있습니다.

AWS Management Console의 태그 편집기는 태그를 생성하고 관리할 수 있는 중앙 통합 방식으로, 이 도구를 사용하면 아주 편리합니다. 자세한 내용은 [AWS Management Console에서 작업의 태그 편집기 사용하기](#)를 참조하세요.

AWS CLI 및 AWS RoboMaker API를 사용하여 태그 관련 작업을 수행할 수 있습니다. 다음 명령에서 태그 필드를 사용하여 태그를 생성할 때 사물 그룹, 사물 유형, 주제 규칙, 작업, 보안 프로필 및 결제 그룹에 태그를 연결할 수 있습니다.

- [CreateRobotApplication](#)
- [CreateSimulationApplication](#)
- [CreateSimulationJob](#)
- [CreateWorldExportJob](#)
- [CreateWorldGenerationJob](#)
- [CreateWorldTemplate](#)
- [StartSimulationJobBatch](#)

다음 명령을 사용하여 태깅을 지원하는 기존 리소스에 대해 태그를 추가, 수정, 삭제할 수 있습니다.

- [TagResource](#)
- [ListTagsForResource](#)
- [UntagResource](#)

태그 키와 값을 편집할 수 있으며 언제든지 리소스에서 태그를 제거할 수 있습니다. 태그의 값을 빈 문자열로 설정할 수 있지만 태그의 값을 Null로 설정할 수는 없습니다. 해당 리소스에 대해 키가 기존 태그와 동일한 태그를 추가하는 경우 새 값이 이전 값을 덮어씁니다. 리소스를 삭제하면, 리소스에 대한 연결이 완료된 태그 또한 삭제됩니다.

## 태그 규제 및 제한

태그에 적용되는 기본 제한은 다음과 같습니다.

- 리소스당 최대 태그 수: 50개
- 최대 키 길이: UTF-8의 유니코드 문자 127자
- 최대 값 길이: 유니코드 문자 255자(UTF-8)
- 태그 키와 값은 대/소문자를 구분합니다.

- 태그 이름이나 값에서 `aws:` 접두사는 사용하지 마세요. 이 단어는 AWS용으로 예약되어 있습니다. 이 접두사가 지정된 태그 이름이나 값은 편집하거나 삭제할 수 없습니다. 이 접두사가 지정된 태그는 리소스당 태그 수 제한에 포함되지 않습니다.
- 태깅 스키마를 여러 서비스와 리소스에서 사용하는 경우 다른 서비스 또한 허용되는 문자에 대한 제한이 있을 수 있음을 유의합니다. 일반적으로 허용되는 문자는 UTF-8로 표현할 수 있는 문자, 공백 및 숫자와 `+ - = . _ : / @` 등의 특수 문자입니다.

## IAM 정책에 태그 사용

AWS RoboMaker API 작업에 사용하는 IAM 정책에 태그 기반의 리소스 수준 권한을 적용할 수 있습니다. 이를 통해 사용자가 생성, 수정 또는 사용할 수 있는 리소스를 더욱 정확하게 제어할 수 있습니다. 리소스 태그를 기반으로 사용자 액세스(권한)를 제어하기 위해 IAM 정책에서 다음 조건 컨텍스트 키 및 값과 함께 Condition 요소(Condition 블록)를 사용합니다.

- `aws:ResourceTag/tag-key: tag-value`를 사용하여 특정 태그가 지정된 리소스에 대한 사용자 작업을 허용 또는 거부합니다.
- `aws:RequestTag/tag-key: tag-value`를 사용하여 태그를 허용하는 리소스를 생성하거나 수정하는 API 요청을 작성할 때 특정 태그를 사용하도록(또는 사용하지 않도록) 요구합니다.
- `aws:TagKeys: [tag-key, ...]`를 사용하여 태깅 가능한 리소스를 생성하거나 수정하는 API 요청을 작성할 때 특정 태그 키 집합을 사용하도록(또는 사용하지 않도록) 요구합니다.

### Note

IAM 정책의 조건 컨텍스트 키와 값은 태깅 가능한 리소스의 ID가 필수 파라미터인 AWS RoboMaker 작업에만 적용됩니다. 예를 들어, 이 요청에서는 태깅 가능한 리소스(플릿, 로봇, 로봇 애플리케이션, 시뮬레이션 애플리케이션, 시뮬레이션 작업, 배포 작업)을 참조하지 않으므로 조건 컨텍스트 키 및 값을 기준으로 [ListFleets](#) 사용을 허용하거나 거부하게 됩니다.

자세한 내용은 AWS Identity and Access Management 사용 설명서의 [AWS 리소스에 대한 액세스 제어](#)를 참조하세요. 이 설명서의 [IAM JSON 정책 참조](#) 단원에서는 IAM에서 JSON 정책의 자세한 구문과 설명, 요소의 예, 변수, 평가 로직을 설명합니다.

다음은 태그 기반 제한 2개를 적용하는 정책 예제입니다. 이 정책으로 제한되는 IAM 사용자는 다음과 같습니다.

- "env=prod" 태그가 있는 로봇을 생성할 수 없습니다(예시에서는 줄 "aws:RequestTag/env" : "prod" 참조).
- 기존 태그("env=prod")가 있는 로봇은 삭제할 수 없습니다(예시에서는 줄 "aws:ResourceTag/env" : "prod" 참조).

```
{
  "Version" : "2012-10-17",
  "Statement" : [
    {
      "Effect" : "Deny",
      "Action" : "robomaker:CreateRobot",
      "Resource" : "*",
      "Condition" : {
        "StringEquals" : {
          "aws:RequestTag/env" : "prod"
        }
      }
    },
    {
      "Effect" : "Deny",
      "Action" : "robomaker>DeleteRobot",
      "Resource" : "*",
      "Condition" : {
        "StringEquals" : {
          "aws:ResourceTag/env" : "prod"
        }
      }
    },
    {
      "Effect": "Allow",
      "Action": "robomaker:*",
      "Resource": "*"
    }
  ]
}
```

또한 다음과 같이 목록에서 태그를 둘러싸 지정된 태그 키에 대해 여러 태그 값을 지정할 수도 있습니다.

```
"StringEquals" : {
```

```

    "aws:ResourceTag/env" : ["dev", "test"]
  }

```

### Note

태그를 기준으로 리소스에 대한 사용자 액세스를 허용 또는 거부하는 경우 동일한 리소스에서 태그를 추가 또는 제거할 수 있도록 사용자를 명시적으로 거부할 것을 고려해야 합니다. 그렇지 않으면 사용자가 제한을 피해 태그를 수정하여 리소스에 대한 액세스 권한을 얻을 수 있습니다.

## 보안 규정 준수

AWS HIPAA 규정 준수 프로그램에는 AWS RoboMaker가 HIPAA 적격 서비스로 포함되어 있습니다. AWS PCI DSS 규정 준수 프로그램에는 AWS RoboMaker가 PCI 규정 준수 서비스로 포함되어 있습니다.

AWS 클라우드 및 HIPAA 규정 준수에 대한 일반적인 내용은 다음 문서를 참조하세요.

- [HIPAA 규정 준수](#)
- [Architecting for HIPAA Security and Compliance on Amazon Web Services](#)

## AWS RoboMaker 복원성

AWS 글로벌 인프라는 AWS 리전 및 가용 영역을 중심으로 구축됩니다. AWS 리전은 물리적으로 분리되고 격리된 다수의 가용 영역을 제공하며 이러한 가용 영역은 짧은 지연 시간, 높은 처리량 및 높은 중복성을 갖춘 네트워크에 연결되어 있습니다. 가용 영역을 사용하면 중단 없이 영역 간에 자동으로 장애 조치가 이루어지는 애플리케이션 및 데이터베이스를 설계하고 운영할 수 있습니다. 가용 영역은 기존의 단일 또는 다중 데이터 센터 인프라보다 가용성, 내결함성, 확장성이 뛰어납니다.

AWS 리전 및 가용 영역에 대한 자세한 내용은 [AWS 글로벌 인프라](#)를 참조하세요.

AWS 글로벌 인프라 뿐만 아니라 AWS RoboMaker 도 데이터 복원력과 백업 요구 사항을 지원하는 다양한 기능을 제공합니다.

## AWS RoboMaker의 인프라 보안

관리형 서비스인 AWS RoboMaker는 AWS 글로벌 네트워크 보안으로 보호됩니다. AWS 보안 서비스와 AWS의 인프라 보호 방법에 대한 자세한 내용은 [AWS 클라우드 보안](#)을 참조하세요. 인프라 보안에 대한 모범 사례를 사용하여 AWS 환경을 설계하려면 보안 원칙 AWS Well-Architected Framework의 [인프라 보호](#)를 참조하세요.

AWS에서 게시한 API 호출을 사용하여 네트워크를 통해 AWS RoboMaker에 액세스합니다. 고객은 다음을 지원해야 합니다.

- 전송 계층 보안(TLS). TLS 1.2는 필수이며 TLS 1.3을 권장합니다.
- DHE(Ephemeral Diffie-Hellman) 또는 ECDHE(Elliptic Curve Ephemeral Diffie-Hellman)와 같은 완전 전송 보안(PFS)이 포함된 암호 제품군. Java 7 이상의 최신 시스템은 대부분 이러한 모드를 지원합니다.

또한 요청은 액세스 키 ID 및 IAM 주체와 관련된 보안 액세스 키를 사용하여 서명해야 합니다. 또는 [AWS Security Token Service](#)(AWS STS)를 사용하여 임시 보안 자격 증명을 생성하여 요청에 서명할 수 있습니다.

## AWS RoboMaker 및 인터페이스 VPC 엔드포인트(AWS PrivateLink)

인터페이스 VPC 종단점을 생성하여 VPC와 AWS RoboMaker 간에 프라이빗 연결을 설정할 수 있습니다. 인터페이스 엔드포인트는 인터넷 게이트웨이, NAT 디바이스, VPN 연결 또는 AWS Direct Connect 연결 없이 비공개로 AWS RoboMaker에 액세스할 수 있도록 지원하는 [AWS PrivateLink](#) 기술에 의해 구동됩니다. VPC의 인스턴스는 AWS RoboMaker API와 통신하는 데 퍼블릭 IP 주소를 필요로 하지 않습니다. VPC와 AWS RoboMaker 간의 트래픽은 Amazon 네트워크를 벗어나지 않습니다.

각 인터페이스 엔드포인트는 서브넷에서 하나 이상의 [탄력적 네트워크 인터페이스](#)로 표현됩니다.

자세한 내용은 AWS PrivateLink 설명서의 [인터페이스 VPC 엔드포인트\(AWS PrivateLink\)](#)를 참조하세요.

## AWS RoboMaker VPC 엔드포인트에 대한 고려사항

AWS RoboMaker에 대한 인터페이스 VPC 엔드포인트를 설정하기 전에 AWS PrivateLink사용 설명서에서 [인터페이스 엔드포인트 속성 및 제한 사항](#)을 검토해야 합니다.

AWS RoboMaker는 VPC에서 모든 API 작업에 대한 호출 수행을 지원합니다.

## AWS RoboMaker에 대한 인터페이스 VPC 엔드포인트 생성

Amazon VPC 콘솔이나 AWS Command Line Interface(AWS CLI)을 사용하여 AWS RoboMaker 서비스에 대한 VPC 엔드포인트를 생성할 수 있습니다. 자세한 내용은 AWS PrivateLink 설명서의 [인터페이스 VPC 엔드포인트 생성](#)을 참조하세요.

다음 서비스 이름을 사용하여 AWS RoboMaker 용 VPC 종단점을 생성합니다.

- `com.amazonaws.region.robomaker`

엔드포인트에 프라이빗 DNS를 사용하도록 설정하는 경우, 리전에 대한 기본 DNS 이름(예: `robomaker.us-east-1.amazonaws.com`)을 사용하여 AWS RoboMaker에 API 요청을 할 수 있습니다.

자세한 내용은 AWS PrivateLink 가이드의 [인터페이스 엔드포인트를 통해 서비스 액세스](#)를 참조하세요.

## AWS RoboMaker에 대한 VPC 엔드포인트 정책 생성

AWS RoboMaker에 대한 액세스를 제어하는 VPC 엔드포인트에 엔드포인트 정책을 연결할 수 있습니다. 이 정책은 다음 정보를 지정합니다.

- 태스크를 수행할 수 있는 보안 주체.
- 수행할 수 있는 작업입니다.
- 태스크를 수행할 있는 리소스.

자세한 내용은 AWS PrivateLink 설명서의 [VPC 엔드포인트를 통해 서비스에 대한 액세스 제어](#)를 참조하세요.

예제: AWS RoboMaker 작업에 대한 VPC 엔드포인트 정책

다음은 AWS RoboMaker 에 대한 엔드포인트 정책의 예입니다. 이 정책은 엔드포인트에 연결될 때 모든 리소스의 모든 보안 주체에 대한 액세스 권한을 나열된 AWS RoboMaker 작업에 부여합니다.

```
{
  "Statement": [
    {
```

```
    "Principal": "*",
    "Effect": "Allow",
    "Action": [
        "robomaker:ListSimulationJobs",
        "robomaker:ListSimulationJobBatches"
    ],
    "Resource": "*"
  }
]
```

# API 참조

이 장에는 에 대한 API 참조 문서가 포함되어 있습니다. AWS RoboMaker 여기에는 다음과 같은 섹션을 포함하고 있습니다.

## Sections

- [작업](#)
- [데이터 유형](#)
- [일반적인 오류](#)
- [공통 파라미터](#)

## 작업

다음 작업이 지원됩니다.

- [BatchDeleteWorlds](#)
- [BatchDescribeSimulationJob](#)
- [CancelDeploymentJob](#)
- [CancelSimulationJob](#)
- [CancelSimulationJobBatch](#)
- [CancelWorldExportJob](#)
- [CancelWorldGenerationJob](#)
- [CreateDeploymentJob](#)
- [CreateFleet](#)
- [CreateRobot](#)
- [CreateRobotApplication](#)
- [CreateRobotApplicationVersion](#)
- [CreateSimulationApplication](#)
- [CreateSimulationApplicationVersion](#)
- [CreateSimulationJob](#)
- [CreateWorldExportJob](#)
- [CreateWorldGenerationJob](#)

- [CreateWorldTemplate](#)
- [DeleteFleet](#)
- [DeleteRobot](#)
- [DeleteRobotApplication](#)
- [DeleteSimulationApplication](#)
- [DeleteWorldTemplate](#)
- [DeregisterRobot](#)
- [DescribeDeploymentJob](#)
- [DescribeFleet](#)
- [DescribeRobot](#)
- [DescribeRobotApplication](#)
- [DescribeSimulationApplication](#)
- [DescribeSimulationJob](#)
- [DescribeSimulationJobBatch](#)
- [DescribeWorld](#)
- [DescribeWorldExportJob](#)
- [DescribeWorldGenerationJob](#)
- [DescribeWorldTemplate](#)
- [GetWorldTemplateBody](#)
- [ListDeploymentJobs](#)
- [ListFleets](#)
- [ListRobotApplications](#)
- [ListRobots](#)
- [ListSimulationApplications](#)
- [ListSimulationJobBatches](#)
- [ListSimulationJobs](#)
- [ListTagsForResource](#)
- [ListWorldExportJobs](#)
- [ListWorldGenerationJobs](#)
- [ListWorlds](#)

- [ListWorldTemplates](#)
- [RegisterRobot](#)
- [RestartSimulationJob](#)
- [StartSimulationJobBatch](#)
- [SyncDeploymentJob](#)
- [TagResource](#)
- [UntagResource](#)
- [UpdateRobotApplication](#)
- [UpdateSimulationApplication](#)
- [UpdateWorldTemplate](#)

## BatchDeleteWorlds

배치 작업에서 하나 이상의 월드를 삭제합니다.

### Request Syntax

```
POST /batchDeleteWorlds HTTP/1.1
Content-type: application/json

{
  "worlds": [ "string" ]
}
```

### URI 요청 파라미터

요청은 URI 파라미터를 사용하지 않습니다.

### 요청 본문

요청은 JSON 형식으로 다음 데이터를 받습니다.

#### worlds

삭제할 월드에 해당하는 Amazon 리소스 이름(ARN) 목록입니다.

유형: 문자열 어레이

배열 멤버: 최소수는 1개입니다. 최대 항목 수는 100개입니다.

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

필수 항목 여부: 예

### 응답 구문

```
HTTP/1.1 200
Content-type: application/json

{
```

```
"unprocessedWorlds": [ "string" ]
}
```

## 응답 요소

작업이 성공하면 서비스가 HTTP 200 응답을 반환합니다.

다음 데이터는 서비스에 의해 JSON 형식으로 반환됩니다.

### unprocessedWorlds

호출과 관련된 처리되지 않은 월드 목록입니다. 이러한 월드는 삭제되지 않았습니다.

유형: 문자열 어레이

배열 멤버: 최소수는 1개입니다. 최대 항목 수는 100개입니다.

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

## Errors

모든 작업에서 발생하는 일반적인 오류에 대한 자세한 내용은 [일반적인 오류](#) 섹션을 참조하세요.

### InternalServerError

RoboMaker AWS에서 서비스 문제가 발생했습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 500

### InvalidParameterException

요청에 지정된 파라미터는 유효하지 않거나, 지원되지 않거나, 사용할 수 없습니다. 반환된 메시지는 오류 값에 대한 설명을 제공합니다.

HTTP 상태 코드: 400

### ThrottlingException

RoboMaker AWS는 일시적으로 요청을 처리할 수 없습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 400

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS Go v2를 위한 SDK](#)
- [AWS Java V2용 SDK](#)
- [AWS V3용 SDK JavaScript](#)
- [AWS PHP V3용 SDK](#)
- [AWS Python용 SDK](#)
- [AWS 루비 V3용 SDK](#)

## BatchDescribeSimulationJob

하나 이상의 시뮬레이션 작업을 설명합니다.

### Request Syntax

```
POST /batchDescribeSimulationJob HTTP/1.1
Content-type: application/json

{
  "jobs": [ "string" ]
}
```

### URI 요청 파라미터

요청은 URI 파라미터를 사용하지 않습니다.

### 요청 본문

요청은 JSON 형식으로 다음 데이터를 받습니다.

#### jobs

설명할 시뮬레이션 작업의 Amazon 리소스 이름(ARN) 목록입니다.

유형: 문자열 어레이

배열 멤버: 최소수는 1개입니다. 최대 항목 수는 100개입니다.

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

필수 항목 여부: 예

### 응답 구문

```
HTTP/1.1 200
Content-type: application/json

{
  "jobs": [
    {
```

```
"arn": "string",
"clientRequestToken": "string",
"compute": {
  "computeType": "string",
  "gpuUnitLimit": number,
  "simulationUnitLimit": number
},
"dataSources": [
  {
    "destination": "string",
    "name": "string",
    "s3Bucket": "string",
    "s3Keys": [
      {
        "etag": "string",
        "s3Key": "string"
      }
    ],
    "type": "string"
  }
],
"failureBehavior": "string",
"failureCode": "string",
"failureReason": "string",
"iamRole": "string",
"lastStartedAt": number,
"lastUpdatedAt": number,
"loggingConfig": {
  "recordAllRosTopics": boolean
},
"maxJobDurationInSeconds": number,
"name": "string",
"networkInterface": {
  "networkInterfaceId": "string",
  "privateIpAddress": "string",
  "publicIpAddress": "string"
},
"outputLocation": {
  "s3Bucket": "string",
  "s3Prefix": "string"
},
"robotApplications": [
  {
    "application": "string",
```

```

    "applicationVersion": "string",
    "launchConfig": {
      "command": [ "string" ],
      "environmentVariables": {
        "string" : "string"
      },
      "launchFile": "string",
      "packageName": "string",
      "portForwardingConfig": {
        "portMappings": [
          {
            "applicationPort": number,
            "enableOnPublicIp": boolean,
            "jobPort": number
          }
        ]
      },
      "streamUI": boolean
    },
    "tools": [
      {
        "command": "string",
        "exitBehavior": "string",
        "name": "string",
        "streamOutputToCloudWatch": boolean,
        "streamUI": boolean
      }
    ],
    "uploadConfigurations": [
      {
        "name": "string",
        "path": "string",
        "uploadBehavior": "string"
      }
    ],
    "useDefaultTools": boolean,
    "useDefaultUploadConfigurations": boolean
  }
],
"simulationApplications": [
  {
    "application": "string",
    "applicationVersion": "string",
    "launchConfig": {

```

```
    "command": [ "string" ],
    "environmentVariables": {
      "string": "string"
    },
    "launchFile": "string",
    "packageName": "string",
    "portForwardingConfig": {
      "portMappings": [
        {
          "applicationPort": number,
          "enableOnPublicIp": boolean,
          "jobPort": number
        }
      ]
    },
    "streamUI": boolean
  },
  "tools": [
    {
      "command": "string",
      "exitBehavior": "string",
      "name": "string",
      "streamOutputToCloudWatch": boolean,
      "streamUI": boolean
    }
  ],
  "uploadConfigurations": [
    {
      "name": "string",
      "path": "string",
      "uploadBehavior": "string"
    }
  ],
  "useDefaultTools": boolean,
  "useDefaultUploadConfigurations": boolean,
  "worldConfigs": [
    {
      "world": "string"
    }
  ]
}
],
"simulationTimeMillis": number,
"status": "string",
```

```

    "tags": {
      "string" : "string"
    },
    "vpcConfig": {
      "assignPublicIp": boolean,
      "securityGroups": [ "string" ],
      "subnets": [ "string" ],
      "vpcId": "string"
    }
  }
],
"unprocessedJobs": [ "string" ]
}

```

## 응답 요소

작업이 성공하면 서비스가 HTTP 200 응답을 반환합니다.

다음 데이터는 서비스에 의해 JSON 형식으로 반환됩니다.

### jobs

시뮬레이션 작업 목록입니다.

유형: [SimulationJob](#) 객체 어레이

### unprocessedJobs

처리되지 않은 시뮬레이션 작업 Amazon 리소스 이름(ARN) 목록입니다.

유형: 문자열 어레이

배열 멤버: 최소수는 1개입니다. 최대 항목 수는 100개입니다.

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

## Errors

모든 작업에서 발생하는 일반적인 오류에 대한 자세한 내용은 [일반적인 오류](#) 섹션을 참조하세요.

### InternalServerError

RoboMaker AWS에서 서비스 문제가 발생했습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 500

#### InvalidParameterException

요청에 지정된 파라미터는 유효하지 않거나, 지원되지 않거나, 사용할 수 없습니다. 반환된 메시지는 오류 값에 대한 설명을 제공합니다.

HTTP 상태 코드: 400

#### ResourceNotFoundException

지정한 리소스가 존재하지 않습니다.

HTTP 상태 코드: 400

#### ThrottlingException

RoboMaker AWS는 일시적으로 요청을 처리할 수 없습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 400

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS Go v2를 위한 SDK](#)
- [AWS Java V2용 SDK](#)
- [AWS V3용 SDK JavaScript](#)
- [AWS PHP V3용 SDK](#)
- [AWS Python용 SDK](#)
- [AWS 루비 V3용 SDK](#)

## CancelDeploymentJob

이 작업은 사용 중단되었습니다.

### Important

해당 API는 더 이상 지원되지 않습니다. 자세한 내용은 [지원 정책](#) 페이지에서 2022년 5월 2일 업데이트를 참조하세요.

지정된 배포 작업을 취소합니다.

## Request Syntax

```
POST /cancelDeploymentJob HTTP/1.1
Content-type: application/json

{
  "job": "string"
}
```

## URI 요청 파라미터

요청은 URI 파라미터를 사용하지 않습니다.

## 요청 본문

요청은 JSON 형식으로 다음 데이터를 받습니다.

### job

취소할 배포 작업 ARN입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

필수 항목 여부: 예

## 응답 구문

```
HTTP/1.1 200
```

### Response Elements

작업이 성공하면 서비스가 비어 있는 HTTP 본문과 함께 HTTP 200 응답을 반환합니다.

### Errors

모든 작업에서 발생하는 일반적인 오류에 대한 자세한 내용은 [일반적인 오류](#) 섹션을 참조하세요.

#### InternalServerError

RoboMaker AWS에서 서비스 문제가 발생했습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 500

#### InvalidParameterException

요청에 지정된 파라미터는 유효하지 않거나, 지원되지 않거나, 사용할 수 없습니다. 반환된 메시지는 오류 값에 대한 설명을 제공합니다.

HTTP 상태 코드: 400

#### ResourceNotFoundException

지정한 리소스가 존재하지 않습니다.

HTTP 상태 코드: 400

#### ThrottlingException

RoboMaker AWS는 일시적으로 요청을 처리할 수 없습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 400

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)

- [AWS SDK for C++](#)
- [AWS Go v2를 위한 SDK](#)
- [AWS Java V2용 SDK](#)
- [AWS V3용 SDK JavaScript](#)
- [AWS PHP V3용 SDK](#)
- [AWS Python용 SDK](#)
- [AWS 루비 V3용 SDK](#)

## CancelSimulationJob

지정된 시뮬레이션 작업을 취소합니다.

### Request Syntax

```
POST /cancelSimulationJob HTTP/1.1
Content-type: application/json

{
  "job": "string"
}
```

### URI 요청 파라미터

요청은 URI 파라미터를 사용하지 않습니다.

### 요청 본문

요청은 JSON 형식으로 다음 데이터를 받습니다.

#### job

취소할 시뮬레이션 작업 ARN입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

필수 항목 여부: 예

### 응답 구문

```
HTTP/1.1 200
```

### Response Elements

작업이 성공하면 서비스가 비어 있는 HTTP 본문과 함께 HTTP 200 응답을 반환합니다.

## Errors

모든 작업에서 발생하는 일반적인 오류에 대한 자세한 내용은 [일반적인 오류](#) 섹션을 참조하세요.

### InternalServerErrorException

RoboMaker AWS에서 서비스 문제가 발생했습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 500

### InvalidParameterException

요청에 지정된 파라미터는 유효하지 않거나, 지원되지 않거나, 사용할 수 없습니다. 반환된 메시지는 오류 값에 대한 설명을 제공합니다.

HTTP 상태 코드: 400

### ResourceNotFoundException

지정한 리소스가 존재하지 않습니다.

HTTP 상태 코드: 400

### ThrottlingException

RoboMaker AWS는 일시적으로 요청을 처리할 수 없습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 400

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS Go v2를 위한 SDK](#)
- [AWS Java V2용 SDK](#)
- [AWS V3용 SDK JavaScript](#)
- [AWS PHP V3용 SDK](#)
- [AWS Python용 SDK](#)

- [AWS 루비 V3용 SDK](#)

## CancelSimulationJobBatch

시뮬레이션 작업 배치를 취소합니다. 시뮬레이션 작업 배치를 취소하면 배치의 일부로 생성된 모든 활성 시뮬레이션 작업도 취소됩니다.

### Request Syntax

```
POST /cancelSimulationJobBatch HTTP/1.1
Content-type: application/json
```

```
{
  "batch": "string"
}
```

### URI 요청 파라미터

요청은 URI 파라미터를 사용하지 않습니다.

### 요청 본문

요청은 JSON 형식으로 다음 데이터를 받습니다.

#### batch

취소할 배치의 ID입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

필수 항목 여부: 예

### 응답 구문

```
HTTP/1.1 200
```

### Response Elements

작업이 성공하면 서비스가 비어 있는 HTTP 본문과 함께 HTTP 200 응답을 반환합니다.

## Errors

모든 작업에서 발생하는 일반적인 오류에 대한 자세한 내용은 [일반적인 오류](#) 섹션을 참조하세요.

### InternalServerErrorException

RoboMaker AWS에서 서비스 문제가 발생했습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 500

### InvalidParameterException

요청에 지정된 파라미터는 유효하지 않거나, 지원되지 않거나, 사용할 수 없습니다. 반환된 메시지는 오류 값에 대한 설명을 제공합니다.

HTTP 상태 코드: 400

### ResourceNotFoundException

지정한 리소스가 존재하지 않습니다.

HTTP 상태 코드: 400

### ThrottlingException

RoboMaker AWS는 일시적으로 요청을 처리할 수 없습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 400

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS Go v2를 위한 SDK](#)
- [AWS Java V2용 SDK](#)
- [AWS V3용 SDK JavaScript](#)
- [AWS PHP V3용 SDK](#)
- [AWS Python용 SDK](#)

- [AWS 루비 V3용 SDK](#)

## CancelWorldExportJob

지정된 내보내기 작업을 취소합니다.

### Request Syntax

```
POST /cancelWorldExportJob HTTP/1.1
Content-type: application/json

{
  "job": "string"
}
```

### URI 요청 파라미터

요청은 URI 파라미터를 사용하지 않습니다.

### 요청 본문

요청은 JSON 형식으로 다음 데이터를 받습니다.

#### job

취소할 전 월드 내보내기 작업의 Amazon 리소스 이름(ARN)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

필수 항목 여부: 예

### 응답 구문

```
HTTP/1.1 200
```

### Response Elements

작업이 성공하면 서비스가 비어 있는 HTTP 본문과 함께 HTTP 200 응답을 반환합니다.

## Errors

모든 작업에서 발생하는 일반적인 오류에 대한 자세한 내용은 [일반적인 오류](#) 섹션을 참조하세요.

### InternalServerErrorException

RoboMaker AWS에서 서비스 문제가 발생했습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 500

### InvalidParameterException

요청에 지정된 파라미터는 유효하지 않거나, 지원되지 않거나, 사용할 수 없습니다. 반환된 메시지는 오류 값에 대한 설명을 제공합니다.

HTTP 상태 코드: 400

### ResourceNotFoundException

지정한 리소스가 존재하지 않습니다.

HTTP 상태 코드: 400

### ThrottlingException

RoboMaker AWS는 일시적으로 요청을 처리할 수 없습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 400

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS Go v2를 위한 SDK](#)
- [AWS Java V2용 SDK](#)
- [AWS V3용 SDK JavaScript](#)
- [AWS PHP V3용 SDK](#)
- [AWS Python용 SDK](#)

- [AWS 루비 V3용 SDK](#)

## CancelWorldGenerationJob

지정된 월드 생성 작업을 취소합니다.

### Request Syntax

```
POST /cancelWorldGenerationJob HTTP/1.1
Content-type: application/json

{
  "job": "string"
}
```

### URI 요청 파라미터

요청은 URI 파라미터를 사용하지 않습니다.

### 요청 본문

요청은 JSON 형식으로 다음 데이터를 받습니다.

#### job

취소할 월드 생성기 작업의 Amazon 리소스 이름(ARN)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

필수 항목 여부: 예

### 응답 구문

```
HTTP/1.1 200
```

### Response Elements

작업이 성공하면 서비스가 비어 있는 HTTP 본문과 함께 HTTP 200 응답을 반환합니다.

## Errors

모든 작업에서 발생하는 일반적인 오류에 대한 자세한 내용은 [일반적인 오류](#) 섹션을 참조하세요.

### InternalServerErrorException

RoboMaker AWS에서 서비스 문제가 발생했습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 500

### InvalidParameterException

요청에 지정된 파라미터는 유효하지 않거나, 지원되지 않거나, 사용할 수 없습니다. 반환된 메시지는 오류 값에 대한 설명을 제공합니다.

HTTP 상태 코드: 400

### ResourceNotFoundException

지정한 리소스가 존재하지 않습니다.

HTTP 상태 코드: 400

### ThrottlingException

RoboMaker AWS는 일시적으로 요청을 처리할 수 없습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 400

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS Go v2를 위한 SDK](#)
- [AWS Java V2용 SDK](#)
- [AWS V3용 SDK JavaScript](#)
- [AWS PHP V3용 SDK](#)
- [AWS Python용 SDK](#)

- [AWS 루비 V3용 SDK](#)

## CreateDeploymentJob

이 작업은 사용 중단되었습니다.

### Important

이 API는 더 이상 지원되지 않으며 사용할 경우 오류가 발생합니다. 자세한 내용은 [지원 정책](#) 페이지의 2022년 1월 31일 업데이트를 참조하세요.

특정 버전의 로봇 애플리케이션을 플릿의 로봇에 배포합니다.

로봇 애플리케이션에는 일관성을 위해 번호가 지정된 `applicationVersion`이 있어야 합니다. 새 버전을 만들려면 `CreateRobotApplicationVersion`을 사용하거나 [로봇 애플리케이션 버전 생성](#)을 참조하세요.

### Note

배포 작업은 90일 후 만료되고 삭제됩니다. 더 이상 액세스할 수 없습니다.

## Request Syntax

```
POST /createDeploymentJob HTTP/1.1
Content-type: application/json

{
  "clientRequestToken": "string",
  "deploymentApplicationConfigs": [
    {
      "application": "string",
      "applicationVersion": "string",
      "launchConfig": {
        "environmentVariables": {
          "string": "string"
        },
        "launchFile": "string",
        "packageName": "string",
        "postLaunchFile": "string",
        "preLaunchFile": "string"
      }
    }
  ]
}
```

```

    }
  ],
  "deploymentConfig": {
    "concurrentDeploymentPercentage": number,
    "downloadConditionFile": {
      "bucket": "string",
      "etag": "string",
      "key": "string"
    },
  },
  "failureThresholdPercentage": number,
  "robotDeploymentTimeoutInSeconds": number
},
"fleet": "string",
"tags": {
  "string" : "string"
}
}
}

```

## URI 요청 파라미터

요청은 URI 파라미터를 사용하지 않습니다.

## 요청 본문

요청은 JSON 형식으로 다음 데이터를 받습니다.

### clientRequestToken

요청 멩등성을 보장하기 위해 제공하는 고유한 대소문자 구분 식별자입니다.

유형: 문자열

길이 제한: 최소 길이는 1. 최대 길이는 64.

패턴: [a-zA-Z0-9\_\-=]\*

필수 사항 여부: Yes

### deploymentApplicationConfigs

배포 애플리케이션 구성입니다.

유형: [DeploymentApplicationConfig](#) 객체 어레이

배열 멤버: 고정된 항목 수는 1개입니다.

필수 여부: 예

### deploymentConfig

요청된 배포 구성입니다.

유형: [DeploymentConfig](#) 객체

필수 항목 여부: 아니요

### fleet

배포할 플릿의 Amazon 리소스 이름(ARN)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

필수 사항 여부: Yes

### tags

배포 작업에 연결된 태그 키 및 태그 값을 포함하는 맵입니다.

유형: 문자열-문자열 맵

맵 항목: 최소 항목 수는 0개입니다. 최대 항목 수는 50개입니다.

키 길이 제약 조건: 최소 길이는 1. 최대 길이 128.

키 패턴: [a-zA-Z0-9 \_.\-\/+=:]\*

값 길이 제약 조건: 최소 길이는 0입니다. 최대 길이는 256.

값 패턴: [a-zA-Z0-9 \_.\-\/+=:]\*

필수 여부: 아니요

## 응답 구문

```
HTTP/1.1 200
```

Content-type: application/json

```
{
  "arn": "string",
  "createdAt": number,
  "deploymentApplicationConfigs": [
    {
      "application": "string",
      "applicationVersion": "string",
      "launchConfig": {
        "environmentVariables": {
          "string" : "string"
        },
        "launchFile": "string",
        "packageName": "string",
        "postLaunchFile": "string",
        "preLaunchFile": "string"
      }
    }
  ],
  "deploymentConfig": {
    "concurrentDeploymentPercentage": number,
    "downloadConditionFile": {
      "bucket": "string",
      "etag": "string",
      "key": "string"
    },
    "failureThresholdPercentage": number,
    "robotDeploymentTimeoutInSeconds": number
  },
  "failureCode": "string",
  "failureReason": "string",
  "fleet": "string",
  "status": "string",
  "tags": {
    "string" : "string"
  }
}
```

## 응답 요소

작업이 성공하면 서비스가 HTTP 200 응답을 반환합니다.

다음 데이터는 서비스에 의해 JSON 형식으로 반환됩니다.

## arn

배포 작업의 Amazon 리소스 이름(ARN)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

## createdAt

플릿이 생성된 epoch 이후 경과 시간(밀리초)입니다.

유형: 타임스탬프

## deploymentApplicationConfigs

배포 애플리케이션 구성입니다.

유형: [DeploymentApplicationConfig](#) 객체 어레이

배열 멤버: 고정된 항목 수는 1개입니다.

## deploymentConfig

배포 구성입니다.

유형: [DeploymentConfig](#) 객체

## failureCode

시뮬레이션 작업 실패 시 발생하는 오류 코드:

BadPermissionError

AWS Greengrass에 다른 서비스에 대한 액세스 권한을 위한 서비스 수준의 역할 권한이 필요합니다. 역할에는 [AWSGreengrassResourceAccessRolePolicy 관리형 정책](#)이 포함되어야 합니다.

ExtractingBundleFailure

번들에서 로봇 애플리케이션을 추출할 수 없습니다.

FailureThresholdBreached

업데이트할 수 없는 로봇의 비율이 배포에 설정된 비율을 초과했습니다.

### GreengrassDeploymentFailed

로봇 애플리케이션을 로봇에 배포할 수 없습니다.

### GreengrassGroupVersionDoesNotExist

로봇과 관련된 AWS Greengrass 그룹 또는 버전이 누락되었습니다.

### InternalServerError

내부 서비스 오류가 발생했습니다. 요청을 다시 시도하세요. 그래도 문제가 계속되면 자세한 내용으로 문의해 주세요.

### MissingRobotApplicationArchitecture

로봇 애플리케이션에 로봇의 아키텍처와 일치하는 소스가 없습니다.

### MissingRobotDeploymentResource

로봇 애플리케이션에 지정된 리소스 중 하나 이상이 누락되었습니다. 예를 들어, 로봇 애플리케이션에 올바른 시작 패키지와 시작 파일이 있습니까?

### PostLaunchFileFailure

실행 후 스크립트가 실패했습니다.

### PreLaunchFileFailure

사전 실행 스크립트가 실패했습니다.

### ResourceNotFound

배포 리소스가 하나 이상 누락되었습니다. 예를 들어, 로봇 애플리케이션 소스 번들이 아직 존재합니까?

### RobotDeploymentNoResponse

로봇으로부터 응답이 없습니다. 전원이 켜져 있지 않거나 인터넷에 연결되어 있지 않을 수 있습니다.

타입: 문자열

유효 값: ResourceNotFound | EnvironmentSetupError | EtagMismatch  
| FailureThresholdBreach | RobotDeploymentAborted |  
RobotDeploymentNoResponse | RobotAgentConnectionTimeout  
| GreengrassDeploymentFailed | InvalidGreengrassGroup |  
MissingRobotArchitecture | MissingRobotApplicationArchitecture |

MissingRobotDeploymentResource | GreengrassGroupVersionDoesNotExist  
 | LambdaDeleted | ExtractingBundleFailure | PreLaunchFileFailure |  
 PostLaunchFileFailure | BadPermissionError | DownloadConditionFailed |  
 BadLambdaAssociated | InternalServerError | RobotApplicationDoesNotExist  
 | DeploymentFleetDoesNotExist | FleetDeploymentTimeout

### [failureReason](#)

배포 작업이 실패한 경우의 실패 이유입니다.

타입: 문자열

길이 제약 조건: 최소 길이는 0입니다. 최대 길이는 1024입니다.

패턴: .\*

### [fleet](#)

배포 작업의 대상 플릿입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

### [status](#)

배포 작업의 상태입니다.

타입: 문자열

유효 값: Pending | Preparing | InProgress | Failed | Succeeded | Canceled

### [tags](#)

배포 작업에 추가된 모든 태그의 목록입니다.

유형: 문자열-문자열 맵

맵 항목: 최소 항목 수는 0개입니다. 최대 항목 수는 50개입니다.

키 길이 제약 조건: 최소 길이는 1. 최대 길이 128.

키 패턴: [a-zA-Z0-9 \_.\-\/+=:]\*

값 길이 제약 조건: 최소 길이는 0입니다. 최대 길이는 256.

가치 패턴: [a-zA-Z0-9 \_.\-\/+=:]\*

## Errors

모든 작업에서 일반적으로 발생하는 오류에 대한 자세한 내용은 [일반적인 오류](#) 섹션을 참조하세요.

### ConcurrentDeploymentException

실패율 임계값 백분율이 충족되었습니다.

HTTP 상태 코드: 400

### IdempotentParameterMismatchException

요청은 이전 요청과 동일한 클라이언트 토큰을 사용하지만, 동일하지는 않습니다. 요청이 동일하지 않는 한, 다른 요청에 클라이언트 토큰을 재사용하지 마세요.

HTTP 상태 코드: 400

### InternalServerErrorException

RoboMaker AWS에서 서비스 문제가 발생했습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 500

### InvalidParameterException

요청에 지정된 파라미터는 유효하지 않거나, 지원되지 않거나, 사용할 수 없습니다. 반환된 메시지는 오류 값에 대한 설명을 제공합니다.

HTTP 상태 코드: 400

### LimitExceededException

요청된 리소스가 허용된 최대 수를 초과하거나 동시 스트림 요청 수가 허용된 최대 수를 초과합니다.

HTTP 상태 코드: 400

### ResourceNotFoundException

지정한 리소스가 존재하지 않습니다.

HTTP 상태 코드: 400

## ThrottlingException

RoboMaker AWS는 일시적으로 요청을 처리할 수 없습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 400

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS Go v2를 위한 SDK](#)
- [AWS Java V2용 SDK](#)
- [AWS V3용 SDK JavaScript](#)
- [AWS PHP V3용 SDK](#)
- [AWS Python용 SDK](#)
- [AWS 루비 V3용 SDK](#)

## CreateFleet

이 작업은 사용 중단되었습니다.

### Important

이 API는 더 이상 지원되지 않으며 사용할 경우 오류가 발생합니다. 자세한 내용은 [지원 정책](#) 페이지의 2022년 1월 31일 업데이트를 참조하세요.

동일한 로봇 애플리케이션을 실행하는 로봇의 논리적 그룹을 나타내는 플릿을 생성합니다.

## Request Syntax

```
POST /createFleet HTTP/1.1
Content-type: application/json

{
  "name": "string",
  "tags": {
    "string" : "string"
  }
}
```

## URI 요청 파라미터

요청은 URI 파라미터를 사용하지 않습니다.

## 요청 본문

요청은 JSON 형식으로 다음 데이터를 받습니다.

### name

플릿의 이름입니다.

유형: 문자열

길이 제약: 최소 길이는 1. 최대 길이는 255.

패턴: [a-zA-Z0-9\_\-]\*

필수 사항 여부: Yes

### tags

플릿에 연결된 태그 키 및 태그 값을 포함하는 맵입니다.

유형: 문자열-문자열 맵

맵 항목: 최소 항목 수는 0개입니다. 최대 항목 수는 50개입니다.

키 길이 제약 조건: 최소 길이는 1. 최대 길이 128.

키 패턴: [a-zA-Z0-9 \_.\-\/+=:]\*

값 길이 제약 조건: 최소 길이는 0입니다. 최대 길이는 256.

값 패턴: [a-zA-Z0-9 \_.\-\/+=:]\*

필수 여부: 아니요

## 응답 구문

```
HTTP/1.1 200
Content-type: application/json

{
  "arn": "string",
  "createdAt": number,
  "name": "string",
  "tags": {
    "string" : "string"
  }
}
```

## 응답 요소

작업이 성공하면 서비스가 HTTP 200 응답을 반환합니다.

다음 데이터는 서비스에 의해 JSON 형식으로 반환됩니다.

### arn

플릿의 Amazon 리소스 이름(ARN)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

### createdAt

플릿이 생성된 epoch 이후 경과 시간(밀리초)입니다.

유형: 타임스탬프

### name

플릿의 이름입니다.

유형: 문자열

길이 제약: 최소 길이는 1. 최대 길이는 255.

패턴: [a-zA-Z0-9\_\-]\*

### tags

플릿에 추가된 모든 태그의 목록입니다.

유형: 문자열-문자열 맵

맵 항목: 최소 항목 수는 0개입니다. 최대 항목 수는 50개입니다.

키 길이 제약 조건: 최소 길이는 1. 최대 길이 128.

키 패턴: [a-zA-Z0-9 \_.\-\/+=:]\*

값 길이 제약 조건: 최소 길이는 0입니다. 최대 길이는 256.

가치 패턴: [a-zA-Z0-9 \_.\-\/+=:]\*

## Errors

모든 작업에서 발생하는 일반적인 오류에 대한 자세한 내용은 [일반적인 오류](#) 섹션을 참조하세요.

### InternalServerError

RoboMaker AWS에서 서비스 문제가 발생했습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 500

#### InvalidParameterException

요청에 지정된 파라미터는 유효하지 않거나, 지원되지 않거나, 사용할 수 없습니다. 반환된 메시지는 오류 값에 대한 설명을 제공합니다.

HTTP 상태 코드: 400

#### LimitExceededException

요청된 리소스가 허용된 최대 수를 초과하거나 동시 스트림 요청 수가 허용된 최대 수를 초과합니다.

HTTP 상태 코드: 400

#### ThrottlingException

RoboMaker AWS는 일시적으로 요청을 처리할 수 없습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 400

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS Go v2를 위한 SDK](#)
- [AWS Java V2용 SDK](#)
- [AWS V3용 SDK JavaScript](#)
- [AWS PHP V3용 SDK](#)
- [AWS Python용 SDK](#)
- [AWS 루비 V3용 SDK](#)

# CreateRobot

이 작업은 사용 중단되었습니다.

## Important

이 API는 더 이상 지원되지 않으며 사용할 경우 오류가 발생합니다. 자세한 내용은 [지원 정책](#) 페이지의 2022년 1월 31일 업데이트를 참조하세요.

로봇을 생성합니다.

## Request Syntax

```
POST /createRobot HTTP/1.1
Content-type: application/json

{
  "architecture": "string",
  "greengrassGroupId": "string",
  "name": "string",
  "tags": {
    "string" : "string"
  }
}
```

## URI 요청 파라미터

요청은 URI 파라미터를 사용하지 않습니다.

## 요청 본문

요청은 JSON 형식으로 다음 데이터를 받습니다.

### architecture

로봇의 대상 아키텍처입니다.

타입: 문자열

유효 값: X86\_64 | ARM64 | ARMHF

필수 사항 여부: 예

### greengrassGroupId

Greengrass 그룹 ID입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: .\*

필수 사항 여부: Yes

### name

로봇의 이름입니다.

유형: 문자열

길이 제약: 최소 길이는 1. 최대 길이는 255.

패턴: [a-zA-Z0-9\_\-]\*

필수 사항 여부: Yes

### tags

로봇에 연결된 태그 키 및 태그 값을 포함하는 맵입니다.

유형: 문자열-문자열 맵

맵 항목: 최소 항목 수는 0개입니다. 최대 항목 수는 50개입니다.

키 길이 제약 조건: 최소 길이는 1. 최대 길이 128.

키 패턴: [a-zA-Z0-9 \_.\-\/+=:]\*

값 길이 제약 조건: 최소 길이는 0입니다. 최대 길이는 256.

값 패턴: [a-zA-Z0-9 \_.\-\/+=:]\*

필수 여부: 아니요

## 응답 구문

```
HTTP/1.1 200
```

```
Content-type: application/json

{
  "architecture": "string",
  "arn": "string",
  "createdAt": number,
  "greengrassGroupId": "string",
  "name": "string",
  "tags": {
    "string" : "string"
  }
}
```

## 응답 요소

작업이 성공하면 서비스가 HTTP 200 응답을 반환합니다.

다음 데이터는 서비스에 의해 JSON 형식으로 반환됩니다.

### architecture

로봇의 대상 아키텍처입니다.

타입: 문자열

유효 값: X86\_64 | ARM64 | ARMHF

### arn

로봇의 Amazon 리소스 이름(ARN)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

### createdAt

로봇이 생성된 epoch 이후 경과 시간(밀리초)입니다.

유형: 타임스탬프

### greengrassGroupId

로봇과 관련된 Greengrass 그룹의 Amazon 리소스 이름(ARN)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: .\*

### name

로봇의 이름입니다.

유형: 문자열

길이 제약: 최소 길이는 1. 최대 길이는 255.

패턴: [a-zA-Z0-9\_\-]\*

### tags

로봇에 추가된 모든 태그의 목록입니다.

유형: 문자열-문자열 맵

맵 항목: 최소 항목 수는 0개입니다. 최대 항목 수는 50개입니다.

키 길이 제약 조건: 최소 길이는 1. 최대 길이 128.

키 패턴: [a-zA-Z0-9 \_.\-\/+=:]\*

값 길이 제약 조건: 최소 길이는 0입니다. 최대 길이는 256.

가치 패턴: [a-zA-Z0-9 \_.\-\/+=:]\*

## Errors

모든 작업에서 발생하는 일반적인 오류에 대한 자세한 내용은 [일반적인 오류](#) 섹션을 참조하세요.

### InternalServerErrorException

RoboMaker AWS에서 서비스 문제가 발생했습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 500

### InvalidParameterException

요청에 지정된 파라미터는 유효하지 않거나, 지원되지 않거나, 사용할 수 없습니다. 반환된 메시지는 오류 값에 대한 설명을 제공합니다.

HTTP 상태 코드: 400

LimitExceededException

요청된 리소스가 허용된 최대 수를 초과하거나 동시 스트림 요청 수가 허용된 최대 수를 초과합니다.

HTTP 상태 코드: 400

ResourceAlreadyExistsException

지정된 리소스가 이미 있습니다.

HTTP 상태 코드: 400

ThrottlingException

RoboMaker AWS는 일시적으로 요청을 처리할 수 없습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 400

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS Go v2를 위한 SDK](#)
- [AWS Java V2용 SDK](#)
- [AWS V3용 SDK JavaScript](#)
- [AWS PHP V3용 SDK](#)
- [AWS Python용 SDK](#)
- [AWS 루비 V3용 SDK](#)

## CreateRobotApplication

로봇 애플리케이션을 생성합니다.

### Request Syntax

```
POST /createRobotApplication HTTP/1.1
Content-type: application/json
```

```
{
  "environment": {
    "uri": "string"
  },
  "name": "string",
  "robotSoftwareSuite": {
    "name": "string",
    "version": "string"
  },
  "sources": [
    {
      "architecture": "string",
      "s3Bucket": "string",
      "s3Key": "string"
    }
  ],
  "tags": {
    "string" : "string"
  }
}
```

### URI 요청 파라미터

요청은 URI 파라미터를 사용하지 않습니다.

### 요청 본문

요청은 JSON 형식으로 다음 데이터를 받습니다.

#### environment

로봇 애플리케이션에 사용하는 도커 이미지의 URI를 포함하는 객체입니다.

유형: [Environment](#) 객체

필수 항목 여부: 아니요

### name

로봇 애플리케이션의 이름입니다.

유형: 문자열

길이 제약: 최소 길이는 1. 최대 길이는 255.

패턴: [a-zA-Z0-9\_\-]\*

필수 사항 여부: Yes

### robotSoftwareSuite

로봇 애플리케이션이 사용하는 로봇 소프트웨어 제품군입니다.

유형: [RobotSoftwareSuite](#) 객체

필수 여부: 예

### sources

로봇 애플리케이션의 소스입니다.

타입: [SourceConfig](#) 객체 배열

필수: 아니요

### tags

로봇 애플리케이션에 연결된 태그 키 및 태그 값을 포함하는 맵입니다.

유형: 문자열-문자열 맵

맵 항목: 최소 항목 수는 0개입니다. 최대 항목 수는 50개입니다.

키 길이 제약 조건: 최소 길이는 1. 최대 길이 128.

키 패턴: [a-zA-Z0-9 \_.\-\/+=:]\*

값 길이 제약 조건: 최소 길이는 0입니다. 최대 길이는 256.

값 패턴: [a-zA-Z0-9 \_.\-\/+=:]\*

필수 여부: 아니요

## 응답 구문

```
HTTP/1.1 200
Content-type: application/json

{
  "arn": "string",
  "environment": {
    "uri": "string"
  },
  "lastUpdatedAt": number,
  "name": "string",
  "revisionId": "string",
  "robotSoftwareSuite": {
    "name": "string",
    "version": "string"
  },
  "sources": [
    {
      "architecture": "string",
      "etag": "string",
      "s3Bucket": "string",
      "s3Key": "string"
    }
  ],
  "tags": {
    "string" : "string"
  },
  "version": "string"
}
```

## 응답 요소

작업이 성공하면 서비스가 HTTP 200 응답을 반환합니다.

다음 데이터는 서비스에 의해 JSON 형식으로 반환됩니다.

### arn

로봇 애플리케이션의 Amazon 리소스 이름(ARN)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: `arn:.*`

### environment

로봇 애플리케이션을 생성하는 데 사용되는 도커 이미지 URI를 포함하는 객체입니다.

유형: [Environment](#) 객체

### lastUpdatedAt

로봇 애플리케이션이 마지막으로 업데이트된 epoch 이후 경과 시간(밀리초)입니다.

유형: 타임스탬프

### name

로봇 애플리케이션의 이름입니다.

유형: 문자열

길이 제약: 최소 길이는 1. 최대 길이는 255.

패턴: `[a-zA-Z0-9_\-\ ]*`

### revisionId

로봇 애플리케이션의 개정 ID입니다.

유형: 문자열

길이 제약: 최소 길이 1자. 최대 길이는 40입니다.

패턴: `[a-zA-Z0-9_.\-\ ]*`

### robotSoftwareSuite

로봇 애플리케이션이 사용하는 로봇 소프트웨어 제품군입니다.

유형: [RobotSoftwareSuite](#) 객체

### sources

로봇 애플리케이션의 소스입니다.

유형: [Source](#) 객체 어레이

### [tags](#)

로봇 애플리케이션에 추가된 모든 태그의 목록입니다.

유형: 문자열-문자열 맵

맵 항목: 최소 항목 수는 0개입니다. 최대 항목 수는 50개입니다.

키 길이 제약 조건: 최소 길이는 1. 최대 길이 128.

키 패턴: [a-zA-Z0-9 \_.\-\/+=:]\*

값 길이 제약 조건: 최소 길이는 0입니다. 최대 길이는 256.

가치 패턴: [a-zA-Z0-9 \_.\-\/+=:]\*

### [version](#)

로봇 애플리케이션의 버전입니다.

유형: 문자열

길이 제약: 최소 길이는 1. 최대 길이는 255.

패턴: (\\$LATEST)|[0-9]\*

## Errors

모든 작업에서 일반적으로 발생하는 오류에 대한 자세한 내용은 [일반적인 오류](#) 섹션을 참조하세요.

### IdempotentParameterMismatchException

요청은 이전 요청과 동일한 클라이언트 토큰을 사용하지만, 동일하지는 않습니다. 요청이 동일하지 않는 한, 다른 요청에 클라이언트 토큰을 재사용하지 마세요.

HTTP 상태 코드: 400

### InternalServerErrorException

RoboMaker AWS에서 서비스 문제가 발생했습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 500

## InvalidParameterException

요청에 지정된 파라미터는 유효하지 않거나, 지원되지 않거나, 사용할 수 없습니다. 반환된 메시지는 오류 값에 대한 설명을 제공합니다.

HTTP 상태 코드: 400

## LimitExceededException

요청된 리소스가 허용된 최대 수를 초과하거나 동시 스트림 요청 수가 허용된 최대 수를 초과합니다.

HTTP 상태 코드: 400

## ResourceAlreadyExistsException

지정된 리소스가 이미 있습니다.

HTTP 상태 코드: 400

## ThrottlingException

RoboMaker AWS는 일시적으로 요청을 처리할 수 없습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 400

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS Go v2를 위한 SDK](#)
- [AWS Java V2용 SDK](#)
- [AWS V3용 SDK JavaScript](#)
- [AWS PHP V3용 SDK](#)
- [AWS Python용 SDK](#)
- [AWS 루비 V3용 SDK](#)

# CreateRobotApplicationVersion

로봇 애플리케이션의 버전을 생성합니다.

## Request Syntax

```
POST /createRobotApplicationVersion HTTP/1.1
Content-type: application/json
```

```
{
  "application": "string",
  "currentRevisionId": "string",
  "imageDigest": "string",
  "s3Etags": [ "string" ]
}
```

## URI 요청 파라미터

요청은 URI 파라미터를 사용하지 않습니다.

## 요청 본문

요청은 JSON 형식으로 다음 데이터를 받습니다.

### application

로봇 애플리케이션에 대한 애플리케이션 정보입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

필수 사항 여부: Yes

### currentRevisionId

로봇 애플리케이션에 대한 현재 개정 ID입니다. 최신 개정 ID와 일치하는 값을 제공할 경우 새 버전이 생성됩니다.

유형: 문자열

길이 제약: 최소 길이 1자. 최대 길이는 40입니다.

패턴: [a-zA-Z0-9\_.\-]\*

Required: No

### imageDigest

로봇 애플리케이션에 사용하는 도커 이미지의 SHA256 식별자입니다.

타입: 문자열

길이 제한: 최소 길이는 0. 최대 길이는 72입니다.

패턴: [Ss][Hh][Aa]256:[0-9a-fA-F]{64}

Required: No

### s3Etags

로봇 애플리케이션에 사용하는 zip 파일 번들의 Amazon S3 식별자입니다.

유형: String 배열

필수 여부: 아니요

## 응답 구문

```
HTTP/1.1 200
Content-type: application/json

{
  "arn": "string",
  "environment": {
    "uri": "string"
  },
  "lastUpdatedAt": number,
  "name": "string",
  "revisionId": "string",
  "robotSoftwareSuite": {
    "name": "string",
    "version": "string"
  },
  "sources": [
```

```

    {
      "architecture": "string",
      "etag": "string",
      "s3Bucket": "string",
      "s3Key": "string"
    }
  ],
  "version": "string"
}

```

## 응답 요소

작업이 성공하면 서비스가 HTTP 200 응답을 반환합니다.

다음 데이터는 서비스에 의해 JSON 형식으로 반환됩니다.

### [arn](#)

로봇 애플리케이션의 Amazon 리소스 이름(ARN)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

### [environment](#)

로봇 애플리케이션을 생성하는 데 사용되는 도커 이미지 URI를 포함하는 객체입니다.

유형: [Environment](#) 객체

### [lastUpdatedAt](#)

로봇 애플리케이션이 마지막으로 업데이트된 epoch 이후 경과 시간(밀리초)입니다.

유형: 타임스탬프

### [name](#)

로봇 애플리케이션의 이름입니다.

유형: 문자열

길이 제약: 최소 길이는 1. 최대 길이는 255.

패턴: [a-zA-Z0-9\_\-]\*

### revisionId

로봇 애플리케이션의 개정 ID입니다.

유형: 문자열

길이 제약: 최소 길이 1자. 최대 길이는 40입니다.

패턴: [a-zA-Z0-9\_.\-]\*

### robotSoftwareSuite

로봇 애플리케이션이 사용하는 로봇 소프트웨어 제품군입니다.

유형: [RobotSoftwareSuite](#) 객체

### sources

로봇 애플리케이션의 소스입니다.

유형: [Source](#) 객체 어레이

### version

로봇 애플리케이션의 버전입니다.

유형: 문자열

길이 제약: 최소 길이는 1. 최대 길이는 255.

패턴: (\\\$LATEST)|[0-9]\*

## Errors

모든 작업에서 일반적으로 발생하는 오류에 대한 자세한 내용은 [일반적인 오류](#) 섹션을 참조하세요.

### IdempotentParameterMismatchException

요청은 이전 요청과 동일한 클라이언트 토큰을 사용하지만, 동일하지는 않습니다. 요청이 동일하지 않는 한, 다른 요청에 클라이언트 토큰을 재사용하지 마세요.

HTTP 상태 코드: 400

## InternalServerErrorException

RoboMaker AWS에서 서비스 문제가 발생했습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 500

## InvalidParameterException

요청에 지정된 파라미터는 유효하지 않거나, 지원되지 않거나, 사용할 수 없습니다. 반환된 메시지는 오류 값에 대한 설명을 제공합니다.

HTTP 상태 코드: 400

## LimitExceededException

요청된 리소스가 허용된 최대 수를 초과하거나 동시 스트림 요청 수가 허용된 최대 수를 초과합니다.

HTTP 상태 코드: 400

## ThrottlingException

RoboMaker AWS는 일시적으로 요청을 처리할 수 없습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 400

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS Go v2를 위한 SDK](#)
- [AWS Java V2용 SDK](#)
- [AWS V3용 SDK JavaScript](#)
- [AWS PHP V3용 SDK](#)
- [AWS Python용 SDK](#)
- [AWS 루비 V3용 SDK](#)

# CreateSimulationApplication

시뮬레이션 애플리케이션을 생성합니다.

## Request Syntax

```
POST /createSimulationApplication HTTP/1.1
Content-type: application/json
```

```
{
  "environment": {
    "uri": "string"
  },
  "name": "string",
  "renderingEngine": {
    "name": "string",
    "version": "string"
  },
  "robotSoftwareSuite": {
    "name": "string",
    "version": "string"
  },
  "simulationSoftwareSuite": {
    "name": "string",
    "version": "string"
  },
  "sources": [
    {
      "architecture": "string",
      "s3Bucket": "string",
      "s3Key": "string"
    }
  ],
  "tags": {
    "string" : "string"
  }
}
```

## URI 요청 파라미터

요청은 URI 파라미터를 사용하지 않습니다.

## 요청 본문

요청은 JSON 형식으로 다음 데이터를 받습니다.

### environment

시뮬레이션 애플리케이션을 만드는 데 사용되는 도커 이미지 URI를 포함하는 객체입니다.

유형: [Environment](#) 객체

필수 항목 여부: 아니요

### name

시뮬레이션 애플리케이션의 이름입니다.

유형: 문자열

길이 제약: 최소 길이는 1. 최대 길이는 255.

패턴: [a-zA-Z0-9\_\-]\*

필수 사항 여부: Yes

### renderingEngine

시뮬레이션 애플리케이션의 렌더링 엔진입니다.

유형: [RenderingEngine](#) 객체

필수 항목 여부: 아니요

### robotSoftwareSuite

시뮬레이션 애플리케이션에서 사용하는 로봇 소프트웨어 제품군입니다.

유형: [RobotSoftwareSuite](#) 객체

필수 여부: 예

### simulationSoftwareSuite

시뮬레이션 애플리케이션이 사용하는 시뮬레이션 소프트웨어 제품군입니다.

유형: [SimulationSoftwareSuite](#) 객체

필수 여부: 예

### sources

시뮬레이션 애플리케이션의 소스입니다.

타입: [SourceConfig](#) 객체 배열

필수: 아니요

### tags

시뮬레이션 애플리케이션에 연결된 태그 키 및 태그 값을 포함하는 맵입니다.

유형: 문자열-문자열 맵

맵 항목: 최소 항목 수는 0개입니다. 최대 항목 수는 50개입니다.

키 길이 제약 조건: 최소 길이는 1. 최대 길이 128.

키 패턴: `[a-zA-Z0-9 _.\-\/+=:]*`

값 길이 제약 조건: 최소 길이는 0입니다. 최대 길이는 256.

값 패턴: `[a-zA-Z0-9 _.\-\/+=:]*`

필수 여부: 아니요

## 응답 구문

```
HTTP/1.1 200
Content-type: application/json
```

```
{
  "arn": "string",
  "environment": {
    "uri": "string"
  },
  "lastUpdatedAt": number,
  "name": "string",
  "renderingEngine": {
    "name": "string",
    "version": "string"
  },
}
```

```

"revisionId": "string",
"robotSoftwareSuite": {
  "name": "string",
  "version": "string"
},
"simulationSoftwareSuite": {
  "name": "string",
  "version": "string"
},
"sources": [
  {
    "architecture": "string",
    "etag": "string",
    "s3Bucket": "string",
    "s3Key": "string"
  }
],
"tags": {
  "string" : "string"
},
"version": "string"
}

```

## 응답 요소

작업이 성공하면 서비스가 HTTP 200 응답을 반환합니다.

다음 데이터는 서비스에 의해 JSON 형식으로 반환됩니다.

### arn

시뮬레이션 애플리케이션의 Amazon 리소스 이름(ARN)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

### environment

시뮬레이션 애플리케이션을 만드는 데 사용한 도커 이미지 URI가 포함된 객체입니다.

유형: Environment 객체

## lastUpdatedAt

시뮬레이션 애플리케이션이 마지막으로 업데이트된 epoch 이후 경과 시간(밀리초).

유형: 타임스탬프

## name

시뮬레이션 애플리케이션의 이름입니다.

유형: 문자열

길이 제약: 최소 길이는 1. 최대 길이는 255.

패턴: [a-zA-Z0-9\_\-]\*

## renderingEngine

시뮬레이션 애플리케이션의 렌더링 엔진입니다.

유형: [RenderingEngine](#) 객체

## revisionId

시뮬레이션 애플리케이션에 대한 개정 ID입니다.

유형: 문자열

길이 제약: 최소 길이 1자. 최대 길이는 40입니다.

패턴: [a-zA-Z0-9\_.\-]\*

## robotSoftwareSuite

로봇 소프트웨어 제품군에 대한 정보입니다.

유형: [RobotSoftwareSuite](#) 객체

## simulationSoftwareSuite

시뮬레이션 애플리케이션이 사용하는 시뮬레이션 소프트웨어 제품군입니다.

유형: [SimulationSoftwareSuite](#) 객체

## sources

시뮬레이션 애플리케이션의 소스입니다.

유형: [Source](#) 객체 어레이

### [tags](#)

시뮬레이션 애플리케이션에 추가된 모든 태그의 목록입니다.

유형: 문자열-문자열 맵

맵 항목: 최소 항목 수는 0개입니다. 최대 항목 수는 50개입니다.

키 길이 제약 조건: 최소 길이는 1. 최대 길이 128.

키 패턴: [a-zA-Z0-9 \_.\-\/+=:]\*

값 길이 제약 조건: 최소 길이는 0입니다. 최대 길이는 256.

가치 패턴: [a-zA-Z0-9 \_.\-\/+=:]\*

### [version](#)

시뮬레이션 애플리케이션 버전입니다.

유형: 문자열

길이 제약: 최소 길이는 1. 최대 길이는 255.

패턴: (\\$LATEST)|[0-9]\*

## Errors

모든 작업에서 일반적으로 발생하는 오류에 대한 자세한 내용은 [일반적인 오류](#) 섹션을 참조하세요.

### IdempotentParameterMismatchException

요청은 이전 요청과 동일한 클라이언트 토큰을 사용하지만, 동일하지는 않습니다. 요청이 동일하지 않는 한, 다른 요청에 클라이언트 토큰을 재사용하지 마세요.

HTTP 상태 코드: 400

### InternalServerErrorException

RoboMaker AWS에서 서비스 문제가 발생했습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 500

## InvalidParameterException

요청에 지정된 파라미터는 유효하지 않거나, 지원되지 않거나, 사용할 수 없습니다. 반환된 메시지는 오류 값에 대한 설명을 제공합니다.

HTTP 상태 코드: 400

## LimitExceededException

요청된 리소스가 허용된 최대 수를 초과하거나 동시 스트림 요청 수가 허용된 최대 수를 초과합니다.

HTTP 상태 코드: 400

## ResourceAlreadyExistsException

지정된 리소스가 이미 있습니다.

HTTP 상태 코드: 400

## ThrottlingException

RoboMaker AWS는 일시적으로 요청을 처리할 수 없습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 400

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS Go v2를 위한 SDK](#)
- [AWS Java V2용 SDK](#)
- [AWS V3용 SDK JavaScript](#)
- [AWS PHP V3용 SDK](#)
- [AWS Python용 SDK](#)
- [AWS 루비 V3용 SDK](#)

## CreateSimulationApplicationVersion

특정 수정 ID로 시뮬레이션 애플리케이션을 만듭니다.

### Request Syntax

```
POST /createSimulationApplicationVersion HTTP/1.1
Content-type: application/json

{
  "application": "string",
  "currentRevisionId": "string",
  "imageDigest": "string",
  "s3Etags": [ "string" ]
}
```

### URI 요청 파라미터

요청은 URI 파라미터를 사용하지 않습니다.

### 요청 본문

요청은 JSON 형식으로 다음 데이터를 받습니다.

#### [application](#)

시뮬레이션 애플리케이션에 대한 애플리케이션 정보입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

필수 사항 여부: Yes

#### [currentRevisionId](#)

시뮬레이션 애플리케이션에 대한 현재 개정 ID입니다. 최신 개정 ID와 일치하는 값을 제공할 경우 새 버전이 생성됩니다.

유형: 문자열

길이 제약: 최소 길이 1자. 최대 길이는 40입니다.

패턴: `[a-zA-Z0-9_.\-]*`

Required: No

### imageDigest

시뮬레이션 애플리케이션을 만드는 데 사용된 도커 이미지 URI를 식별하는 데 사용되는 SHA256 다이제스트입니다.

타입: 문자열

길이 제한: 최소 길이는 0. 최대 길이는 72입니다.

패턴: `[Ss][Hh][Aa]256:[0-9a-fA-F]{64}`

Required: No

### s3Etags

시뮬레이션 애플리케이션을 생성하는 데 사용하는 zip 파일 번들의 Amazon S3 ETag 식별자입니다.

유형: String 배열

필수 여부: 아니요

## 응답 구문

```
HTTP/1.1 200
Content-type: application/json

{
  "arn": "string",
  "environment": {
    "uri": "string"
  },
  "lastUpdatedAt": number,
  "name": "string",
  "renderingEngine": {
    "name": "string",
    "version": "string"
  },
  "revisionId": "string",
  "robotSoftwareSuite": {
```

```

    "name": "string",
    "version": "string"
  },
  "simulationSoftwareSuite": {
    "name": "string",
    "version": "string"
  },
  "sources": [
    {
      "architecture": "string",
      "etag": "string",
      "s3Bucket": "string",
      "s3Key": "string"
    }
  ],
  "version": "string"
}

```

## 응답 요소

작업이 성공하면 서비스가 HTTP 200 응답을 반환합니다.

다음 데이터는 서비스에 의해 JSON 형식으로 반환됩니다.

### arn

시뮬레이션 애플리케이션의 Amazon 리소스 이름(ARN)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

### environment

시뮬레이션 애플리케이션을 만드는 데 사용되는 도커 이미지 URI가 포함된 객체입니다.

유형: [Environment](#) 객체

### lastUpdatedAt

시뮬레이션 애플리케이션이 마지막으로 업데이트된 epoch 이후 경과 시간(밀리초).

유형: 타임스탬프

## name

시뮬레이션 애플리케이션의 이름입니다.

유형: 문자열

길이 제약: 최소 길이는 1. 최대 길이는 255.

패턴: [a-zA-Z0-9\_\-]\*

## renderingEngine

시뮬레이션 애플리케이션의 렌더링 엔진입니다.

유형: [RenderingEngine](#) 객체

## revisionId

시뮬레이션 애플리케이션에 대한 개정 ID입니다.

유형: 문자열

길이 제약: 최소 길이 1자. 최대 길이는 40입니다.

패턴: [a-zA-Z0-9\_.\-]\*

## robotSoftwareSuite

로봇 소프트웨어 제품군에 대한 정보입니다.

유형: [RobotSoftwareSuite](#) 객체

## simulationSoftwareSuite

시뮬레이션 애플리케이션이 사용하는 시뮬레이션 소프트웨어 제품군입니다.

유형: [SimulationSoftwareSuite](#) 객체

## sources

시뮬레이션 애플리케이션의 소스입니다.

유형: [Source](#) 객체 어레이

## version

시뮬레이션 애플리케이션 버전입니다.

유형: 문자열

길이 제약: 최소 길이는 1. 최대 길이는 255.

패턴: (`\$LATEST`)|[0-9]\*

## Errors

모든 작업에서 일반적으로 발생하는 오류에 대한 자세한 내용은 [일반적인 오류](#) 섹션을 참조하세요.

### IdempotentParameterMismatchException

요청은 이전 요청과 동일한 클라이언트 토큰을 사용하지만, 동일하지는 않습니다. 요청이 동일하지 않는 한, 다른 요청에 클라이언트 토큰을 재사용하지 마세요.

HTTP 상태 코드: 400

### InternalServerErrorException

RoboMaker AWS에서 서비스 문제가 발생했습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 500

### InvalidParameterException

요청에 지정된 파라미터는 유효하지 않거나, 지원되지 않거나, 사용할 수 없습니다. 반환된 메시지는 오류 값에 대한 설명을 제공합니다.

HTTP 상태 코드: 400

### LimitExceededException

요청된 리소스가 허용된 최대 수를 초과하거나 동시 스트림 요청 수가 허용된 최대 수를 초과합니다.

HTTP 상태 코드: 400

### ThrottlingException

RoboMaker AWS는 일시적으로 요청을 처리할 수 없습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 400

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS Go v2를 위한 SDK](#)
- [AWS Java V2용 SDK](#)
- [AWS V3용 SDK JavaScript](#)
- [AWS PHP V3용 SDK](#)
- [AWS Python용 SDK](#)
- [AWS 루비 V3용 SDK](#)

## CreateSimulationJob

시뮬레이션 작업을 생성합니다.

### Note

시뮬레이션 작업은 90일 후에 만료되고 삭제됩니다. 더 이상 액세스할 수 없습니다.

## Request Syntax

```
POST /createSimulationJob HTTP/1.1
Content-type: application/json

{
  "clientRequestToken": "string",
  "compute": {
    "computeType": "string",
    "gpuUnitLimit": number,
    "simulationUnitLimit": number
  },
  "dataSources": [
    {
      "destination": "string",
      "name": "string",
      "s3Bucket": "string",
      "s3Keys": [ "string" ],
      "type": "string"
    }
  ],
  "failureBehavior": "string",
  "iamRole": "string",
  "loggingConfig": {
    "recordAllRosTopics": boolean
  },
  "maxJobDurationInSeconds": number,
  "outputLocation": {
    "s3Bucket": "string",
    "s3Prefix": "string"
  },
  "robotApplications": [
    {
```

```
"application": "string",
"applicationVersion": "string",
"launchConfig": {
  "command": [ "string" ],
  "environmentVariables": {
    "string": "string"
  },
  "launchFile": "string",
  "packageName": "string",
  "portForwardingConfig": {
    "portMappings": [
      {
        "applicationPort": number,
        "enableOnPublicIp": boolean,
        "jobPort": number
      }
    ]
  },
  "streamUI": boolean
},
"tools": [
  {
    "command": "string",
    "exitBehavior": "string",
    "name": "string",
    "streamOutputToCloudWatch": boolean,
    "streamUI": boolean
  }
],
"uploadConfigurations": [
  {
    "name": "string",
    "path": "string",
    "uploadBehavior": "string"
  }
],
"useDefaultTools": boolean,
"useDefaultUploadConfigurations": boolean
}
],
"simulationApplications": [
  {
    "application": "string",
    "applicationVersion": "string",
```

```
"launchConfig": {
  "command": [ "string" ],
  "environmentVariables": {
    "string" : "string"
  },
  "launchFile": "string",
  "packageName": "string",
  "portForwardingConfig": {
    "portMappings": [
      {
        "applicationPort": number,
        "enableOnPublicIp": boolean,
        "jobPort": number
      }
    ]
  },
  "streamUI": boolean
},
"tools": [
  {
    "command": "string",
    "exitBehavior": "string",
    "name": "string",
    "streamOutputToCloudWatch": boolean,
    "streamUI": boolean
  }
],
"uploadConfigurations": [
  {
    "name": "string",
    "path": "string",
    "uploadBehavior": "string"
  }
],
"useDefaultTools": boolean,
"useDefaultUploadConfigurations": boolean,
"worldConfigs": [
  {
    "world": "string"
  }
]
}
],
"tags": {
```

```

    "string" : "string"
  },
  "vpcConfig": {
    "assignPublicIp": boolean,
    "securityGroups": [ "string" ],
    "subnets": [ "string" ]
  }
}

```

## URI 요청 파라미터

요청은 URI 파라미터를 사용하지 않습니다.

## 요청 본문

요청은 JSON 형식으로 다음 데이터를 받습니다.

### clientRequestToken

요청 멩등성을 보장하기 위해 제공하는 고유한 대소문자 구분 식별자입니다.

유형: 문자열

길이 제한: 최소 길이는 1. 최대 길이는 64.

패턴: [a-zA-Z0-9\_\-=]\*

Required: No

### compute

시뮬레이션 작업에 대한 컴퓨팅 정보입니다.

유형: [Compute](#) 객체

필수 항목 여부: 아니요

### dataSources

S3의 읽기 전용 파일을 시뮬레이션에 마운트할 데이터 소스를 지정합니다. 이러한 파일은 /opt/robomaker/datasources/data\_source\_name에서 사용할 수 있습니다.

#### Note

파일 수는 100개로 제한되며 모든 DataSourceConfig 객체의 총 크기는 25GB입니다.

유형: [DataSourceConfig](#) 객체 어레이

어레이 멤버: 최소 항목 수 1개. 최대 항목 수는 6개입니다.

필수 여부: 아니요

### [failureBehavior](#)

시뮬레이션 작업의 실패 동작입니다.

계속

4XX 오류 코드가 발생한 후 최대 제한 시간 동안 인스턴스를 계속 실행합니다.

Fail

시뮬레이션 작업을 중지하고 인스턴스를 종료합니다.

타입: 문자열

유효 값: Fail | Continue

필수 여부: 아니요

### [iamRole](#)

시뮬레이션 인스턴스가 사용자 대신 연결된 정책에 지정된 AWS API를 호출하도록 허용하는 IAM 역할 이름입니다. 이것이 시뮬레이션 작업에 보안 자격 증명이 전달되는 과정입니다.

유형: 문자열

길이 제약: 최소 길이는 1. 최대 길이는 255.

패턴: arn:aws:iam::\w+:role/.\*

필수 사항 여부: Yes

### [loggingConfig](#)

로깅 구성입니다.

유형: [LoggingConfig](#) 객체

필수 항목 여부: 아니요

### [maxJobDurationInSeconds](#)

초 단위의 최대 시뮬레이션 작업 기간(최대 14일 또는 1,209,600초)

maxJobDurationInSeconds에 도달하면 시뮬레이션 작업 상태가 Completed로 전환됩니다.

타입: Long

필수 여부: 예

### outputLocation

시뮬레이션 작업에서 생성된 출력 파일의 위치입니다.

유형: [OutputLocation](#) 객체

필수 항목 여부: 아니요

### robotApplications

시뮬레이션 작업에서 사용할 로봇 애플리케이션입니다.

유형: [RobotApplicationConfig](#) 객체 어레이

배열 멤버: 고정된 항목 수는 1개입니다.

필수 여부: 아니요

### simulationApplications

시뮬레이션 작업에서 사용할 시뮬레이션 애플리케이션입니다.

유형: [SimulationApplicationConfig](#) 객체 어레이

배열 멤버: 고정된 항목 수는 1개입니다.

필수 여부: 아니요

### tags

시뮬레이션 작업에 연결된 태그 키 및 태그 값을 포함하는 맵입니다.

유형: 문자열-문자열 맵

맵 항목: 최소 항목 수는 0개입니다. 최대 항목 수는 50개입니다.

키 길이 제약 조건: 최소 길이는 1. 최대 길이 128.

키 패턴: [a-zA-Z0-9 \_.\-\/+=:]\*

값 길이 제약 조건: 최소 길이는 0입니다. 최대 길이는 256.

값 패턴: [a-zA-Z0-9 \_.\-\/+=:]\*

필수 여부: 아니요

## vpcConfig

시뮬레이션 작업이 VPC의 리소스에 액세스하는 경우 보안 그룹 ID 및 서브넷 ID 목록을 식별하는 이 파라미터를 제공합니다. 동일한 VPC에 속해야 합니다. 하나 이상의 보안 그룹과 하나의 서브넷 ID를 제공해야 합니다.

유형: [VPCConfig](#) 객체

필수 항목 여부: 아니요

## 응답 구문

```
HTTP/1.1 200
Content-type: application/json

{
  "arn": "string",
  "clientRequestToken": "string",
  "compute": {
    "computeType": "string",
    "gpuUnitLimit": number,
    "simulationUnitLimit": number
  },
  "dataSources": [
    {
      "destination": "string",
      "name": "string",
      "s3Bucket": "string",
      "s3Keys": [
        {
          "etag": "string",
          "s3Key": "string"
        }
      ],
      "type": "string"
    }
  ],
  "failureBehavior": "string",
  "failureCode": "string",
  "iamRole": "string",
  "lastStartedAt": number,
  "lastUpdatedAt": number,
  "loggingConfig": {
```

```

    "recordAllRosTopics": boolean
  },
  "maxJobDurationInSeconds": number,
  "outputLocation": {
    "s3Bucket": "string",
    "s3Prefix": "string"
  },
  "robotApplications": [
    {
      "application": "string",
      "applicationVersion": "string",
      "launchConfig": {
        "command": [ "string ],
        "environmentVariables": {
          "string": "string"
        },
        "launchFile": "string",
        "packageName": "string",
        "portForwardingConfig": {
          "portMappings": [
            {
              "applicationPort": number,
              "enableOnPublicIp": boolean,
              "jobPort": number
            }
          ]
        },
        "streamUI": boolean
      },
      "tools": [
        {
          "command": "string",
          "exitBehavior": "string",
          "name": "string",
          "streamOutputToCloudWatch": boolean,
          "streamUI": boolean
        }
      ],
      "uploadConfigurations": [
        {
          "name": "string",
          "path": "string",
          "uploadBehavior": "string"
        }
      ]
    }
  ]
}

```

```

    ],
    "useDefaultTools": boolean,
    "useDefaultUploadConfigurations": boolean
  }
],
"simulationApplications": [
  {
    "application": "string",
    "applicationVersion": "string",
    "launchConfig": {
      "command": [ "string " ],
      "environmentVariables": {
        "string": "string"
      },
      "launchFile": "string",
      "packageName": "string",
      "portForwardingConfig": {
        "portMappings": [
          {
            "applicationPort": number,
            "enableOnPublicIp": boolean,
            "jobPort": number
          }
        ]
      },
      "streamUI": boolean
    },
    "tools": [
      {
        "command": "string",
        "exitBehavior": "string",
        "name": "string",
        "streamOutputToCloudWatch": boolean,
        "streamUI": boolean
      }
    ],
    "uploadConfigurations": [
      {
        "name": "string",
        "path": "string",
        "uploadBehavior": "string"
      }
    ],
    "useDefaultTools": boolean,

```

```

    "useDefaultUploadConfigurations": boolean,
    "worldConfigs": [
      {
        "world": "string"
      }
    ]
  }
],
"simulationTimeMillis": number,
"status": "string",
"tags": {
  "string" : "string"
},
"vpcConfig": {
  "assignPublicIp": boolean,
  "securityGroups": [ "string" ],
  "subnets": [ "string" ],
  "vpcId": "string"
}
}

```

## 응답 요소

작업이 성공하면 서비스가 HTTP 200 응답을 반환합니다.

다음 데이터는 서비스에 의해 JSON 형식으로 반환됩니다.

### [arn](#)

시뮬레이션 작업의 Amazon 리소스 이름(ARN)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

### [clientRequestToken](#)

요청 멱등성을 보장하기 위해 제공하는 고유한 대소문자 구분 식별자입니다.

유형: 문자열

길이 제한: 최소 길이는 1. 최대 길이는 64.

패턴: [a-zA-Z0-9\_\-=]\*

### compute

시뮬레이션 작업에 대한 컴퓨팅 정보입니다.

유형: [ComputeResponse](#) 객체

### dataSources

시뮬레이션 작업의 데이터 소스입니다.

유형: [DataSource](#) 객체 어레이

### failureBehavior

시뮬레이션 작업의 실패 동작입니다.

타입: 문자열

유효 값: Fail | Continue

### failureCode

시뮬레이션 작업 실패 시 발생하는 오류 코드:

InternalServerError

내부 서비스 오류.

RobotApplicationCrash

로봇 애플리케이션이 비정상적으로 종료되었습니다.

SimulationApplicationCrash

시뮬레이션 애플리케이션이 비정상적으로 종료되었습니다.

BadPermissionsRobotApplication

로봇 애플리케이션 번들을 다운로드할 수 없습니다.

BadPermissionsSimulationApplication

시뮬레이션 애플리케이션 번들을 다운로드할 수 없습니다.

BadPermissionsS3 출력

출력을 고객이 제공한 S3 버킷에 게시할 수 없습니다.

## BadPermissionsCloudwatchLogs

고객이 CloudWatch 제공한 Logs 리소스에 로그를 게시할 수 없습니다.

## SubnetIpLimitExceeded

서브넷 IP 제한을 초과했습니다.

## ENI LimitExceeded

ENI 제한을 초과했습니다.

## BadPermissionsUserCredentials

제공된 역할을 사용할 수 없습니다.

## InvalidBundleRobotApplication

로봇 번들을 추출할 수 없습니다(잘못된 형식, 번들링 오류 또는 기타 문제).

## InvalidBundleSimulationApplication

시뮬레이션 번들을 추출할 수 없습니다(잘못된 형식, 번들링 오류 또는 기타 문제).

## RobotApplicationVersionMismatchedEtag

버전 생성 중 Etag for 값이 일치하지 RobotApplication 않습니다.

## SimulationApplicationVersionMismatchedEtag

버전 생성 시 Etag for 값이 일치하지 SimulationApplication 않습니다.

타입: 문자열

유효 값: InternalServiceError | RobotApplicationCrash | SimulationApplicationCrash | RobotApplicationHealthCheckFailure | SimulationApplicationHealthCheckFailure | BadPermissionsRobotApplication | BadPermissionsSimulationApplication | BadPermissionsS3Object | BadPermissionsS3Output | BadPermissionsCloudwatchLogs | SubnetIpLimitExceeded | ENILimitExceeded | BadPermissionsUserCredentials | InvalidBundleRobotApplication | InvalidBundleSimulationApplication | InvalidS3Resource | ThrottlingError | LimitExceeded | MismatchedEtag | RobotApplicationVersionMismatchedEtag | SimulationApplicationVersionMismatchedEtag | ResourceNotFound | RequestThrottled | BatchTimedOut | BatchCanceled | InvalidInput |

WrongRegionS3Bucket | WrongRegionS3Output | WrongRegionRobotApplication  
| WrongRegionSimulationApplication | UploadContentMismatchError

### [iamRole](#)

시뮬레이션 작업이 사용자 대신 연결된 정책에 지정된 AWS API를 호출하도록 허용하는 IAM 역할입니다.

유형: 문자열

길이 제약: 최소 길이는 1. 최대 길이는 255.

패턴: arn:aws:iam::\w+:role/.\*

### [lastStartedAt](#)

시뮬레이션 작업이 마지막으로 시작된 epoch 이후 경과 시간(밀리초)

유형: 타임스탬프

### [lastUpdatedAt](#)

시뮬레이션 작업이 마지막으로 업데이트된 epoch 이후 경과 시간(밀리초)입니다.

유형: 타임스탬프

### [loggingConfig](#)

로깅 구성입니다.

유형: [LoggingConfig](#) 객체

### [maxJobDurationInSeconds](#)

초 단위 최대 시뮬레이션 작업 시간입니다.

타입: Long

### [outputLocation](#)

시뮬레이션 작업 출력 파일 위치입니다.

유형: [OutputLocation](#) 객체

### [robotApplications](#)

시뮬레이션 작업에서 사용할 로봇 애플리케이션입니다.

유형: [RobotApplicationConfig](#) 객체 어레이

배열 멤버: 고정된 항목 수는 1개입니다.

### [simulationApplications](#)

시뮬레이션 작업에서 사용할 시뮬레이션 애플리케이션입니다.

유형: [SimulationApplicationConfig](#) 객체 어레이

배열 멤버: 고정된 항목 수는 1개입니다.

### [simulationTimeMillis](#)

시뮬레이션 작업 실행 기간(밀리초)

타입: Long

### [status](#)

시뮬레이션 작업의 상태입니다.

타입: 문자열

유효 값: Pending | Preparing | Running | Restarting | Completed | Failed | RunningFailed | Terminating | Terminated | Canceled

### [tags](#)

시뮬레이션 작업에 추가된 모든 태그의 목록입니다.

유형: 문자열-문자열 맵

맵 항목: 최소 항목 수는 0개입니다. 최대 항목 수는 50개입니다.

키 길이 제약 조건: 최소 길이는 1. 최대 길이 128.

키 패턴: [a-zA-Z0-9 \_.\-\/+=:]\*

값 길이 제약 조건: 최소 길이는 0입니다. 최대 길이는 256.

가치 패턴: [a-zA-Z0-9 \_.\-\/+=:]\*

### [vpcConfig](#)

vpc 구성에 대한 정보입니다.

유형: [VPCConfigResponse](#) 객체

## Errors

모든 작업에서 일반적으로 발생하는 오류에 대한 자세한 내용은 [일반적인 오류](#) 섹션을 참조하세요.

### IdempotentParameterMismatchException

요청은 이전 요청과 동일한 클라이언트 토큰을 사용하지만, 동일하지는 않습니다. 요청이 동일하지 않는 한, 다른 요청에 클라이언트 토큰을 재사용하지 마세요.

HTTP 상태 코드: 400

### InternalServerErrorException

RoboMaker AWS에서 서비스 문제가 발생했습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 500

### InvalidParameterException

요청에 지정된 파라미터는 유효하지 않거나, 지원되지 않거나, 사용할 수 없습니다. 반환된 메시지는 오류 값에 대한 설명을 제공합니다.

HTTP 상태 코드: 400

### LimitExceededException

요청된 리소스가 허용된 최대 수를 초과하거나 동시 스트림 요청 수가 허용된 최대 수를 초과합니다.

HTTP 상태 코드: 400

### ResourceNotFoundException

지정한 리소스가 존재하지 않습니다.

HTTP 상태 코드: 400

### ServiceUnavailableException

서버의 일시적 장애로 인해 요청이 실패했습니다.

HTTP 상태 코드: 503

### ThrottlingException

RoboMaker AWS는 일시적으로 요청을 처리할 수 없습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 400

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS Go v2를 위한 SDK](#)
- [AWS Java V2용 SDK](#)
- [AWS V3용 SDK JavaScript](#)
- [AWS PHP V3용 SDK](#)
- [AWS Python용 SDK](#)
- [AWS 루비 V3용 SDK](#)

## CreateWorldExportJob

월드 내보내기 작업을 생성합니다.

### Request Syntax

```
POST /createWorldExportJob HTTP/1.1
Content-type: application/json

{
  "clientRequestToken": "string",
  "iamRole": "string",
  "outputLocation": {
    "s3Bucket": "string",
    "s3Prefix": "string"
  },
  "tags": {
    "string" : "string"
  },
  "worlds": [ "string" ]
}
```

### URI 요청 파라미터

요청은 URI 파라미터를 사용하지 않습니다.

### 요청 본문

요청은 JSON 형식으로 다음 데이터를 받습니다.

#### clientRequestToken

요청 멩등성을 보장하기 위해 제공하는 고유한 대소문자 구분 식별자입니다.

유형: 문자열

길이 제한: 최소 길이는 1. 최대 길이는 64.

패턴: [a-zA-Z0-9\_\-=]\*

Required: No

## iamRole

월드 내보내기 프로세스에서 Amazon S3 버킷에 액세스하여 내보내기를 전송하는 데 사용하는 IAM 역할입니다.

유형: 문자열

길이 제약: 최소 길이는 1. 최대 길이는 255.

패턴: `arn:aws:iam::\w+:role/.*`

필수 사항 여부: Yes

## outputLocation

출력 위치입니다.

유형: [OutputLocation](#) 객체

필수 여부: 예

## tags

월드 내보내기 작업에 연결된 태그 키 및 태그 값을 포함하는 맵입니다.

유형: 문자열-문자열 맵

맵 항목: 최소 항목 수는 0개입니다. 최대 항목 수는 50개입니다.

키 길이 제약 조건: 최소 길이는 1. 최대 길이 128.

키 패턴: `[a-zA-Z0-9 _.\-\/+=:]*`

값 길이 제약 조건: 최소 길이는 0입니다. 최대 길이는 256.

값 패턴: `[a-zA-Z0-9 _.\-\/+=:]*`

필수 여부: 아니요

## worlds

내보낼 월드에 해당하는 Amazon 리소스 이름(arn) 목록입니다.

유형: 문자열 어레이

배열 멤버: 최소수는 1개입니다. 최대 항목 수는 100개입니다.

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

필수 항목 여부: 예

## 응답 구문

```
HTTP/1.1 200
Content-type: application/json

{
  "arn": "string",
  "clientRequestToken": "string",
  "createdAt": number,
  "failureCode": "string",
  "iamRole": "string",
  "outputLocation": {
    "s3Bucket": "string",
    "s3Prefix": "string"
  },
  "status": "string",
  "tags": {
    "string" : "string"
  }
}
```

## 응답 요소

작업이 성공하면 서비스가 HTTP 200 응답을 반환합니다.

다음 데이터는 서비스에 의해 JSON 형식으로 반환됩니다.

### arn

월드 내보내기 작업 Amazon 리소스 이름(ARN)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

## clientRequestToken

요청 멱등성을 보장하기 위해 제공하는 고유한 대소문자 구분 식별자입니다.

유형: 문자열

길이 제한: 최소 길이는 1. 최대 길이는 64.

패턴: [a-zA-Z0-9\_\-=]\*

## createdAt

월드 내보내기 작업이 생성된 epoch 이후 경과 시간(밀리초)입니다.

유형: 타임스탬프

## failureCode

월드 내보내기 작업이 실패한 경우의 실패 코드:

InternalServerError

내부 서비스 오류.

LimitExceeded

요청된 리소스가 허용된 최대 수를 초과하거나 동시 스트림 요청 수가 허용된 최대 수를 초과합니다.

ResourceNotFound

특정 리소스를 찾을 수 없습니다.

RequestThrottled

요청에 병목 현상이 발생했습니다.

InvalidInput

요청의 입력 파라미터가 유효하지 않습니다.

AllWorldGenerationFailed

월드 생성 작업에 참여한 모든 월드가 실패했습니다. worldCount가 50보다 크거나 1보다 작을 때 발생할 수 있습니다.

문제 WorldForge 해결에 대한 자세한 내용은 [시뮬레이션 문제 해결](#)을 참조하십시오 WorldForge.

타입: 문자열

유효 값: `InternalServiceError` | `LimitExceeded` | `ResourceNotFound` | `RequestThrottled` | `InvalidInput` | `AccessDenied`

### iamRole

월드 내보내기 프로세스에서 Amazon S3 버킷에 액세스하여 내보내기를 전송하는 데 사용하는 IAM 역할입니다.

유형: 문자열

길이 제약: 최소 길이는 1. 최대 길이는 255.

패턴: `arn:aws:iam::\w+:role/.*`

### outputLocation

출력 위치입니다.

유형: OutputLocation 객체

### status

월드 내보내기 작업의 상태입니다.

보류중

월드 내보내기 작업 요청이 보류 중입니다.

실행 중

월드 내보내기 작업이 실행 중입니다.

완료됨

월드 내보내기 작업이 완료되었습니다.

Failed

월드 내보내기 작업이 실패했습니다. 자세한 정보는 `failureCode`을 참조하세요.

취소됨

월드 내보내기 작업이 취소되었습니다.

취소 중

월드 내보내기 작업을 취소 중입니다.

타입: 문자열

유효 값: Pending | Running | Completed | Failed | Canceling | Canceled

## [tags](#)

월드 내보내기 작업에 연결된 태그 키 및 태그 값을 포함하는 맵입니다.

유형: 문자열-문자열 맵

맵 항목: 최소 항목 수는 0개입니다. 최대 항목 수는 50개입니다.

키 길이 제약 조건: 최소 길이는 1. 최대 길이 128.

키 패턴: [a-zA-Z0-9 \_.\-\/+=:]\*

값 길이 제약 조건: 최소 길이는 0입니다. 최대 길이는 256.

가치 패턴: [a-zA-Z0-9 \_.\-\/+=:]\*

## Errors

모든 작업에서 일반적으로 발생하는 오류에 대한 자세한 내용은 [일반적인 오류](#) 섹션을 참조하세요.

### IdempotentParameterMismatchException

요청은 이전 요청과 동일한 클라이언트 토큰을 사용하지만, 동일하지는 않습니다. 요청이 동일하지 않는 한, 다른 요청에 클라이언트 토큰을 재사용하지 마세요.

HTTP 상태 코드: 400

### InternalServerErrorException

RoboMaker AWS에서 서비스 문제가 발생했습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 500

### InvalidParameterException

요청에 지정된 파라미터는 유효하지 않거나, 지원되지 않거나, 사용할 수 없습니다. 반환된 메시지는 오류 값에 대한 설명을 제공합니다.

HTTP 상태 코드: 400

### ResourceNotFoundException

지정한 리소스가 존재하지 않습니다.

HTTP 상태 코드: 400

ServiceUnavailableException

서버의 일시적 장애로 인해 요청이 실패했습니다.

HTTP 상태 코드: 503

ThrottlingException

RoboMaker AWS는 일시적으로 요청을 처리할 수 없습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 400

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS Go v2를 위한 SDK](#)
- [AWS Java V2용 SDK](#)
- [AWS V3용 SDK JavaScript](#)
- [AWS PHP V3용 SDK](#)
- [AWS Python용 SDK](#)
- [AWS 루비 V3용 SDK](#)

## CreateWorldGenerationJob

지정된 템플릿을 사용하여 월드를 생성합니다.

### Request Syntax

```
POST /createWorldGenerationJob HTTP/1.1
Content-type: application/json

{
  "clientRequestToken": "string",
  "tags": {
    "string" : "string"
  },
  "template": "string",
  "worldCount": {
    "floorplanCount": number,
    "interiorCountPerFloorplan": number
  },
  "worldTags": {
    "string" : "string"
  }
}
```

### URI 요청 파라미터

요청은 URI 파라미터를 사용하지 않습니다.

### 요청 본문

요청은 JSON 형식으로 다음 데이터를 받습니다.

#### clientRequestToken

요청 멱등성을 보장하기 위해 제공하는 고유한 대소문자 구분 식별자입니다.

유형: 문자열

길이 제한: 최소 길이는 1. 최대 길이는 64.

패턴: [a-zA-Z0-9\_\-=]\*

Required: No

## tags

월드 생성기 작업에 연결된 태그 키 및 태그 값을 포함하는 맵입니다.

유형: 문자열-문자열 맵

맵 항목: 최소 항목 수는 0개입니다. 최대 항목 수는 50개입니다.

키 길이 제약 조건: 최소 길이는 1. 최대 길이 128.

키 패턴: [a-zA-Z0-9 \_.\-\/+=:]\*

값 길이 제약 조건: 최소 길이는 0입니다. 최대 길이는 256.

값 패턴: [a-zA-Z0-9 \_.\-\/+=:]\*

필수 여부: 아니요

## template

생성하려는 월드를 설명하는 월드 템플릿의 Amazon 리소스 이름(ARN)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

필수 사항 여부: Yes

## worldCount

월드 수에 대한 정보입니다.

유형: [WorldCount](#) 객체

필수 여부: 예

## worldTags

생성된 월드 작업에 연결된 태그 키 및 태그 값을 포함하는 맵입니다.

유형: 문자열-문자열 맵

맵 항목: 최소 항목 수는 0개입니다. 최대 항목 수는 50개입니다.

키 길이 제약 조건: 최소 길이는 1. 최대 길이 128.

키 패턴: [a-zA-Z0-9 \_.\-\/+=:]\*

값 길이 제약 조건: 최소 길이는 0입니다. 최대 길이는 256.

값 패턴: [a-zA-Z0-9 \_.\-\/+=:]\*

필수 여부: 아니요

## 응답 구문

```
HTTP/1.1 200
Content-type: application/json

{
  "arn": "string",
  "clientRequestToken": "string",
  "createdAt": number,
  "failureCode": "string",
  "status": "string",
  "tags": {
    "string" : "string"
  },
  "template": "string",
  "worldCount": {
    "floorplanCount": number,
    "interiorCountPerFloorplan": number
  },
  "worldTags": {
    "string" : "string"
  }
}
```

## 응답 요소

작업이 성공하면 서비스가 HTTP 200 응답을 반환합니다.

다음 데이터는 서비스에 의해 JSON 형식으로 반환됩니다.

### arn

월드 생성기 작업의 Amazon 리소스 이름(ARN)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: `arn:.*`

### clientRequestToken

요청 멱등성을 보장하기 위해 제공하는 고유한 대소문자 구분 식별자입니다.

유형: 문자열

길이 제한: 최소 길이는 1. 최대 길이는 64.

패턴: `[a-zA-Z0-9_\-=]*`

### createdAt

월드 생성 작업이 생성된 epoch 이후 경과 시간(밀리초)입니다.

유형: 타임스탬프

### failureCode

월드 생성기 작업이 실패한 경우의 실패 코드:

`InternalServerError`

내부 서비스 오류.

`LimitExceeded`

요청된 리소스가 허용된 최대 수를 초과하거나 동시 스트림 요청 수가 허용된 최대 수를 초과합니다.

`ResourceNotFound`

특정 리소스를 찾을 수 없습니다.

`RequestThrottled`

요청에 병목 현상이 발생했습니다.

`InvalidInput`

요청의 입력 파라미터가 유효하지 않습니다.

타입: 문자열

유효 값: `InternalServerError | LimitExceeded | ResourceNotFound | RequestThrottled | InvalidInput | AllWorldGenerationFailed`

## status

월드 생성기 작업의 상태입니다.

### 보류중

월드 생성기 작업 요청이 보류 중입니다.

### 실행 중

월드 생성기 작업이 실행 중입니다.

### 완료됨

월드 생성기 작업이 완료되었습니다.

### Failed

월드 생성기 작업이 실패했습니다. 자세한 정보는 `failureCode`을 참조하세요.

### PartialFailed

일부 월드가 생성되지 않았습니다.

### 취소됨

월드 생성기 작업이 취소되었습니다.

### 취소 중

월드 생성기 작업이 취소되고 있습니다.

타입: 문자열

유효 값: Pending | Running | Completed | Failed | PartialFailed | Canceling  
| Canceled

## tags

월드 생성기 작업에 연결된 태그 키 및 태그 값을 포함하는 맵입니다.

유형: 문자열-문자열 맵

맵 항목: 최소 항목 수는 0개입니다. 최대 항목 수는 50개입니다.

키 길이 제약 조건: 최소 길이는 1. 최대 길이 128.

키 패턴: `[a-zA-Z0-9 _.\-\/+=:]*`

값 길이 제약 조건: 최소 길이는 0입니다. 최대 길이는 256.

가치 패턴: [a-zA-Z0-9 \_.\-\/+=:]\*

### template

월드 템플릿의 Amazon 리소스 이름(ARN)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

### worldCount

월드 수에 대한 정보입니다.

유형: [WorldCount](#) 객체

### worldTags

생성된 월드 작업에 연결된 태그 키 및 태그 값을 포함하는 맵입니다.

유형: 문자열-문자열 맵

맵 항목: 최소 항목 수는 0개입니다. 최대 항목 수는 50개입니다.

키 길이 제약 조건: 최소 길이는 1. 최대 길이 128.

키 패턴: [a-zA-Z0-9 \_.\-\/+=:]\*

값 길이 제약 조건: 최소 길이는 0입니다. 최대 길이는 256.

가치 패턴: [a-zA-Z0-9 \_.\-\/+=:]\*

## Errors

모든 작업에서 일반적으로 발생하는 오류에 대한 자세한 내용은 [일반적인 오류](#) 섹션을 참조하세요.

### IdempotentParameterMismatchException

요청은 이전 요청과 동일한 클라이언트 토큰을 사용하지만, 동일하지는 않습니다. 요청이 동일하지 않는 한, 다른 요청에 클라이언트 토큰을 재사용하지 마세요.

HTTP 상태 코드: 400

#### InternalServerError

RoboMaker AWS에서 서비스 문제가 발생했습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 500

#### InvalidParameterException

요청에 지정된 파라미터는 유효하지 않거나, 지원되지 않거나, 사용할 수 없습니다. 반환된 메시지는 오류 값에 대한 설명을 제공합니다.

HTTP 상태 코드: 400

#### LimitExceededException

요청된 리소스가 허용된 최대 수를 초과하거나 동시 스트림 요청 수가 허용된 최대 수를 초과합니다.

HTTP 상태 코드: 400

#### ResourceNotFoundException

지정한 리소스가 존재하지 않습니다.

HTTP 상태 코드: 400

#### ServiceUnavailableException

서버의 일시적 장애로 인해 요청이 실패했습니다.

HTTP 상태 코드: 503

#### ThrottlingException

RoboMaker AWS는 일시적으로 요청을 처리할 수 없습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 400

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)

- [AWS SDK for C++](#)
- [AWS Go v2를 위한 SDK](#)
- [AWS Java V2용 SDK](#)
- [AWS V3용 SDK JavaScript](#)
- [AWS PHP V3용 SDK](#)
- [AWS Python용 SDK](#)
- [AWS 루비 V3용 SDK](#)

# CreateWorldTemplate

월드 템플릿을 생성합니다.

## Request Syntax

```
POST /createWorldTemplate HTTP/1.1
Content-type: application/json
```

```
{
  "clientRequestToken": "string",
  "name": "string",
  "tags": {
    "string" : "string"
  },
  "templateBody": "string",
  "templateLocation": {
    "s3Bucket": "string",
    "s3Key": "string"
  }
}
```

## URI 요청 파라미터

요청은 URI 파라미터를 사용하지 않습니다.

## 요청 본문

요청은 JSON 형식으로 다음 데이터를 받습니다.

### clientRequestToken

요청 멱등성을 보장하기 위해 제공하는 고유한 대소문자 구분 식별자입니다.

유형: 문자열

길이 제한: 최소 길이는 1. 최대 길이는 64.

패턴: [a-zA-Z0-9\_\-=]\*

Required: No

## name

월드 템플릿의 이름입니다.

타입: 문자열

길이 제약 조건: 최소 길이는 0입니다. 최대 길이는 255.

패턴: .\*

Required: No

## tags

월드 템플릿에 연결된 태그 키 및 태그 값을 포함하는 맵입니다.

유형: 문자열-문자열 맵

맵 항목: 최소 항목 수는 0개입니다. 최대 항목 수는 50개입니다.

키 길이 제약 조건: 최소 길이는 1. 최대 길이 128.

키 패턴: [a-zA-Z0-9 \_.\-\/+=:]\*

값 길이 제약 조건: 최소 길이는 0입니다. 최대 길이는 256.

값 패턴: [a-zA-Z0-9 \_.\-\/+=:]\*

필수 여부: 아니요

## templateBody

월드 템플릿 본문입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 262144자입니다.

패턴: [\\S\\s]+

Required: No

## templateLocation

월드 템플릿의 위치입니다.

유형: [TemplateLocation](#) 객체

필수 항목 여부: 아니요

## 응답 구문

```
HTTP/1.1 200
Content-type: application/json

{
  "arn": "string",
  "clientRequestToken": "string",
  "createdAt": number,
  "name": "string",
  "tags": {
    "string" : "string"
  }
}
```

## 응답 요소

작업이 성공하면 서비스가 HTTP 200 응답을 반환합니다.

다음 데이터는 서비스에 의해 JSON 형식으로 반환됩니다.

### arn

월드 템플릿의 Amazon 리소스 이름(ARN)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

### clientRequestToken

요청 멱등성을 보장하기 위해 제공하는 고유한 대소문자 구분 식별자입니다.

유형: 문자열

길이 제한: 최소 길이는 1. 최대 길이는 64.

패턴: [a-zA-Z0-9\_\-=]\*

## createdAt

월드 템플릿이 생성된 epoch 이후 경과 시간(밀리초)입니다.

유형: 타임스탬프

## name

월드 템플릿의 이름입니다.

타입: 문자열

길이 제약 조건: 최소 길이는 0입니다. 최대 길이는 255.

패턴: .\*

## tags

월드 템플릿에 연결된 태그 키 및 태그 값을 포함하는 맵입니다.

유형: 문자열-문자열 맵

맵 항목: 최소 항목 수는 0개입니다. 최대 항목 수는 50개입니다.

키 길이 제약 조건: 최소 길이는 1. 최대 길이 128.

키 패턴: [a-zA-Z0-9 \_.\-\/+=:]\*

값 길이 제약 조건: 최소 길이는 0입니다. 최대 길이는 256.

가치 패턴: [a-zA-Z0-9 \_.\-\/+=:]\*

## Errors

모든 작업에서 발생하는 일반적인 오류에 대한 자세한 내용은 [일반적인 오류](#) 섹션을 참조하세요.

### InternalServerErrorException

RoboMaker AWS에서 서비스 문제가 발생했습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 500

### InvalidParameterException

요청에 지정된 파라미터는 유효하지 않거나, 지원되지 않거나, 사용할 수 없습니다. 반환된 메시지는 오류 값에 대한 설명을 제공합니다.

HTTP 상태 코드: 400

LimitExceededException

요청된 리소스가 허용된 최대 수를 초과하거나 동시 스트림 요청 수가 허용된 최대 수를 초과합니다.

HTTP 상태 코드: 400

ResourceAlreadyExistsException

지정된 리소스가 이미 있습니다.

HTTP 상태 코드: 400

ResourceNotFoundException

지정한 리소스가 존재하지 않습니다.

HTTP 상태 코드: 400

ThrottlingException

RoboMaker AWS는 일시적으로 요청을 처리할 수 없습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 400

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS Go v2를 위한 SDK](#)
- [AWS Java V2용 SDK](#)
- [AWS V3용 SDK JavaScript](#)
- [AWS PHP V3용 SDK](#)
- [AWS Python용 SDK](#)
- [AWS 루비 V3용 SDK](#)

## DeleteFleet

이 작업은 사용 중단되었습니다.

### Important

해당 API는 더 이상 지원되지 않습니다. 자세한 내용은 [지원 정책](#) 페이지에서 2022년 5월 2일 업데이트를 참조하세요.

집합을 삭제합니다.

## Request Syntax

```
POST /deleteFleet HTTP/1.1
Content-type: application/json

{
  "fleet": "string"
}
```

## URI 요청 파라미터

요청은 URI 파라미터를 사용하지 않습니다.

## 요청 본문

요청은 JSON 형식으로 다음 데이터를 받습니다.

### fleet

플릿의 Amazon 리소스 이름(ARN)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

필수 항목 여부: 예

## 응답 구문

```
HTTP/1.1 200
```

### Response Elements

작업이 성공하면 서비스가 비어 있는 HTTP 본문과 함께 HTTP 200 응답을 반환합니다.

### Errors

모든 작업에서 발생하는 일반적인 오류에 대한 자세한 내용은 [일반적인 오류](#) 섹션을 참조하세요.

#### InternalServerError

RoboMaker AWS에서 서비스 문제가 발생했습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 500

#### InvalidParameterException

요청에 지정된 파라미터는 유효하지 않거나, 지원되지 않거나, 사용할 수 없습니다. 반환된 메시지는 오류 값에 대한 설명을 제공합니다.

HTTP 상태 코드: 400

#### ThrottlingException

RoboMaker AWS는 일시적으로 요청을 처리할 수 없습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 400

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS Go v2를 위한 SDK](#)
- [AWS Java V2용 SDK](#)
- [AWS V3용 SDK JavaScript](#)

- [AWS PHP V3용 SDK](#)
- [AWS Python용 SDK](#)
- [AWS 루비 V3용 SDK](#)

## DeleteRobot

이 작업은 사용 중단되었습니다.

### Important

해당 API는 더 이상 지원되지 않습니다. 자세한 내용은 [지원 정책](#) 페이지에서 2022년 5월 2일 업데이트를 참조하세요.

로봇을 삭제합니다.

## Request Syntax

```
POST /deleteRobot HTTP/1.1
Content-type: application/json

{
  "robot": "string"
}
```

## URI 요청 파라미터

요청은 URI 파라미터를 사용하지 않습니다.

## 요청 본문

요청은 JSON 형식으로 다음 데이터를 받습니다.

### robot

로봇의 Amazon 리소스 이름(ARN)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

필수 항목 여부: 예

## 응답 구문

```
HTTP/1.1 200
```

### Response Elements

작업이 성공하면 서비스가 비어 있는 HTTP 본문과 함께 HTTP 200 응답을 반환합니다.

### Errors

모든 작업에서 발생하는 일반적인 오류에 대한 자세한 내용은 [일반적인 오류](#) 섹션을 참조하세요.

#### InternalServerError

RoboMaker AWS에서 서비스 문제가 발생했습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 500

#### InvalidParameterException

요청에 지정된 파라미터는 유효하지 않거나, 지원되지 않거나, 사용할 수 없습니다. 반환된 메시지는 오류 값에 대한 설명을 제공합니다.

HTTP 상태 코드: 400

#### ThrottlingException

RoboMaker AWS는 일시적으로 요청을 처리할 수 없습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 400

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS Go v2를 위한 SDK](#)
- [AWS Java V2용 SDK](#)
- [AWS V3용 SDK JavaScript](#)

- [AWS PHP V3용 SDK](#)
- [AWS Python용 SDK](#)
- [AWS 루비 V3용 SDK](#)

## DeleteRobotApplication

로봇 애플리케이션을 삭제합니다.

### Request Syntax

```
POST /deleteRobotApplication HTTP/1.1
Content-type: application/json

{
  "application": "string",
  "applicationVersion": "string"
}
```

### URI 요청 파라미터

요청은 URI 파라미터를 사용하지 않습니다.

### 요청 본문

요청은 JSON 형식으로 다음 데이터를 받습니다.

#### application

로봇 애플리케이션의 Amazon 리소스 이름(ARN)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

필수 사항 여부: Yes

#### applicationVersion

설명할 로봇 애플리케이션의 버전입니다.

유형: 문자열

길이 제약: 최소 길이는 1. 최대 길이는 255.

패턴: (\\$LATEST)|[0-9]\*

필수 여부: 아니요

## 응답 구문

```
HTTP/1.1 200
```

## Response Elements

작업이 성공하면 서비스가 비어 있는 HTTP 본문과 함께 HTTP 200 응답을 반환합니다.

## Errors

모든 작업에서 발생하는 일반적인 오류에 대한 자세한 내용은 [일반적인 오류](#) 섹션을 참조하세요.

### InternalServerError

RoboMaker AWS에서 서비스 문제가 발생했습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 500

### InvalidParameterException

요청에 지정된 파라미터는 유효하지 않거나, 지원되지 않거나, 사용할 수 없습니다. 반환된 메시지는 오류 값에 대한 설명을 제공합니다.

HTTP 상태 코드: 400

### ThrottlingException

RoboMaker AWS는 일시적으로 요청을 처리할 수 없습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 400

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS Go v2를 위한 SDK](#)

- [AWS Java V2용 SDK](#)
- [AWS V3용 SDK JavaScript](#)
- [AWS PHP V3용 SDK](#)
- [AWS Python용 SDK](#)
- [AWS 루비 V3용 SDK](#)

## DeleteSimulationApplication

시뮬레이션 애플리케이션을 삭제합니다.

### Request Syntax

```
POST /deleteSimulationApplication HTTP/1.1
Content-type: application/json

{
  "application": "string",
  "applicationVersion": "string"
}
```

### URI 요청 파라미터

요청은 URI 파라미터를 사용하지 않습니다.

### 요청 본문

요청은 JSON 형식으로 다음 데이터를 받습니다.

#### application

삭제할 시뮬레이션 애플리케이션에 대한 애플리케이션 정보입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

필수 사항 여부: Yes

#### applicationVersion

삭제할 시뮬레이션 애플리케이션 버전입니다.

유형: 문자열

길이 제약: 최소 길이는 1. 최대 길이는 255.

패턴: (\\$LATEST)|[0-9]\*

필수 여부: 아니요

## 응답 구문

```
HTTP/1.1 200
```

## Response Elements

작업이 성공하면 서비스가 비어 있는 HTTP 본문과 함께 HTTP 200 응답을 반환합니다.

## Errors

모든 작업에서 발생하는 일반적인 오류에 대한 자세한 내용은 [일반적인 오류](#) 섹션을 참조하세요.

### InternalServerError

RoboMaker AWS에서 서비스 문제가 발생했습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 500

### InvalidParameterException

요청에 지정된 파라미터는 유효하지 않거나, 지원되지 않거나, 사용할 수 없습니다. 반환된 메시지는 오류 값에 대한 설명을 제공합니다.

HTTP 상태 코드: 400

### ThrottlingException

RoboMaker AWS는 일시적으로 요청을 처리할 수 없습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 400

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS Go v2를 위한 SDK](#)

- [AWS Java V2용 SDK](#)
- [AWS V3용 SDK JavaScript](#)
- [AWS PHP V3용 SDK](#)
- [AWS Python용 SDK](#)
- [AWS 루비 V3용 SDK](#)

## DeleteWorldTemplate

월드 템플릿을 삭제합니다.

### Request Syntax

```
POST /deleteWorldTemplate HTTP/1.1
Content-type: application/json

{
  "template": "string"
}
```

### URI 요청 파라미터

요청은 URI 파라미터를 사용하지 않습니다.

### 요청 본문

요청은 JSON 형식으로 다음 데이터를 받습니다.

#### template

삭제하려는 월드 템플릿의 Amazon 리소스 이름(arn)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

필수 항목 여부: 예

### 응답 구문

```
HTTP/1.1 200
```

### Response Elements

작업이 성공하면 서비스가 비어 있는 HTTP 본문과 함께 HTTP 200 응답을 반환합니다.

## Errors

모든 작업에서 발생하는 일반적인 오류에 대한 자세한 내용은 [일반적인 오류](#) 섹션을 참조하세요.

### InternalServerErrorException

RoboMaker AWS에서 서비스 문제가 발생했습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 500

### InvalidParameterException

요청에 지정된 파라미터는 유효하지 않거나, 지원되지 않거나, 사용할 수 없습니다. 반환된 메시지는 오류 값에 대한 설명을 제공합니다.

HTTP 상태 코드: 400

### ResourceNotFoundException

지정한 리소스가 존재하지 않습니다.

HTTP 상태 코드: 400

### ThrottlingException

RoboMaker AWS는 일시적으로 요청을 처리할 수 없습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 400

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS Go v2를 위한 SDK](#)
- [AWS Java V2용 SDK](#)
- [AWS V3용 SDK JavaScript](#)
- [AWS PHP V3용 SDK](#)
- [AWS Python용 SDK](#)

- [AWS 루비 V3용 SDK](#)

## DeregisterRobot

이 작업은 사용 중단되었습니다.

### Important

해당 API는 더 이상 지원되지 않습니다. 자세한 내용은 [지원 정책](#) 페이지에서 2022년 5월 2일 업데이트를 참조하세요.

로봇 등록을 취소합니다.

## Request Syntax

```
POST /deregisterRobot HTTP/1.1
Content-type: application/json

{
  "fleet": "string",
  "robot": "string"
}
```

## URI 요청 파라미터

요청은 URI 파라미터를 사용하지 않습니다.

## 요청 본문

요청은 JSON 형식으로 다음 데이터를 받습니다.

### fleet

플릿의 Amazon 리소스 이름(ARN)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

필수 사항 여부: Yes

## robot

로봇의 Amazon 리소스 이름(ARN)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

필수 항목 여부: 예

## 응답 구문

```
HTTP/1.1 200
Content-type: application/json

{
  "fleet": "string",
  "robot": "string"
}
```

## 응답 요소

작업이 성공하면 서비스가 HTTP 200 응답을 반환합니다.

다음 데이터는 서비스에 의해 JSON 형식으로 반환됩니다.

## fleet

플릿의 Amazon 리소스 이름(ARN)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

## robot

로봇의 Amazon 리소스 이름(ARN)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

## Errors

모든 작업에서 발생하는 일반적인 오류에 대한 자세한 내용은 [일반적인 오류](#) 섹션을 참조하세요.

### InternalServerErrorException

RoboMaker AWS에서 서비스 문제가 발생했습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 500

### InvalidParameterException

요청에 지정된 파라미터는 유효하지 않거나, 지원되지 않거나, 사용할 수 없습니다. 반환된 메시지는 오류 값에 대한 설명을 제공합니다.

HTTP 상태 코드: 400

### ResourceNotFoundException

지정한 리소스가 존재하지 않습니다.

HTTP 상태 코드: 400

### ThrottlingException

RoboMaker AWS는 일시적으로 요청을 처리할 수 없습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 400

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS Go v2를 위한 SDK](#)
- [AWS Java V2용 SDK](#)

- [AWS V3용 SDK JavaScript](#)
- [AWS PHP V3용 SDK](#)
- [AWS Python용 SDK](#)
- [AWS 루비 V3용 SDK](#)

## DescribeDeploymentJob

이 작업은 사용 중단되었습니다.

### Important

해당 API는 더 이상 지원되지 않습니다. 자세한 내용은 [지원 정책](#) 페이지에서 2022년 5월 2일 업데이트를 참조하세요.

배포 작업을 설명합니다.

## Request Syntax

```
POST /describeDeploymentJob HTTP/1.1
Content-type: application/json

{
  "job": "string"
}
```

## URI 요청 파라미터

요청은 URI 파라미터를 사용하지 않습니다.

## 요청 본문

요청은 JSON 형식으로 다음 데이터를 받습니다.

### job

배포 작업의 Amazon 리소스 이름(ARN)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

필수 항목 여부: 예

## 응답 구문

```

HTTP/1.1 200
Content-type: application/json

{
  "arn": "string",
  "createdAt": number,
  "deploymentApplicationConfigs": [
    {
      "application": "string",
      "applicationVersion": "string",
      "launchConfig": {
        "environmentVariables": {
          "string": "string"
        },
        "launchFile": "string",
        "packageName": "string",
        "postLaunchFile": "string",
        "preLaunchFile": "string"
      }
    }
  ],
  "deploymentConfig": {
    "concurrentDeploymentPercentage": number,
    "downloadConditionFile": {
      "bucket": "string",
      "etag": "string",
      "key": "string"
    },
    "failureThresholdPercentage": number,
    "robotDeploymentTimeoutInSeconds": number
  },
  "failureCode": "string",
  "failureReason": "string",
  "fleet": "string",
  "robotDeploymentSummary": [
    {
      "arn": "string",
      "deploymentFinishTime": number,
      "deploymentStartTime": number,
      "failureCode": "string",
      "failureReason": "string",

```

```

    "progressDetail": {
      "currentProgress": "string",
      "estimatedTimeRemainingSeconds": number,
      "percentDone": number,
      "targetResource": "string"
    },
    "status": "string"
  }
],
"status": "string",
"tags": {
  "string" : "string"
}
}

```

## 응답 요소

작업이 성공하면 서비스가 HTTP 200 응답을 반환합니다.

다음 데이터는 서비스에 의해 JSON 형식으로 반환됩니다.

### arn

배포 작업의 Amazon 리소스 이름(ARN)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

### createdAt

배포 작업이 생성된 epoch 이후 경과 시간(밀리초)입니다.

유형: 타임스탬프

### deploymentApplicationConfigs

배포 애플리케이션 구성입니다.

유형: [DeploymentApplicationConfig](#) 객체 어레이

배열 멤버: 고정된 항목 수는 1개입니다.

## deploymentConfig

배포 구성입니다.

유형: [DeploymentConfig](#) 객체

## failureCode

배포 작업 실패 코드입니다.

타입: 문자열

유효 값: ResourceNotFound | EnvironmentSetupError | EtagMismatch  
| FailureThresholdBreached | RobotDeploymentAborted |  
RobotDeploymentNoResponse | RobotAgentConnectionTimeout  
| GreengrassDeploymentFailed | InvalidGreengrassGroup |  
MissingRobotArchitecture | MissingRobotApplicationArchitecture |  
MissingRobotDeploymentResource | GreengrassGroupVersionDoesNotExist  
| LambdaDeleted | ExtractingBundleFailure | PreLaunchFileFailure |  
PostLaunchFileFailure | BadPermissionError | DownloadConditionFailed |  
BadLambdaAssociated | InternalServerError | RobotApplicationDoesNotExist  
| DeploymentFleetDoesNotExist | FleetDeploymentTimeout

## failureReason

배포 작업이 실패한 이유에 대한 간략한 설명입니다.

타입: 문자열

길이 제약 조건: 최소 길이는 0입니다. 최대 길이는 1024입니다.

패턴: .\*

## fleet

플릿의 Amazon 리소스 이름(ARN)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

## robotDeploymentSummary

로봇 배포 요약 목록입니다.

유형: [RobotDeployment](#) 객체 어레이

### [status](#)

배포 작업의 상태입니다.

타입: 문자열

유효 값: Pending | Preparing | InProgress | Failed | Succeeded | Canceled

### [tags](#)

지정된 배포 작업에 추가된 모든 태그의 상태입니다.

유형: 문자열-문자열 맵

맵 항목: 최소 항목 수는 0개입니다. 최대 항목 수는 50개입니다.

키 길이 제약 조건: 최소 길이는 1. 최대 길이 128.

키 패턴: [a-zA-Z0-9 \_.\-\/+=:]\*

값 길이 제약 조건: 최소 길이는 0입니다. 최대 길이는 256.

가치 패턴: [a-zA-Z0-9 \_.\-\/+=:]\*

## Errors

모든 작업에서 발생하는 일반적인 오류에 대한 자세한 내용은 [일반적인 오류](#) 섹션을 참조하세요.

### InternalServerError

RoboMaker AWS에서 서비스 문제가 발생했습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 500

### InvalidParameterException

요청에 지정된 파라미터는 유효하지 않거나, 지원되지 않거나, 사용할 수 없습니다. 반환된 메시지는 오류 값에 대한 설명을 제공합니다.

HTTP 상태 코드: 400

### ResourceNotFoundException

지정한 리소스가 존재하지 않습니다.

HTTP 상태 코드: 400

ThrottlingException

RoboMaker AWS는 일시적으로 요청을 처리할 수 없습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 400

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS Go v2를 위한 SDK](#)
- [AWS Java V2용 SDK](#)
- [AWS V3용 SDK JavaScript](#)
- [AWS PHP V3용 SDK](#)
- [AWS Python용 SDK](#)
- [AWS 루비 V3용 SDK](#)

## DescribeFleet

이 작업은 사용 중단되었습니다.

### Important

해당 API는 더 이상 지원되지 않습니다. 자세한 내용은 [지원 정책](#) 페이지에서 2022년 5월 2일 업데이트를 참조하세요.

플릿을 설명합니다.

## Request Syntax

```
POST /describeFleet HTTP/1.1
Content-type: application/json

{
  "fleet": "string"
}
```

## URI 요청 파라미터

요청은 URI 파라미터를 사용하지 않습니다.

## 요청 본문

요청은 JSON 형식으로 다음 데이터를 받습니다.

### fleet

플릿의 Amazon 리소스 이름(ARN)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

필수 항목 여부: 예

## 응답 구문

```

HTTP/1.1 200
Content-type: application/json

{
  "arn": "string",
  "createdAt": number,
  "lastDeploymentJob": "string",
  "lastDeploymentStatus": "string",
  "lastDeploymentTime": number,
  "name": "string",
  "robots": [
    {
      "architecture": "string",
      "arn": "string",
      "createdAt": number,
      "fleetArn": "string",
      "greenGrassGroupId": "string",
      "lastDeploymentJob": "string",
      "lastDeploymentTime": number,
      "name": "string",
      "status": "string"
    }
  ],
  "tags": {
    "string" : "string"
  }
}

```

## 응답 요소

작업이 성공하면 서비스가 HTTP 200 응답을 반환합니다.

다음 데이터는 서비스에 의해 JSON 형식으로 반환됩니다.

### arn

플릿의 Amazon 리소스 이름(ARN)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

### createdAt

플릿이 생성된 epoch 이후 경과 시간(밀리초)입니다.

유형: 타임스탬프

### lastDeploymentJob

마지막 배포 작업의 Amazon 리소스 이름(ARN)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

### lastDeploymentStatus

마지막 배포 상태입니다.

타입: 문자열

유효 값: Pending | Preparing | InProgress | Failed | Succeeded | Canceled

### lastDeploymentTime

마지막 배포 시간입니다.

유형: 타임스탬프

### name

플릿의 이름입니다.

유형: 문자열

길이 제약: 최소 길이는 1. 최대 길이는 255.

패턴: [a-zA-Z0-9\_\-]\*

### robots

로봇 목록입니다.

유형: [Robot](#) 객체 어레이

어레이 멤버: 최소 항목 수 0개. 최대 항목 수는 1000개입니다.

## tags

지정된 플릿에 추가된 모든 태그의 목록입니다.

유형: 문자열-문자열 맵

맵 항목: 최소 항목 수는 0개입니다. 최대 항목 수는 50개입니다.

키 길이 제약 조건: 최소 길이는 1. 최대 길이 128.

키 패턴: [a-zA-Z0-9 \_.\-\/+=:]\*

값 길이 제약 조건: 최소 길이는 0입니다. 최대 길이는 256.

가치 패턴: [a-zA-Z0-9 \_.\-\/+=:]\*

## Errors

모든 작업에서 발생하는 일반적인 오류에 대한 자세한 내용은 [일반적인 오류](#) 섹션을 참조하세요.

### InternalServerError

RoboMaker AWS에서 서비스 문제가 발생했습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 500

### InvalidParameterException

요청에 지정된 파라미터는 유효하지 않거나, 지원되지 않거나, 사용할 수 없습니다. 반환된 메시지는 오류 값에 대한 설명을 제공합니다.

HTTP 상태 코드: 400

### ResourceNotFoundException

지정한 리소스가 존재하지 않습니다.

HTTP 상태 코드: 400

### ThrottlingException

RoboMaker AWS는 일시적으로 요청을 처리할 수 없습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 400

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS Go v2를 위한 SDK](#)
- [AWS Java V2용 SDK](#)
- [AWS V3용 SDK JavaScript](#)
- [AWS PHP V3용 SDK](#)
- [AWS Python용 SDK](#)
- [AWS 루비 V3용 SDK](#)

## DescribeRobot

이 작업은 사용 중단되었습니다.

### Important

해당 API는 더 이상 지원되지 않습니다. 자세한 내용은 [지원 정책](#) 페이지에서 2022년 5월 2일 업데이트를 참조하세요.

로봇을 설명합니다.

## Request Syntax

```
POST /describeRobot HTTP/1.1
Content-type: application/json

{
  "robot": "string"
}
```

## URI 요청 파라미터

요청은 URI 파라미터를 사용하지 않습니다.

## 요청 본문

요청은 JSON 형식으로 다음 데이터를 받습니다.

### robot

설명되는 로봇의 Amazon 리소스 이름(ARN)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

필수 항목 여부: 예

## 응답 구문

```
HTTP/1.1 200
Content-type: application/json

{
  "architecture": "string",
  "arn": "string",
  "createdAt": number,
  "fleetArn": "string",
  "greengrassGroupId": "string",
  "lastDeploymentJob": "string",
  "lastDeploymentTime": number,
  "name": "string",
  "status": "string",
  "tags": {
    "string" : "string"
  }
}
```

## 응답 요소

작업이 성공하면 서비스가 HTTP 200 응답을 반환합니다.

다음 데이터는 서비스에 의해 JSON 형식으로 반환됩니다.

### architecture

로봇 애플리케이션의 대상 아키텍처입니다.

타입: 문자열

유효 값: X86\_64 | ARM64 | ARMHF

### arn

로봇의 Amazon 리소스 이름(ARN)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

### createdAt

로봇이 생성된 epoch 이후 경과 시간(밀리초)입니다.

유형: 타임스탬프

### fleetArn

플릿의 Amazon 리소스 이름(ARN)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

### greengrassGroupId

Greengrass 그룹 ID입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: .\*

### lastDeploymentJob

마지막 배포 작업의 Amazon 리소스 이름(ARN)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

### lastDeploymentTime

마지막 배포 작업의 시간입니다.

유형: 타임스탬프

### name

로봇의 이름입니다.

유형: 문자열

길이 제약: 최소 길이는 1. 최대 길이는 255.

패턴: [a-zA-Z0-9\_\-]\*

### status

플릿의 상태입니다.

타입: 문자열

유효 값: Available | Registered | PendingNewDeployment | Deploying | Failed  
| InSync | NoResponse

### tags

지정된 로봇에 추가된 모든 태그의 목록입니다.

유형: 문자열-문자열 맵

맵 항목: 최소 항목 수는 0개입니다. 최대 항목 수는 50개입니다.

키 길이 제약 조건: 최소 길이는 1. 최대 길이 128.

키 패턴: [a-zA-Z0-9 \_.\-\/+=:]\*

값 길이 제약 조건: 최소 길이는 0입니다. 최대 길이는 256.

가치 패턴: [a-zA-Z0-9 \_.\-\/+=:]\*

## Errors

모든 작업에서 발생하는 일반적인 오류에 대한 자세한 내용은 [일반적인 오류](#) 섹션을 참조하세요.

### InternalServerError

RoboMaker AWS에서 서비스 문제가 발생했습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 500

### InvalidParameterException

요청에 지정된 파라미터는 유효하지 않거나, 지원되지 않거나, 사용할 수 없습니다. 반환된 메시지는 오류 값에 대한 설명을 제공합니다.

HTTP 상태 코드: 400

## ResourceNotFoundException

지정한 리소스가 존재하지 않습니다.

HTTP 상태 코드: 400

## ThrottlingException

RoboMaker AWS는 일시적으로 요청을 처리할 수 없습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 400

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS Go v2를 위한 SDK](#)
- [AWS Java V2용 SDK](#)
- [AWS V3용 SDK JavaScript](#)
- [AWS PHP V3용 SDK](#)
- [AWS Python용 SDK](#)
- [AWS 루비 V3용 SDK](#)

## DescribeRobotApplication

로봇 애플리케이션을 설명합니다.

### Request Syntax

```
POST /describeRobotApplication HTTP/1.1
Content-type: application/json

{
  "application": "string",
  "applicationVersion": "string"
}
```

### URI 요청 파라미터

요청은 URI 파라미터를 사용하지 않습니다.

### 요청 본문

요청은 JSON 형식으로 다음 데이터를 받습니다.

#### [application](#)

로봇 애플리케이션의 Amazon 리소스 이름(ARN)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

필수 사항 여부: Yes

#### [applicationVersion](#)

설명할 로봇 애플리케이션의 버전입니다.

유형: 문자열

길이 제약: 최소 길이는 1. 최대 길이는 255.

패턴: (\\$LATEST)|[0-9]\*

필수 여부: 아니요

## 응답 구문

```

HTTP/1.1 200
Content-type: application/json

{
  "arn": "string",
  "environment": {
    "uri": "string"
  },
  "imageDigest": "string",
  "lastUpdatedAt": number,
  "name": "string",
  "revisionId": "string",
  "robotSoftwareSuite": {
    "name": "string",
    "version": "string"
  },
  "sources": [
    {
      "architecture": "string",
      "etag": "string",
      "s3Bucket": "string",
      "s3Key": "string"
    }
  ],
  "tags": {
    "string" : "string"
  },
  "version": "string"
}

```

## 응답 요소

작업이 성공하면 서비스가 HTTP 200 응답을 반환합니다.

다음 데이터는 서비스에 의해 JSON 형식으로 반환됩니다.

### arn

로봇 애플리케이션의 Amazon 리소스 이름(ARN)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: `arn:.*`

### environment

로봇 애플리케이션을 만드는 데 사용된 도커 이미지 URI를 포함하는 객체입니다.

유형: [Environment](#) 객체

### imageDigest

로봇 애플리케이션에 사용하는 도커 이미지의 SHA256 식별자입니다.

타입: 문자열

길이 제한: 최소 길이는 0. 최대 길이는 72입니다.

패턴: `[Ss][Hh][Aa]256:[0-9a-fA-F]{64}`

### lastUpdatedAt

로봇 애플리케이션이 마지막으로 업데이트된 epoch 이후 경과 시간(밀리초)입니다.

유형: 타임스탬프

### name

로봇 애플리케이션의 이름입니다.

유형: 문자열

길이 제약: 최소 길이는 1. 최대 길이는 255.

패턴: `[a-zA-Z0-9_\-\-]*`

### revisionId

로봇 애플리케이션의 개정 ID입니다.

유형: 문자열

길이 제약: 최소 길이 1자. 최대 길이는 40입니다.

패턴: `[a-zA-Z0-9_.\-\-]*`

## [robotSoftwareSuite](#)

로봇 애플리케이션이 사용하는 로봇 소프트웨어 제품군입니다.

유형: [RobotSoftwareSuite](#) 객체

### [sources](#)

로봇 애플리케이션의 소스입니다.

유형: [Source](#) 객체 어레이

### [tags](#)

지정된 로봇 애플리케이션에 추가된 모든 태그의 목록입니다.

유형: 문자열-문자열 맵

맵 항목: 최소 항목 수는 0개입니다. 최대 항목 수는 50개입니다.

키 길이 제약 조건: 최소 길이는 1. 최대 길이 128.

키 패턴: [a-zA-Z0-9 \_.\-\/+=:]\*

값 길이 제약 조건: 최소 길이는 0입니다. 최대 길이는 256.

가치 패턴: [a-zA-Z0-9 \_.\-\/+=:]\*

### [version](#)

로봇 애플리케이션의 버전입니다.

유형: 문자열

길이 제약: 최소 길이는 1. 최대 길이는 255.

패턴: (\\$LATEST)|[0-9]\*

## Errors

모든 작업에서 발생하는 일반적인 오류에 대한 자세한 내용은 [일반적인 오류](#) 섹션을 참조하세요.

### InternalServerErrorException

RoboMaker AWS에서 서비스 문제가 발생했습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 500

#### InvalidParameterException

요청에 지정된 파라미터는 유효하지 않거나, 지원되지 않거나, 사용할 수 없습니다. 반환된 메시지는 오류 값에 대한 설명을 제공합니다.

HTTP 상태 코드: 400

#### ResourceNotFoundException

지정한 리소스가 존재하지 않습니다.

HTTP 상태 코드: 400

#### ThrottlingException

RoboMaker AWS는 일시적으로 요청을 처리할 수 없습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 400

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS Go v2를 위한 SDK](#)
- [AWS Java V2용 SDK](#)
- [AWS V3용 SDK JavaScript](#)
- [AWS PHP V3용 SDK](#)
- [AWS Python용 SDK](#)
- [AWS 루비 V3용 SDK](#)

## DescribeSimulationApplication

시뮬레이션 애플리케이션을 설명합니다.

### Request Syntax

```
POST /describeSimulationApplication HTTP/1.1
Content-type: application/json

{
  "application": "string",
  "applicationVersion": "string"
}
```

### URI 요청 파라미터

요청은 URI 파라미터를 사용하지 않습니다.

### 요청 본문

요청은 JSON 형식으로 다음 데이터를 받습니다.

#### [application](#)

시뮬레이션 애플리케이션에 대한 애플리케이션 정보입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

필수 사항 여부: Yes

#### [applicationVersion](#)

설명할 시뮬레이션 애플리케이션의 버전입니다.

유형: 문자열

길이 제약: 최소 길이는 1. 최대 길이는 255.

패턴: (\\$LATEST)|[0-9]\*

필수 여부: 아니요

## 응답 구문

```

HTTP/1.1 200
Content-type: application/json

{
  "arn": "string",
  "environment": {
    "uri": "string"
  },
  "imageDigest": "string",
  "lastUpdatedAt": number,
  "name": "string",
  "renderingEngine": {
    "name": "string",
    "version": "string"
  },
  "revisionId": "string",
  "robotSoftwareSuite": {
    "name": "string",
    "version": "string"
  },
  "simulationSoftwareSuite": {
    "name": "string",
    "version": "string"
  },
  "sources": [
    {
      "architecture": "string",
      "etag": "string",
      "s3Bucket": "string",
      "s3Key": "string"
    }
  ],
  "tags": {
    "string" : "string"
  },
  "version": "string"
}

```

## 응답 요소

작업이 성공하면 서비스가 HTTP 200 응답을 반환합니다.

다음 데이터는 서비스에 의해 JSON 형식으로 반환됩니다.

### arn

시뮬레이션 애플리케이션의 Amazon 리소스 이름(ARN)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

### environment

시뮬레이션 애플리케이션을 만드는 데 사용되는 도커 이미지 URI가 포함된 객체입니다.

유형: [Environment](#) 객체

### imageDigest

시뮬레이션 애플리케이션에 사용하는 도커 이미지의 SHA256 식별자입니다.

타입: 문자열

길이 제한: 최소 길이는 0. 최대 길이는 72입니다.

패턴: [Ss][Hh][Aa]256:[0-9a-fA-F]{64}

### lastUpdatedAt

시뮬레이션 애플리케이션이 마지막으로 업데이트된 epoch 이후 경과 시간(밀리초).

유형: 타임스탬프

### name

시뮬레이션 애플리케이션의 이름입니다.

유형: 문자열

길이 제약: 최소 길이는 1. 최대 길이는 255.

패턴: [a-zA-Z0-9\_\-]\*

### renderingEngine

시뮬레이션 애플리케이션의 렌더링 엔진입니다.

유형: [RenderingEngine](#) 객체

### [revisionId](#)

시뮬레이션 애플리케이션에 대한 개정 ID입니다.

유형: 문자열

길이 제약: 최소 길이 1자. 최대 길이는 40입니다.

패턴: `[a-zA-Z0-9_.\-]*`

### [robotSoftwareSuite](#)

로봇 소프트웨어 제품군에 대한 정보입니다.

유형: [RobotSoftwareSuite](#) 객체

### [simulationSoftwareSuite](#)

시뮬레이션 애플리케이션이 사용하는 시뮬레이션 소프트웨어 제품군입니다.

유형: [SimulationSoftwareSuite](#) 객체

### [sources](#)

시뮬레이션 애플리케이션의 소스입니다.

유형: [Source](#) 객체 어레이

### [tags](#)

지정된 시뮬레이션 애플리케이션에 추가된 모든 태그의 목록입니다.

유형: 문자열-문자열 맵

맵 항목: 최소 항목 수는 0개입니다. 최대 항목 수는 50개입니다.

키 길이 제약 조건: 최소 길이는 1. 최대 길이 128.

키 패턴: `[a-zA-Z0-9 _.\-\/+=:]*`

값 길이 제약 조건: 최소 길이는 0입니다. 최대 길이는 256.

가치 패턴: `[a-zA-Z0-9 _.\-\/+=:]*`

### [version](#)

시뮬레이션 애플리케이션 버전입니다.

유형: 문자열

길이 제약: 최소 길이는 1. 최대 길이는 255.

패턴: (\\$LATEST)|[0-9]\*

## Errors

모든 작업에서 발생하는 일반적인 오류에 대한 자세한 내용은 [일반적인 오류](#) 섹션을 참조하세요.

### InternalServerErrorException

RoboMaker AWS에서 서비스 문제가 발생했습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 500

### InvalidParameterException

요청에 지정된 파라미터는 유효하지 않거나, 지원되지 않거나, 사용할 수 없습니다. 반환된 메시지는 오류 값에 대한 설명을 제공합니다.

HTTP 상태 코드: 400

### ResourceNotFoundException

지정한 리소스가 존재하지 않습니다.

HTTP 상태 코드: 400

### ThrottlingException

RoboMaker AWS는 일시적으로 요청을 처리할 수 없습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 400

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)

- [AWS Go v2를 위한 SDK](#)
- [AWS Java V2용 SDK](#)
- [AWS V3용 SDK JavaScript](#)
- [AWS PHP V3용 SDK](#)
- [AWS Python용 SDK](#)
- [AWS 루비 V3용 SDK](#)

## DescribeSimulationJob

시뮬레이션 작업을 설명합니다.

### Request Syntax

```
POST /describeSimulationJob HTTP/1.1
Content-type: application/json

{
  "job": "string"
}
```

### URI 요청 파라미터

요청은 URI 파라미터를 사용하지 않습니다.

### 요청 본문

요청은 JSON 형식으로 다음 데이터를 받습니다.

#### job

설명할 시뮬레이션 작업의 Amazon 리소스 이름(ARN)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

필수 항목 여부: 예

### 응답 구문

```
HTTP/1.1 200
Content-type: application/json

{
  "arn": "string",
  "clientRequestToken": "string",
  "compute": {
    "computeType": "string",
```

```

    "gpuUnitLimit": number,
    "simulationUnitLimit": number
  },
  "dataSources": [
    {
      "destination": "string",
      "name": "string",
      "s3Bucket": "string",
      "s3Keys": [
        {
          "etag": "string",
          "s3Key": "string"
        }
      ],
      "type": "string"
    }
  ],
  "failureBehavior": "string",
  "failureCode": "string",
  "failureReason": "string",
  "iamRole": "string",
  "lastStartedAt": number,
  "lastUpdatedAt": number,
  "loggingConfig": {
    "recordAllRosTopics": boolean
  },
  "maxJobDurationInSeconds": number,
  "name": "string",
  "networkInterface": {
    "networkInterfaceId": "string",
    "privateIpAddress": "string",
    "publicIpAddress": "string"
  },
  "outputLocation": {
    "s3Bucket": "string",
    "s3Prefix": "string"
  },
  "robotApplications": [
    {
      "application": "string",
      "applicationVersion": "string",
      "launchConfig": {
        "command": [ "string" ],
        "environmentVariables": {

```

```

        "string" : "string"
    },
    "launchFile": "string",
    "packageName": "string",
    "portForwardingConfig": {
        "portMappings": [
            {
                "applicationPort": number,
                "enableOnPublicIp": boolean,
                "jobPort": number
            }
        ]
    },
    "streamUI": boolean
},
"tools": [
    {
        "command": "string",
        "exitBehavior": "string",
        "name": "string",
        "streamOutputToCloudWatch": boolean,
        "streamUI": boolean
    }
],
"uploadConfigurations": [
    {
        "name": "string",
        "path": "string",
        "uploadBehavior": "string"
    }
],
"useDefaultTools": boolean,
"useDefaultUploadConfigurations": boolean
}
],
"simulationApplications": [
    {
        "application": "string",
        "applicationVersion": "string",
        "launchConfig": {
            "command": [ "string" ],
            "environmentVariables": {
                "string" : "string"
            }
        }
    },

```

```

    "launchFile": "string",
    "packageName": "string",
    "portForwardingConfig": {
      "portMappings": [
        {
          "applicationPort": number,
          "enableOnPublicIp": boolean,
          "jobPort": number
        }
      ]
    },
    "streamUI": boolean
  },
  "tools": [
    {
      "command": "string",
      "exitBehavior": "string",
      "name": "string",
      "streamOutputToCloudWatch": boolean,
      "streamUI": boolean
    }
  ],
  "uploadConfigurations": [
    {
      "name": "string",
      "path": "string",
      "uploadBehavior": "string"
    }
  ],
  "useDefaultTools": boolean,
  "useDefaultUploadConfigurations": boolean,
  "worldConfigs": [
    {
      "world": "string"
    }
  ]
}
],
"simulationTimeMillis": number,
"status": "string",
"tags": {
  "string" : "string"
},
"vpcConfig": {

```

```

    "assignPublicIp": boolean,
    "securityGroups": [ "string" ],
    "subnets": [ "string" ],
    "vpcId": "string"
  }
}

```

## 응답 요소

작업이 성공하면 서비스가 HTTP 200 응답을 반환합니다.

다음 데이터는 서비스에 의해 JSON 형식으로 반환됩니다.

### [arn](#)

시뮬레이션 작업의 Amazon 리소스 이름(ARN)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

### [clientRequestToken](#)

요청 멱등성을 보장하기 위해 제공하는 고유한 대소문자 구분 식별자입니다.

유형: 문자열

길이 제한: 최소 길이는 1. 최대 길이는 64.

패턴: [a-zA-Z0-9\_\-=]\*

### [compute](#)

시뮬레이션 작업에 대한 컴퓨팅 정보입니다.

유형: [ComputeResponse](#) 객체

### [dataSources](#)

시뮬레이션 작업의 데이터 소스입니다.

유형: [DataSource](#) 객체 어레이

## failureBehavior

시뮬레이션 작업의 실패 동작입니다.

타입: 문자열

유효 값: Fail | Continue

## failureCode

시뮬레이션 작업 실패 시 발생하는 오류 코드:

InternalServerError

내부 서비스 오류.

RobotApplicationCrash

로봇 애플리케이션이 비정상적으로 종료되었습니다.

SimulationApplicationCrash

시뮬레이션 애플리케이션이 비정상적으로 종료되었습니다.

BadPermissionsRobotApplication

로봇 애플리케이션 번들을 다운로드할 수 없습니다.

BadPermissionsSimulationApplication

시뮬레이션 애플리케이션 번들을 다운로드할 수 없습니다.

BadPermissionsS3 출력

출력을 고객이 제공한 S3 버킷에 게시할 수 없습니다.

BadPermissionsCloudwatchLogs

고객이 CloudWatch 제공한 Logs 리소스에 로그를 게시할 수 없습니다.

SubnetIpLimitExceeded

서브넷 IP 제한을 초과했습니다.

ENI LimitExceeded

ENI 제한을 초과했습니다.

BadPermissionsUserCredentials

제공된 역할을 사용할 수 없습니다.

## InvalidBundleRobotApplication

로봇 번들을 추출할 수 없습니다(잘못된 형식, 번들링 오류 또는 기타 문제).

## InvalidBundleSimulationApplication

시뮬레이션 번들을 추출할 수 없습니다(잘못된 형식, 번들링 오류 또는 기타 문제).

## RobotApplicationVersionMismatchedEtag

버전 생성 중 Etag for 값이 일치하지 RobotApplication 않습니다.

## SimulationApplicationVersionMismatchedEtag

버전 생성 시 Etag for 값이 일치하지 SimulationApplication 않습니다.

타입: 문자열

유효 값: InternalServiceError | RobotApplicationCrash | SimulationApplicationCrash | RobotApplicationHealthCheckFailure | SimulationApplicationHealthCheckFailure | BadPermissionsRobotApplication | BadPermissionsSimulationApplication | BadPermissionsS3Object | BadPermissionsS3Output | BadPermissionsCloudwatchLogs | SubnetIpLimitExceeded | ENILimitExceeded | BadPermissionsUserCredentials | InvalidBundleRobotApplication | InvalidBundleSimulationApplication | InvalidS3Resource | ThrottlingError | LimitExceeded | MismatchedEtag | RobotApplicationVersionMismatchedEtag | SimulationApplicationVersionMismatchedEtag | ResourceNotFound | RequestThrottled | BatchTimedOut | BatchCanceled | InvalidInput | WrongRegionS3Bucket | WrongRegionS3Output | WrongRegionRobotApplication | WrongRegionSimulationApplication | UploadContentMismatchError

### failureReason

시뮬레이션 작업이 실패한 이유에 대한 세부 정보입니다. 흐름 문제 해결에 대한 자세한 내용은 [문제 해결](#)을 참조하세요.

타입: 문자열

길이 제약 조건: 최소 길이는 0입니다. 최대 길이는 1024입니다.

패턴: .\*

## iamRole

시뮬레이션 인스턴스가 사용자 대신 연결된 정책에 지정된 AWS API를 호출하도록 허용하는 IAM 역할입니다.

유형: 문자열

길이 제약: 최소 길이는 1. 최대 길이는 255.

패턴: `arn:aws:iam::\w+:role/.*`

## lastStartedAt

물레이션 작업이 마지막으로 시작된 epoch 이후 경과 시간(밀리초)

유형: 타임스탬프

## lastUpdatedAt

시뮬레이션 작업이 마지막으로 업데이트된 epoch 이후 경과 시간(밀리초)입니다.

유형: 타임스탬프

## loggingConfig

로깅 구성입니다.

유형: [LoggingConfig](#) 객체

## maxJobDurationInSeconds

최대 작업 시간(초)입니다. 값은 8일(691,200초) 이하여야 합니다.

타입: Long

## name

시뮬레이션 작업의 이름입니다.

유형: 문자열

길이 제약: 최소 길이는 1. 최대 길이는 255.

패턴: `[a-zA-Z0-9_\-\ ]*`

## networkInterface

시뮬레이션 작업에 대한 네트워크 인터페이스 정보입니다.

유형: [NetworkInterface](#) 객체

### [outputLocation](#)

시뮬레이션 작업에서 생성된 출력 파일의 위치입니다.

유형: [OutputLocation](#) 객체

### [robotApplications](#)

로봇 애플리케이션 목록입니다.

유형: [RobotApplicationConfig](#) 객체 어레이

배열 멤버: 고정된 항목 수는 1개입니다.

### [simulationApplications](#)

시뮬레이션 애플리케이션 목록입니다.

유형: [SimulationApplicationConfig](#) 객체 어레이

배열 멤버: 고정된 항목 수는 1개입니다.

### [simulationTimeMillis](#)

시뮬레이션 작업 실행 기간(밀리초)

타입: Long

### [status](#)

시뮬레이션 작업의 상태입니다.

타입: 문자열

유효 값: Pending | Preparing | Running | Restarting | Completed | Failed | RunningFailed | Terminating | Terminated | Canceled

### [tags](#)

지정된 시뮬레이션 작업에 추가된 모든 태그의 목록입니다.

유형: 문자열-문자열 맵

맵 항목: 최소 항목 수는 0개입니다. 최대 항목 수는 50개입니다.

키 길이 제약 조건: 최소 길이는 1. 최대 길이 128.

키 패턴: [a-zA-Z0-9 \_.\-\/+=:]\*

값 길이 제약 조건: 최소 길이는 0입니다. 최대 길이는 256.

가치 패턴: [a-zA-Z0-9 \_.\-\/+=:]\*

## [vpcConfig](#)

VPC 구성입니다.

유형: [VPCConfigResponse](#) 객체

## Errors

모든 작업에서 발생하는 일반적인 오류에 대한 자세한 내용은 [일반적인 오류](#) 섹션을 참조하세요.

### InternalServerError

RoboMaker AWS에서 서비스 문제가 발생했습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 500

### InvalidParameterException

요청에 지정된 파라미터는 유효하지 않거나, 지원되지 않거나, 사용할 수 없습니다. 반환된 메시지는 오류 값에 대한 설명을 제공합니다.

HTTP 상태 코드: 400

### ResourceNotFoundException

지정한 리소스가 존재하지 않습니다.

HTTP 상태 코드: 400

### ThrottlingException

RoboMaker AWS는 일시적으로 요청을 처리할 수 없습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 400

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS Go v2를 위한 SDK](#)
- [AWS Java V2용 SDK](#)
- [AWS V3용 SDK JavaScript](#)
- [AWS PHP V3용 SDK](#)
- [AWS Python용 SDK](#)
- [AWS 루비 V3용 SDK](#)

## DescribeSimulationJobBatch

시뮬레이션 작업 배치를 설명합니다.

### Request Syntax

```
POST /describeSimulationJobBatch HTTP/1.1
Content-type: application/json

{
  "batch": "string"
}
```

### URI 요청 파라미터

요청은 URI 파라미터를 사용하지 않습니다.

### 요청 본문

요청은 JSON 형식으로 다음 데이터를 받습니다.

#### batch

설명할 배치의 ID입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

필수 항목 여부: 예

### 응답 구문

```
HTTP/1.1 200
Content-type: application/json

{
  "arn": "string",
  "batchPolicy": {
    "maxConcurrency": number,
    "timeoutInSeconds": number
  }
}
```

```

},
"clientRequestToken": "string",
"createdAt": number,
"createdRequests": [
  {
    "arn": "string",
    "computeType": "string",
    "dataSourceNames": [ "string" ],
    "lastUpdatedAt": number,
    "name": "string",
    "robotApplicationNames": [ "string" ],
    "simulationApplicationNames": [ "string" ],
    "status": "string"
  }
],
"failedRequests": [
  {
    "failedAt": number,
    "failureCode": "string",
    "failureReason": "string",
    "request": {
      "compute": {
        "computeType": "string",
        "gpuUnitLimit": number,
        "simulationUnitLimit": number
      },
      "dataSources": [
        {
          "destination": "string",
          "name": "string",
          "s3Bucket": "string",
          "s3Keys": [ "string" ],
          "type": "string"
        }
      ],
      "failureBehavior": "string",
      "iamRole": "string",
      "loggingConfig": {
        "recordAllRosTopics": boolean
      },
      "maxJobDurationInSeconds": number,
      "outputLocation": {
        "s3Bucket": "string",
        "s3Prefix": "string"
      }
    }
  }
]

```

```

    },
    "robotApplications": [
      {
        "application": "string",
        "applicationVersion": "string",
        "launchConfig": {
          "command": [ "string" ],
          "environmentVariables": {
            "string" : "string"
          },
          "launchFile": "string",
          "packageName": "string",
          "portForwardingConfig": {
            "portMappings": [
              {
                "applicationPort": number,
                "enableOnPublicIp": boolean,
                "jobPort": number
              }
            ]
          },
          "streamUI": boolean
        },
        "tools": [
          {
            "command": "string",
            "exitBehavior": "string",
            "name": "string",
            "streamOutputToCloudWatch": boolean,
            "streamUI": boolean
          }
        ],
        "uploadConfigurations": [
          {
            "name": "string",
            "path": "string",
            "uploadBehavior": "string"
          }
        ],
        "useDefaultTools": boolean,
        "useDefaultUploadConfigurations": boolean
      }
    ],
    "simulationApplications": [

```

```
{
  "application": "string",
  "applicationVersion": "string",
  "launchConfig": {
    "command": [ "string" ],
    "environmentVariables": {
      "string" : "string"
    },
    "launchFile": "string",
    "packageName": "string",
    "portForwardingConfig": {
      "portMappings": [
        {
          "applicationPort": number,
          "enableOnPublicIp": boolean,
          "jobPort": number
        }
      ]
    },
    "streamUI": boolean
  },
  "tools": [
    {
      "command": "string",
      "exitBehavior": "string",
      "name": "string",
      "streamOutputToCloudWatch": boolean,
      "streamUI": boolean
    }
  ],
  "uploadConfigurations": [
    {
      "name": "string",
      "path": "string",
      "uploadBehavior": "string"
    }
  ],
  "useDefaultTools": boolean,
  "useDefaultUploadConfigurations": boolean,
  "worldConfigs": [
    {
      "world": "string"
    }
  ]
}
```

```

    }
  ],
  "tags": {
    "string" : "string"
  },
  "useDefaultApplications": boolean,
  "vpcConfig": {
    "assignPublicIp": boolean,
    "securityGroups": [ "string" ],
    "subnets": [ "string" ]
  }
}
],
"failureCode": "string",
"failureReason": "string",
"lastUpdatedAt": number,
"pendingRequests": [
  {
    "compute": {
      "computeType": "string",
      "gpuUnitLimit": number,
      "simulationUnitLimit": number
    },
    "dataSources": [
      {
        "destination": "string",
        "name": "string",
        "s3Bucket": "string",
        "s3Keys": [ "string" ],
        "type": "string"
      }
    ],
    "failureBehavior": "string",
    "iamRole": "string",
    "loggingConfig": {
      "recordAllRosTopics": boolean
    },
    "maxJobDurationInSeconds": number,
    "outputLocation": {
      "s3Bucket": "string",
      "s3Prefix": "string"
    },
    "robotApplications": [

```

```

{
  "application": "string",
  "applicationVersion": "string",
  "launchConfig": {
    "command": [ "string" ],
    "environmentVariables": {
      "string": "string"
    },
    "launchFile": "string",
    "packageName": "string",
    "portForwardingConfig": {
      "portMappings": [
        {
          "applicationPort": number,
          "enableOnPublicIp": boolean,
          "jobPort": number
        }
      ]
    },
    "streamUI": boolean
  },
  "tools": [
    {
      "command": "string",
      "exitBehavior": "string",
      "name": "string",
      "streamOutputToCloudWatch": boolean,
      "streamUI": boolean
    }
  ],
  "uploadConfigurations": [
    {
      "name": "string",
      "path": "string",
      "uploadBehavior": "string"
    }
  ],
  "useDefaultTools": boolean,
  "useDefaultUploadConfigurations": boolean
}
],
"simulationApplications": [
  {
    "application": "string",

```

```

    "applicationVersion": "string",
    "launchConfig": {
      "command": [ "string" ],
      "environmentVariables": {
        "string" : "string"
      },
      "launchFile": "string",
      "packageName": "string",
      "portForwardingConfig": {
        "portMappings": [
          {
            "applicationPort": number,
            "enableOnPublicIp": boolean,
            "jobPort": number
          }
        ]
      },
      "streamUI": boolean
    },
    "tools": [
      {
        "command": "string",
        "exitBehavior": "string",
        "name": "string",
        "streamOutputToCloudWatch": boolean,
        "streamUI": boolean
      }
    ],
    "uploadConfigurations": [
      {
        "name": "string",
        "path": "string",
        "uploadBehavior": "string"
      }
    ],
    "useDefaultTools": boolean,
    "useDefaultUploadConfigurations": boolean,
    "worldConfigs": [
      {
        "world": "string"
      }
    ]
  }
],

```

```

    "tags": {
      "string" : "string"
    },
    "useDefaultApplications": boolean,
    "vpcConfig": {
      "assignPublicIp": boolean,
      "securityGroups": [ "string" ],
      "subnets": [ "string" ]
    }
  }
],
"status": "string",
"tags": {
  "string" : "string"
}
}

```

## 응답 요소

작업이 성공하면 서비스가 HTTP 200 응답을 반환합니다.

다음 데이터는 서비스에 의해 JSON 형식으로 반환됩니다.

### [arn](#)

배치의 Amazon 리소스 이름(ARN)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

### [batchPolicy](#)

배치 정책입니다.

유형: [BatchPolicy](#) 객체

### [clientRequestToken](#)

요청 멱등성을 보장하기 위해 제공하는 고유한 대소문자 구분 식별자입니다.

유형: 문자열

길이 제한: 최소 길이는 1. 최대 길이는 64.

패턴: [a-zA-Z0-9\_\-=]\*

### createdAt

시뮬레이션 작업 배치가 마지막으로 생성된 epoch 이후 경과 시간(밀리초)입니다.

유형: 타임스탬프

### createdRequests

생성된 시뮬레이션 작업 요약 목록입니다.

유형: [SimulationJobSummary](#) 객체 어레이

어레이 멤버: 최소 항목 수 0개. 최대 항목 수는 100개입니다.

### failedRequests

실패한 시뮬레이션 작업 생성 요청 목록입니다. 요청이 시뮬레이션 작업을 생성하는데 실패했습니다. 실패한 요청에는 시뮬레이션 작업 ID가 없습니다.

유형: [FailedCreateSimulationJobRequest](#) 객체 어레이

### failureCode

시뮬레이션 작업 배치의 실패 코드입니다.

타입: 문자열

유효 값: InternalServiceError

### failureReason

시뮬레이션 작업 배치가 실패한 이유입니다.

타입: 문자열

길이 제약 조건: 최소 길이는 0입니다. 최대 길이는 1024입니다.

패턴: .\*

### lastUpdatedAt

시뮬레이션 작업 배치가 마지막으로 업데이트된 epoch 이후 경과 시간(밀리초)입니다.

유형: 타임스탬프

## pendingRequests

보류 중인 시뮬레이션 작업 요청 목록입니다. 이러한 요청은 아직 시뮬레이션 작업에 생성되지 않았습니다.

유형: [SimulationJobRequest](#) 객체 어레이

배열 멤버: 최소 항목 수는 1입니다. 최대 항목 수는 1000개입니다.

## status

배치의 상태입니다.

### 보류중

시뮬레이션 작업 배치 요청이 보류 중입니다.

### InProgress

시뮬레이션 작업 배치가 진행 중입니다.

### Failed

시뮬레이션 작업 배치가 실패했습니다. 내부 장애(예: `InternalServerError`)로 인해 하나 이상의 시뮬레이션 작업 요청을 완료할 수 없습니다. 자세한 내용은 `failureCode` 및 `failureReason` 섹션을 참조하세요.

### 완료됨

시뮬레이션 배치 작업이 완료되었습니다. 다음의 경우 배치가 완료됩니다. (1) 배치에 보류 중인 시뮬레이션 작업 요청이 없고 실패한 시뮬레이션 작업 요청 모두가 `InternalServerError`로 인한 것이 아닌 경우, (2) 생성된 모든 시뮬레이션 작업이 최종 상태(예: `Completed` 또는 `Failed`)에 도달한 경우.

### 취소됨

시뮬레이션 배치 작업이 취소되었습니다.

### 취소 중

시뮬레이션 배치 작업을 취소 중입니다.

### 완료 중

시뮬레이션 배치 작업을 완료 중입니다.

### TimingOut

시뮬레이션 작업 배치 시간이 초과되었습니다.

배치 시간이 초과되고 내부 장애(예: InternalServiceError)로 인해 실패한 보류 중인 요청이 있는 경우, 배치 상태는 Failed입니다. 이러한 실패한 요청이 없는 경우 배치 상태는 TimedOut입니다.

TimedOut

시뮬레이션 배치 작업 시간이 초과되었습니다.

타입: 문자열

유효 값: Pending | InProgress | Failed | Completed | Canceled | Canceling | Completing | TimingOut | TimedOut

## tags

시뮬레이션 작업 배치에 연결된 태그 키 및 태그 값을 포함하는 맵입니다.

유형: 문자열-문자열 맵

맵 항목: 최소 항목 수는 0개입니다. 최대 항목 수는 50개입니다.

키 길이 제약 조건: 최소 길이는 1. 최대 길이 128.

키 패턴: [a-zA-Z0-9 \_.\-\/+=:]\*

값 길이 제약 조건: 최소 길이는 0입니다. 최대 길이는 256.

가치 패턴: [a-zA-Z0-9 \_.\-\/+=:]\*

## Errors

모든 작업에서 발생하는 일반적인 오류에 대한 자세한 내용은 [일반적인 오류](#) 섹션을 참조하세요.

### InternalServerErrorException

RoboMaker AWS에서 서비스 문제가 발생했습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 500

### InvalidParameterException

요청에 지정된 파라미터는 유효하지 않거나, 지원되지 않거나, 사용할 수 없습니다. 반환된 메시지는 오류 값에 대한 설명을 제공합니다.

HTTP 상태 코드: 400

ResourceNotFoundException

지정한 리소스가 존재하지 않습니다.

HTTP 상태 코드: 400

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS Go v2를 위한 SDK](#)
- [AWS Java V2용 SDK](#)
- [AWS V3용 SDK JavaScript](#)
- [AWS PHP V3용 SDK](#)
- [AWS Python용 SDK](#)
- [AWS 루비 V3용 SDK](#)

## DescribeWorld

월드 설명입니다.

### Request Syntax

```
POST /describeWorld HTTP/1.1
Content-type: application/json

{
  "world": "string"
}
```

### URI 요청 파라미터

요청은 URI 파라미터를 사용하지 않습니다.

### 요청 본문

요청은 JSON 형식으로 다음 데이터를 받습니다.

#### world

설명하려는 월드의 Amazon 리소스 이름(ARN)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

필수 항목 여부: 예

### 응답 구문

```
HTTP/1.1 200
Content-type: application/json

{
  "arn": "string",
  "createdAt": number,
```

```

"generationJob": "string",
"tags": {
  "string" : "string"
},
"template": "string",
"worldDescriptionBody": "string"
}

```

## 응답 요소

작업이 성공하면 서비스가 HTTP 200 응답을 반환합니다.

다음 데이터는 서비스에 의해 JSON 형식으로 반환됩니다.

### arn

월드 of Amazon 리소스 이름(ARN)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

### createdAt

월드가 생성된 epoch 이후 경과 시간(밀리초)입니다.

유형: 타임스탬프

### generationJob

월드 생성 작업을 생성한 월드 생성 작업의 Amazon 리소스 이름(ARN)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

### tags

월드에 연결된 태그 키 및 태그 값을 포함하는 맵입니다.

유형: 문자열-문자열 맵

맵 항목: 최소 항목 수는 0개입니다. 최대 항목 수는 50개입니다.

키 길이 제약 조건: 최소 길이는 1. 최대 길이 128.

키 패턴: [a-zA-Z0-9 \_.\-\/+=:]\*

값 길이 제약 조건: 최소 길이는 0입니다. 최대 길이는 256.

가치 패턴: [a-zA-Z0-9 \_.\-\/+=:]\*

## template

월드 템플릿.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

## worldDescriptionBody

월드의 콘텐츠를 설명하는 JSON 형식의 문자열을 반환합니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 262144자입니다.

패턴: [\\S\\s]+

## Errors

모든 작업에서 발생하는 일반적인 오류에 대한 자세한 내용은 [일반적인 오류](#) 섹션을 참조하세요.

### InternalServerErrorException

RoboMaker AWS에서 서비스 문제가 발생했습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 500

### InvalidParameterException

요청에 지정된 파라미터는 유효하지 않거나, 지원되지 않거나, 사용할 수 없습니다. 반환된 메시지는 오류 값에 대한 설명을 제공합니다.

HTTP 상태 코드: 400

ResourceNotFoundException

지정한 리소스가 존재하지 않습니다.

HTTP 상태 코드: 400

ThrottlingException

RoboMaker AWS는 일시적으로 요청을 처리할 수 없습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 400

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS Go v2를 위한 SDK](#)
- [AWS Java V2용 SDK](#)
- [AWS V3용 SDK JavaScript](#)
- [AWS PHP V3용 SDK](#)
- [AWS Python용 SDK](#)
- [AWS 루비 V3용 SDK](#)

## DescribeWorldExportJob

월드 내보내기 작업 설명입니다.

### Request Syntax

```
POST /describeWorldExportJob HTTP/1.1
Content-type: application/json

{
  "job": "string"
}
```

### URI 요청 파라미터

요청은 URI 파라미터를 사용하지 않습니다.

### 요청 본문

요청은 JSON 형식으로 다음 데이터를 받습니다.

#### job

설명할 월드 내보내기 작업의 Amazon 리소스 이름(ARN)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

필수 항목 여부: 예

### 응답 구문

```
HTTP/1.1 200
Content-type: application/json

{
  "arn": "string",
  "clientRequestToken": "string",
}
```

```

    "createdAt": number,
    "failureCode": "string",
    "failureReason": "string",
    "iamRole": "string",
    "outputLocation": {
      "s3Bucket": "string",
      "s3Prefix": "string"
    },
    "status": "string",
    "tags": {
      "string" : "string"
    },
    "worlds": [ "string" ]
  }

```

## 응답 요소

작업이 성공하면 서비스가 HTTP 200 응답을 반환합니다.

다음 데이터는 서비스에 의해 JSON 형식으로 반환됩니다.

### arn

월드 내보내기 작업 Amazon 리소스 이름(ARN)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

### clientRequestToken

요청 멱등성을 보장하기 위해 제공하는 고유한 대소문자 구분 식별자입니다.

유형: 문자열

길이 제한: 최소 길이는 1. 최대 길이는 64.

패턴: [a-zA-Z0-9\_\-=]\*

### createdAt

월드 내보내기 작업이 생성된 epoch 이후 경과 시간(밀리초)입니다.

유형: 타임스탬프

### [failureCode](#)

월드 내보내기 작업이 실패한 경우의 실패 코드:

InternalServerError

내부 서비스 오류.

LimitExceeded

요청된 리소스가 허용된 최대 수를 초과하거나 동시 스트림 요청 수가 허용된 최대 수를 초과합니다.

ResourceNotFound

특정 리소스를 찾을 수 없습니다.

RequestThrottled

요청에 병목 현상이 발생했습니다.

InvalidInput

요청의 입력 파라미터가 유효하지 않습니다.

타입: 문자열

유효 값: InternalServerError | LimitExceeded | ResourceNotFound | RequestThrottled | InvalidInput | AccessDenied

### [failureReason](#)

월드 내보내기 작업이 실패한 이유입니다.

타입: 문자열

길이 제약 조건: 최소 길이는 0입니다. 최대 길이는 1024입니다.

패턴: .\*

### [iamRole](#)

월드 내보내기 프로세스에서 Amazon S3 버킷에 액세스하여 내보내기를 전송하는 데 사용하는 IAM 역할입니다.

유형: 문자열

길이 제약: 최소 길이는 1. 최대 길이는 255.

패턴: `arn:aws:iam::\w+:role/.*`

### outputLocation

출력 위치입니다.

유형: [OutputLocation](#) 객체

### status

월드 내보내기 작업의 상태입니다.

보류중

월드 내보내기 작업 요청이 보류 중입니다.

실행 중

월드 내보내기 작업이 실행 중입니다.

완료됨

월드 내보내기 작업이 완료되었습니다.

Failed

월드 내보내기 작업이 실패했습니다. 자세한 내용은 `failureCode` 및 `failureReason` 섹션을 참조하세요.

취소됨

월드 내보내기 작업이 취소되었습니다.

취소 중

월드 내보내기 작업을 취소 중입니다.

타입: 문자열

유효 값: `Pending` | `Running` | `Completed` | `Failed` | `Canceling` | `Canceled`

### tags

월드 내보내기 작업에 연결된 태그 키 및 태그 값을 포함하는 맵입니다.

유형: 문자열-문자열 맵

맵 항목: 최소 항목 수는 0개입니다. 최대 항목 수는 50개입니다.

키 길이 제약 조건: 최소 길이는 1. 최대 길이 128.

키 패턴: [a-zA-Z0-9 \_.\-\/+=:]\*

값 길이 제약 조건: 최소 길이는 0입니다. 최대 길이는 256.

가치 패턴: [a-zA-Z0-9 \_.\-\/+=:]\*

## worlds

내보낼 월드에 해당하는 Amazon 리소스 이름(arn) 목록입니다.

유형: 문자열 어레이

배열 멤버: 최소수는 1개입니다. 최대 항목 수는 100개입니다.

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

## Errors

모든 작업에서 발생하는 일반적인 오류에 대한 자세한 내용은 [일반적인 오류](#) 섹션을 참조하세요.

### InternalServerError

RoboMaker AWS에서 서비스 문제가 발생했습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 500

### InvalidParameterException

요청에 지정된 파라미터는 유효하지 않거나, 지원되지 않거나, 사용할 수 없습니다. 반환된 메시지는 오류 값에 대한 설명을 제공합니다.

HTTP 상태 코드: 400

### ResourceNotFoundException

지정한 리소스가 존재하지 않습니다.

HTTP 상태 코드: 400

## ThrottlingException

RoboMaker AWS는 일시적으로 요청을 처리할 수 없습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 400

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS Go v2를 위한 SDK](#)
- [AWS Java V2용 SDK](#)
- [AWS V3용 SDK JavaScript](#)
- [AWS PHP V3용 SDK](#)
- [AWS Python용 SDK](#)
- [AWS 루비 V3용 SDK](#)

## DescribeWorldGenerationJob

월드 생성 작업 설명입니다.

### Request Syntax

```
POST /describeWorldGenerationJob HTTP/1.1
Content-type: application/json

{
  "job": "string"
}
```

### URI 요청 파라미터

요청은 URI 파라미터를 사용하지 않습니다.

### 요청 본문

요청은 JSON 형식으로 다음 데이터를 받습니다.

#### job

설명할 월드 생성 작업의 Amazon 리소스 이름(arn)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

필수 항목 여부: 예

### 응답 구문

```
HTTP/1.1 200
Content-type: application/json

{
  "arn": "string",
  "clientRequestToken": "string",
}
```

```

"createdAt": number,
"failureCode": "string",
"failureReason": "string",
"finishedWorldsSummary": {
  "failureSummary": {
    "failures": [
      {
        "failureCode": "string",
        "failureCount": number,
        "sampleFailureReason": "string"
      }
    ],
    "totalFailureCount": number
  },
  "finishedCount": number,
  "succeededWorlds": [ "string" ]
},
"status": "string",
"tags": {
  "string" : "string"
},
"template": "string",
"worldCount": {
  "floorplanCount": number,
  "interiorCountPerFloorplan": number
},
"worldTags": {
  "string" : "string"
}
}

```

## 응답 요소

작업이 성공하면 서비스가 HTTP 200 응답을 반환합니다.

다음 데이터는 서비스에 의해 JSON 형식으로 반환됩니다.

### arn

월드 생성 작업의 Amazon 리소스 이름(arn)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: `arn:.*`

### clientRequestToken

요청 멱등성을 보장하기 위해 제공하는 고유한 대소문자 구분 식별자입니다.

유형: 문자열

길이 제한: 최소 길이는 1. 최대 길이는 64.

패턴: `[a-zA-Z0-9_\-=]*`

### createdAt

월드 생성 작업이 생성된 epoch 이후 경과 시간(밀리초)입니다.

유형: 타임스탬프

### failureCode

월드 생성 작업이 실패한 경우의 실패 코드:

`InternalServerError`

내부 서비스 오류.

`LimitExceeded`

요청된 리소스가 허용된 최대 수를 초과하거나 동시 스트림 요청 수가 허용된 최대 수를 초과합니다.

`ResourceNotFound`

특정 리소스를 찾을 수 없습니다.

`RequestThrottled`

요청에 병목 현상이 발생했습니다.

`InvalidInput`

요청의 입력 파라미터가 유효하지 않습니다.

타입: 문자열

유효 값: `InternalServerError | LimitExceeded | ResourceNotFound | RequestThrottled | InvalidInput | AllWorldGenerationFailed`

## failureReason

월드 생성 작업이 실패한 이유입니다.

타입: 문자열

길이 제약 조건: 최소 길이는 0입니다. 최대 길이는 1024입니다.

패턴: .\*

## finishedWorldsSummary

완성된 월드에 대한 요약 정보입니다.

유형: [FinishedWorldsSummary](#) 객체

## status

월드 생성 작업 상태:

보류중

월드 생성 작업 요청이 보류 중입니다.

실행 중

월드 생성 작업이 실행 중입니다.

완료됨

월드 생성 작업이 완료되었습니다.

Failed

월드 생성 작업이 실패했습니다. 자세한 정보는 `failureCode`을 참조하세요.

PartialFailed

일부 월드가 생성되지 않았습니다.

취소됨

월드 생성 작업이 취소되었습니다.

취소 중

월드 생성 작업을 취소 중입니다.

타입: 문자열

유효 값: Pending | Running | Completed | Failed | PartialFailed | Canceling  
| Canceled

### tags

월드 생성 작업에 연결된 태그 키 및 태그 값을 포함하는 맵입니다.

유형: 문자열-문자열 맵

맵 항목: 최소 항목 수는 0개입니다. 최대 항목 수는 50개입니다.

키 길이 제약 조건: 최소 길이는 1. 최대 길이 128.

키 패턴: [a-zA-Z0-9 \_.\-\/+=:]\*

값 길이 제약 조건: 최소 길이는 0입니다. 최대 길이는 256.

가치 패턴: [a-zA-Z0-9 \_.\-\/+=:]\*

### template

월드 템플릿의 Amazon 리소스 이름(ARN)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

### worldCount

월드 수에 대한 정보입니다.

유형: [WorldCount](#) 객체

### worldTags

생성된 월드 작업에 연결된 태그 키 및 태그 값을 포함하는 맵입니다.

유형: 문자열-문자열 맵

맵 항목: 최소 항목 수는 0개입니다. 최대 항목 수는 50개입니다.

키 길이 제약 조건: 최소 길이는 1. 최대 길이 128.

키 패턴: [a-zA-Z0-9 \_.\-\/+=:]\*

값 길이 제약 조건: 최소 길이는 0입니다. 최대 길이는 256.

가치 패턴: [a-zA-Z0-9 \_.\-\/+=:]\*

## Errors

모든 작업에서 발생하는 일반적인 오류에 대한 자세한 내용은 [일반적인 오류](#) 섹션을 참조하세요.

### InternalServerErrorException

RoboMaker AWS에서 서비스 문제가 발생했습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 500

### InvalidParameterException

요청에 지정된 파라미터는 유효하지 않거나, 지원되지 않거나, 사용할 수 없습니다. 반환된 메시지는 오류 값에 대한 설명을 제공합니다.

HTTP 상태 코드: 400

### ResourceNotFoundException

지정한 리소스가 존재하지 않습니다.

HTTP 상태 코드: 400

### ThrottlingException

RoboMaker AWS는 일시적으로 요청을 처리할 수 없습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 400

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS Go v2를 위한 SDK](#)
- [AWS Java V2용 SDK](#)

- [AWS V3용 SDK JavaScript](#)
- [AWS PHP V3용 SDK](#)
- [AWS Python용 SDK](#)
- [AWS 루비 V3용 SDK](#)

# DescribeWorldTemplate

월드 템플릿 설명입니다.

## Request Syntax

```
POST /describeWorldTemplate HTTP/1.1
Content-type: application/json

{
  "template": "string"
}
```

## URI 요청 파라미터

요청은 URI 파라미터를 사용하지 않습니다.

## 요청 본문

요청은 JSON 형식으로 다음 데이터를 받습니다.

### template

설명하려는 월드 템플릿의 Amazon 리소스 이름(arn)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

필수 항목 여부: 예

## 응답 구문

```
HTTP/1.1 200
Content-type: application/json

{
  "arn": "string",
  "clientRequestToken": "string",
}
```

```

    "createdAt": number,
    "lastUpdatedAt": number,
    "name": "string",
    "tags": {
      "string" : "string"
    },
    "version": "string"
  }

```

## 응답 요소

작업이 성공하면 서비스가 HTTP 200 응답을 반환합니다.

다음 데이터는 서비스에 의해 JSON 형식으로 반환됩니다.

### [arn](#)

월드 템플릿의 Amazon 리소스 이름(ARN)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

### [clientRequestToken](#)

요청 멱등성을 보장하기 위해 제공하는 고유한 대소문자 구분 식별자입니다.

유형: 문자열

길이 제한: 최소 길이는 1. 최대 길이는 64.

패턴: [a-zA-Z0-9\_\-=]\*

### [createdAt](#)

월드 템플릿이 생성된 epoch 이후 경과 시간(밀리초)입니다.

유형: 타임스탬프

### [lastUpdatedAt](#)

월드 템플릿이 업데이트된 epoch 이후 경과 시간(밀리초)입니다.

유형: 타임스탬프

## name

월드 템플릿의 이름입니다.

타입: 문자열

길이 제약 조건: 최소 길이는 0입니다. 최대 길이는 255.

패턴: .\*

## tags

월드 템플릿에 연결된 태그 키 및 태그 값을 포함하는 맵입니다.

유형: 문자열-문자열 맵

맵 항목: 최소 항목 수는 0개입니다. 최대 항목 수는 50개입니다.

키 길이 제약 조건: 최소 길이는 1. 최대 길이 128.

키 패턴: [a-zA-Z0-9 \_.\-\/+=:]\*

값 길이 제약 조건: 최소 길이는 0입니다. 최대 길이는 256.

가치 패턴: [a-zA-Z0-9 \_.\-\/+=:]\*

## version

사용 중인 월드 템플릿의 버전입니다.

타입: 문자열

길이 제약 조건: 최소 길이는 0입니다. 최대 길이는 1024입니다.

패턴: .\*

## Errors

모든 작업에서 발생하는 일반적인 오류에 대한 자세한 내용은 [일반적인 오류](#) 섹션을 참조하세요.

### InternalServerError

RoboMaker AWS에서 서비스 문제가 발생했습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 500

## InvalidParameterException

요청에 지정된 파라미터는 유효하지 않거나, 지원되지 않거나, 사용할 수 없습니다. 반환된 메시지는 오류 값에 대한 설명을 제공합니다.

HTTP 상태 코드: 400

## ResourceNotFoundException

지정한 리소스가 존재하지 않습니다.

HTTP 상태 코드: 400

## ThrottlingException

RoboMaker AWS는 일시적으로 요청을 처리할 수 없습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 400

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS Go v2를 위한 SDK](#)
- [AWS Java V2용 SDK](#)
- [AWS V3용 SDK JavaScript](#)
- [AWS PHP V3용 SDK](#)
- [AWS Python용 SDK](#)
- [AWS 루비 V3용 SDK](#)

## GetWorldTemplateBody

월드 템플릿 본문을 가져옵니다.

### Request Syntax

```
POST /getWorldTemplateBody HTTP/1.1
Content-type: application/json

{
  "generationJob": "string",
  "template": "string"
}
```

### URI 요청 파라미터

요청은 URI 파라미터를 사용하지 않습니다.

### 요청 본문

요청은 JSON 형식으로 다음 데이터를 받습니다.

#### generationJob

월드 생성기 작업의 Amazon 리소스 이름(ARN)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

Required: No

#### template

월드 템플릿의 Amazon 리소스 이름(ARN)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

필수 여부: 아니요

## 응답 구문

```
HTTP/1.1 200
Content-type: application/json

{
  "templateBody": "string"
}
```

## 응답 요소

작업이 성공하면 서비스가 HTTP 200 응답을 반환합니다.

다음 데이터는 서비스에 의해 JSON 형식으로 반환됩니다.

### templateBody

월드 템플릿 본문입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 262144자입니다.

패턴: `[\S\s]+`

## Errors

모든 작업에서 발생하는 일반적인 오류에 대한 자세한 내용은 [일반적인 오류](#) 섹션을 참조하세요.

### InternalServerErrorException

RoboMaker AWS에서 서비스 문제가 발생했습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 500

### InvalidParameterException

요청에 지정된 파라미터는 유효하지 않거나, 지원되지 않거나, 사용할 수 없습니다. 반환된 메시지는 오류 값에 대한 설명을 제공합니다.

HTTP 상태 코드: 400

ResourceNotFoundException

지정한 리소스가 존재하지 않습니다.

HTTP 상태 코드: 400

ThrottlingException

RoboMaker AWS는 일시적으로 요청을 처리할 수 없습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 400

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS Go v2를 위한 SDK](#)
- [AWS Java V2용 SDK](#)
- [AWS V3용 SDK JavaScript](#)
- [AWS PHP V3용 SDK](#)
- [AWS Python용 SDK](#)
- [AWS 루비 V3용 SDK](#)

## ListDeploymentJobs

이 작업은 사용 중단되었습니다.

### ⚠ Important

해당 API는 더 이상 지원되지 않습니다. 자세한 내용은 [지원 정책](#) 페이지에서 2022년 5월 2일 업데이트를 참조하세요.

플릿의 배포 목록을 반환합니다. 선택적으로 필터를 제공하여 특정 배포 작업을 검색할 수 있습니다.

## Request Syntax

```
POST /listDeploymentJobs HTTP/1.1
Content-type: application/json

{
  "filters": [
    {
      "name": "string",
      "values": [ "string" ]
    }
  ],
  "maxResults": number,
  "nextToken": "string"
}
```

## URI 요청 파라미터

요청은 URI 파라미터를 사용하지 않습니다.

## 요청 본문

요청은 JSON 형식으로 다음 데이터를 받습니다.

### filters

결과를 제한하기 위한 선택적 필터입니다.

지원되는 필터 키는 `status` 및 `fleetName` 입니다. 필터링할 때는 필터링된 항목의 전체 값을 사용해야 합니다. 필터는 최대 3개까지 사용할 수 있지만, 이름이 동일한 항목에 대한 필터여야 합니다. 예를 들어, 상태 `InProgress` 또는 상태 `Pending`가 있는 항목을 검색하는 경우가 있습니다.

유형: [Filter](#) 객체 어레이

배열 멤버: 고정된 항목 수는 1개입니다.

필수 여부: 아니요

### [maxResults](#)

이 파라미터를 사용하면 `ListDeploymentJobs`은 `maxResults` 응답 요소와 함께 단일 페이지에서 `nextToken` 결과만 반환합니다. 반환된 `ListDeploymentJobs` 값과 함께 다른 `nextToken` 요청을 보내면 초기 요청의 나머지 결과를 확인할 수 있습니다. 이 값은 1~200이어야 합니다. 이 파라미터를 사용하지 않는 경우 `ListDeploymentJobs`는 최대 200개의 결과와 해당하는 경우 `nextToken` 값을 반환합니다.

유형: 정수

필수 항목 여부: 아니요

### [nextToken](#)

이전 페이지 매김 요청에서 나머지 결과를 모두 반환하지 않은 경우, 응답 객체의 `nextToken` 파라미터 값이 토큰으로 설정됩니다. 다음 결과 세트를 검색하려면 `ListDeploymentJobs`를 다시 호출하여 해당 토큰을 요청 객체의 `nextToken` 파라미터에 할당합니다. 남은 결과가 없는 경우 이전 응답 객체의 `NextToken` 파라미터는 `null`로 설정됩니다.

유형: 문자열

길이 제약: 최소 길이는 1. 최대 길이는 2,048.

패턴: `[a-zA-Z0-9_.\-\/+=]*`

필수 여부: 아니요

## 응답 구문

```
HTTP/1.1 200
Content-type: application/json
```

```

{
  "deploymentJobs": [
    {
      "arn": "string",
      "createdAt": number,
      "deploymentApplicationConfigs": [
        {
          "application": "string",
          "applicationVersion": "string",
          "launchConfig": {
            "environmentVariables": {
              "string": "string"
            },
            "launchFile": "string",
            "packageName": "string",
            "postLaunchFile": "string",
            "preLaunchFile": "string"
          }
        }
      ],
      "deploymentConfig": {
        "concurrentDeploymentPercentage": number,
        "downloadConditionFile": {
          "bucket": "string",
          "etag": "string",
          "key": "string"
        },
        "failureThresholdPercentage": number,
        "robotDeploymentTimeoutInSeconds": number
      },
      "failureCode": "string",
      "failureReason": "string",
      "fleet": "string",
      "status": "string"
    }
  ],
  "nextToken": "string"
}

```

## 응답 요소

작업이 성공하면 서비스가 HTTP 200 응답을 반환합니다.

다음 데이터는 서비스에 의해 JSON 형식으로 반환됩니다.

## [deploymentJobs](#)

요청 기준을 충족하는 배포 작업 목록입니다.

유형: [DeploymentJob](#) 객체 어레이

어레이 멤버: 최소 항목 수 0개. 최대 항목 수는 200개입니다.

## [nextToken](#)

이전 페이지 매김 요청에서 나머지 결과를 모두 반환하지 않은 경우, 응답 객체의 nextToken 파라미터 값이 토큰으로 설정됩니다. 다음 결과 세트를 검색하려면 ListDeploymentJobs를 다시 호출하여 해당 토큰을 요청 객체의 nextToken 파라미터에 할당합니다. 남은 결과가 없는 경우 이전 응답 객체의 NextToken 매개 변수는 null로 설정됩니다.

유형: 문자열

길이 제약: 최소 길이는 1. 최대 길이는 2,048.

패턴: [a-zA-Z0-9\_.\-\/+=]\*

## Errors

모든 작업에서 발생하는 일반적인 오류에 대한 자세한 내용은 [일반적인 오류](#) 섹션을 참조하세요.

### InternalServerErrorException

RoboMaker AWS에서 서비스 문제가 발생했습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 500

### InvalidParameterException

요청에 지정된 파라미터는 유효하지 않거나, 지원되지 않거나, 사용할 수 없습니다. 반환된 메시지는 오류 값에 대한 설명을 제공합니다.

HTTP 상태 코드: 400

### ResourceNotFoundException

지정한 리소스가 존재하지 않습니다.

HTTP 상태 코드: 400

## ThrottlingException

RoboMaker AWS는 일시적으로 요청을 처리할 수 없습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 400

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS Go v2를 위한 SDK](#)
- [AWS Java V2용 SDK](#)
- [AWS V3용 SDK JavaScript](#)
- [AWS PHP V3용 SDK](#)
- [AWS Python용 SDK](#)
- [AWS 루비 V3용 SDK](#)

## ListFleets

이 작업은 사용 중단되었습니다.

### Important

해당 API는 더 이상 지원되지 않습니다. 자세한 내용은 [지원 정책](#) 페이지에서 2022년 5월 2일 업데이트를 참조하세요.

플릿 목록을 반환합니다. 선택적으로 필터를 제공하여 특정 플릿을 검색할 수 있습니다.

## Request Syntax

```
POST /listFleets HTTP/1.1
Content-type: application/json
```

```
{
  "filters": [
    {
      "name": "string",
      "values": [ "string" ]
    }
  ],
  "maxResults": number,
  "nextToken": "string"
}
```

## URI 요청 파라미터

요청은 URI 파라미터를 사용하지 않습니다.

## 요청 본문

요청은 JSON 형식으로 다음 데이터를 받습니다.

### filters

결과를 제한하기 위한 선택적 필터입니다.

필터 이름 name이 지원됩니다. 필터링할 때는 필터링된 항목의 전체 값을 사용해야 합니다. 최대 3개의 필터를 사용할 수 있습니다.

유형: [Filter](#) 객체 어레이

배열 멤버: 고정된 항목 수는 1개입니다.

필수 여부: 아니요

### [maxResults](#)

이 파라미터를 사용하면 ListFleets은 maxResults 응답 요소와 함께 단일 페이지에서 nextToken 결과만 반환합니다. 반환된 ListFleets 값과 함께 다른 nextToken 요청을 보내면 초기 요청의 나머지 결과를 확인할 수 있습니다. 이 값은 1~200이어야 합니다. 이 파라미터를 사용하지 않는 경우 ListFleets는 최대 200개의 결과와 해당하는 경우 nextToken 값을 반환합니다.

유형: 정수

필수 항목 여부: 아니요

### [nextToken](#)

이전 페이지 매김 요청에서 나머지 결과를 모두 반환하지 않은 경우, 응답 객체의 nextToken 파라미터 값이 토큰으로 설정됩니다. 다음 결과 세트를 검색하려면 ListFleets를 다시 호출하여 해당 토큰을 요청 객체의 nextToken 파라미터에 할당합니다. 남은 결과가 없는 경우 이전 응답 객체의 NextToken 파라미터는 null로 설정됩니다.

#### Note

이 토큰은 목록의 다음 항목을 검색하는 데만 사용되는 불투명 식별자로 취급되어야 하며, 다른 프로그래밍 용도로는 사용되지 않습니다.

유형: 문자열

길이 제약: 최소 길이는 1. 최대 길이는 2,048.

패턴: `[a-zA-Z0-9_.\-\/+=]*`

필수 여부: 아니요

## 응답 구문

```
HTTP/1.1 200
```

```
Content-type: application/json

{
  "fleetDetails": [
    {
      "arn": "string",
      "createdAt": number,
      "lastDeploymentJob": "string",
      "lastDeploymentStatus": "string",
      "lastDeploymentTime": number,
      "name": "string"
    }
  ],
  "nextToken": "string"
}
```

## 응답 요소

작업이 성공하면 서비스가 HTTP 200 응답을 반환합니다.

다음 데이터는 서비스에 의해 JSON 형식으로 반환됩니다.

### fleetDetails

요청 기준을 충족하는 플릿 세부 정보 목록입니다.

유형: [Fleet](#) 객체 어레이

어레이 멤버: 최소 항목 수 0개. 최대 항목 수는 200개입니다.

### nextToken

이전 페이지 매김 요청에서 나머지 결과를 모두 반환하지 않은 경우, 응답 객체의 nextToken 파라미터 값이 토큰으로 설정됩니다. 다음 결과 세트를 검색하려면 ListFleets를 다시 호출하여 해당 토큰을 요청 객체의 nextToken 파라미터에 할당합니다. 남은 결과가 없는 경우 이전 응답 객체의 NextToken 매개 변수는 null로 설정됩니다.

유형: 문자열

길이 제약: 최소 길이는 1. 최대 길이는 2,048.

패턴: [a-zA-Z0-9\_.\-\/+=]\*

## Errors

모든 작업에서 발생하는 일반적인 오류에 대한 자세한 내용은 [일반적인 오류](#) 섹션을 참조하세요.

### InternalServerErrorException

RoboMaker AWS에서 서비스 문제가 발생했습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 500

### InvalidParameterException

요청에 지정된 파라미터는 유효하지 않거나, 지원되지 않거나, 사용할 수 없습니다. 반환된 메시지는 오류 값에 대한 설명을 제공합니다.

HTTP 상태 코드: 400

### ResourceNotFoundException

지정한 리소스가 존재하지 않습니다.

HTTP 상태 코드: 400

### ThrottlingException

RoboMaker AWS는 일시적으로 요청을 처리할 수 없습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 400

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS Go v2를 위한 SDK](#)
- [AWS Java V2용 SDK](#)
- [AWS V3용 SDK JavaScript](#)
- [AWS PHP V3용 SDK](#)
- [AWS Python용 SDK](#)

- [AWS 루비 V3용 SDK](#)

## ListRobotApplications

로봇 애플리케이션의 목록을 반환합니다. 선택적으로 필터를 제공하여 특정 로봇 애플리케이션을 검색할 수 있습니다.

### Request Syntax

```
POST /listRobotApplications HTTP/1.1
Content-type: application/json

{
  "filters": [
    {
      "name": "string",
      "values": [ "string" ]
    }
  ],
  "maxResults": number,
  "nextToken": "string",
  "versionQualifier": "string"
}
```

### URI 요청 파라미터

요청은 URI 파라미터를 사용하지 않습니다.

### 요청 본문

요청은 JSON 형식으로 다음 데이터를 받습니다.

#### filters

결과를 제한하기 위한 선택적 필터입니다.

필터 이름 name이 지원됩니다. 필터링할 때는 필터링된 항목의 전체 값을 사용해야 합니다. 최대 3개의 필터를 사용할 수 있습니다.

유형: [Filter](#) 객체 어레이

배열 멤버: 고정된 항목 수는 1개입니다.

필수 여부: 아니요

## maxResults

이 파라미터를 사용하면 ListRobotApplications은 maxResults 응답 요소와 함께 단일 페이지에서 nextToken 결과만 반환합니다. 반환된 nextToken 값과 함께 다른 ListRobotApplications 요청을 보내면 초기 요청의 나머지 결과를 확인할 수 있습니다. 이 값은 1~100이어야 합니다. 이 파라미터를 사용하지 않는 경우 ListRobotApplications은 최대 100개의 결과와 해당하는 경우 nextToken 값을 반환합니다.

유형: 정수

필수 항목 여부: 아니요

## nextToken

이전 페이지 매김 요청에서 나머지 결과를 모두 반환하지 않은 경우, 응답 객체의 nextToken 파라미터 값이 토큰으로 설정됩니다. 다음 결과 세트를 검색하려면 ListRobotApplications를 다시 호출하여 해당 토큰을 요청 객체의 nextToken 파라미터에 할당합니다. 남은 결과가 없는 경우 이전 응답 객체의 NextToken 파라미터는 null로 설정됩니다.

유형: 문자열

길이 제약: 최소 길이는 1. 최대 길이는 2,048.

패턴: [a-zA-Z0-9\_.\-\/+=]\*

Required: No

## versionQualifier

로봇 애플리케이션의 버전 검증입니다.

유형: 문자열

길이 제약: 최소 길이는 1. 최대 길이는 255.

패턴: ALL

필수 여부: 아니요

## 응답 구문

```
HTTP/1.1 200
```

```
Content-type: application/json

{
  "nextToken": "string",
  "robotApplicationSummaries": [
    {
      "arn": "string",
      "lastUpdatedAt": number,
      "name": "string",
      "robotSoftwareSuite": {
        "name": "string",
        "version": "string"
      },
      "version": "string"
    }
  ]
}
```

## 응답 요소

작업이 성공하면 서비스가 HTTP 200 응답을 반환합니다.

다음 데이터는 서비스에 의해 JSON 형식으로 반환됩니다.

### [nextToken](#)

이전 페이지 매김 요청에서 나머지 결과를 모두 반환하지 않은 경우, 응답 객체의 nextToken 파라미터 값이 토큰으로 설정됩니다. 다음 결과 세트를 검색하려면 ListRobotApplications를 다시 호출하여 해당 토큰을 요청 객체의 nextToken 파라미터에 할당합니다. 남은 결과가 없는 경우 이전 응답 객체의 NextToken 매개 변수는 null로 설정됩니다.

유형: 문자열

길이 제약: 최소 길이는 1. 최대 길이는 2,048.

패턴: [a-zA-Z0-9\_.\-\/+=]\*

### [robotApplicationSummaries](#)

요청 기준을 충족하는 로봇 애플리케이션 요약 목록입니다.

유형: [RobotApplicationSummary](#) 객체 어레이

어레이 멤버: 최소 항목 수 0개. 최대 항목 수는 100개입니다.

## Errors

모든 작업에서 발생하는 일반적인 오류에 대한 자세한 내용은 [일반적인 오류](#) 섹션을 참조하세요.

### InternalServerErrorException

RoboMaker AWS에서 서비스 문제가 발생했습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 500

### InvalidParameterException

요청에 지정된 파라미터는 유효하지 않거나, 지원되지 않거나, 사용할 수 없습니다. 반환된 메시지는 오류 값에 대한 설명을 제공합니다.

HTTP 상태 코드: 400

### ThrottlingException

RoboMaker AWS는 일시적으로 요청을 처리할 수 없습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 400

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS Go v2를 위한 SDK](#)
- [AWS Java V2용 SDK](#)
- [AWS V3용 SDK JavaScript](#)
- [AWS PHP V3용 SDK](#)
- [AWS Python용 SDK](#)
- [AWS 루비 V3용 SDK](#)

## ListRobots

이 작업은 사용 중단되었습니다.

### Important

해당 API는 더 이상 지원되지 않습니다. 자세한 내용은 [지원 정책](#) 페이지에서 2022년 5월 2일 업데이트를 참조하세요.

로봇 목록을 반환합니다. 선택적으로 특정 로봇을 검색하기 위한 필터를 제공할 수 있습니다.

## Request Syntax

```
POST /listRobots HTTP/1.1
Content-type: application/json
```

```
{
  "filters": [
    {
      "name": "string",
      "values": [ "string" ]
    }
  ],
  "maxResults": number,
  "nextToken": "string"
}
```

## URI 요청 파라미터

요청은 URI 파라미터를 사용하지 않습니다.

## 요청 본문

요청은 JSON 형식으로 다음 데이터를 받습니다.

### filters

결과를 제한하기 위한 선택적 필터입니다.

지원되는 필터 키는 status 및 fleetName 입니다. 필터링할 때는 필터링된 항목의 전체 값을 사용해야 합니다. 필터는 최대 3개까지 사용할 수 있지만, 이름이 동일한 항목에 대한 필터여야 합니다.

다. 예를 들어, 상태 Registered 또는 상태 Available가 있는 항목을 검색하는 경우가 있습니다.

유형: [Filter](#) 객체 어레이

배열 멤버: 고정된 항목 수는 1개입니다.

필수 여부: 아니요

### [maxResults](#)

이 파라미터를 사용하면 ListRobots은 maxResults 응답 요소와 함께 단일 페이지에서 nextToken 결과만 반환합니다. 반환된 ListRobots 값과 함께 다른 nextToken 요청을 보내면 초기 요청의 나머지 결과를 확인할 수 있습니다. 이 값은 1~200이어야 합니다. 이 파라미터를 사용하지 않는 경우 ListRobots는 최대 200개의 결과와 해당하는 경우 nextToken 값을 반환합니다.

유형: 정수

필수 항목 여부: 아니요

### [nextToken](#)

이전 페이지 매김 요청에서 나머지 결과를 모두 반환하지 않은 경우, 응답 객체의 nextToken 파라미터 값이 토큰으로 설정됩니다. 다음 결과 세트를 검색하려면 ListRobots를 다시 호출하여 해당 토큰을 요청 객체의 nextToken 파라미터에 할당합니다. 남은 결과가 없는 경우 이전 응답 객체의 NextToken 파라미터는 null로 설정됩니다.

유형: 문자열

길이 제약: 최소 길이는 1. 최대 길이는 2,048.

패턴: [a-zA-Z0-9\_.\-\/+=]\*

필수 여부: 아니요

## 응답 구문

```
HTTP/1.1 200
Content-type: application/json

{
  "nextToken": "string",
  "robots": [
    {
```

```

    "architecture": "string",
    "arn": "string",
    "createdAt": number,
    "fleetArn": "string",
    "greenGrassGroupId": "string",
    "lastDeploymentJob": "string",
    "lastDeploymentTime": number,
    "name": "string",
    "status": "string"
  }
]
}

```

## 응답 요소

작업이 성공하면 서비스가 HTTP 200 응답을 반환합니다.

다음 데이터는 서비스에 의해 JSON 형식으로 반환됩니다.

### [nextToken](#)

이전 페이지 매김 요청에서 나머지 결과를 모두 반환하지 않은 경우, 응답 객체의 nextToken 파라미터 값이 토큰으로 설정됩니다. 다음 결과 세트를 검색하려면 ListRobots를 다시 호출하여 해당 토큰을 요청 객체의 nextToken 파라미터에 할당합니다. 남은 결과가 없는 경우 이전 응답 개체의 NextToken 매개 변수는 null로 설정됩니다.

유형: 문자열

길이 제약: 최소 길이는 1. 최대 길이는 2,048.

패턴: [a-zA-Z0-9\_.\-\/+=]\*

### [robots](#)

요청 기준을 충족하는 로봇 목록입니다.

유형: [Robot](#) 객체 어레이

어레이 멤버: 최소 항목 수 0개. 최대 항목 수는 1000개입니다.

## Errors

모든 작업에서 발생하는 일반적인 오류에 대한 자세한 내용은 [일반적인 오류](#) 섹션을 참조하세요.

## InternalServerErrorException

RoboMaker AWS에서 서비스 문제가 발생했습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 500

## InvalidParameterException

요청에 지정된 파라미터는 유효하지 않거나, 지원되지 않거나, 사용할 수 없습니다. 반환된 메시지는 오류 값에 대한 설명을 제공합니다.

HTTP 상태 코드: 400

## ResourceNotFoundException

지정한 리소스가 존재하지 않습니다.

HTTP 상태 코드: 400

## ThrottlingException

RoboMaker AWS는 일시적으로 요청을 처리할 수 없습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 400

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS Go v2를 위한 SDK](#)
- [AWS Java V2용 SDK](#)
- [AWS V3용 SDK JavaScript](#)
- [AWS PHP V3용 SDK](#)
- [AWS Python용 SDK](#)
- [AWS 루비 V3용 SDK](#)

## ListSimulationApplications

시뮬레이션 애플리케이션의 목록을 반환합니다. 선택적으로 필터를 제공하여 특정 시뮬레이션 애플리케이션을 검색할 수 있습니다.

### Request Syntax

```
POST /listSimulationApplications HTTP/1.1
Content-type: application/json

{
  "filters": [
    {
      "name": "string",
      "values": [ "string" ]
    }
  ],
  "maxResults": number,
  "nextToken": "string",
  "versionQualifier": "string"
}
```

### URI 요청 파라미터

요청은 URI 파라미터를 사용하지 않습니다.

### 요청 본문

요청은 JSON 형식으로 다음 데이터를 받습니다.

#### filters

결과를 제한하기 위한 선택적 필터 목록입니다.

필터 이름 `name`이 지원됩니다. 필터링할 때는 필터링된 항목의 전체 값을 사용해야 합니다. 최대 3개의 필터를 사용할 수 있습니다.

유형: [Filter](#) 객체 어레이

배열 멤버: 고정된 항목 수는 1개입니다.

필수 여부: 아니요

## maxResults

이 파라미터를 사용하면 ListSimulationApplications은 maxResults 응답 요소와 함께 단일 페이지에서 nextToken 결과만 반환합니다. 반환된 nextToken 값과 함께 다른 ListSimulationApplications 요청을 보내면 초기 요청의 나머지 결과를 확인할 수 있습니다. 이 값은 1~100이어야 합니다. 이 파라미터를 사용하지 않는 경우 ListSimulationApplications은 최대 100개의 결과와 해당하는 경우 nextToken 값을 반환합니다.

유형: 정수

필수 항목 여부: 아니요

## nextToken

이전 페이지 매김 요청에서 나머지 결과를 모두 반환하지 않은 경우, 응답 객체의 nextToken 파라미터 값이 토큰으로 설정됩니다. 다음 결과 세트를 검색하려면 ListSimulationApplications를 다시 호출하여 해당 토큰을 요청 객체의 nextToken 파라미터에 할당합니다. 남은 결과가 없는 경우 이전 응답 객체의 NextToken 파라미터는 null로 설정됩니다.

유형: 문자열

길이 제약: 최소 길이는 1. 최대 길이는 2,048.

패턴: [a-zA-Z0-9\_.\-\/+=]\*

Required: No

## versionQualifier

시뮬레이션 애플리케이션의 버전 한정자입니다.

유형: 문자열

길이 제약: 최소 길이는 1. 최대 길이는 255.

패턴: ALL

필수 여부: 아니요

## 응답 구문

```
HTTP/1.1 200
```

```

Content-type: application/json

{
  "nextToken": "string",
  "simulationApplicationSummaries": [
    {
      "arn": "string",
      "lastUpdatedAt": number,
      "name": "string",
      "robotSoftwareSuite": {
        "name": "string",
        "version": "string"
      },
      "simulationSoftwareSuite": {
        "name": "string",
        "version": "string"
      },
      "version": "string"
    }
  ]
}

```

## 응답 요소

작업이 성공하면 서비스가 HTTP 200 응답을 반환합니다.

다음 데이터는 서비스에 의해 JSON 형식으로 반환됩니다.

### [nextToken](#)

이전 페이지 매김 요청에서 나머지 결과를 모두 반환하지 않은 경우, 응답 객체의 nextToken 파라미터 값이 토큰으로 설정됩니다. 다음 결과 세트를 검색하려면 ListSimulationApplications를 다시 호출하여 해당 토큰을 요청 객체의 nextToken 파라미터에 할당합니다. 남은 결과가 없는 경우 이전 응답 객체의 NextToken 매개 변수는 null로 설정됩니다.

유형: 문자열

길이 제약: 최소 길이는 1. 최대 길이는 2,048.

패턴: [a-zA-Z0-9\_.\-\/+=]\*

## [simulationApplicationSummaries](#)

요청 기준을 충족하는 시뮬레이션 애플리케이션의 목록입니다.

유형: [SimulationApplicationSummary](#) 객체 어레이

어레이 멤버: 최소 항목 수 0개. 최대 항목 수는 100개입니다.

## Errors

모든 작업에서 발생하는 일반적인 오류에 대한 자세한 내용은 [일반적인 오류](#) 섹션을 참조하세요.

### InternalServerErrorException

RoboMaker AWS에서 서비스 문제가 발생했습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 500

### InvalidParameterException

요청에 지정된 파라미터는 유효하지 않거나, 지원되지 않거나, 사용할 수 없습니다. 반환된 메시지는 오류 값에 대한 설명을 제공합니다.

HTTP 상태 코드: 400

### ThrottlingException

RoboMaker AWS는 일시적으로 요청을 처리할 수 없습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 400

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS Go v2를 위한 SDK](#)
- [AWS Java V2용 SDK](#)
- [AWS V3용 SDK JavaScript](#)

- [AWS PHP V3용 SDK](#)
- [AWS Python용 SDK](#)
- [AWS 루비 V3용 SDK](#)

## ListSimulationJobBatches

시뮬레이션 작업 배치 나열을 리턴합니다. 선택적으로 필터를 제공하여 특정 시뮬레이션 배치 작업을 검색할 수 있습니다.

### Request Syntax

```
POST /listSimulationJobBatches HTTP/1.1
Content-type: application/json
```

```
{
  "filters": [
    {
      "name": "string",
      "values": [ "string" ]
    }
  ],
  "maxResults": number,
  "nextToken": "string"
}
```

### URI 요청 파라미터

요청은 URI 파라미터를 사용하지 않습니다.

### 요청 본문

요청은 JSON 형식으로 다음 데이터를 받습니다.

#### filters

결과를 제한하기 위한 선택적 필터입니다.

유형: [Filter](#) 객체 어레이

배열 멤버: 고정된 항목 수는 1개입니다.

필수 여부: 아니요

#### maxResults

이 파라미터를 사용하면 ListSimulationJobBatches은 maxResults 응답 요소와 함께 단일 페이지에서 nextToken 결과만 반환합니다. 반환된 ListSimulationJobBatches 값과 함께 다른 nextToken 요청을 보내면 초기 요청의 나머지 결과를 확인할 수 있습니다.

유형: 정수

필수 항목 여부: 아니요

### [nextToken](#)

이전 페이지 매김 요청에서 나머지 결과를 모두 반환하지 않은 경우, 응답 객체의 nextToken 파라미터 값이 토큰으로 설정됩니다. 다음 결과 세트를 검색하려면 ListSimulationJobBatches를 다시 호출하여 해당 토큰을 요청 객체의 nextToken 파라미터에 할당합니다. 남은 결과가 없는 경우 이전 응답 개체의 NextToken 파라미터는 null로 설정됩니다.

유형: 문자열

길이 제약: 최소 길이는 1. 최대 길이는 2,048.

패턴: [a-zA-Z0-9\_.\-\/+=]\*

필수 여부: 아니요

## 응답 구문

```
HTTP/1.1 200
Content-type: application/json

{
  "nextToken": "string",
  "simulationJobBatchSummaries": [
    {
      "arn": "string",
      "createdAt": number,
      "createdRequestCount": number,
      "failedRequestCount": number,
      "lastUpdatedAt": number,
      "pendingRequestCount": number,
      "status": "string"
    }
  ]
}
```

## 응답 요소

작업이 성공하면 서비스가 HTTP 200 응답을 반환합니다.

다음 데이터는 서비스에 의해 JSON 형식으로 반환됩니다.

### [nextToken](#)

이전 페이지 매김 요청에서 나머지 결과를 모두 반환하지 않은 경우, 응답 객체의 nextToken 파라미터 값이 토큰으로 설정됩니다. 다음 결과 세트를 검색하려면 ListSimulationJobBatches를 다시 호출하여 해당 토큰을 요청 객체의 nextToken 파라미터에 할당합니다. 남은 결과가 없는 경우 이전 응답 객체의 NextToken 매개 변수는 null로 설정됩니다.

유형: 문자열

길이 제약: 최소 길이는 1. 최대 길이는 2,048.

패턴: [a-zA-Z0-9\_.\-\/+=]\*

### [simulationJobBatchSummaries](#)

시뮬레이션 작업 배치 요약 목록입니다.

타입: [SimulationJobBatchSummary](#) 객체 배열

## Errors

모든 작업에서 발생하는 일반적인 오류에 대한 자세한 내용은 [일반적인 오류](#) 섹션을 참조하세요.

### InternalServerError

RoboMaker AWS에서 서비스 문제가 발생했습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 500

### InvalidParameterException

요청에 지정된 파라미터는 유효하지 않거나, 지원되지 않거나, 사용할 수 없습니다. 반환된 메시지는 오류 값에 대한 설명을 제공합니다.

HTTP 상태 코드: 400

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS Command Line Interface](#)

- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS Go v2를 위한 SDK](#)
- [AWS Java V2용 SDK](#)
- [AWS V3용 SDK JavaScript](#)
- [AWS PHP V3용 SDK](#)
- [AWS Python용 SDK](#)
- [AWS 루비 V3용 SDK](#)

## ListSimulationJobs

시뮬레이션 작업 목록을 반환합니다. 선택적으로 필터를 제공하여 특정 시뮬레이션 작업을 검색할 수 있습니다.

### Request Syntax

```
POST /listSimulationJobs HTTP/1.1
Content-type: application/json

{
  "filters": [
    {
      "name": "string",
      "values": [ "string" ]
    }
  ],
  "maxResults": number,
  "nextToken": "string"
}
```

### URI 요청 파라미터

요청은 URI 파라미터를 사용하지 않습니다.

### 요청 본문

요청은 JSON 형식으로 다음 데이터를 받습니다.

#### [filters](#)

결과를 제한하기 위한 선택적 필터입니다.

필터 이름 `status`, `simulationApplicationName`, `robotApplicationName`가 지원됩니다. 필터링할 때는 필터링된 항목의 전체 값을 사용해야 합니다. 필터는 최대 3개까지 사용할 수 있지만, 이름이 동일한 항목에 대한 필어야 합니다. 예를 들어, 상태 `Preparing` 또는 상태 `Running`가 있는 항목을 검색하는 경우가 있습니다.

유형: [Filter](#) 객체 어레이

배열 멤버: 고정된 항목 수는 1개입니다.

필수 여부: 아니요

### maxResults

이 파라미터를 사용하면 ListSimulationJobs은 maxResults 응답 요소와 함께 단일 페이지에서 nextToken 결과만 반환합니다. 반환된 ListSimulationJobs 값과 함께 다른 nextToken 요청을 보내면 초기 요청의 나머지 결과를 확인할 수 있습니다. 이 값은 1~1000이어야 합니다. 이 파라미터를 사용하지 않는 경우 ListSimulationJobs는 최대 1000개의 결과와 해당하는 경우 nextToken 값을 반환합니다.

유형: 정수

필수 항목 여부: 아니요

### nextToken

이전 페이지 매김 요청에서 나머지 결과를 모두 반환하지 않은 경우, 응답 객체의 nextToken 파라미터 값이 토큰으로 설정됩니다. 다음 결과 세트를 검색하려면 ListSimulationJobs를 다시 호출하여 해당 토큰을 요청 객체의 nextToken 파라미터에 할당합니다. 남은 결과가 없는 경우 이전 응답 개체의 NextToken 파라미터는 null로 설정됩니다.

유형: 문자열

길이 제약: 최소 길이는 1. 최대 길이는 2,048.

패턴: [a-zA-Z0-9\_.\-\/+=]\*

필수 여부: 아니요

## 응답 구문

```
HTTP/1.1 200
Content-type: application/json

{
  "nextToken": "string",
  "simulationJobSummaries": [
    {
      "arn": "string",
      "computeType": "string",
      "dataSourceNames": [ "string" ],
      "lastUpdatedAt": number,
      "name": "string",
```

```

    "robotApplicationNames": [ "string" ],
    "simulationApplicationNames": [ "string" ],
    "status": "string"
  }
]
}

```

## 응답 요소

작업이 성공하면 서비스가 HTTP 200 응답을 반환합니다.

다음 데이터는 서비스에 의해 JSON 형식으로 반환됩니다.

### [nextToken](#)

이전 페이지 매김 요청에서 나머지 결과를 모두 반환하지 않은 경우, 응답 객체의 nextToken 파라미터 값이 토큰으로 설정됩니다. 다음 결과 세트를 검색하려면 ListSimulationJobs를 다시 호출하여 해당 토큰을 요청 객체의 nextToken 파라미터에 할당합니다. 남은 결과가 없는 경우 이전 응답 객체의 NextToken 매개 변수는 null로 설정됩니다.

유형: 문자열

길이 제약: 최소 길이는 1. 최대 길이는 2,048.

패턴: [a-zA-Z0-9\_.\-\/+=]\*

### [simulationJobSummaries](#)

요청 기준을 충족하는 시뮬레이션 작업 요약 목록입니다.

유형: [SimulationJobSummary](#) 객체 어레이

어레이 멤버: 최소 항목 수 0개. 최대 항목 수는 100개입니다.

## Errors

모든 작업에서 발생하는 일반적인 오류에 대한 자세한 내용은 [일반적인 오류](#) 섹션을 참조하세요.

### InternalServerError

RoboMaker AWS에서 서비스 문제가 발생했습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 500

## InvalidParameterException

요청에 지정된 파라미터는 유효하지 않거나, 지원되지 않거나, 사용할 수 없습니다. 반환된 메시지는 오류 값에 대한 설명을 제공합니다.

HTTP 상태 코드: 400

## ThrottlingException

RoboMaker AWS는 일시적으로 요청을 처리할 수 없습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 400

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS Go v2를 위한 SDK](#)
- [AWS Java V2용 SDK](#)
- [AWS V3용 SDK JavaScript](#)
- [AWS PHP V3용 SDK](#)
- [AWS Python용 SDK](#)
- [AWS 루비 V3용 SDK](#)

## ListTagsForResource

AWS RoboMaker 리소스의 모든 태그를 나열합니다.

### Request Syntax

```
GET /tags/resourceArn HTTP/1.1
```

### URI 요청 파라미터

요청은 다음 URI 파라미터를 사용합니다.

#### resourceArn

나열할 태그가 포함된 AWS RoboMaker Amazon 리소스 이름 (ARN)

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

필수 사항 여부: Yes

### Request Body

해당 요청에는 본문이 없습니다.

### Response Syntax

```
HTTP/1.1 200
Content-type: application/json

{
  "tags": {
    "string" : "string"
  }
}
```

### 응답 요소

작업이 성공하면 서비스가 HTTP 200 응답을 반환합니다.

다음 데이터는 서비스에 의해 JSON 형식으로 반환됩니다.

## tags

지정된 리소스에 추가된 모든 태그의 목록입니다.

유형: 문자열-문자열 맵

맵 항목: 최소 항목 수는 0개입니다. 최대 항목 수는 50개입니다.

키 길이 제약 조건: 최소 길이는 1. 최대 길이 128.

키 패턴: [a-zA-Z0-9 \_.\-\/+=:]\*

값 길이 제약 조건: 최소 길이는 0입니다. 최대 길이는 256.

가치 패턴: [a-zA-Z0-9 \_.\-\/+=:]\*

## Errors

모든 작업에서 발생하는 일반적인 오류에 대한 자세한 내용은 [일반적인 오류](#) 섹션을 참조하세요.

### InternalServerError

RoboMaker AWS에서 서비스 문제가 발생했습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 500

### InvalidParameterException

요청에 지정된 파라미터는 유효하지 않거나, 지원되지 않거나, 사용할 수 없습니다. 반환된 메시지는 오류 값에 대한 설명을 제공합니다.

HTTP 상태 코드: 400

### ResourceNotFoundException

지정한 리소스가 존재하지 않습니다.

HTTP 상태 코드: 400

### ThrottlingException

RoboMaker AWS는 일시적으로 요청을 처리할 수 없습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 400

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS Go v2를 위한 SDK](#)
- [AWS Java V2용 SDK](#)
- [AWS V3용 SDK JavaScript](#)
- [AWS PHP V3용 SDK](#)
- [AWS Python용 SDK](#)
- [AWS 루비 V3용 SDK](#)

## ListWorldExportJobs

월드 내보내기 작업 목록입니다.

### Request Syntax

```
POST /listWorldExportJobs HTTP/1.1
Content-type: application/json
```

```
{
  "filters": [
    {
      "name": "string",
      "values": [ "string" ]
    }
  ],
  "maxResults": number,
  "nextToken": "string"
}
```

### URI 요청 파라미터

요청은 URI 파라미터를 사용하지 않습니다.

### 요청 본문

요청은 JSON 형식으로 다음 데이터를 받습니다.

#### filters

결과를 제한하기 위한 선택적 필터입니다. generationJobId 및 templateId를 사용할 수 있습니다.

유형: [Filter](#) 객체 어레이

배열 멤버: 고정된 항목 수는 1개입니다.

필수 여부: 아니요

#### maxResults

이 파라미터를 사용하면 ListWorldExportJobs은 maxResults 응답 요소와 함께 단일 페이지에서 nextToken 결과만 반환합니다. 반환된 nextToken 값과 함께 다른

ListWorldExportJobs 요청을 보내면 초기 요청의 나머지 결과를 확인할 수 있습니다. 이 값은 1~100이어야 합니다. 이 파라미터를 사용하지 않는 경우 ListWorldExportJobs은 최대 100개의 결과와 해당하는 경우 nextToken 값을 반환합니다.

유형: 정수

필수 항목 여부: 아니요

### nextToken

이전 페이지 매김 요청에서 나머지 결과를 모두 반환하지 않은 경우, 응답 객체의 nextToken 파라미터 값이 토큰으로 설정됩니다. 다음 결과 세트를 검색하려면 ListWorldExportJobs를 다시 호출하여 해당 토큰을 요청 객체의 nextToken 파라미터에 할당합니다. 남은 결과가 없는 경우 이전 응답 개체의 NextToken 파라미터는 null로 설정됩니다.

유형: 문자열

길이 제약: 최소 길이는 1. 최대 길이는 2,048.

패턴: [a-zA-Z0-9\_.\-\/+=]\*

필수 여부: 아니요

## 응답 구문

```
HTTP/1.1 200
Content-type: application/json

{
  "nextToken": "string",
  "worldExportJobSummaries": [
    {
      "arn": "string",
      "createdAt": number,
      "outputLocation": {
        "s3Bucket": "string",
        "s3Prefix": "string"
      },
      "status": "string",
      "worlds": [ "string" ]
    }
  ]
}
```

}

## 응답 요소

작업이 성공하면 서비스가 HTTP 200 응답을 반환합니다.

다음 데이터는 서비스에 의해 JSON 형식으로 반환됩니다.

### [nextToken](#)

이전 페이지 매김 요청에서 나머지 결과를 모두 반환하지 않은 경우, 응답 객체의 nextToken 파라미터 값이 토큰으로 설정됩니다. 다음 결과 세트를 검색하려면 ListWorldExportJobsRequest를 다시 호출하여 해당 토큰을 요청 객체의 nextToken 파라미터에 할당합니다. 남은 결과가 없는 경우 이전 응답 개체의 NextToken 매개 변수는 null로 설정됩니다.

유형: 문자열

길이 제약: 최소 길이는 1. 최대 길이는 2,048.

패턴: [a-zA-Z0-9\_.\-\/+=]\*

### [worldExportJobSummaries](#)

월드 내보내기 작업에 대한 요약 정보입니다.

유형: [WorldExportJobSummary](#) 객체 어레이

어레이 멤버: 최소 항목 수 0개. 최대 항목 수는 100개입니다.

## Errors

모든 작업에서 발생하는 일반적인 오류에 대한 자세한 내용은 [일반적인 오류](#) 섹션을 참조하세요.

### InternalServerErrorException

RoboMaker AWS에서 서비스 문제가 발생했습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 500

### InvalidParameterException

요청에 지정된 파라미터는 유효하지 않거나, 지원되지 않거나, 사용할 수 없습니다. 반환된 메시지는 오류 값에 대한 설명을 제공합니다.

HTTP 상태 코드: 400

ThrottlingException

RoboMaker AWS는 일시적으로 요청을 처리할 수 없습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 400

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS Go v2를 위한 SDK](#)
- [AWS Java V2용 SDK](#)
- [AWS V3용 SDK JavaScript](#)
- [AWS PHP V3용 SDK](#)
- [AWS Python용 SDK](#)
- [AWS 루비 V3용 SDK](#)

## ListWorldGenerationJobs

월드 생성 작업을 나열합니다.

### Request Syntax

```
POST /listWorldGenerationJobs HTTP/1.1
Content-type: application/json

{
  "filters": [
    {
      "name": "string",
      "values": [ "string" ]
    }
  ],
  "maxResults": number,
  "nextToken": "string"
}
```

### URI 요청 파라미터

요청은 URI 파라미터를 사용하지 않습니다.

### 요청 본문

요청은 JSON 형식으로 다음 데이터를 받습니다.

#### [filters](#)

결과를 제한하기 위한 선택적 필터입니다. status 및 templateId를 사용할 수 있습니다.

유형: [Filter](#) 객체 어레이

배열 멤버: 고정된 항목 수는 1개입니다.

필수 여부: 아니요

#### [maxResults](#)

이 파라미터를 사용하면 ListWorldGeneratorJobs은 maxResults 응답 요소와 함께 단일 페이지에서 nextToken 결과만 반환합니다. 반환된 nextToken 값과 함께 다른

ListWorldGeneratorJobs 요청을 보내면 초기 요청의 나머지 결과를 확인할 수 있습니다. 이 값은 1~100이어야 합니다. 이 파라미터를 사용하지 않는 경우 ListWorldGeneratorJobs은 최대 100개의 결과와 해당하는 경우 nextToken 값을 반환합니다.

유형: 정수

필수 항목 여부: 아니요

### nextToken

이전 페이지 매김 요청에서 나머지 결과를 모두 반환하지 않은 경우, 응답 객체의 nextToken 파라미터 값이 토큰으로 설정됩니다. 다음 결과 세트를 검색하려면 ListWorldGenerationJobsRequest를 다시 호출하여 해당 토큰을 요청 객체의 nextToken 파라미터에 할당합니다. 남은 결과가 없는 경우 이전 응답 개체의 NextToken 파라미터는 null로 설정됩니다.

유형: 문자열

길이 제약: 최소 길이는 1. 최대 길이는 2,048.

패턴: [a-zA-Z0-9\_.\-\/+=]\*

필수 여부: 아니요

## 응답 구문

```
HTTP/1.1 200
Content-type: application/json

{
  "nextToken": "string",
  "worldGenerationJobSummaries": [
    {
      "arn": "string",
      "createdAt": number,
      "failedWorldCount": number,
      "status": "string",
      "succeededWorldCount": number,
      "template": "string",
      "worldCount": {
        "floorplanCount": number,
        "interiorCountPerFloorplan": number
      }
    }
  ]
}
```

```

    }
  }
]
}

```

## 응답 요소

작업이 성공하면 서비스가 HTTP 200 응답을 반환합니다.

다음 데이터는 서비스에 의해 JSON 형식으로 반환됩니다.

### [nextToken](#)

이전 페이지 매김 요청에서 나머지 결과를 모두 반환하지 않은 경우, 응답 객체의 nextToken 파라미터 값이 토큰으로 설정됩니다. 다음 결과 세트를 검색하려면 ListWorldGeneratorJobsRequest를 다시 호출하여 해당 토큰을 요청 객체의 nextToken 파라미터에 할당합니다. 남은 결과가 없는 경우 이전 응답 개체의 NextToken 매개 변수는 null로 설정됩니다.

유형: 문자열

길이 제약: 최소 길이는 1. 최대 길이는 2,048.

패턴: [a-zA-Z0-9\_.\-\/+=]\*

### [worldGenerationJobSummaries](#)

월드 생성 작업에 대한 요약 정보입니다.

유형: [WorldGenerationJobSummary](#) 객체 어레이

어레이 멤버: 최소 항목 수 0개. 최대 항목 수는 100개입니다.

## Errors

모든 작업에서 발생하는 일반적인 오류에 대한 자세한 내용은 [일반적인 오류](#) 섹션을 참조하세요.

### InternalServerErrorException

RoboMaker AWS에서 서비스 문제가 발생했습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 500

## InvalidParameterException

요청에 지정된 파라미터는 유효하지 않거나, 지원되지 않거나, 사용할 수 없습니다. 반환된 메시지는 오류 값에 대한 설명을 제공합니다.

HTTP 상태 코드: 400

## ThrottlingException

RoboMaker AWS는 일시적으로 요청을 처리할 수 없습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 400

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS Go v2를 위한 SDK](#)
- [AWS Java V2용 SDK](#)
- [AWS V3용 SDK JavaScript](#)
- [AWS PHP V3용 SDK](#)
- [AWS Python용 SDK](#)
- [AWS 루비 V3용 SDK](#)

## ListWorlds

월드름 나열합니다.

### Request Syntax

```
POST /listWorlds HTTP/1.1
Content-type: application/json

{
  "filters": [
    {
      "name": "string",
      "values": [ "string" ]
    }
  ],
  "maxResults": number,
  "nextToken": "string"
}
```

### URI 요청 파라미터

요청은 URI 파라미터를 사용하지 않습니다.

### 요청 본문

요청은 JSON 형식으로 다음 데이터를 받습니다.

#### [filters](#)

결과를 제한하기 위한 선택적 필터입니다. `status`를 사용할 수 있습니다.

유형: [Filter](#) 객체 어레이

배열 멤버: 고정된 항목 수는 1개입니다.

필수 여부: 아니요

#### [maxResults](#)

이 파라미터를 사용하면 ListWorlds은 maxResults 응답 요소와 함께 단일 페이지에서 nextToken 결과만 반환합니다. 반환된 nextToken 값과 함께 다른 ListWorlds 요청을 보내면

초기 요청의 나머지 결과를 확인할 수 있습니다. 이 값은 1~100이어야 합니다. 이 파라미터를 사용하지 않는 경우 ListWorlds은 최대 100개의 결과와 해당하는 경우 nextToken 값을 반환합니다.

유형: 정수

필수 항목 여부: 아니요

### nextToken

이전 페이지 매김 요청에서 나머지 결과를 모두 반환하지 않은 경우, 응답 객체의 nextToken 파라미터 값이 토큰으로 설정됩니다. 다음 결과 세트를 검색하려면 ListWorlds를 다시 호출하여 해당 토큰을 요청 객체의 nextToken 파라미터에 할당합니다. 남은 결과가 없는 경우 이전 응답 객체의 NextToken 파라미터는 null로 설정됩니다.

유형: 문자열

길이 제약: 최소 길이는 1. 최대 길이는 2,048.

패턴: [a-zA-Z0-9\_.\-\/+=]\*

필수 여부: 아니요

## 응답 구문

```
HTTP/1.1 200
Content-type: application/json

{
  "nextToken": "string",
  "worldSummaries": [
    {
      "arn": "string",
      "createdAt": number,
      "generationJob": "string",
      "template": "string"
    }
  ]
}
```

## 응답 요소

작업이 성공하면 서비스가 HTTP 200 응답을 반환합니다.

다음 데이터는 서비스에 의해 JSON 형식으로 반환됩니다.

### nextToken

이전 페이지 매김 요청에서 나머지 결과를 모두 반환하지 않은 경우, 응답 객체의 nextToken 파라미터 값이 토큰으로 설정됩니다. 다음 결과 세트를 검색하려면 ListWorlds를 다시 호출하여 해당 토큰을 요청 객체의 nextToken 파라미터에 할당합니다. 남은 결과가 없는 경우 이전 응답 객체의 NextToken 매개 변수는 null로 설정됩니다.

유형: 문자열

길이 제약: 최소 길이는 1. 최대 길이는 2,048.

패턴: [a-zA-Z0-9\_.\-\/+=]\*

### worldSummaries

월드에 대한 요약 정보입니다.

타입: [WorldSummary](#) 객체 배열

## Errors

모든 작업에서 발생하는 일반적인 오류에 대한 자세한 내용은 [일반적인 오류](#) 섹션을 참조하세요.

### InternalServerError

RoboMaker AWS에서 서비스 문제가 발생했습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 500

### InvalidParameterException

요청에 지정된 파라미터는 유효하지 않거나, 지원되지 않거나, 사용할 수 없습니다. 반환된 메시지는 오류 값에 대한 설명을 제공합니다.

HTTP 상태 코드: 400

### ThrottlingException

RoboMaker AWS는 일시적으로 요청을 처리할 수 없습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 400

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS Go v2를 위한 SDK](#)
- [AWS Java V2용 SDK](#)
- [AWS V3용 SDK JavaScript](#)
- [AWS PHP V3용 SDK](#)
- [AWS Python용 SDK](#)
- [AWS 루비 V3용 SDK](#)

## ListWorldTemplates

월드 템플릿을 나열합니다.

### Request Syntax

```
POST /listWorldTemplates HTTP/1.1
Content-type: application/json

{
  "maxResults": number,
  "nextToken": "string"
}
```

### URI 요청 파라미터

요청은 URI 파라미터를 사용하지 않습니다.

### 요청 본문

요청은 JSON 형식으로 다음 데이터를 받습니다.

#### [maxResults](#)

이 파라미터를 사용하면 ListWorldTemplates은 maxResults 응답 요소와 함께 단일 페이지에서 nextToken 결과만 반환합니다. 반환된 nextToken 값과 함께 다른 ListWorldTemplates 요청을 보내면 초기 요청의 나머지 결과를 확인할 수 있습니다. 이 값은 1~100이어야 합니다. 이 파라미터를 사용하지 않는 경우 ListWorldTemplates은 최대 100개의 결과와 해당하는 경우 nextToken 값을 반환합니다.

유형: 정수

필수 항목 여부: 아니요

#### [nextToken](#)

이전 페이지 매김 요청에서 나머지 결과를 모두 반환하지 않은 경우, 응답 객체의 nextToken 파라미터 값이 토큰으로 설정됩니다. 다음 결과 세트를 검색하려면 ListWorldTemplates를 다시 호출하여 해당 토큰을 요청 객체의 nextToken 파라미터에 할당합니다. 남은 결과가 없는 경우 이전 응답 개체의 NextToken 파라미터는 null로 설정됩니다.

유형: 문자열

길이 제약: 최소 길이는 1. 최대 길이는 2,048.

패턴: [a-zA-Z0-9\_.\-\/+=]\*

필수 여부: 아니요

## 응답 구문

```
HTTP/1.1 200
Content-type: application/json

{
  "nextToken": "string",
  "templateSummaries": [
    {
      "arn": "string",
      "createdAt": number,
      "lastUpdatedAt": number,
      "name": "string",
      "version": "string"
    }
  ]
}
```

## 응답 요소

작업이 성공하면 서비스가 HTTP 200 응답을 반환합니다.

다음 데이터는 서비스에 의해 JSON 형식으로 반환됩니다.

### nextToken

이전 페이지 매김 요청에서 나머지 결과를 모두 반환하지 않은 경우, 응답 객체의 nextToken 파라미터 값이 토큰으로 설정됩니다. 다음 결과 세트를 검색하려면 ListWorldTemplates를 다시 호출하여 해당 토큰을 요청 객체의 nextToken 파라미터에 할당합니다. 남은 결과가 없는 경우 이전 응답 객체의 NextToken 매개 변수는 null로 설정됩니다.

유형: 문자열

길이 제약: 최소 길이는 1. 최대 길이는 2,048.

패턴: [a-zA-Z0-9\_.\-\/+=]\*

## [templateSummaries](#)

템플릿에 대한 요약 정보입니다.

타입: [TemplateSummary](#) 객체 배열

## Errors

모든 작업에서 발생하는 일반적인 오류에 대한 자세한 내용은 [일반적인 오류](#) 섹션을 참조하세요.

### InternalServerErrorException

RoboMaker AWS에서 서비스 문제가 발생했습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 500

### InvalidParameterException

요청에 지정된 파라미터는 유효하지 않거나, 지원되지 않거나, 사용할 수 없습니다. 반환된 메시지는 오류 값에 대한 설명을 제공합니다.

HTTP 상태 코드: 400

### ThrottlingException

RoboMaker AWS는 일시적으로 요청을 처리할 수 없습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 400

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS Go v2를 위한 SDK](#)
- [AWS Java V2용 SDK](#)
- [AWS V3용 SDK JavaScript](#)
- [AWS PHP V3용 SDK](#)

- [AWS Python용 SDK](#)
- [AWS 루비 V3용 SDK](#)

## RegisterRobot

이 작업은 사용 중단되었습니다.

플릿으로 로봇을 등록합니다.

### Important

이 API는 더 이상 지원되지 않으며 사용할 경우 오류가 발생합니다. 자세한 내용은 [지원 정책](#) 페이지의 2022년 1월 31일 업데이트를 참조하세요.

## Request Syntax

```
POST /registerRobot HTTP/1.1
Content-type: application/json

{
  "fleet": "string",
  "robot": "string"
}
```

## URI 요청 파라미터

요청은 URI 파라미터를 사용하지 않습니다.

## 요청 본문

요청은 JSON 형식으로 다음 데이터를 받습니다.

### fleet

플릿의 Amazon 리소스 이름(ARN)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

필수 사항 여부: Yes

## robot

로봇의 Amazon 리소스 이름(ARN)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

필수 항목 여부: 예

## 응답 구문

```

HTTP/1.1 200
Content-type: application/json

{
  "fleet": "string",
  "robot": "string"
}

```

## 응답 요소

작업이 성공하면 서비스가 HTTP 200 응답을 반환합니다.

다음 데이터는 서비스에 의해 JSON 형식으로 반환됩니다.

## fleet

로봇이 등록할 플릿의 Amazon 리소스 이름(ARN)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

## robot

로봇 등록에 대한 정보입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

## Errors

모든 작업에서 발생하는 일반적인 오류에 대한 자세한 내용은 [일반적인 오류](#) 섹션을 참조하세요.

### InternalServerErrorException

RoboMaker AWS에서 서비스 문제가 발생했습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 500

### InvalidParameterException

요청에 지정된 파라미터는 유효하지 않거나, 지원되지 않거나, 사용할 수 없습니다. 반환된 메시지는 오류 값에 대한 설명을 제공합니다.

HTTP 상태 코드: 400

### LimitExceededException

요청된 리소스가 허용된 최대 수를 초과하거나 동시 스트림 요청 수가 허용된 최대 수를 초과합니다.

HTTP 상태 코드: 400

### ResourceNotFoundException

지정한 리소스가 존재하지 않습니다.

HTTP 상태 코드: 400

### ThrottlingException

RoboMaker AWS는 일시적으로 요청을 처리할 수 없습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 400

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS Go v2를 위한 SDK](#)
- [AWS Java V2용 SDK](#)
- [AWS V3용 SDK JavaScript](#)
- [AWS PHP V3용 SDK](#)
- [AWS Python용 SDK](#)
- [AWS 루비 V3용 SDK](#)

## RestartSimulationJob

실행 중인 시뮬레이션 작업을 다시 시작합니다.

### Request Syntax

```
POST /restartSimulationJob HTTP/1.1
Content-type: application/json

{
  "job": "string"
}
```

### URI 요청 파라미터

요청은 URI 파라미터를 사용하지 않습니다.

### 요청 본문

요청은 JSON 형식으로 다음 데이터를 받습니다.

#### job

시뮬레이션 작업의 Amazon 리소스 이름(ARN)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

필수 항목 여부: 예

### 응답 구문

```
HTTP/1.1 200
```

### Response Elements

작업이 성공하면 서비스가 비어 있는 HTTP 본문과 함께 HTTP 200 응답을 반환합니다.

## Errors

모든 작업에서 발생하는 일반적인 오류에 대한 자세한 내용은 [일반적인 오류](#) 섹션을 참조하세요.

### InternalServerErrorException

RoboMaker AWS에서 서비스 문제가 발생했습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 500

### InvalidParameterException

요청에 지정된 파라미터는 유효하지 않거나, 지원되지 않거나, 사용할 수 없습니다. 반환된 메시지는 오류 값에 대한 설명을 제공합니다.

HTTP 상태 코드: 400

### LimitExceededException

요청된 리소스가 허용된 최대 수를 초과하거나 동시 스트림 요청 수가 허용된 최대 수를 초과합니다.

HTTP 상태 코드: 400

### ResourceNotFoundException

지정한 리소스가 존재하지 않습니다.

HTTP 상태 코드: 400

### ThrottlingException

RoboMaker AWS는 일시적으로 요청을 처리할 수 없습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 400

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS Go v2를 위한 SDK](#)

- [AWS Java V2용 SDK](#)
- [AWS V3용 SDK JavaScript](#)
- [AWS PHP V3용 SDK](#)
- [AWS Python용 SDK](#)
- [AWS 루비 V3용 SDK](#)

## StartSimulationJobBatch

새 시뮬레이션 작업 배치를 시작합니다. 배치는 하나 이상의 SimulationJobRequest 객체를 사용하여 정의됩니다.

### Request Syntax

```
POST /startSimulationJobBatch HTTP/1.1
Content-type: application/json

{
  "batchPolicy": {
    "maxConcurrency": number,
    "timeoutInSeconds": number
  },
  "clientRequestToken": "string",
  "createSimulationJobRequests": [
    {
      "compute": {
        "computeType": "string",
        "gpuUnitLimit": number,
        "simulationUnitLimit": number
      },
      "dataSources": [
        {
          "destination": "string",
          "name": "string",
          "s3Bucket": "string",
          "s3Keys": [ "string " ],
          "type": "string"
        }
      ],
      "failureBehavior": "string",
      "iamRole": "string",
      "loggingConfig": {
        "recordAllRosTopics": boolean
      },
      "maxJobDurationInSeconds": number,
      "outputLocation": {
        "s3Bucket": "string",
        "s3Prefix": "string"
      },
      "robotApplications": [
```

```
{
  "application": "string",
  "applicationVersion": "string",
  "launchConfig": {
    "command": [ "string" ],
    "environmentVariables": {
      "string": "string"
    },
    "launchFile": "string",
    "packageName": "string",
    "portForwardingConfig": {
      "portMappings": [
        {
          "applicationPort": number,
          "enableOnPublicIp": boolean,
          "jobPort": number
        }
      ]
    },
    "streamUI": boolean
  },
  "tools": [
    {
      "command": "string",
      "exitBehavior": "string",
      "name": "string",
      "streamOutputToCloudWatch": boolean,
      "streamUI": boolean
    }
  ],
  "uploadConfigurations": [
    {
      "name": "string",
      "path": "string",
      "uploadBehavior": "string"
    }
  ],
  "useDefaultTools": boolean,
  "useDefaultUploadConfigurations": boolean
},
"simulationApplications": [
  {
    "application": "string",
```

```

    "applicationVersion": "string",
    "launchConfig": {
      "command": [ "string" ],
      "environmentVariables": {
        "string" : "string"
      },
      "launchFile": "string",
      "packageName": "string",
      "portForwardingConfig": {
        "portMappings": [
          {
            "applicationPort": number,
            "enableOnPublicIp": boolean,
            "jobPort": number
          }
        ]
      },
      "streamUI": boolean
    },
    "tools": [
      {
        "command": "string",
        "exitBehavior": "string",
        "name": "string",
        "streamOutputToCloudWatch": boolean,
        "streamUI": boolean
      }
    ],
    "uploadConfigurations": [
      {
        "name": "string",
        "path": "string",
        "uploadBehavior": "string"
      }
    ],
    "useDefaultTools": boolean,
    "useDefaultUploadConfigurations": boolean,
    "worldConfigs": [
      {
        "world": "string"
      }
    ]
  }
],
}

```

```

    "tags": {
      "string" : "string"
    },
    "useDefaultApplications": boolean,
    "vpcConfig": {
      "assignPublicIp": boolean,
      "securityGroups": [ "string" ],
      "subnets": [ "string" ]
    }
  }
],
"tags": {
  "string" : "string"
}
}

```

## URI 요청 파라미터

요청은 URI 파라미터를 사용하지 않습니다.

## 요청 본문

요청은 JSON 형식으로 다음 데이터를 받습니다.

### batchPolicy

배치 정책입니다.

유형: [BatchPolicy](#) 객체

필수 항목 여부: 아니요

### clientRequestToken

요청 멱등성을 보장하기 위해 제공하는 고유한 대소문자 구분 식별자입니다.

유형: 문자열

길이 제한: 최소 길이는 1. 최대 길이는 64.

패턴: `[a-zA-Z0-9_\-=]*`

Required: No

## createSimulationJobRequests

배치에서 생성해야 하는 시뮬레이션 작업 요청 목록입니다.

유형: [SimulationJobRequest](#) 객체 어레이

배열 멤버: 최소 항목 수는 1입니다. 최대 항목 수는 1000입니다.

필수 여부: 예

### tags

배포 작업 배치에 연결된 태그 키 및 태그 값을 포함하는 맵입니다.

유형: 문자열-문자열 맵

맵 항목: 최소 항목 수는 0개입니다. 최대 항목 수는 50개입니다.

키 길이 제약 조건: 최소 길이는 1. 최대 길이 128.

키 패턴: [a-zA-Z0-9 \_.\-\/+=:]\*

값 길이 제약 조건: 최소 길이는 0입니다. 최대 길이는 256.

값 패턴: [a-zA-Z0-9 \_.\-\/+=:]\*

필수 여부: 아니요

## 응답 구문

```
HTTP/1.1 200
Content-type: application/json

{
  "arn": "string",
  "batchPolicy": {
    "maxConcurrency": number,
    "timeoutInSeconds": number
  },
  "clientRequestToken": "string",
  "createdAt": number,
  "createdRequests": [
    {
      "arn": "string",
      "computeType": "string",
```

```

    "dataSourceNames": [ "string" ],
    "lastUpdatedAt": number,
    "name": "string",
    "robotApplicationNames": [ "string" ],
    "simulationApplicationNames": [ "string" ],
    "status": "string"
  }
],
"failedRequests": [
  {
    "failedAt": number,
    "failureCode": "string",
    "failureReason": "string",
    "request": {
      "compute": {
        "computeType": "string",
        "gpuUnitLimit": number,
        "simulationUnitLimit": number
      },
      "dataSources": [
        {
          "destination": "string",
          "name": "string",
          "s3Bucket": "string",
          "s3Keys": [ "string" ],
          "type": "string"
        }
      ],
      "failureBehavior": "string",
      "iamRole": "string",
      "loggingConfig": {
        "recordAllRosTopics": boolean
      },
      "maxJobDurationInSeconds": number,
      "outputLocation": {
        "s3Bucket": "string",
        "s3Prefix": "string"
      },
      "robotApplications": [
        {
          "application": "string",
          "applicationVersion": "string",
          "launchConfig": {
            "command": [ "string" ],

```

```

    "environmentVariables": {
      "string" : "string"
    },
    "launchFile": "string",
    "packageName": "string",
    "portForwardingConfig": {
      "portMappings": [
        {
          "applicationPort": number,
          "enableOnPublicIp": boolean,
          "jobPort": number
        }
      ]
    },
    "streamUI": boolean
  },
  "tools": [
    {
      "command": "string",
      "exitBehavior": "string",
      "name": "string",
      "streamOutputToCloudWatch": boolean,
      "streamUI": boolean
    }
  ],
  "uploadConfigurations": [
    {
      "name": "string",
      "path": "string",
      "uploadBehavior": "string"
    }
  ],
  "useDefaultTools": boolean,
  "useDefaultUploadConfigurations": boolean
}
],
"simulationApplications": [
  {
    "application": "string",
    "applicationVersion": "string",
    "launchConfig": {
      "command": [ "string" ],
      "environmentVariables": {
        "string" : "string"
      }
    }
  }
]

```

```

    },
    "launchFile": "string",
    "packageName": "string",
    "portForwardingConfig": {
      "portMappings": [
        {
          "applicationPort": number,
          "enableOnPublicIp": boolean,
          "jobPort": number
        }
      ]
    },
    "streamUI": boolean
  },
  "tools": [
    {
      "command": "string",
      "exitBehavior": "string",
      "name": "string",
      "streamOutputToCloudWatch": boolean,
      "streamUI": boolean
    }
  ],
  "uploadConfigurations": [
    {
      "name": "string",
      "path": "string",
      "uploadBehavior": "string"
    }
  ],
  "useDefaultTools": boolean,
  "useDefaultUploadConfigurations": boolean,
  "worldConfigs": [
    {
      "world": "string"
    }
  ]
}
],
"tags": {
  "string" : "string"
},
"useDefaultApplications": boolean,
"vpcConfig": {

```

```

        "assignPublicIp": boolean,
        "securityGroups": [ "string" ],
        "subnets": [ "string" ]
    }
}
],
"failureCode": "string",
"failureReason": "string",
"pendingRequests": [
    {
        "compute": {
            "computeType": "string",
            "gpuUnitLimit": number,
            "simulationUnitLimit": number
        },
        "dataSources": [
            {
                "destination": "string",
                "name": "string",
                "s3Bucket": "string",
                "s3Keys": [ "string" ],
                "type": "string"
            }
        ],
        "failureBehavior": "string",
        "iamRole": "string",
        "loggingConfig": {
            "recordAllRosTopics": boolean
        },
        "maxJobDurationInSeconds": number,
        "outputLocation": {
            "s3Bucket": "string",
            "s3Prefix": "string"
        },
        "robotApplications": [
            {
                "application": "string",
                "applicationVersion": "string",
                "launchConfig": {
                    "command": [ "string" ],
                    "environmentVariables": {
                        "string" : "string"
                    }
                }
            }
        ],

```

```

    "launchFile": "string",
    "packageName": "string",
    "portForwardingConfig": {
      "portMappings": [
        {
          "applicationPort": number,
          "enableOnPublicIp": boolean,
          "jobPort": number
        }
      ]
    },
    "streamUI": boolean
  },
  "tools": [
    {
      "command": "string",
      "exitBehavior": "string",
      "name": "string",
      "streamOutputToCloudWatch": boolean,
      "streamUI": boolean
    }
  ],
  "uploadConfigurations": [
    {
      "name": "string",
      "path": "string",
      "uploadBehavior": "string"
    }
  ],
  "useDefaultTools": boolean,
  "useDefaultUploadConfigurations": boolean
}
],
"simulationApplications": [
  {
    "application": "string",
    "applicationVersion": "string",
    "launchConfig": {
      "command": [ "string ],
      "environmentVariables": {
        "string": "string"
      },
    },
    "launchFile": "string",
    "packageName": "string",

```

```

    "portForwardingConfig": {
      "portMappings": [
        {
          "applicationPort": number,
          "enableOnPublicIp": boolean,
          "jobPort": number
        }
      ]
    },
    "streamUI": boolean
  },
  "tools": [
    {
      "command": "string",
      "exitBehavior": "string",
      "name": "string",
      "streamOutputToCloudWatch": boolean,
      "streamUI": boolean
    }
  ],
  "uploadConfigurations": [
    {
      "name": "string",
      "path": "string",
      "uploadBehavior": "string"
    }
  ],
  "useDefaultTools": boolean,
  "useDefaultUploadConfigurations": boolean,
  "worldConfigs": [
    {
      "world": "string"
    }
  ]
}
],
"tags": {
  "string" : "string"
},
"useDefaultApplications": boolean,
"vpcConfig": {
  "assignPublicIp": boolean,
  "securityGroups": [ "string" ],
  "subnets": [ "string" ]
}

```

```

    }
  }
],
"status": "string",
"tags": {
  "string" : "string"
}
}

```

## 응답 요소

작업이 성공하면 서비스가 HTTP 200 응답을 반환합니다.

다음 데이터는 서비스에 의해 JSON 형식으로 반환됩니다.

### [arn](#)

배치의 Amazon 리소스 이름(ARN)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

### [batchPolicy](#)

배치 정책입니다.

유형: [BatchPolicy](#) 객체

### [clientRequestToken](#)

요청 멱등성을 보장하기 위해 제공하는 고유한 대소문자 구분 식별자입니다.

유형: 문자열

길이 제한: 최소 길이는 1. 최대 길이는 64.

패턴: [a-zA-Z0-9\_\-=]\*

### [createdAt](#)

시뮬레이션 작업 배치가 마지막으로 생성된 epoch 이후 경과 시간(밀리초)입니다.

유형: 타임스탬프

## createdRequests

생성된 시뮬레이션 작업 요청 요약 목록입니다.

유형: [SimulationJobSummary](#) 객체 어레이

어레이 멤버: 최소 항목 수 0개. 최대 항목 수는 100개입니다.

## failedRequests

실패한 시뮬레이션 작업 요청 목록입니다. 요청이 시뮬레이션 작업을 생성하는데 실패했습니다. 실패한 요청에는 시뮬레이션 작업 ID가 없습니다.

유형: [FailedCreateSimulationJobRequest](#) 객체 어레이

## failureCode

시뮬레이션 작업 배치가 실패한 경우의 실패 코드입니다.

타입: 문자열

유효 값: InternalServiceError

## failureReason

시뮬레이션 작업 배치가 실패한 이유입니다.

타입: 문자열

길이 제약 조건: 최소 길이는 0입니다. 최대 길이는 1024입니다.

패턴: .\*

## pendingRequests

보류 중인 시뮬레이션 작업 요청 목록입니다. 이러한 요청은 아직 시뮬레이션 작업에 생성되지 않았습니다.

유형: [SimulationJobRequest](#) 객체 어레이

배열 멤버: 최소 항목 수는 1입니다. 최대 항목 수는 1,000개입니다.

## status

시뮬레이션 작업 배치의 상태입니다.

보류중

시뮬레이션 작업 배치 요청이 보류 중입니다.

## InProgress

시뮬레이션 작업 배치가 진행 중입니다.

## Failed

시뮬레이션 작업 배치가 실패했습니다. 내부 장애(예: `InternalServerError`)로 인해 하나 이상의 시뮬레이션 작업 요청을 완료할 수 없습니다. 자세한 내용은 `failureCode` 및 `failureReason` 섹션을 참조하세요.

## 완료됨

시뮬레이션 배치 작업이 완료되었습니다. 다음의 경우 배치가 완료됩니다. (1) 배치에 보류 중인 시뮬레이션 작업 요청이 없고 실패한 시뮬레이션 작업 요청 모두가 `InternalServerError`로 인한 것이 아닌 경우, (2) 생성된 모든 시뮬레이션 작업이 최종 상태(예: `Completed` 또는 `Failed`)에 도달한 경우.

## 취소됨

시뮬레이션 배치 작업이 취소되었습니다.

## 취소 중

시뮬레이션 배치 작업을 취소 중입니다.

## 완료 중

시뮬레이션 배치 작업을 완료 중입니다.

## TimingOut

시뮬레이션 작업 배치 시간이 초과되었습니다.

배치 시간이 초과되고 내부 장애(예: `InternalServerError`)로 인해 실패한 보류 중인 요청이 있는 경우, 배치 상태는 `Failed`입니다. 이러한 실패한 요청이 없는 경우 배치 상태는 `TimedOut`입니다.

## TimedOut

시뮬레이션 배치 작업 시간이 초과되었습니다.

## 타입: 문자열

유효 값: `Pending` | `InProgress` | `Failed` | `Completed` | `Canceled` | `Canceling` | `Completing` | `TimingOut` | `TimedOut`

## tags

배포 작업 배치에 연결된 태그 키 및 태그 값을 포함하는 맵입니다.

유형: 문자열-문자열 맵

맵 항목: 최소 항목 수는 0개입니다. 최대 항목 수는 50개입니다.

키 길이 제약 조건: 최소 길이는 1. 최대 길이 128.

키 패턴: [a-zA-Z0-9 \_.\-\/+=:]\*

값 길이 제약 조건: 최소 길이는 0입니다. 최대 길이는 256.

가치 패턴: [a-zA-Z0-9 \_.\-\/+=:]\*

## Errors

모든 작업에서 일반적으로 발생하는 오류에 대한 자세한 내용은 [일반적인 오류](#) 섹션을 참조하세요.

### IdempotentParameterMismatchException

요청은 이전 요청과 동일한 클라이언트 토큰을 사용하지만, 동일하지는 않습니다. 요청이 동일하지 않는 한, 다른 요청에 클라이언트 토큰을 재사용하지 마세요.

HTTP 상태 코드: 400

### InternalServerErrorException

RoboMaker AWS에서 서비스 문제가 발생했습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 500

### InvalidParameterException

요청에 지정된 파라미터는 유효하지 않거나, 지원되지 않거나, 사용할 수 없습니다. 반환된 메시지는 오류 값에 대한 설명을 제공합니다.

HTTP 상태 코드: 400

### LimitExceededException

요청된 리소스가 허용된 최대 수를 초과하거나 동시 스트림 요청 수가 허용된 최대 수를 초과합니다.

HTTP 상태 코드: 400

ThrottlingException

RoboMaker AWS는 일시적으로 요청을 처리할 수 없습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 400

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS Go v2를 위한 SDK](#)
- [AWS Java V2용 SDK](#)
- [AWS V3용 SDK JavaScript](#)
- [AWS PHP V3용 SDK](#)
- [AWS Python용 SDK](#)
- [AWS 루비 V3용 SDK](#)

## SyncDeploymentJob

이 작업은 사용 중단되었습니다.

### Important

해당 API는 더 이상 지원되지 않습니다. 자세한 내용은 [지원 정책](#) 페이지에서 2022년 5월 2일 업데이트를 참조하세요.

플릿의 로봇을 최신 배포와 동기화합니다. 이는 배포 후 로봇을 추가한 경우에 유용합니다.

## Request Syntax

```
POST /syncDeploymentJob HTTP/1.1
Content-type: application/json

{
  "clientRequestToken": "string",
  "fleet": "string"
}
```

## URI 요청 파라미터

요청은 URI 파라미터를 사용하지 않습니다.

## 요청 본문

요청은 JSON 형식으로 다음 데이터를 받습니다.

### clientRequestToken

요청 멱등성을 보장하기 위해 제공하는 고유한 대소문자 구분 식별자입니다.

유형: 문자열

길이 제한: 최소 길이는 1. 최대 길이는 64.

패턴: [a-zA-Z0-9\_\-=]\*

필수 사항 여부: Yes

## fleet

동기화를 위한 대상 플릿입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

필수 항목 여부: 예

## 응답 구문

```

HTTP/1.1 200
Content-type: application/json

{
  "arn": "string",
  "createdAt": number,
  "deploymentApplicationConfigs": [
    {
      "application": "string",
      "applicationVersion": "string",
      "launchConfig": {
        "environmentVariables": {
          "string" : "string"
        },
        "launchFile": "string",
        "packageName": "string",
        "postLaunchFile": "string",
        "preLaunchFile": "string"
      }
    }
  ],
  "deploymentConfig": {
    "concurrentDeploymentPercentage": number,
    "downloadConditionFile": {
      "bucket": "string",
      "etag": "string",
      "key": "string"
    },
    "failureThresholdPercentage": number,

```

```

    "robotDeploymentTimeoutInSeconds": number
  },
  "failureCode": "string",
  "failureReason": "string",
  "fleet": "string",
  "status": "string"
}

```

## 응답 요소

작업이 성공하면 서비스가 HTTP 200 응답을 반환합니다.

다음 데이터는 서비스에 의해 JSON 형식으로 반환됩니다.

### arn

동기화 요청의 Amazon 리소스 이름(ARN)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

### createdAt

플릿이 생성된 epoch 이후 경과 시간(밀리초)입니다.

유형: 타임스탬프

### deploymentApplicationConfigs

배포 애플리케이션 구성에 대한 정보입니다.

유형: [DeploymentApplicationConfig](#) 객체 어레이

배열 멤버: 고정된 항목 수는 1개입니다.

### deploymentConfig

배포 구성에 대한 정보입니다.

유형: [DeploymentConfig](#) 객체

### failureCode

작업 실패 시 오류 코드:

### InternalServiceError

내부 서비스 오류.

### RobotApplicationCrash

로봇 애플리케이션이 비정상적으로 종료되었습니다.

### SimulationApplicationCrash

시뮬레이션 애플리케이션이 비정상적으로 종료되었습니다.

### BadPermissionsRobotApplication

로봇 애플리케이션 번들을 다운로드할 수 없습니다.

### BadPermissionsSimulationApplication

시뮬레이션 애플리케이션 번들을 다운로드할 수 없습니다.

### BadPermissionsS3 출력

출력을 고객이 제공한 S3 버킷에 게시할 수 없습니다.

### BadPermissionsCloudwatchLogs

고객이 CloudWatch 제공한 Logs 리소스에 로그를 게시할 수 없습니다.

### SubnetIpLimitExceeded

서브넷 IP 제한을 초과했습니다.

### ENI LimitExceeded

ENI 제한을 초과했습니다.

### BadPermissionsUserCredentials

제공된 역할을 사용할 수 없습니다.

### InvalidBundleRobotApplication

로봇 번들을 추출할 수 없습니다(잘못된 형식, 번들링 오류 또는 기타 문제).

### InvalidBundleSimulationApplication

시뮬레이션 번들을 추출할 수 없습니다(잘못된 형식, 번들링 오류 또는 기타 문제).

### RobotApplicationVersionMismatchedEtag

버전 생성 중 Etag for 값이 일치하지 RobotApplication 않습니다.

## SimulationApplicationVersionMismatchedEtag

버전 생성 시 Etag for 값이 일치하지 SimulationApplication 않습니다.

타입: 문자열

유효 값: ResourceNotFound | EnvironmentSetupError | EtagMismatch  
 | FailureThresholdBreached | RobotDeploymentAborted |  
 RobotDeploymentNoResponse | RobotAgentConnectionTimeout  
 | GreengrassDeploymentFailed | InvalidGreengrassGroup |  
 MissingRobotArchitecture | MissingRobotApplicationArchitecture |  
 MissingRobotDeploymentResource | GreengrassGroupVersionDoesNotExist  
 | LambdaDeleted | ExtractingBundleFailure | PreLaunchFileFailure |  
 PostLaunchFileFailure | BadPermissionError | DownloadConditionFailed |  
 BadLambdaAssociated | InternalServerError | RobotApplicationDoesNotExist  
 | DeploymentFleetDoesNotExist | FleetDeploymentTimeout

### [failureReason](#)

작업이 실패한 경우의 실패 원인입니다.

타입: 문자열

길이 제약 조건: 최소 길이는 0입니다. 최대 길이는 1024입니다.

패턴: .\*

### [fleet](#)

플릿의 Amazon 리소스 이름(ARN)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

### [status](#)

동기화 작업의 상태입니다.

타입: 문자열

유효 값: Pending | Preparing | InProgress | Failed | Succeeded | Canceled

## Errors

모든 작업에서 일반적으로 발생하는 오류에 대한 자세한 내용은 [일반적인 오류](#) 섹션을 참조하세요.

### ConcurrentDeploymentException

실패율 임계값 백분율이 충족되었습니다.

HTTP 상태 코드: 400

### IdempotentParameterMismatchException

요청은 이전 요청과 동일한 클라이언트 토큰을 사용하지만, 동일하지는 않습니다. 요청이 동일하지 않는 한, 다른 요청에 클라이언트 토큰을 재사용하지 마세요.

HTTP 상태 코드: 400

### InternalServerErrorException

RoboMaker AWS에서 서비스 문제가 발생했습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 500

### InvalidParameterException

요청에 지정된 파라미터는 유효하지 않거나, 지원되지 않거나, 사용할 수 없습니다. 반환된 메시지는 오류 값에 대한 설명을 제공합니다.

HTTP 상태 코드: 400

### LimitExceededException

요청된 리소스가 허용된 최대 수를 초과하거나 동시 스트림 요청 수가 허용된 최대 수를 초과합니다.

HTTP 상태 코드: 400

### ResourceNotFoundException

지정한 리소스가 존재하지 않습니다.

HTTP 상태 코드: 400

### ThrottlingException

RoboMaker AWS는 일시적으로 요청을 처리할 수 없습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 400

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS Go v2를 위한 SDK](#)
- [AWS Java V2용 SDK](#)
- [AWS V3용 SDK JavaScript](#)
- [AWS PHP V3용 SDK](#)
- [AWS Python용 SDK](#)
- [AWS 루비 V3용 SDK](#)

## TagResource

AWS RoboMaker 리소스의 태그를 추가하거나 편집합니다.

각 태그는 태그 키와 태그 값으로 구성됩니다. 태그 키와 태그 값 모두 필요하지만 태그 값은 빈 문자열일 수 있습니다.

태그 키 및 태그 값에 적용되는 규칙에 대한 자세한 내용은 [AWS 과금 정보 및 비용 관리 사용 설명서의 사용자 지정 태그 제한](#)을 참조하세요.

### Request Syntax

```
POST /tags/resourceArn HTTP/1.1
Content-type: application/json

{
  "tags": {
    "string" : "string"
  }
}
```

### URI 요청 파라미터

요청은 다음 URI 파라미터를 사용합니다.

#### resourceArn

태그를 지정하는 AWS 리소스의 Amazon RoboMaker 리소스 이름 (ARN).

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

필수 사항 여부: Yes

### 요청 본문

요청은 JSON 형식으로 다음 데이터를 받습니다.

#### tags

리소스에 연결된 태그 키 및 태그 값을 포함하는 맵입니다.

유형: 문자열-문자열 맵

맵 항목: 최소 항목 수는 0개입니다. 최대 항목 수는 50개입니다.

키 길이 제약 조건: 최소 길이는 1. 최대 길이 128.

키 패턴: [a-zA-Z0-9 \_.\-\/+=:]\*

값 길이 제약 조건: 최소 길이는 0입니다. 최대 길이는 256.

가치 패턴: [a-zA-Z0-9 \_.\-\/+=:]\*

필수 여부: 예

## 응답 구문

```
HTTP/1.1 200
```

## Response Elements

작업이 성공하면 서비스가 비어 있는 HTTP 본문과 함께 HTTP 200 응답을 반환합니다.

## Errors

모든 작업에서 발생하는 일반적인 오류에 대한 자세한 내용은 [일반적인 오류](#) 섹션을 참조하세요.

### InternalServerErrorException

RoboMaker AWS에서 서비스 문제가 발생했습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 500

### InvalidParameterException

요청에 지정된 파라미터는 유효하지 않거나, 지원되지 않거나, 사용할 수 없습니다. 반환된 메시지는 오류 값에 대한 설명을 제공합니다.

HTTP 상태 코드: 400

### ResourceNotFoundException

지정한 리소스가 존재하지 않습니다.

HTTP 상태 코드: 400

ThrottlingException

RoboMaker AWS는 일시적으로 요청을 처리할 수 없습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 400

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS Go v2를 위한 SDK](#)
- [AWS Java V2용 SDK](#)
- [AWS V3용 SDK JavaScript](#)
- [AWS PHP V3용 SDK](#)
- [AWS Python용 SDK](#)
- [AWS 루비 V3용 SDK](#)

## UntagResource

지정된 AWS RoboMaker 리소스에서 지정된 태그를 제거합니다.

태그를 제거하려면 태그 키를 지정합니다. 기존 태그 키의 태그 값을 변경하려면 [TagResource](#)를 사용합니다.

### Request Syntax

```
DELETE /tags/resourceArn?tagKeys=tagKeys HTTP/1.1
```

### URI 요청 파라미터

요청은 다음 URI 파라미터를 사용합니다.

#### [resourceArn](#)

태그를 제거하려는 AWS 리소스의 Amazon RoboMaker 리소스 이름 (ARN)

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: `arn:.*`

필수 사항 여부: Yes

#### [tagKeys](#)

리소스에서 연결 해제되는 태그 키 및 태그 값을 포함하는 맵입니다.

길이 제약: 최소 길이 1. 최대 길이 128.

패턴: `[a-zA-Z0-9 _.\-\/+=:]*`

필수 사항 여부: Yes

### Request Body

해당 요청에는 본문이 없습니다.

### Response Syntax

```
HTTP/1.1 200
```

## Response Elements

작업이 성공하면 서비스가 비어 있는 HTTP 본문과 함께 HTTP 200 응답을 반환합니다.

## Errors

모든 작업에서 발생하는 일반적인 오류에 대한 자세한 내용은 [일반적인 오류](#) 섹션을 참조하세요.

### InternalServerErrorException

RoboMaker AWS에서 서비스 문제가 발생했습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 500

### InvalidParameterException

요청에 지정된 파라미터는 유효하지 않거나, 지원되지 않거나, 사용할 수 없습니다. 반환된 메시지는 오류 값에 대한 설명을 제공합니다.

HTTP 상태 코드: 400

### ResourceNotFoundException

지정한 리소스가 존재하지 않습니다.

HTTP 상태 코드: 400

### ThrottlingException

RoboMaker AWS는 일시적으로 요청을 처리할 수 없습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 400

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS Go v2를 위한 SDK](#)
- [AWS Java V2용 SDK](#)

- [AWS V3용 SDK JavaScript](#)
- [AWS PHP V3용 SDK](#)
- [AWS Python용 SDK](#)
- [AWS 루비 V3용 SDK](#)

# UpdateRobotApplication

로봇 애플리케이션을 업데이트합니다.

## Request Syntax

```
POST /updateRobotApplication HTTP/1.1
Content-type: application/json
```

```
{
  "application": "string",
  "currentRevisionId": "string",
  "environment": {
    "uri": "string"
  },
  "robotSoftwareSuite": {
    "name": "string",
    "version": "string"
  },
  "sources": [
    {
      "architecture": "string",
      "s3Bucket": "string",
      "s3Key": "string"
    }
  ]
}
```

## URI 요청 파라미터

요청은 URI 파라미터를 사용하지 않습니다.

## 요청 본문

요청은 JSON 형식으로 다음 데이터를 받습니다.

### application

로봇 애플리케이션에 대한 애플리케이션 정보입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: `arn:.*`

필수 사항 여부: Yes

### currentRevisionId

로봇 애플리케이션에 대한 개정 ID입니다.

유형: 문자열

길이 제약: 최소 길이 1자. 최대 길이는 40입니다.

패턴: `[a-zA-Z0-9_.\-]*`

Required: No

### environment

로봇 애플리케이션의 도커 이미지 URI를 포함하는 객체입니다.

유형: [Environment](#) 객체

필수 항목 여부: 아니요

### robotSoftwareSuite

로봇 애플리케이션이 사용하는 로봇 소프트웨어 제품군입니다.

유형: [RobotSoftwareSuite](#) 객체

필수 여부: 예

### sources

로봇 애플리케이션의 소스입니다.

타입: [SourceConfig](#) 객체 배열

필수: 아니요

## 응답 구문

```
HTTP/1.1 200
Content-type: application/json
```

```

{
  "arn": "string",
  "environment": {
    "uri": "string"
  },
  "lastUpdatedAt": number,
  "name": "string",
  "revisionId": "string",
  "robotSoftwareSuite": {
    "name": "string",
    "version": "string"
  },
  "sources": [
    {
      "architecture": "string",
      "etag": "string",
      "s3Bucket": "string",
      "s3Key": "string"
    }
  ],
  "version": "string"
}

```

## 응답 요소

작업이 성공하면 서비스가 HTTP 200 응답을 반환합니다.

다음 데이터는 서비스에 의해 JSON 형식으로 반환됩니다.

### arn

업데이트된 로봇 애플리케이션의 Amazon 리소스 이름(ARN)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

### environment

로봇 애플리케이션의 도커 이미지 URI를 포함하는 객체입니다.

유형: Environment 객체

## lastUpdatedAt

로봇 애플리케이션이 마지막으로 업데이트된 epoch 이후 경과 시간(밀리초)입니다.

유형: 타임스탬프

## name

로봇 애플리케이션의 이름입니다.

유형: 문자열

길이 제약: 최소 길이는 1. 최대 길이는 255.

패턴: [a-zA-Z0-9\_\-]\*

## revisionId

로봇 애플리케이션의 개정 ID입니다.

유형: 문자열

길이 제약: 최소 길이 1자. 최대 길이는 40입니다.

패턴: [a-zA-Z0-9\_.\-]\*

## robotSoftwareSuite

로봇 애플리케이션이 사용하는 로봇 소프트웨어 제품군입니다.

유형: [RobotSoftwareSuite](#) 객체

## sources

로봇 애플리케이션의 소스입니다.

유형: [Source](#) 객체 어레이

## version

로봇 애플리케이션의 버전입니다.

유형: 문자열

길이 제약: 최소 길이는 1. 최대 길이는 255.

패턴: (\\$LATEST)|[0-9]\*

## Errors

모든 작업에서 발생하는 일반적인 오류에 대한 자세한 내용은 [일반적인 오류](#) 섹션을 참조하세요.

### InternalServerErrorException

RoboMaker AWS에서 서비스 문제가 발생했습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 500

### InvalidParameterException

요청에 지정된 파라미터는 유효하지 않거나, 지원되지 않거나, 사용할 수 없습니다. 반환된 메시지는 오류 값에 대한 설명을 제공합니다.

HTTP 상태 코드: 400

### LimitExceededException

요청된 리소스가 허용된 최대 수를 초과하거나 동시 스트림 요청 수가 허용된 최대 수를 초과합니다.

HTTP 상태 코드: 400

### ResourceNotFoundException

지정한 리소스가 존재하지 않습니다.

HTTP 상태 코드: 400

### ThrottlingException

RoboMaker AWS는 일시적으로 요청을 처리할 수 없습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 400

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS Go v2를 위한 SDK](#)

- [AWS Java V2용 SDK](#)
- [AWS V3용 SDK JavaScript](#)
- [AWS PHP V3용 SDK](#)
- [AWS Python용 SDK](#)
- [AWS 루비 V3용 SDK](#)

# UpdateSimulationApplication

시뮬레이션 애플리케이션을 업데이트합니다.

## Request Syntax

```
POST /updateSimulationApplication HTTP/1.1
Content-type: application/json
```

```
{
  "application": "string",
  "currentRevisionId": "string",
  "environment": {
    "uri": "string"
  },
  "renderingEngine": {
    "name": "string",
    "version": "string"
  },
  "robotSoftwareSuite": {
    "name": "string",
    "version": "string"
  },
  "simulationSoftwareSuite": {
    "name": "string",
    "version": "string"
  },
  "sources": [
    {
      "architecture": "string",
      "s3Bucket": "string",
      "s3Key": "string"
    }
  ]
}
```

## URI 요청 파라미터

요청은 URI 파라미터를 사용하지 않습니다.

## 요청 본문

요청은 JSON 형식으로 다음 데이터를 받습니다.

## application

시뮬레이션 애플리케이션에 대한 애플리케이션 정보입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

필수 사항 여부: Yes

## currentRevisionId

로봇 애플리케이션에 대한 개정 ID입니다.

유형: 문자열

길이 제약: 최소 길이 1자. 최대 길이는 40입니다.

패턴: [a-zA-Z0-9\_.\-]\*

Required: No

## environment

시뮬레이션 애플리케이션의 도커 이미지 URI를 포함하는 객체입니다.

유형: [Environment](#) 객체

필수 항목 여부: 아니요

## renderingEngine

시뮬레이션 애플리케이션의 렌더링 엔진입니다.

유형: [RenderingEngine](#) 객체

필수 항목 여부: 아니요

## robotSoftwareSuite

로봇 소프트웨어 제품군에 대한 정보입니다.

유형: [RobotSoftwareSuite](#) 객체

필수 여부: 예

## simulationSoftwareSuite

시뮬레이션 애플리케이션이 사용하는 시뮬레이션 소프트웨어 제품군입니다.

유형: [SimulationSoftwareSuite](#) 객체

필수 여부: 예

### sources

시뮬레이션 애플리케이션의 소스입니다.

타입: [SourceConfig](#) 객체 배열

필수: 아니요

## 응답 구문

```

HTTP/1.1 200
Content-type: application/json

{
  "arn": "string",
  "environment": {
    "uri": "string"
  },
  "lastUpdatedAt": number,
  "name": "string",
  "renderingEngine": {
    "name": "string",
    "version": "string"
  },
  "revisionId": "string",
  "robotSoftwareSuite": {
    "name": "string",
    "version": "string"
  },
  "simulationSoftwareSuite": {
    "name": "string",
    "version": "string"
  },
  "sources": [
    {

```

```

    "architecture": "string",
    "etag": "string",
    "s3Bucket": "string",
    "s3Key": "string"
  }
],
"version": "string"
}

```

## 응답 요소

작업이 성공하면 서비스가 HTTP 200 응답을 반환합니다.

다음 데이터는 서비스에 의해 JSON 형식으로 반환됩니다.

### [arn](#)

업데이트된 시뮬레이션 애플리케이션의 Amazon 리소스 이름(ARN)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

### [environment](#)

시뮬레이션 애플리케이션에 사용되는 도커 이미지 URI를 포함하는 객체입니다.

유형: [Environment](#) 객체

### [lastUpdatedAt](#)

시뮬레이션 애플리케이션이 마지막으로 업데이트된 epoch 이후 경과 시간(밀리초).

유형: 타임스탬프

### [name](#)

시뮬레이션 애플리케이션의 이름입니다.

유형: 문자열

길이 제약: 최소 길이는 1. 최대 길이는 255.

패턴: [a-zA-Z0-9\_\-]\*

## [renderingEngine](#)

시뮬레이션 애플리케이션의 렌더링 엔진입니다.

유형: [RenderingEngine](#) 객체

## [revisionId](#)

시뮬레이션 애플리케이션에 대한 개정 ID입니다.

유형: 문자열

길이 제약: 최소 길이 1자. 최대 길이는 40입니다.

패턴: `[a-zA-Z0-9_.\-]*`

## [robotSoftwareSuite](#)

로봇 소프트웨어 제품군에 대한 정보입니다.

유형: [RobotSoftwareSuite](#) 객체

## [simulationSoftwareSuite](#)

시뮬레이션 애플리케이션이 사용하는 시뮬레이션 소프트웨어 제품군입니다.

유형: [SimulationSoftwareSuite](#) 객체

## [sources](#)

시뮬레이션 애플리케이션의 소스입니다.

유형: [Source](#) 객체 어레이

## [version](#)

로봇 애플리케이션의 버전입니다.

유형: 문자열

길이 제약: 최소 길이는 1. 최대 길이는 255.

패턴: `(\$LATEST)|[0-9]*`

## Errors

모든 작업에서 발생하는 일반적인 오류에 대한 자세한 내용은 [일반적인 오류](#) 섹션을 참조하세요.

## InternalServerErrorException

RoboMaker AWS에서 서비스 문제가 발생했습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 500

## InvalidParameterException

요청에 지정된 파라미터는 유효하지 않거나, 지원되지 않거나, 사용할 수 없습니다. 반환된 메시지는 오류 값에 대한 설명을 제공합니다.

HTTP 상태 코드: 400

## LimitExceededException

요청된 리소스가 허용된 최대 수를 초과하거나 동시 스트림 요청 수가 허용된 최대 수를 초과합니다.

HTTP 상태 코드: 400

## ResourceNotFoundException

지정한 리소스가 존재하지 않습니다.

HTTP 상태 코드: 400

## ThrottlingException

RoboMaker AWS는 일시적으로 요청을 처리할 수 없습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 400

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS Go v2를 위한 SDK](#)
- [AWS Java V2용 SDK](#)
- [AWS V3용 SDK JavaScript](#)

- [AWS PHP V3용 SDK](#)
- [AWS Python용 SDK](#)
- [AWS 루비 V3용 SDK](#)

# UpdateWorldTemplate

월드 템플릿을 업데이트합니다.

## Request Syntax

```
POST /updateWorldTemplate HTTP/1.1
Content-type: application/json

{
  "name": "string",
  "template": "string",
  "templateBody": "string",
  "templateLocation": {
    "s3Bucket": "string",
    "s3Key": "string"
  }
}
```

## URI 요청 파라미터

요청은 URI 파라미터를 사용하지 않습니다.

## 요청 본문

요청은 JSON 형식으로 다음 데이터를 받습니다.

### name

템플릿의 이름입니다.

타입: 문자열

길이 제약 조건: 최소 길이는 0입니다. 최대 길이는 255.

패턴: .\*

Required: No

### template

업데이트할 월드 템플릿의 Amazon 리소스 이름(ARN)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: `arn:.*`

필수 사항 여부: Yes

### templateBody

월드 템플릿 본문입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 262144자입니다.

패턴: `[\S\s]+`

Required: No

### templateLocation

월드 템플릿의 위치입니다.

유형: [TemplateLocation](#) 객체

필수 항목 여부: 아니요

## 응답 구문

```
HTTP/1.1 200
Content-type: application/json

{
  "arn": "string",
  "createdAt": number,
  "lastUpdatedAt": number,
  "name": "string"
}
```

## 응답 요소

작업이 성공하면 서비스가 HTTP 200 응답을 반환합니다.

다음 데이터는 서비스에 의해 JSON 형식으로 반환됩니다.

## arn

월드 템플릿의 Amazon 리소스 이름(ARN)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

## createdAt

월드 템플릿이 생성된 epoch 이후 경과 시간(밀리초)입니다.

유형: 타임스탬프

## lastUpdatedAt

월드 템플릿이 업데이트된 epoch 이후 경과 시간(밀리초)입니다.

유형: 타임스탬프

## name

월드 템플릿의 이름입니다.

타입: 문자열

길이 제약 조건: 최소 길이는 0입니다. 최대 길이는 255.

패턴: .\*

## Errors

모든 작업에서 발생하는 일반적인 오류에 대한 자세한 내용은 [일반적인 오류](#) 섹션을 참조하세요.

### InternalServerErrorException

RoboMaker AWS에서 서비스 문제가 발생했습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 500

### InvalidParameterException

요청에 지정된 파라미터는 유효하지 않거나, 지원되지 않거나, 사용할 수 없습니다. 반환된 메시지는 오류 값에 대한 설명을 제공합니다.

HTTP 상태 코드: 400

ResourceNotFoundException

지정한 리소스가 존재하지 않습니다.

HTTP 상태 코드: 400

ThrottlingException

RoboMaker AWS는 일시적으로 요청을 처리할 수 없습니다. 호출을 다시 시도하세요.

HTTP 상태 코드: 400

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS Go v2를 위한 SDK](#)
- [AWS Java V2용 SDK](#)
- [AWS V3용 SDK JavaScript](#)
- [AWS PHP V3용 SDK](#)
- [AWS Python용 SDK](#)
- [AWS 루비 V3용 SDK](#)

## 데이터 유형

다음 데이터 유형이 지원됩니다.

- [BatchPolicy](#)
- [Compute](#)
- [ComputeResponse](#)
- [DataSource](#)
- [DataSourceConfig](#)

- [DeploymentApplicationConfig](#)
- [DeploymentConfig](#)
- [DeploymentJob](#)
- [DeploymentLaunchConfig](#)
- [Environment](#)
- [FailedCreateSimulationJobRequest](#)
- [FailureSummary](#)
- [Filter](#)
- [FinishedWorldsSummary](#)
- [Fleet](#)
- [LaunchConfig](#)
- [LoggingConfig](#)
- [NetworkInterface](#)
- [OutputLocation](#)
- [PortForwardingConfig](#)
- [PortMapping](#)
- [ProgressDetail](#)
- [RenderingEngine](#)
- [Robot](#)
- [RobotApplicationConfig](#)
- [RobotApplicationSummary](#)
- [RobotDeployment](#)
- [RobotSoftwareSuite](#)
- [S3KeyOutput](#)
- [S3Object](#)
- [SimulationApplicationConfig](#)
- [SimulationApplicationSummary](#)
- [SimulationJob](#)
- [SimulationJobBatchSummary](#)
- [SimulationJobRequest](#)

- [SimulationJobSummary](#)
- [SimulationSoftwareSuite](#)
- [Source](#)
- [SourceConfig](#)
- [TemplateLocation](#)
- [TemplateSummary](#)
- [Tool](#)
- [UploadConfiguration](#)
- [VPCConfig](#)
- [VPCConfigResponse](#)
- [WorldConfig](#)
- [WorldCount](#)
- [WorldExportJobSummary](#)
- [WorldFailure](#)
- [WorldGenerationJobSummary](#)
- [WorldSummary](#)

## BatchPolicy

배치 정책에 대한 정보입니다.

### 내용

#### maxConcurrency

동시에 활성 상태에 있을 수 있는 배치의 일부로 생성되는 활성 시뮬레이션 작업의 수입니다.

활성 상태 포함 항목: Pending, Preparing, Running, Restarting, RunningFailed 및 Terminating. 다른 모든 상태는 터미널 상태입니다.

유형: 정수

필수 항목 여부: 아니요

#### timeoutInSeconds

배치가 완료될 때까지 대기하는 시간(초)입니다.

배치 제한 시간이 초과되고 내부 장애(예: InternalServiceError)로 인해 실패한 대기 중인 요청이 있는 경우 해당 요청은 실패 목록으로 이동되고 배치 상태는 Failed가 됩니다. 대기 중인 요청이 다른 이유로 실패한 경우, 실패한 대기 중인 요청은 실패 목록으로 이동되고 배치 상태는 TimedOut이 됩니다.

유형: Long

필수 여부: 아니요

### 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS SDK for C++](#)
- [AWS Java V2용 SDK](#)
- [AWS 루비 V3용 SDK](#)

## Compute

시뮬레이션 작업에 대한 컴퓨팅 정보입니다.

### 내용

#### computeType

시뮬레이션 작업에 대한 컴퓨팅 유형 정보입니다.

타입: 문자열

유효 값: CPU | GPU\_AND\_CPU

필수 여부: 아니요

#### gpuUnitLimit

시뮬레이션 작업의 GPU 단위 제한을 컴퓨팅합니다. 에 할당된 GPU 수와 동일합니다.

#### SimulationJob

유형: 정수

유효한 범위: 최소값 0. 최댓값은 1.

필수 여부: 아니요

#### simulationUnitLimit

시뮬레이션 단위 제한입니다. 시뮬레이션에 제공된 시뮬레이션 단위 제한에 비례하여 CPU와 메모리가 할당됩니다. 시뮬레이션 단위는 1 vcpu 및 2GB의 메모리입니다. 제공된 최댓값까지 소비한 SU 사용량에 대해서만 요금이 청구됩니다. 기본값은 15입니다.

타입: 정수

유효 범위: 최소값 1. 최댓값은 15입니다.

필수 여부: 아니요

### 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS SDK for C++](#)

- [AWS Java V2용 SDK](#)
- [AWS 루비 V3용 SDK](#)

# ComputeResponse

시뮬레이션 작업에 대한 컴퓨팅 정보입니다.

## 내용

### computeType

시뮬레이션 작업에 대한 컴퓨팅 유형 응답 정보입니다.

타입: 문자열

유효 값: CPU | GPU\_AND\_CPU

필수 여부: 아니요

### gpuUnitLimit

시뮬레이션 작업의 GPU 단위 제한을 컴퓨팅합니다. 에 할당된 GPU 수와 동일합니다.

#### SimulationJob

유형: 정수

유효한 범위: 최소값 0. 최댓값은 1.

필수 여부: 아니요

### simulationUnitLimit

시뮬레이션 단위 제한입니다. 시뮬레이션에 제공된 시뮬레이션 단위 제한에 비례하여 CPU와 메모리가 할당됩니다. 시뮬레이션 단위는 1 vcpu 및 2GB의 메모리입니다. 제공된 최댓값까지 소비한 SU 사용량에 대해서만 요금이 청구됩니다. 기본값은 15입니다.

타입: 정수

유효 범위: 최소값 1. 최댓값은 15입니다.

필수 여부: 아니요

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS SDK for C++](#)

- [AWS Java V2용 SDK](#)
- [AWS 루비 V3용 SDK](#)

## DataSource

데이터 소스에 대한 정보입니다.

### 내용

#### destination

컨테이너 이미지에서 파일이 마운트되는 위치입니다.

데이터 소스의 type를 Archive로 지정한 경우 Amazon S3 객체 키를 아카이브에 제공해야 합니다. 객체 키는 .zip 또는 .tar.gz 파일을 가리켜야 합니다.

데이터 소스의 type를 Prefix로 지정한 경우 데이터 소스로 사용 중인 파일을 가리키는 Amazon S3 접두사를 제공해야 합니다.

데이터 소스의 type를 File로 지정한 경우 데이터 소스로 사용하는 파일의 Amazon S3 경로를 제공해야 합니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1024입니다.

패턴: .\*

Required: No

#### name

데이터 원본의 이름입니다.

유형: 문자열

길이 제약: 최소 길이는 1. 최대 길이는 255.

패턴: [a-zA-Z0-9\_\-\ ]\*

Required: No

#### s3Bucket

데이터 파일이 위치한 Amazon S3 버킷입니다.

타입: 문자열

길이 제약 조건: 최소 길이는 3입니다. 최대 길이는 63입니다.

패턴: `[a-z0-9][a-z0-9.\-]*[a-z0-9]`

Required: No

### s3Keys

데이터 소스 파일을 식별하는 S3 키 목록입니다.

타입: [S3KeyOutput](#) 객체 배열

필수: 아니요

### type

컨테이너 이미지 또는 시뮬레이션 작업에 사용하는 데이터 소스의 데이터 유형입니다. 이 필드를 사용하여 데이터 소스가 아카이브인지, Amazon S3 접두사인지 또는 파일인지를 지정할 수 있습니다.

필드를 지정하지 않으면 기본값은 File입니다.

타입: 문자열

유효 값: Prefix | Archive | File

필수 여부: 아니요

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS SDK for C++](#)
- [AWS Java V2용 SDK](#)
- [AWS 루비 V3용 SDK](#)

## DataSourceConfig

데이터 소스에 대한 정보입니다.

### 내용

#### name

데이터 원본의 이름입니다.

유형: 문자열

길이 제약: 최소 길이는 1. 최대 길이는 255.

패턴: [a-zA-Z0-9\_\-]\*

필수 사항 여부: Yes

#### s3Bucket

데이터 파일이 위치한 Amazon S3 버킷입니다.

타입: 문자열

길이 제약 조건: 최소 길이는 3입니다. 최대 길이는 63입니다.

패턴: [a-z0-9][a-z0-9.\-]\*[a-z0-9]

필수 사항 여부: Yes

#### s3Keys

데이터 소스 파일을 식별하는 S3 키 목록입니다.

유형: 문자열 어레이

배열 멤버: 최소수는 1개입니다. 최대 항목 수는 100개입니다.

길이 제한: 최소 길이는 0. 최대 길이는 1024입니다.

패턴: .\*

필수 사항 여부: Yes

## destination

컨테이너 이미지에서 파일이 마운트되는 위치입니다.

데이터 소스의 type를 Archive로 지정한 경우 Amazon S3 객체 키를 아카이브에 제공해야 합니다. 객체 키는 .zip 또는 .tar.gz 파일을 가리켜야 합니다.

데이터 소스의 type를 Prefix로 지정한 경우 데이터 소스로 사용 중인 파일을 가리키는 Amazon S3 접두사를 제공해야 합니다.

데이터 소스의 type를 File로 지정한 경우 데이터 소스로 사용하는 파일의 Amazon S3 경로를 제공해야 합니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1024입니다.

패턴: .\*

Required: No

## type

컨테이너 이미지 또는 시뮬레이션 작업에 사용하는 데이터 소스의 데이터 유형입니다. 이 필드를 사용하여 데이터 소스가 아카이브인지, Amazon S3 접두사인지 또는 파일인지를 지정할 수 있습니다.

필드를 지정하지 않으면 기본값은 File입니다.

타입: 문자열

유효 값: Prefix | Archive | File

필수 여부: 아니요

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS SDK for C++](#)
- [AWS Java V2용 SDK](#)
- [AWS 루비 V3용 SDK](#)



# DeploymentApplicationConfig

배포 애플리케이션 구성에 대한 정보입니다.

## 내용

### application

로봇 애플리케이션의 Amazon 리소스 이름(ARN)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

필수 사항 여부: Yes

### applicationVersion

애플리케이션의 버전입니다

유형: 문자열

길이 제약: 최소 길이는 1. 최대 길이는 255.

패턴: [0-9]\*

필수 사항 여부: Yes

### launchConfig

시작 구성입니다.

유형: [DeploymentLaunchConfig](#) 객체

필수 여부: 예

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS SDK for C++](#)

- [AWS Java V2용 SDK](#)
- [AWS 루비 V3용 SDK](#)

## DeploymentConfig

배포 구성에 대한 정보입니다.

### 내용

#### concurrentDeploymentPercentage

동시에 배포를 받는 로봇의 비율입니다.

타입: 정수

유효 범위: 최소값 1. 최대값 100.

필수 여부: 아니요

#### downloadConditionFile

다운로드 조건 파일입니다.

유형: [S3Object](#) 객체

필수 항목 여부: 아니요

#### failureThresholdPercentage

배포를 중지하기 전에 실패해야 하는 배포의 비율입니다.

타입: 정수

유효 범위: 최소값 1. 최대값 100.

필수 여부: 아니요

#### robotDeploymentTimeoutInSeconds

단일 로봇에 대한 배포가 완료될 때까지 대기하는 시간입니다. 1분~7일 중에서 시간을 선택하세요. 기본값은 5시간입니다.

유형: Long

필수 여부: 아니요

### 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS SDK for C++](#)
- [AWS Java V2용 SDK](#)
- [AWS 루비 V3용 SDK](#)

## DeploymentJob

배포 작업에 대한 정보입니다.

### 내용

#### arn

배포 작업의 Amazon 리소스 이름(ARN)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

Required: No

#### createdAt

배포 작업이 생성된 epoch 이후 경과 시간(밀리초)입니다.

유형: 타임스탬프

필수 여부: 아니요

#### deploymentApplicationConfigs

배포 애플리케이션 구성입니다.

유형: [DeploymentApplicationConfig](#) 객체 어레이

배열 멤버: 고정된 항목 수는 1개입니다.

필수 여부: 아니요

#### deploymentConfig

배포 구성입니다.

유형: [DeploymentConfig](#) 객체

필수 항목 여부: 아니요

#### failureCode

배포 작업 실패 코드입니다.

타입: 문자열

유효 값: ResourceNotFound | EnvironmentSetupError | EtagMismatch  
 | FailureThresholdBreached | RobotDeploymentAborted |  
 RobotDeploymentNoResponse | RobotAgentConnectionTimeout  
 | GreengrassDeploymentFailed | InvalidGreengrassGroup |  
 MissingRobotArchitecture | MissingRobotApplicationArchitecture |  
 MissingRobotDeploymentResource | GreengrassGroupVersionDoesNotExist  
 | LambdaDeleted | ExtractingBundleFailure | PreLaunchFileFailure |  
 PostLaunchFileFailure | BadPermissionError | DownloadConditionFailed |  
 BadLambdaAssociated | InternalServerError | RobotApplicationDoesNotExist  
 | DeploymentFleetDoesNotExist | FleetDeploymentTimeout

필수 여부: 아니요

failureReason

배포 작업이 실패한 이유에 대한 간략한 설명입니다.

타입: 문자열

길이 제약 조건: 최소 길이는 0입니다. 최대 길이는 1024입니다.

패턴: .\*

Required: No

fleet

플릿의 Amazon 리소스 이름(ARN)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

Required: No

status

배포 작업의 상태입니다.

타입: 문자열

유효 값: Pending | Preparing | InProgress | Failed | Succeeded | Canceled

필수 여부: 아니요

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS SDK for C++](#)
- [AWS Java V2용 SDK](#)
- [AWS 루비 V3용 SDK](#)

## DeploymentLaunchConfig

배포 시작을 위한 구성 정보입니다.

### 내용

#### launchFile

시작 파일 이름입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1024입니다.

패턴: [a-zA-Z0-9\_.\-]\*

필수 사항 여부: Yes

#### packageName

패키지 이름입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1024입니다.

패턴: [a-zA-Z0-9\_.\-]\*

필수 사항 여부: Yes

#### environmentVariables

로봇 애플리케이션의 환경 변수를 지정하는 키/값 쌍의 배열

유형: 문자열-문자열 맵

맵 항목: 최소 항목 수는 0개입니다. 최대 항목 수는 20개.

키 길이 제약: 최소 길이는 1입니다. 최대 길이 1024.

키 패턴: [A-Z\_][A-Z0-9\_]\*

값 길이 제약 조건: 최소 길이는 1. 최대 길이 1024.

가치 패턴: .\*

필수 여부: 아니요

#### postLaunchFile

시작 후 배포 파일입니다. 이 파일은 시작 파일 이후에 실행됩니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1024입니다.

패턴: .\*

Required: No

#### preLaunchFile

시작 전 배포 파일입니다. 이 파일은 시작 파일보다 먼저 실행됩니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1024입니다.

패턴: .\*

필수 여부: 아니요

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS SDK for C++](#)
- [AWS Java V2용 SDK](#)
- [AWS 루비 V3용 SDK](#)

## Environment

로봇 또는 시뮬레이션 애플리케이션의 도커 이미지 URI를 포함하는 객체입니다.

### 내용

uri

로봇 또는 시뮬레이션 애플리케이션을 위한 도커 이미지 URI입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1024입니다.

패턴: .+

필수 여부: 아니요

### 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS SDK for C++](#)
- [AWS Java V2용 SDK](#)
- [AWS 루비 V3용 SDK](#)

## FailedCreateSimulationJobRequest

실패한 시뮬레이션 작업 생성 요청에 대한 정보입니다.

### 내용

#### failedAt

시뮬레이션 작업 배치가 실패한 epoch 이후 경과 시간(밀리초)입니다.

유형: 타임스탬프

필수 여부: 아니요

#### failureCode

실패 코드입니다.

타입: 문자열

유효 값: InternalServiceError | RobotApplicationCrash | SimulationApplicationCrash | RobotApplicationHealthCheckFailure | SimulationApplicationHealthCheckFailure | BadPermissionsRobotApplication | BadPermissionsSimulationApplication | BadPermissionsS3Object | BadPermissionsS3Output | BadPermissionsCloudwatchLogs | SubnetIpLimitExceeded | ENILimitExceeded | BadPermissionsUserCredentials | InvalidBundleRobotApplication | InvalidBundleSimulationApplication | InvalidS3Resource | ThrottlingError | LimitExceeded | MismatchedEtag | RobotApplicationVersionMismatchedEtag | SimulationApplicationVersionMismatchedEtag | ResourceNotFound | RequestThrottled | BatchTimedOut | BatchCanceled | InvalidInput | WrongRegionS3Bucket | WrongRegionS3Output | WrongRegionRobotApplication | WrongRegionSimulationApplication | UploadContentMismatchError

필수 여부: 아니요

#### failureReason

시뮬레이션 작업 요청의 실패 이유입니다.

타입: 문자열

길이 제약 조건: 최소 길이는 0입니다. 최대 길이는 1024입니다.

패턴: .\*

Required: No

request

시뮬레이션 작업 요청입니다.

유형: [SimulationJobRequest](#) 객체

필수 여부: 아니요

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS SDK for C++](#)
- [AWS Java V2용 SDK](#)
- [AWS 루비 V3용 SDK](#)

## FailureSummary

실패한 월드에 대한 정보입니다.

### 내용

#### failures

실패한 월드입니다.

유형: [WorldFailure](#) 객체 어레이

어레이 멤버: 최소 항목 수 0개. 최대 항목 수 100개.

필수 여부: 아니요

#### totalFailureCount

총 실패 수입니다.

유형: 정수

필수 항목 여부: 아니요

### 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS SDK for C++](#)
- [AWS Java V2용 SDK](#)
- [AWS 루비 V3용 SDK](#)

## Filter

필터에 대한 정보입니다.

### 내용

#### name

필터의 이름.

유형: 문자열

길이 제약: 최소 길이는 1. 최대 길이는 255.

패턴: [a-zA-Z0-9\_\-]\*

Required: No

#### values

값의 목록입니다.

유형: 문자열 어레이

배열 멤버: 고정된 항목 수는 1개입니다.

길이 제약: 최소 길이 1. 최대 길이는 255.

패턴: [a-zA-Z0-9\_\-]\*

필수 여부: 아니요

### 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS SDK for C++](#)
- [AWS Java V2용 SDK](#)
- [AWS 루비 V3용 SDK](#)

## FinishedWorldsSummary

완료한 월드에 대한 정보입니다.

### 내용

#### failureSummary

실패한 월드에 대한 정보입니다.

유형: [FailureSummary](#) 객체

필수 항목 여부: 아니요

#### finishedCount

완료된 월드의 총 수입니다.

유형: 정수

필수 항목 여부: 아니요

#### succeededWorlds

성공한 월드 목록입니다.

유형: 문자열 어레이

배열 멤버: 최소수는 1개입니다. 최대 항목 수는 100개입니다.

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: `arn:.*`

필수 여부: 아니요

### 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS SDK for C++](#)
- [AWS Java V2용 SDK](#)
- [AWS 루비 V3용 SDK](#)



## Fleet

플릿에 대한 정보입니다.

### 내용

arn

플릿의 Amazon 리소스 이름(ARN)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

Required: No

createdAt

플릿이 생성된 epoch 이후 경과 시간(밀리초)입니다.

유형: 타임스탬프

필수 여부: 아니요

lastDeploymentJob

마지막 배포 작업의 Amazon 리소스 이름(ARN)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

Required: No

lastDeploymentStatus

마지막 플릿 배포 상태입니다..

타입: 문자열

유효 값: Pending | Preparing | InProgress | Failed | Succeeded | Canceled

필수 여부: 아니요

lastDeploymentTime

마지막 배포 시간입니다.

유형: 타임스탬프

필수 여부: 아니요

name

플릿의 이름입니다.

유형: 문자열

길이 제약: 최소 길이는 1. 최대 길이는 255.

패턴: [a-zA-Z0-9\_\-]\*

필수 여부: 아니요

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS SDK for C++](#)
- [AWS Java V2용 SDK](#)
- [AWS 루비 V3용 SDK](#)

# LaunchConfig

시작 구성에 대한 정보입니다.

## 내용

### command

General를 RobotSoftwareSuite의 값으로 지정한 경우, 이 필드를 사용하여 컨테이너 이미지에 대한 명령 목록을 지정할 수 있습니다.

SimulationRuntime를 SimulationSoftwareSuite의 값으로 지정한 경우, 이 필드를 사용하여 컨테이너 이미지에 대한 명령 목록을 지정할 수 있습니다.

유형: 문자열 어레이

길이 제약: 최소 길이 1. 최대 길이는 255.

패턴: .+

Required: No

### environmentVariables

애플리케이션 실행을 위한 환경 변수입니다.

유형: 문자열-문자열 맵

맵 항목: 최소 항목 수는 0개입니다. 최대 항목 수는 20개.

키 길이 제약: 최소 길이는 1입니다. 최대 길이 1024.

키 패턴: [A-Z\_][A-Z0-9\_]\*

값 길이 제약 조건: 최소 길이는 1. 최대 길이 1024.

가치 패턴: .\*

필수 여부: 아니요

### launchFile

시작 파일 이름입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1024입니다.

패턴: [a-zA-Z0-9\_.\-]\*

Required: No

#### packageName

패키지 이름입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1024입니다.

패턴: [a-zA-Z0-9\_.\-]\*

Required: No

#### portForwardingConfig

포트 전송 구성입니다.

유형: [PortForwardingConfig](#) 객체

필수 항목 여부: 아니요

#### streamUI

스트리밍 세션을 애플리케이션에 맞게 구성할지 여부를 나타내는 부울입니다. 만약 True, RoboMaker AWS가 연결을 구성하여 시뮬레이션에서 실행 중인 애플리케이션과 상호 작용할 수 있도록 합니다. 구성 요소를 구성하고 시작해야 합니다. 그래픽 사용자 인터페이스가 있어야 합니다.

타입: 부울

필수 항목 여부: 아니요

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS SDK for C++](#)
- [AWS Java V2용 SDK](#)
- [AWS 루비 V3용 SDK](#)



## LoggingConfig

로깅 구성입니다.

### 내용

recordAllRosTopics

이 멤버는 사용 중단되었습니다.

모든 ROS 주제를 기록할지 여부를 나타내는 부울입니다.

#### Important

이 API는 더 이상 지원되지 않으며 사용할 경우 오류가 발생합니다.

타입: 부울

필수 항목 여부: 아니요

### 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS SDK for C++](#)
- [AWS Java V2용 SDK](#)
- [AWS 루비 V3용 SDK](#)

# NetworkInterface

네트워크 인터페이스를 서술합니다.

## 내용

### networkInterfaceId

네트워크 인터페이스의 ID입니다.

타입: 문자열

길이 제약 조건: 최소 길이는 0입니다. 최대 길이는 1024입니다.

패턴: .\*

Required: No

### privateIpAddress

서브넷 내 네트워크 인터페이스의 IPv4 주소입니다.

타입: 문자열

길이 제약 조건: 최소 길이는 0입니다. 최대 길이는 1024입니다.

패턴: .\*

Required: No

### publicIpAddress

네트워크 인터페이스의 퍼블릭 IPv4 주소입니다.

타입: 문자열

길이 제약 조건: 최소 길이는 0입니다. 최대 길이는 1024입니다.

패턴: .\*

필수 여부: 아니요

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS SDK for C++](#)
- [AWS Java V2용 SDK](#)
- [AWS 루비 V3용 SDK](#)

## OutputLocation

출력 위치입니다.

### 내용

#### s3Bucket

출력용 S3 버킷입니다.

타입: 문자열

길이 제약 조건: 최소 길이는 3입니다. 최대 길이는 63입니다.

패턴: [a-z0-9][a-z0-9.\-]\*[a-z0-9]

Required: No

#### s3Prefix

출력 파일이 배치될 s3Bucket의 S3 폴더입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1024입니다.

패턴: .\*

필수 여부: 아니요

### 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS SDK for C++](#)
- [AWS Java V2용 SDK](#)
- [AWS 루비 V3용 SDK](#)

# PortForwardingConfig

포트 전송을 위한 구성 정보입니다.

## 내용

### portMappings

구성에 대한 포트 매핑입니다.

유형: [PortMapping](#) 객체 어레이

배열 멤버: 최소 항목 수는 0개. 최대 항목 수 10개.

필수 여부: 아니요

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS SDK for C++](#)
- [AWS Java V2용 SDK](#)
- [AWS 루비 V3용 SDK](#)

# PortMapping

포트 매핑을 나타내는 객체입니다.

## 내용

### applicationPort

애플리케이션의 포트 번호입니다.

유형: 정수

유효한 범위: 최소값은 1024입니다. 최댓값은 65535입니다.

필수 여부: 예

### jobPort

원격 연결 지점으로 사용하려는 시뮬레이션 작업 인스턴스의 포트 번호입니다.

타입: 정수

유효 범위: 최소값 1. 최댓값은 65535입니다.

필수 여부: 예

### enableOnPublicIp

퍼블릭 IP에서 이 포트 매핑을 활성화할지 여부를 나타내는 부울입니다.

타입: 부울

필수 항목 여부: 아니요

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS SDK for C++](#)
- [AWS Java V2용 SDK](#)
- [AWS 루비 V3용 SDK](#)

## ProgressDetail

배포 작업의 진행 상황에 대한 정보입니다.

### 내용

#### currentProgress

현재 진행 상태입니다.

#### 검증

배포를 검증하는 중입니다.

#### DownloadingExtracting

로봇에서 번들 다운로드 및 추출.

#### ExecutingPreLaunch

시작 전 스크립트 실행(제공된 경우).

#### 실행

로봇 애플리케이션의 실행입니다.

#### ExecutingPostLaunch

시작 후 스크립트 실행(제공된 경우)

#### 완료

배포가 완료되었습니다.

타입: 문자열

유효 값: Validating | DownloadingExtracting | ExecutingDownloadCondition | ExecutingPreLaunch | Launching | ExecutingPostLaunch | Finished

필수 여부: 아니요

#### estimatedTimeRemainingSeconds

단계에서 남아 있는 예상 시간(초). 이는 현재 배포의 Downloading/Extracting 단계에만 적용됩니다. 다른 단계에 대해서는 비어 있습니다.

유형: 정수

필수 항목 여부: 아니요

#### percentDone

수행한 단계의 백분율. 이는 현재 배포의 Downloading/Extracting 단계에만 적용됩니다. 다른 단계에 대해서는 비어 있습니다.

유형: Float

유효한 범위: 최소값은 0.0입니다. 최댓값은 100.0입니다.

필수 여부: 아니요

#### targetResource

배포 작업의 Amazon 리소스 이름(ARN)입니다.

타입: 문자열

길이 제약 조건: 최소 길이는 0입니다. 최대 길이는 1024입니다.

패턴: .\*

필수 여부: 아니요

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS SDK for C++](#)
- [AWS Java V2용 SDK](#)
- [AWS 루비 V3용 SDK](#)

# RenderingEngine

렌더링 엔진에 대한 정보입니다.

## 내용

### name

렌더링 엔진의 이름입니다.

타입: 문자열

유효 값: OGRE

필수 여부: 아니요

### version

렌더링 엔진의 버전입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 4입니다.

패턴: 1.x

필수 여부: 아니요

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS SDK for C++](#)
- [AWS Java V2용 SDK](#)
- [AWS 루비 V3용 SDK](#)

# Robot

로봇에 대한 정보입니다.

## 내용

### architecture

로봇의 아키텍처입니다.

타입: 문자열

유효 값: X86\_64 | ARM64 | ARMHF

필수 여부: 아니요

### arn

로봇의 Amazon 리소스 이름(ARN)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

Required: No

### createdAt

로봇이 생성된 epoch 이후 경과 시간(밀리초)입니다.

유형: 타임스탬프

필수 여부: 아니요

### fleetArn

플릿의 Amazon 리소스 이름(ARN)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

Required: No

greenGrassGroupId

로봇과 연결된 Greengrass 그룹입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: .\*

Required: No

lastDeploymentJob

마지막 배포 작업의 Amazon 리소스 이름(ARN)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

Required: No

lastDeploymentTime

마지막 배포 시간입니다.

유형: 타임스탬프

필수 여부: 아니요

name

로봇의 이름입니다.

유형: 문자열

길이 제약: 최소 길이는 1. 최대 길이는 255.

패턴: [a-zA-Z0-9\_\-]\*

Required: No

## status

로봇의 상태입니다.

타입: 문자열

유효 값: Available | Registered | PendingNewDeployment | Deploying | Failed  
| InSync | NoResponse

필수 여부: 아니요

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS SDK for C++](#)
- [AWS Java V2용 SDK](#)
- [AWS 루비 V3용 SDK](#)

# RobotApplicationConfig

로봇의 애플리케이션 구성 정보입니다.

## 내용

### application

로봇 애플리케이션에 대한 애플리케이션 정보입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: `arn:.*`

필수 사항 여부: Yes

### launchConfig

로봇 애플리케이션에 대한 시작 구성입니다.

유형: [LaunchConfig](#) 객체

필수 여부: 예

### applicationVersion

로봇 애플리케이션의 버전입니다.

유형: 문자열

길이 제약: 최소 길이는 1. 최대 길이는 255.

패턴: `(\\$LATEST)|[0-9]*`

Required: No

### tools

로봇 애플리케이션에 대해 구성된 도구에 대한 정보입니다.

유형: [Tool](#) 객체 어레이

배열 멤버: 최소 항목 수는 0개. 최대 항목 수 10개.

필수 여부: 아니요

## uploadConfigurations

로봇 애플리케이션에 대한 업로드 구성입니다.

유형: [UploadConfiguration](#) 객체 어레이

배열 멤버: 최소 항목 수는 0개. 최대 항목 수 10개.

필수 여부: 아니요

## useDefaultTools

이 멤버는 사용 중단되었습니다.

기본 로봇 애플리케이션 도구를 사용할지 여부를 나타내는 부울입니다. 기본 도구는 rviz, rqt, 터미널, rosbag 레코드입니다. 기본값은 False입니다.

### Important

이 API는 더 이상 지원되지 않으며 사용할 경우 오류가 발생합니다.

타입: 부울

필수 항목 여부: 아니요

## useDefaultUploadConfigurations

이 멤버는 사용 중단되었습니다.

기본 업로드 구성을 사용할지 여부를 나타내는 부울입니다. 기본적으로 애플리케이션이 종료되면 .ros 및 .gazebo 파일이 업로드되며 모든 ROS 주제가 기록됩니다.

이 값을 설정한 경우 outputLocation를 지정해야 합니다.

### Important

이 API는 더 이상 지원되지 않으며 사용할 경우 오류가 발생합니다.

타입: 부울

필수 항목 여부: 아니요

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS SDK for C++](#)
- [AWS Java V2용 SDK](#)
- [AWS 루비 V3용 SDK](#)

## RobotApplicationSummary

로봇 애플리케이션을 위한 요약 정보입니다.

### 내용

#### arn

로봇의 Amazon 리소스 이름(ARN)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

Required: No

#### lastUpdatedAt

로봇 애플리케이션이 마지막으로 업데이트된 epoch 이후 경과 시간(밀리초)입니다.

유형: 타임스탬프

필수 여부: 아니요

#### name

로봇 애플리케이션의 이름입니다.

유형: 문자열

길이 제약: 최소 길이는 1. 최대 길이는 255.

패턴: [a-zA-Z0-9\_\-]\*

Required: No

#### robotSoftwareSuite

로봇 소프트웨어 제품군에 대한 정보입니다.

유형: [RobotSoftwareSuite](#) 객체

필수 항목 여부: 아니요

## version

로봇 애플리케이션의 버전입니다.

유형: 문자열

길이 제약: 최소 길이는 1. 최대 길이는 255.

패턴: ( $\$LATEST$ )|[0-9]\*

필수 여부: 아니요

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS SDK for C++](#)
- [AWS Java V2용 SDK](#)
- [AWS 루비 V3용 SDK](#)

## RobotDeployment

로봇 배포에 대한 정보입니다.

### 내용

arn

로봇의 배포 Amazon 리소스 이름(ARN)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

Required: No

deploymentFinishTime

배포가 완료된 epoch 이후 경과 시간(밀리초)입니다.

유형: 타임스탬프

필수 여부: 아니요

deploymentStartTime

배포가 시작된 epoch 이후 경과 시간(밀리초)입니다.

유형: 타임스탬프

필수 여부: 아니요

failureCode

로봇 배포 실패 코드입니다.

타입: 문자열

유효 값: ResourceNotFound | EnvironmentSetupError | EtagMismatch  
| FailureThresholdBreached | RobotDeploymentAborted |  
RobotDeploymentNoResponse | RobotAgentConnectionTimeout  
| GreengrassDeploymentFailed | InvalidGreengrassGroup |

MissingRobotArchitecture | MissingRobotApplicationArchitecture |  
 MissingRobotDeploymentResource | GreengrassGroupVersionDoesNotExist  
 | LambdaDeleted | ExtractingBundleFailure | PreLaunchFileFailure |  
 PostLaunchFileFailure | BadPermissionError | DownloadConditionFailed |  
 BadLambdaAssociated | InternalServerError | RobotApplicationDoesNotExist  
 | DeploymentFleetDoesNotExist | FleetDeploymentTimeout

필수 여부: 아니요

#### failureReason

로봇 배포가 실패한 이유에 대한 간략한 설명입니다.

타입: 문자열

길이 제약 조건: 최소 길이는 0입니다. 최대 길이는 1024입니다.

패턴: .\*

Required: No

#### progressDetail

배포 진행 상황 대한 자세한 내용입니다.

유형: [ProgressDetail](#) 객체

필수 항목 여부: 아니요

#### status

로봇의 배포 상태입니다.

타입: 문자열

유효 값: Available | Registered | PendingNewDeployment | Deploying | Failed  
 | InSync | NoResponse

필수 여부: 아니요

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS SDK for C++](#)
- [AWS Java V2용 SDK](#)
- [AWS 루비 V3용 SDK](#)

# RobotSoftwareSuite

로봇 소프트웨어 제품군에 대한 정보입니다.

## 내용

### name

로봇 소프트웨어 제품군의 이름입니다. General만 지원되는 값입니다.

타입: 문자열

유효 값: ROS | ROS2 | General

필수 여부: 아니요

### version

로봇 소프트웨어 제품군의 버전입니다. 일반 소프트웨어 제품군에는 적용되지 않습니다.

타입: 문자열

유효 값: Kinetic | Melodic | Dashing | Foxy

필수 여부: 아니요

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS SDK for C++](#)
- [AWS Java V2용 SDK](#)
- [AWS 루비 V3용 SDK](#)

## S3KeyOutput

S3 키에 대한 정보입니다.

### 내용

#### etag

객체에 대한 e태그입니다.

타입: 문자열

필수사항: 아니요

#### s3Key

S3 키입니다.

타입: 문자열

길이 제약 조건: 최소 길이는 0입니다. 최대 길이는 1024입니다.

패턴: .\*

필수 여부: 아니요

### 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS SDK for C++](#)
- [AWS Java V2용 SDK](#)
- [AWS 루비 V3용 SDK](#)

## S3Object

S3 객체에 대한 정보입니다.

### 내용

#### bucket

객체를 포함하는 버킷입니다.

타입: 문자열

길이 제약 조건: 최소 길이는 3입니다. 최대 길이는 63입니다.

패턴: [a-z0-9][a-z0-9.\-]\*[a-z0-9]

필수 사항 여부: Yes

#### key

객체의 키입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1024입니다.

패턴: .\*

필수 사항 여부: Yes

#### etag

객체의 Etag입니다.

타입: 문자열

필수 항목 여부: 아니요

### 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS SDK for C++](#)

- [AWS Java V2용 SDK](#)
- [AWS 루비 V3용 SDK](#)

## SimulationApplicationConfig

시뮬레이션 애플리케이션 구성에 대한 정보입니다.

### 내용

#### application

시뮬레이션 애플리케이션에 대한 애플리케이션 정보입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: `arn:.*`

필수 사항 여부: Yes

#### launchConfig

시뮬레이션 애플리케이션에 대한 시작 구성입니다.

유형: [LaunchConfig](#) 객체

필수 여부: 예

#### applicationVersion

시뮬레이션 애플리케이션 버전입니다.

유형: 문자열

길이 제약: 최소 길이는 1. 최대 길이는 255.

패턴: `(\$\{LATEST\})|[0-9]*`

Required: No

#### tools

시뮬레이션 애플리케이션에 대해 구성된 도구에 대한 정보입니다.

유형: [Tool](#) 객체 어레이

배열 멤버: 최소 항목 수는 0개. 최대 항목 수 10개.

필수 여부: 아니요

## uploadConfigurations

시뮬레이션 애플리케이션에 대한 업로드 구성 정보입니다.

유형: [UploadConfiguration](#) 객체 어레이

배열 멤버: 최소 항목 수는 0개. 최대 항목 수 10개.

필수 여부: 아니요

## useDefaultTools

이 멤버는 사용 중단되었습니다.

기본 시뮬레이션 애플리케이션 도구를 사용할지 여부를 나타내는 부울입니다. 기본 도구는 rviz, rqt, 터미널, rosbag 레코드입니다. 기본값은 False입니다.

### Important

이 API는 더 이상 지원되지 않으며 사용할 경우 오류가 발생합니다.

타입: 부울

필수 항목 여부: 아니요

## useDefaultUploadConfigurations

이 멤버는 사용 중단되었습니다.

기본 업로드 구성을 사용할지 여부를 나타내는 부울입니다. 기본적으로 애플리케이션이 종료되면 .ros 및 .gazebo 파일이 업로드되며 모든 ROS 주제가 기록됩니다.

이 값을 설정한 경우 outputLocation를 지정해야 합니다.

### Important

이 API는 더 이상 지원되지 않으며 사용할 경우 오류가 발생합니다.

타입: 부울

필수 항목 여부: 아니요

worldConfigs

월드 구성 목록입니다.

 Important

이 API는 더 이상 지원되지 않으며 사용할 경우 오류가 발생합니다.

유형: [WorldConfig](#) 객체 어레이

어레이 멤버: 최소 항목 수 0개. 최대 항목 수는 1개입니다.

필수 여부: 아니요

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS SDK for C++](#)
- [AWS Java V2용 SDK](#)
- [AWS 루비 V3용 SDK](#)

## SimulationApplicationSummary

시뮬레이션 애플리케이션에 대한 요약 정보입니다.

### 내용

#### arn

시뮬레이션 애플리케이션의 Amazon 리소스 이름(ARN)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

Required: No

#### lastUpdatedAt

시뮬레이션 애플리케이션이 마지막으로 업데이트된 epoch 이후 경과 시간(밀리초).

유형: 타임스탬프

필수 여부: 아니요

#### name

시뮬레이션 애플리케이션의 이름입니다.

유형: 문자열

길이 제약: 최소 길이는 1. 최대 길이는 255.

패턴: [a-zA-Z0-9\_\-]\*

Required: No

#### robotSoftwareSuite

로봇 소프트웨어 제품군에 대한 정보입니다.

유형: [RobotSoftwareSuite](#) 객체

필수 항목 여부: 아니요

## simulationSoftwareSuite

시뮬레이션 소프트웨어 제품군에 대한 정보입니다.

유형: [SimulationSoftwareSuite](#) 객체

필수 항목 여부: 아니요

## version

시뮬레이션 애플리케이션 버전입니다.

유형: 문자열

길이 제약: 최소 길이는 1. 최대 길이는 255.

패턴: (\\$LATEST)|[0-9]\*

필수 여부: 아니요

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS SDK for C++](#)
- [AWS Java V2용 SDK](#)
- [AWS 루비 V3용 SDK](#)

## SimulationJob

시뮬레이션 작업에 대한 정보입니다.

### 내용

#### arn

시뮬레이션 작업의 Amazon 리소스 이름(ARN)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

Required: No

#### clientRequestToken

이 SimulationJob 요청에 대한 고유 식별자입니다.

유형: 문자열

길이 제한: 최소 길이는 1. 최대 길이는 64.

패턴: [a-zA-Z0-9\_\-=]\*

Required: No

#### compute

시뮬레이션 작업에 대한 컴퓨팅 정보입니다.

유형: [ComputeResponse](#) 객체

필수 항목 여부: 아니요

#### dataSources

시뮬레이션 작업의 데이터 소스입니다.

타입: [DataSource](#) 객체 배열

필수: 아니요

## failureBehavior

시뮬레이션 작업의 실패 동작입니다.

계속

4XX 오류 코드가 발생한 후 최대 제한 시간 동안 호스트를 계속 실행합니다.

Fail

시뮬레이션 작업을 중지하고 인스턴스를 종료합니다.

타입: 문자열

유효 값: Fail | Continue

필수 여부: 아니요

## failureCode

시뮬레이션 작업이 실패한 경우의 실패 코드입니다.

타입: 문자열

유효 값: InternalServiceError | RobotApplicationCrash | SimulationApplicationCrash | RobotApplicationHealthCheckFailure | SimulationApplicationHealthCheckFailure | BadPermissionsRobotApplication | BadPermissionsSimulationApplication | BadPermissionsS3Object | BadPermissionsS3Output | BadPermissionsCloudwatchLogs | SubnetIpLimitExceeded | ENILimitExceeded | BadPermissionsUserCredentials | InvalidBundleRobotApplication | InvalidBundleSimulationApplication | InvalidS3Resource | ThrottlingError | LimitExceeded | MismatchedEtag | RobotApplicationVersionMismatchedEtag | SimulationApplicationVersionMismatchedEtag | ResourceNotFound | RequestThrottled | BatchTimedOut | BatchCanceled | InvalidInput | WrongRegionS3Bucket | WrongRegionS3Output | WrongRegionRobotApplication | WrongRegionSimulationApplication | UploadContentMismatchError

필수 여부: 아니요

## failureReason

시뮬레이션 작업이 실패한 이유입니다.

타입: 문자열

길이 제약 조건: 최소 길이는 0입니다. 최대 길이는 1024입니다.

패턴: .\*

Required: No

#### iamRole

시뮬레이션 인스턴스가 사용자 대신 연결된 정책에 지정된 AWS API를 호출하도록 허용하는 IAM 역할입니다. 이것이 시뮬레이션 작업에 보안 자격 증명이 전달되는 과정입니다.

유형: 문자열

길이 제약: 최소 길이는 1. 최대 길이는 255.

패턴: arn:aws:iam::\w+:role/.\*

Required: No

#### lastStartedAt

물레이션 작업이 마지막으로 시작된 epoch 이후 경과 시간(밀리초)

유형: 타임스탬프

필수 여부: 아니요

#### lastUpdatedAt

시뮬레이션 작업이 마지막으로 업데이트된 epoch 이후 경과 시간(밀리초)입니다.

유형: 타임스탬프

필수 여부: 아니요

#### loggingConfig

로깅 구성입니다.

유형: [LoggingConfig](#) 객체

필수 항목 여부: 아니요

#### maxJobDurationInSeconds

초 단위 최대 시뮬레이션 작업 시간입니다. 값은 8일(691,200초) 이하여야 합니다.

유형: Long

필수 여부: 아니요

#### name

시뮬레이션 작업의 이름입니다.

유형: 문자열

길이 제약: 최소 길이는 1. 최대 길이는 255.

패턴: [a-zA-Z0-9\_\-]\*

Required: No

#### networkInterface

네트워크 인터페이스 정보입니다.

유형: [NetworkInterface](#) 객체

필수 항목 여부: 아니요

#### outputLocation

시뮬레이션 작업에서 생성된 출력 파일의 위치입니다.

유형: [OutputLocation](#) 객체

필수 항목 여부: 아니요

#### robotApplications

로봇 애플리케이션 목록입니다.

유형: [RobotApplicationConfig](#) 객체 어레이

배열 멤버: 고정된 항목 수는 1개입니다.

필수 여부: 아니요

#### simulationApplications

시뮬레이션 애플리케이션 목록입니다.

유형: [SimulationApplicationConfig](#) 객체 어레이

배열 멤버: 고정된 항목 수는 1개입니다.

필수 여부: 아니요

#### simulationTimeMillis

시뮬레이션 작업 실행 기간(밀리초)

유형: Long

필수 여부: 아니요

#### status

시뮬레이션 작업의 상태입니다.

타입: 문자열

유효 값: Pending | Preparing | Running | Restarting | Completed | Failed | RunningFailed | Terminating | Terminated | Canceled

필수 여부: 아니요

#### tags

시뮬레이션 작업에 연결된 태그 키 및 태그 값을 포함하는 맵입니다.

유형: 문자열-문자열 맵

맵 항목: 최소 항목 수는 0개입니다. 최대 항목 수는 50개입니다.

키 길이 제약 조건: 최소 길이는 1. 최대 길이 128.

키 패턴: [a-zA-Z0-9 \_.\-\/+=:]\*

값 길이 제약 조건: 최소 길이는 0입니다. 최대 길이는 256.

값 패턴: [a-zA-Z0-9 \_.\-\/+=:]\*

필수 여부: 아니요

#### vpcConfig

VPC 구성 정보입니다.

유형: [VPCConfigResponse](#) 객체

필수 여부: 아니요

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS SDK for C++](#)
- [AWS Java V2용 SDK](#)
- [AWS 루비 V3용 SDK](#)

## SimulationJobBatchSummary

시뮬레이션 작업 배치에 대한 정보입니다.

### 내용

arn

배치의 Amazon 리소스 이름(ARN)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

Required: No

createdAt

시뮬레이션 작업 배치가 마지막으로 생성된 epoch 이후 경과 시간(밀리초)입니다.

유형: 타임스탬프

필수 여부: 아니요

createdRequestCount

생성된 시뮬레이션 작업 요청 수입니다.

유형: 정수

필수 항목 여부: 아니요

failedRequestCount

실패한 시뮬레이션 작업 요청 수입니다.

유형: 정수

필수 항목 여부: 아니요

lastUpdatedAt

시뮬레이션 작업 배치가 마지막으로 업데이트된 epoch 이후 경과 시간(밀리초)입니다.

유형: 타임스탬프

필수 여부: 아니요

pendingRequestCount

보류 중인 시뮬레이션 작업 요청 수입니다.

유형: 정수

필수 항목 여부: 아니요

status

시뮬레이션 작업 배치의 상태입니다.

보류중

시뮬레이션 작업 배치 요청이 보류 중입니다.

InProgress

시뮬레이션 작업 배치가 진행 중입니다.

Failed

시뮬레이션 작업 배치가 실패했습니다. 내부 장애(예: `InternalServerError`)로 인해 하나 이상의 시뮬레이션 작업 요청을 완료할 수 없습니다. 자세한 내용은 `failureCode` 및 `failureReason` 섹션을 참조하세요.

완료됨

시뮬레이션 배치 작업이 완료되었습니다. 다음의 경우 배치가 완료됩니다. (1) 배치에 보류 중인 시뮬레이션 작업 요청이 없고 실패한 시뮬레이션 작업 요청 모두가 `InternalServerError`로 인한 것이 아닌 경우, (2) 생성된 모든 시뮬레이션 작업이 최종 상태(예: `Completed` 또는 `Failed`)에 도달한 경우.

취소됨

시뮬레이션 배치 작업이 취소되었습니다.

취소 중

시뮬레이션 배치 작업을 취소 중입니다.

완료 중

시뮬레이션 배치 작업을 완료 중입니다.

## TimingOut

시뮬레이션 작업 배치 시간이 초과되었습니다.

배치 시간이 초과되고 내부 장애(예: `InternalServerError`)로 인해 실패한 보류 중인 요청이 있는 경우, 배치 상태는 `Failed`입니다. 이러한 실패한 요청이 없는 경우 배치 상태는 `TimedOut`입니다.

## TimedOut

시뮬레이션 배치 작업 시간이 초과되었습니다.

타입: 문자열

유효 값: `Pending` | `InProgress` | `Failed` | `Completed` | `Canceled` | `Canceling` | `Completing` | `TimingOut` | `TimedOut`

필수 여부: 아니요

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS SDK for C++](#)
- [AWS Java V2용 SDK](#)
- [AWS 루비 V3용 SDK](#)

## SimulationJobRequest

시뮬레이션 작업 요청에 대한 정보입니다.

### 내용

#### maxJobDurationInSeconds

초 단위 최대 시뮬레이션 작업 시간입니다. 값은 8일(691,200초) 이하여야 합니다.

타입: Long

필수 여부: 예

#### compute

시뮬레이션 작업에 대한 컴퓨팅 정보입니다.

유형: [Compute](#) 객체

필수 항목 여부: 아니요

#### dataSources

S3의 읽기 전용 파일을 시뮬레이션에 마운트할 데이터 소스를 지정합니다. 이러한 파일은 /opt/robomaker/datasources/data\_source\_name에서 사용할 수 있습니다.

#### Note

파일 수는 100개로 제한되며 모든 DataSourceConfig 객체의 총 크기는 25GB입니다.

유형: [DataSourceConfig](#) 객체 어레이

어레이 멤버: 최소 항목 수 1개. 최대 항목 수는 6개입니다.

필수 여부: 아니요

#### failureBehavior

시뮬레이션 작업의 실패 동작입니다.

계속

4XX 오류 코드가 발생한 후 최대 제한 시간 동안 호스트를 계속 실행합니다.

## Fail

시뮬레이션 작업을 중지하고 인스턴스를 종료합니다.

타입: 문자열

유효 값: Fail | Continue

필수 여부: 아니요

## iamRole

시뮬레이션 인스턴스가 사용자 대신 연결된 정책에 지정된 AWS API를 호출하도록 허용하는 IAM 역할 이름입니다. 이것이 시뮬레이션 작업에 보안 자격 증명이 전달되는 과정입니다.

유형: 문자열

길이 제약: 최소 길이는 1. 최대 길이는 255.

패턴: arn:aws:iam::\w+:role/.\*

Required: No

## loggingConfig

로깅 구성입니다.

유형: [LoggingConfig](#) 객체

필수 항목 여부: 아니요

## outputLocation

출력 위치입니다.

유형: [OutputLocation](#) 객체

필수 항목 여부: 아니요

## robotApplications

시뮬레이션 작업에 사용할 로봇 애플리케이션입니다.

유형: [RobotApplicationConfig](#) 객체 어레이

배열 멤버: 고정된 항목 수는 1개입니다.

필수 여부: 아니요

## simulationApplications

시뮬레이션 작업에서 사용할 시뮬레이션 애플리케이션입니다.

유형: [SimulationApplicationConfig](#) 객체 어레이

배열 멤버: 고정된 항목 수는 1개입니다.

필수 여부: 아니요

## tags

시뮬레이션 작업 요청에 연결된 태그 키 및 태그 값을 포함하는 맵입니다.

유형: 문자열-문자열 맵

맵 항목: 최소 항목 수는 0개입니다. 최대 항목 수는 50개입니다.

키 길이 제약 조건: 최소 길이는 1. 최대 길이 128.

키 패턴: [a-zA-Z0-9 \_.\-\/+=:]\*

값 길이 제약 조건: 최소 길이는 0입니다. 최대 길이는 256.

값 패턴: [a-zA-Z0-9 \_.\-\/+=:]\*

필수 여부: 아니요

## useDefaultApplications

시뮬레이션 작업에서 기본 애플리케이션을 사용할지 여부를 나타내는 부울입니다. 기본 애플리케이션에는 Gazebo, rqt, rviz, 터미널 액세스가 포함됩니다.

타입: 부울

필수 항목 여부: 아니요

## vpcConfig

시뮬레이션 작업이 VPC의 리소스에 액세스하는 경우 보안 그룹 ID 및 서브넷 ID 목록을 식별하는 이 파라미터를 제공합니다. 동일한 VPC에 속해야 합니다. 보안 그룹 하나와 서브넷 ID 두 개를 제공해야 합니다.

유형: [VPCConfig](#) 객체

필수 여부: 아니요

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS SDK for C++](#)
- [AWS Java V2용 SDK](#)
- [AWS 루비 V3용 SDK](#)

## SimulationJobSummary

시뮬레이션 작업에 대한 요약 정보입니다.

### 내용

#### arn

시뮬레이션 작업의 Amazon 리소스 이름(ARN)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

Required: No

#### computeType

시뮬레이션 작업 요약의 컴퓨팅 유형입니다.

타입: 문자열

유효 값: CPU | GPU\_AND\_CPU

필수 여부: 아니요

#### dataSourceNames

데이터 원본의 이름입니다.

유형: 문자열 어레이

길이 제약: 최소 길이 1. 최대 길이는 255.

패턴: [a-zA-Z0-9\_\-]\*

Required: No

#### lastUpdatedAt

시뮬레이션 작업이 마지막으로 업데이트된 epoch 이후 경과 시간(밀리초)입니다.

유형: 타임스탬프

필수 여부: 아니요

#### name

시뮬레이션 작업의 이름입니다.

유형: 문자열

길이 제약: 최소 길이는 1. 최대 길이는 255.

패턴: [a-zA-Z0-9\_\-]\*

Required: No

#### robotApplicationNames

시뮬레이션 작업 로봇 애플리케이션 이름의 목록입니다.

유형: 문자열 어레이

길이 제약: 최소 길이 1. 최대 길이는 255.

패턴: [a-zA-Z0-9\_\-]\*

Required: No

#### simulationApplicationNames

시뮬레이션 작업 시뮬레이션 애플리케이션 이름의 목록입니다.

유형: 문자열 어레이

길이 제약: 최소 길이 1. 최대 길이는 255.

패턴: [a-zA-Z0-9\_\-]\*

Required: No

#### status

시뮬레이션 작업의 상태입니다.

타입: 문자열

유효 값: Pending | Preparing | Running | Restarting | Completed | Failed | RunningFailed | Terminating | Terminated | Canceled

필수 여부: 아니요

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS SDK for C++](#)
- [AWS Java V2용 SDK](#)
- [AWS 루비 V3용 SDK](#)

# SimulationSoftwareSuite

시뮬레이션 소프트웨어 제품군에 대한 정보입니다.

## 내용

### name

시뮬레이션 소프트웨어 제품군의 이름입니다. SimulationRuntime만 지원되는 값입니다.

타입: 문자열

유효 값: Gazebo | RosbagPlay | SimulationRuntime

필수 여부: 아니요

### version

시뮬레이션 소프트웨어 제품군의 버전입니다. SimulationRuntime에는 해당되지 않습니다.

타입: 문자열

길이 제약 조건: 최소 길이는 0입니다. 최대 길이는 1024입니다.

패턴: 7|9|11|Kinetic|Melodic|Dashing|Foxy

필수 여부: 아니요

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS SDK for C++](#)
- [AWS Java V2용 SDK](#)
- [AWS 루비 V3용 SDK](#)

## Source

소스에 대한 정보입니다.

### 내용

#### architecture

애플리케이션의 대상 프로세서 아키텍처입니다.

타입: 문자열

유효 값: X86\_64 | ARM64 | ARMHF

필수 여부: 아니요

#### etag

s3Bucket 및 s3Key로 지정된 객체의 해시입니다.

타입: 문자열

필수사항: 아니요

#### s3Bucket

S3 버킷 이름입니다.

타입: 문자열

길이 제약 조건: 최소 길이는 3입니다. 최대 길이는 63입니다.

패턴: [a-z0-9][a-z0-9.\-]\*[a-z0-9]

Required: No

#### s3Key

s3 객체 키입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1024입니다.

패턴: .\*

필수 여부: 아니요

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS SDK for C++](#)
- [AWS Java V2용 SDK](#)
- [AWS 루비 V3용 SDK](#)

## SourceConfig

소스 구성에 대한 정보입니다.

### 내용

#### architecture

애플리케이션의 대상 프로세서 아키텍처입니다.

타입: 문자열

유효 값: X86\_64 | ARM64 | ARMHF

필수 여부: 아니요

#### s3Bucket

Amazon S3 버킷 이름입니다.

타입: 문자열

길이 제약 조건: 최소 길이는 3입니다. 최대 길이는 63입니다.

패턴: [a-z0-9][a-z0-9.\-]\*[a-z0-9]

Required: No

#### s3Key

s3 객체 키입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1024입니다.

패턴: .\*

필수 여부: 아니요

### 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS SDK for C++](#)
- [AWS Java V2용 SDK](#)
- [AWS 루비 V3용 SDK](#)

## TemplateLocation

템플릿 위치에 대한 정보입니다.

### 내용

#### s3Bucket

Amazon S3 버킷 이름입니다.

타입: 문자열

길이 제약 조건: 최소 길이는 3입니다. 최대 길이는 63입니다.

패턴: [a-z0-9][a-z0-9.\-]\*[a-z0-9]

필수 사항 여부: Yes

#### s3Key

데이터 소스 파일을 식별하는 S3 키 목록입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1024입니다.

패턴: .\*

필수 여부: 예

### 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS SDK for C++](#)
- [AWS Java V2용 SDK](#)
- [AWS 루비 V3용 SDK](#)

## TemplateSummary

템플릿에 대한 요약 정보입니다.

### 내용

#### arn

템플릿의 Amazon 리소스 이름(ARN)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

Required: No

#### createdAt

템플릿이 생성된 epoch 이후 경과 시간(밀리초)입니다.

유형: 타임스탬프

필수 여부: 아니요

#### lastUpdatedAt

템플릿이 마지막으로 업데이트된 epoch 이후 경과 시간(밀리초)입니다.

유형: 타임스탬프

필수 여부: 아니요

#### name

템플릿의 이름입니다.

타입: 문자열

길이 제약 조건: 최소 길이는 0입니다. 최대 길이는 255.

패턴: .\*

Required: No

## version

사용 중인 템플릿의 버전입니다.

타입: 문자열

길이 제약 조건: 최소 길이는 0입니다. 최대 길이는 1024입니다.

패턴: .\*

필수 여부: 아니요

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS SDK for C++](#)
- [AWS Java V2용 SDK](#)
- [AWS 루비 V3용 SDK](#)

## Tool

도구에 대한 정보입니다. 도구는 시뮬레이션 작업에 사용됩니다.

### 내용

#### command

도구의 명령줄 인수입니다. 도구 실행 파일 이름을 포함해야 합니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1024입니다.

패턴: .\*

필수 사항 여부: Yes

#### name

도구의 이름입니다.

유형: 문자열

길이 제약: 최소 길이는 1. 최대 길이는 255.

패턴: [a-zA-Z0-9\_\-]\*

필수 사항 여부: Yes

#### exitBehavior

종료 동작에 따라 도구 실행이 중단될 때 어떤 일이 발생하는지가 결정됩니다. RESTART 시 도구가 다시 시작됩니다. FAIL 시 작업이 종료됩니다. 기본값은 RESTART입니다.

타입: 문자열

유효 값: FAIL | RESTART

필수 여부: 아니요

#### streamOutputToCloudWatch

도구에 CloudWatch 대한 로그를 기록할지 여부를 나타내는 Boolean입니다. 기본값은 False입니다.

타입: 부울

필수 항목 여부: 아니요

## streamUI

도구에 대해 스트리밍 세션을 구성할지 여부를 나타내는 부울입니다. 만약 True, RoboMaker AWS가 연결을 구성하여 시뮬레이션에서 실행 중인 도구와 상호 작용할 수 있도록 합니다. 그래픽 사용자 인터페이스가 있어야 합니다. 기본값은 False입니다.

타입: 부울

필수 항목 여부: 아니요

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS SDK for C++](#)
- [AWS Java V2용 SDK](#)
- [AWS 루비 V3용 SDK](#)

## UploadConfiguration

업로드 구성 정보를 제공합니다. 시뮬레이션 작업에서 지정한 위치로 파일이 업로드됩니다.

### 내용

#### name

Amazon S3에서 파일을 업로드할 위치를 지정하는 접두사입니다. 최종 경로를 결정하기 위해 시뮬레이션 출력 위치에 추가됩니다.

예를 들어 시뮬레이션 출력 위치가 `s3://my-bucket`이고 업로드 구성 이름이 `robot-test`인 경우 파일이 `s3://my-bucket/<simid>/<runid>/robot-test`에 업로드됩니다.

유형: 문자열

길이 제약: 최소 길이는 1. 최대 길이는 255.

패턴: `[a-zA-Z0-9_\-]*`

필수 사항 여부: Yes

#### path

업로드할 파일의 경로를 지정합니다. 표준 Unix 글로브 일치 규칙이 허용되며 `**`를 '슈퍼 별표'로 추가합니다. 예를 들어 `/var/log/**/*.log`를 지정하면 `/var/log` 디렉터리 트리의 모든 `.log` 파일이 수집됩니다. 보다 자세한 예는 [Glob Library](#)를 참조하세요.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1024입니다.

패턴: `.*`

필수 사항 여부: Yes

#### uploadBehavior

파일을 업로드할 시기를 지정합니다.

`UPLOAD_ON_TERMINATE`

시뮬레이션이 `TERMINATING` 상태에 들어가면 일치하는 파일이 업로드됩니다. 일치하는 파일은 모든 코드(도구 포함)가 중지될 때까지 업로드되지 않습니다.

파일을 업로드하는 데 문제가 있는 경우 업로드가 다시 시도됩니다. 문제가 지속되면 더 이상 업로드가 시도되지 않습니다.

#### UPLOAD\_ROLLING\_AUTO\_REMOVE

일치하는 파일은 생성되는 대로 업로드됩니다. 업로드된 후에는 삭제됩니다. 5초마다 지정한 경로를 확인합니다. 모든 코드(도구 포함)가 중지되면 최종 확인이 이루어집니다.

타입: 문자열

유효 값: UPLOAD\_ON\_TERMINATE | UPLOAD\_ROLLING\_AUTO\_REMOVE

필수 여부: 예

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS SDK for C++](#)
- [AWS Java V2용 SDK](#)
- [AWS 루비 V3용 SDK](#)

## VPCConfig

시뮬레이션 작업이 VPC의 리소스에 액세스하는 경우 보안 그룹 ID 및 서브넷 ID 목록을 식별하는 이 파라미터를 제공합니다. 동일한 VPC에 속해야 합니다. 보안 그룹 하나와 서브넷 ID 두 개를 제공해야 합니다.

### 내용

#### subnets

VPC에서 서브넷 ID 하나 이상의 목록입니다.

유형: 문자열 어레이

배열 멤버: 최소 항목 수는 1개입니다. 최대 항목 수는 16개입니다.

길이 제약: 최소 길이 1. 최대 길이는 255.

패턴: .+

필수 사항 여부: Yes

#### assignPublicIp

퍼블릭 IP 주소 할당 여부를 나타내는 부울입니다.

타입: 부울

필수 항목 여부: 아니요

#### securityGroups

해당 VPC의 한 개 이상 보안 그룹 ID의 목록입니다.

유형: 문자열 어레이

배열 구성원: 최소수는 1개입니다. 최대 항목 수는 5개.

길이 제약: 최소 길이 1. 최대 길이는 255.

패턴: .+

필수 여부: 아니요

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS SDK for C++](#)
- [AWS Java V2용 SDK](#)
- [AWS 루비 V3용 SDK](#)

## VPCConfigResponse

시뮬레이션 작업과 관련된 VPC 구성입니다.

### 내용

#### assignPublicIp

퍼블릭 IP가 할당되었는지 여부를 나타내는 부울입니다.

타입: 부울

필수 항목 여부: 아니요

#### securityGroups

시뮬레이션 작업과 연결된 보안 그룹 ID의 목록입니다.

유형: 문자열 어레이

배열 구성원: 최소수는 1개입니다. 최대 항목 수는 5개.

길이 제약: 최소 길이 1. 최대 길이는 255.

패턴: .+

Required: No

#### subnets

시뮬레이션 작업과 연결된 서브넷 ID의 목록입니다.

유형: 문자열 어레이

배열 멤버: 최소 항목 수는 1개입니다. 최대 항목 수는 16개입니다.

길이 제약: 최소 길이 1. 최대 길이는 255.

패턴: .+

Required: No

#### vpclId

시뮬레이션 작업과 관련된 VPC ID입니다.

타입: 문자열

길이 제약 조건: 최소 길이는 0입니다. 최대 길이는 1024입니다.

패턴: .\*

필수 여부: 아니요

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS SDK for C++](#)
- [AWS Java V2용 SDK](#)
- [AWS 루비 V3용 SDK](#)

# WorldConfig

월드에 대한 구성 정보입니다.

## 내용

world

시뮬레이션을 통해 생성된 세계 WorldForge.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

필수 여부: 아니요

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS SDK for C++](#)
- [AWS Java V2용 SDK](#)
- [AWS 루비 V3용 SDK](#)

## WorldCount

생성할 월드 수입니다. 각 평면도별로 고유한 평면도 수와 고유한 인테리어 수를 구성할 수 있습니다. 예를 들어 20개의 고유한 인테리어가 있는 하나의 월드를 원하는 경우 `floorplanCount = 1` 및 `interiorCountPerFloorplan = 20`을 설정합니다. 그러면 20개의 월드가 생성됩니다 (`floorplanCount * interiorCountPerFloorplan`).

`floorplanCount = 4`와 `interiorCountPerFloorplan = 5`를 설정하면 5개의 고유한 평면도를 가진 20개의 월드가 생깁니다.

### 내용

#### `floorplanCount`

고유한 평면도 수입니다.

유형: 정수

필수 항목 여부: 아니요

#### `interiorCountPerFloorplan`

평면도별 고유한 인테리어 수입니다.

유형: 정수

필수 항목 여부: 아니요

### 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS SDK for C++](#)
- [AWS Java V2용 SDK](#)
- [AWS 루비 V3용 SDK](#)

## WorldExportJobSummary

월드 내보내기 작업에 대한 정보입니다.

### 내용

#### arn

월드 내보내기 작업 Amazon 리소스 이름(ARN)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

Required: No

#### createdAt

월드 내보내기 작업이 생성된 epoch 이후 경과 시간(밀리초)입니다.

유형: 타임스탬프

필수 여부: 아니요

#### outputLocation

출력 위치입니다.

유형: [OutputLocation](#) 객체

필수 항목 여부: 아니요

#### status

월드 내보내기 작업의 상태입니다.

보류 중

월드 내보내기 작업 요청이 보류 중입니다.

실행 중

월드 내보내기 작업이 실행 중입니다.

## 완료됨

월드 내보내기 작업이 완료되었습니다.

## Failed

월드 내보내기 작업이 실패했습니다. 자세한 정보는 `failureCode`을 참조하세요.

## 취소됨

월드 내보내기 작업이 취소되었습니다.

## 취소 중

월드 내보내기 작업을 취소 중입니다.

타입: 문자열

유효 값: Pending | Running | Completed | Failed | Canceling | Canceled

필수 여부: 아니요

## worlds

월드 목록입니다.

유형: 문자열 어레이

배열 멤버: 최소수는 1개입니다. 최대 항목 수는 100개입니다.

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: `arn:.*`

필수 여부: 아니요

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS SDK for C++](#)
- [AWS Java V2용 SDK](#)
- [AWS 루비 V3용 SDK](#)

## WorldFailure

실패한 월드에 대한 정보입니다.

### 내용

#### failureCode

월드 내보내기 작업이 실패한 경우의 실패 코드:

`InternalServerError`

내부 서비스 오류.

`LimitExceeded`

요청된 리소스가 허용된 최대 수를 초과하거나 동시 스트림 요청 수가 허용된 최대 수를 초과합니다.

`ResourceNotFound`

특정 리소스를 찾을 수 없습니다.

`RequestThrottled`

요청에 병목 현상이 발생했습니다.

`InvalidInput`

요청의 입력 파라미터가 유효하지 않습니다.

타입: 문자열

유효 값: `InternalServerError` | `LimitExceeded` | `ResourceNotFound` | `RequestThrottled` | `InvalidInput` | `AllWorldGenerationFailed`

필수 여부: 아니요

#### failureCount

실패한 월드 수입니다.

유형: 정수

필수 항목 여부: 아니요

## sampleFailureReason

월드가 실패한 이유의 예시입니다. 월드 오류가 집계됩니다. 샘플은 sampleFailureReason으로 사용됩니다.

타입: 문자열

길이 제약 조건: 최소 길이는 0입니다. 최대 길이는 1024입니다.

패턴: .\*

필수 여부: 아니요

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS SDK for C++](#)
- [AWS Java V2용 SDK](#)
- [AWS 루비 V3용 SDK](#)

## WorldGenerationJobSummary

월드 생성 작업에 대한 정보입니다.

### 내용

#### arn

월드 생성기 작업의 Amazon 리소스 이름(ARN)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

Required: No

#### createdAt

월드 생성 작업이 생성된 epoch 이후 경과 시간(밀리초)입니다.

유형: 타임스탬프

필수 여부: 아니요

#### failedWorldCount

실패한 월드 수입니다.

유형: 정수

필수 항목 여부: 아니요

#### status

월드 생성기 작업의 상태입니다.

보류중

월드 생성기 작업 요청이 보류 중입니다.

실행 중

월드 생성기 작업이 실행 중입니다.

## 완료됨

월드 생성기 작업이 완료되었습니다.

## Failed

월드 생성기 작업이 실패했습니다. 자세한 정보는 `failureCode`을 참조하세요.

## PartialFailed

일부 월드가 생성되지 않았습니다.

## 취소됨

월드 생성기 작업이 취소되었습니다.

## 취소 중

월드 생성기 작업이 취소되고 있습니다.

타입: 문자열

유효 값: Pending | Running | Completed | Failed | PartialFailed | Canceling  
| Canceled

필수 여부: 아니요

## succeededWorldCount

생성된 월드 수입니다.

유형: 정수

필수 항목 여부: 아니요

## template

월드 템플릿의 Amazon 리소스 이름(ARN)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: `arn:.*`

Required: No

## worldCount

월드 수에 대한 정보입니다.

유형: [WorldCount](#) 객체

필수 여부: 아니요

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS SDK for C++](#)
- [AWS Java V2용 SDK](#)
- [AWS 루비 V3용 SDK](#)

## WorldSummary

월드에 대한 정보입니다.

### 내용

#### arn

월드의 Amazon 리소스 이름(ARN)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

Required: No

#### createdAt

월드가 생성된 epoch 이후 경과 시간(밀리초)입니다.

유형: 타임스탬프

필수 여부: 아니요

#### generationJob

월드 생성 작업의 Amazon 리소스 이름(ARN)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

Required: No

#### template

월드 템플릿의 Amazon 리소스 이름(ARN)입니다.

유형: 문자열

길이 제약: 최소 길이 1. 최대 길이는 1224입니다.

패턴: arn:.\*

필수 여부: 아니요

## 참고

언어별 AWS SDK 중 하나에서 이 API를 사용하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- [AWS SDK for C++](#)
- [AWS Java V2용 SDK](#)
- [AWS 루비 V3용 SDK](#)

## 일반적인 오류

이 단원에는 모든 AWS 서비스의 API 작업에 대한 일반 오류가 나와 있습니다. 이 서비스의 API 작업에 대한 오류는 해당 API 작업 항목을 참조하십시오.

### AccessDeniedException

이 작업을 수행할 수 있는 충분한 액세스 권한이 없습니다.

HTTP 상태 코드: 400

### IncompleteSignature

요청 서명이 AWS 표준을 준수하지 않습니다.

HTTP 상태 코드: 400

### InternalFailure

알 수 없는 오류, 예외 또는 장애 때문에 요청 처리가 실패했습니다.

HTTP 상태 코드: 500

### InvalidAction

요청된 동작 또는 작업이 유효하지 않습니다. 작업을 올바르게 입력했는지 확인합니다.

HTTP 상태 코드: 400

### InvalidClientTokenId

제공된 X.509 인증서 또는 AWS 액세스 키 ID가 AWS의 레코드에 존재하지 않습니다.

HTTP 상태 코드: 403

## NotAuthorized

이 작업을 수행하려면 권한이 있어야 합니다.

HTTP 상태 코드: 400

## OptInRequired

AWS 액세스 키 ID는 서비스에 대한 구독이 필요합니다.

HTTP 상태 코드: 403

## RequestExpired

요청이 요청상의 날짜 스탬프로부터 15분 이상, 또는 요청 만료 날짜(예: 미리 서명된 URL)로부터 15분 이상 경과한 후 서비스에 도달했거나, 요청상의 날짜 스탬프가 15분 이상 미래입니다.

HTTP 상태 코드: 400

## ServiceUnavailable

서버의 일시적 장애로 인해 요청이 실패하였습니다.

HTTP 상태 코드: 503

## ThrottlingException

요청 제한 때문에 요청이 거부되었습니다.

HTTP 상태 코드: 400

## ValidationError

입력이 AWS 서비스에서 지정한 제약에 충족되지 않습니다.

HTTP 상태 코드: 400

## 공통 파라미터

다음 목록에는 모든 작업이 쿼리 문자열을 사용하여 Signature Version 4 요청에 서명하는 데 사용하는 파라미터가 포함되어 있습니다. 작업별 파라미터는 그 작업에 대한 항목에 나열되어 있습니다. Signature Version 4에 대한 자세한 내용은 IAM 사용 설명서의 [AWS API 요청에 서명](#)을 참조하세요.

## Action

수행할 작업입니다.

유형: 문자열

필수 항목 여부: 예

### Version

요청이 작성되는 API 버전으로 YYYY-MM-DD 형식으로 표시됩니다.

유형: 문자열

필수 항목 여부: 예

### X-Amz-Algorithm

요청 서명을 생성하는 데 사용된 해시 알고리즘입니다.

조건: HTTP 권한 부여 헤더 대신 쿼리 문자열에 인증 정보를 포함하는 경우 이 파라미터를 지정합니다.

유형: 문자열

유효한 값: AWS4-HMAC-SHA256

필수 항목 여부: 조건부

### X-Amz-Credential

자격 증명 범위 값이며 액세스 키, 날짜, 대상으로 하는 리전, 요청하는 서비스 및 종료 문자열("aws4\_request")이 포함된 문자열입니다. 값은 다음 형식으로 표시됩니다. access\_key/YYYYMMDD/region/service/aws4\_request.

자세한 내용은 IAM 사용 설명서의 [서명된 AWS API 요청 생성](#)을 참조하세요.

조건: HTTP 권한 부여 헤더 대신 쿼리 문자열에 인증 정보를 포함하는 경우 이 파라미터를 지정합니다.

유형: 문자열

필수 항목 여부: 조건부

### X-Amz-Date

서명을 만드는 데 사용되는 날짜입니다. 형식은 ISO 8601 기본 형식(YYYYMMDD'T'HHMMSS'Z')이어야 합니다. 예를 들어 다음 날짜 시간은 유효한 X-Amz-Date 값: 20120325T120000Z.

조건: X-Amz-Date는 모든 요청에서 옵션이지만 서명 요청에 사용되는 날짜보다 우선할 때 사용됩니다. 날짜 헤더가 ISO 8601 기본 형식으로 지정된 경우 X-Amz-Date가 필요하지 않습니다. X-Amz-Date를 사용하는 경우 항상 Date 헤더의 값을 재정의합니다. 자세한 내용은 IAM 사용 설명서의 [AWS API 요청 서명의 요소](#)를 참조하세요.

유형: 문자열

필수 항목 여부: 조건부

### X-Amz-Security-Token

AWS Security Token Service(AWS STS)에 대한 호출을 통해 받은 임시 보안 토큰입니다. AWS STS의 임시 보안 인증 정보를 지원하는 서비스 목록은 IAM 사용 설명서의 [IAM으로 작업하는 AWS 서비스](#)를 참조하세요.

조건: AWS STS의 임시 보안 인증 정보를 사용하는 경우 보안 토큰을 포함시켜야 합니다.

유형: 문자열

필수 항목 여부: 조건부

### X-Amz-Signature

서명할 문자열과 파생된 서명 키에서 계산된 16진수로 인코딩된 서명을 지정합니다.

조건: HTTP 권한 부여 헤더 대신 쿼리 문자열에 인증 정보를 포함하는 경우 이 파라미터를 지정합니다.

유형: 문자열

필수 항목 여부: 조건부

### X-Amz-SignedHeaders

표준 요청의 일부로 포함된 모든 HTTP 헤더를 지정합니다. 서명된 헤더 지정에 대한 자세한 내용은 IAM 사용 설명서의 [서명된 AWS API 요청 생성](#)을 참조하세요.

조건: HTTP 권한 부여 헤더 대신 쿼리 문자열에 인증 정보를 포함하는 경우 이 파라미터를 지정합니다.

유형: 문자열

필수 항목 여부: 조건부

## AWS RoboMaker 엔드포인트 및 할당량

다음은 AWS RoboMaker에 대한 서비스 엔드포인트 및 Service Quotas입니다. AWS 서비스에 프로그래밍 방식으로 연결하려면 엔드포인트를 사용해야 합니다. 표준 AWS 엔드포인트 외에도 일부 AWS 서비스는 일부 리전에서 FIPS 엔드포인트를 제공합니다. 자세한 내용은 [AWS 서비스 엔드포인트](#)를 참조하세요.

Service Quotas(제한이라고도 함)는 AWS 계정의 최대 서비스 리소스 또는 작업 수입니다. 자세한 내용은 [AWS 서비스 할당량](#)을 참조하세요.

### Service endpoints

리전 이름	Region	Endpoint	프로토콜
미국 동부 (오하이오)	us-east-2	robomaker.us-east-2.amazonaws.com	HTTPS
미국 동부 (버지니아 북부)	us-east-1	robomaker.us-east-1.amazonaws.com	HTTPS
미국 서부 (오레곤)	us-west-2	robomaker.us-west-2.amazonaws.com	HTTPS
아시아 태평양 (싱가포르)	ap-southeast-1	robomaker.ap-southeast-1.amazonaws.com	HTTPS
아시아 태평양 (도쿄)	ap-northeast-1	robomaker.ap-northeast-1.amazonaws.com	HTTPS
유럽(프랑크푸르트)	eu-central-1	robomaker.eu-central-1.amazonaws.com	HTTPS
유럽(아일랜드)	eu-west-1	robomaker.eu-west-1.amazonaws.com	HTTPS

리전 이름	Region	Endpoint	프로토콜
AWS GovCloud (US-West)	us-gov-west-1	robomaker.us-gov-west-1.amazonaws.com	HTTPS

## 서비스 할당량

이름	기본값	조정 가능	설명
배치 제한 시간	지원되는 각 리전: 14	아니요	시뮬레이션 작업 배치의 최대 제한 시간(일)
동시 GPU 시뮬레이션 작업	각 지원되는 리전: 1	<a href="#">예</a>	현재 리전의 이 계정에서 실행할 수 있는 동시 GPU 시뮬레이션 작업의 최대 수입니다.
동시 월드 내보내기 작업	지원되는 각 리전: 3	<a href="#">예</a>	현재 리전의 이 계정에서 실행할 수 있는 동시 월드 내보내기 작업의 최대 수입니다.
동시 월드 생성 작업	지원되는 각 리전: 3	<a href="#">예</a>	이 리전의 이 계정에서 실행할 수 있는 동시 월드 생성 작업의 최대 수입니다.
동시 배포 작업	각 지원되는 리전: 20	<a href="#">예</a>	현재 리전의 이 계정에서 실행할 수 있는 동시 배포 작업의 최대 수.

이름	기본값	조정 가능	설명
동시 시뮬레이션 작업 배치	지원되는 각 리전: 5	<a href="#">예</a>	현재 리전의 이 계정에서 실행할 수 있는 동시 시뮬레이션 작업 배치의 최대 수.
동시 시뮬레이션 작업	각 지원되는 리전: 1	<a href="#">예</a>	현재 리전의 이 계정에서 실행할 수 있는 동시 시뮬레이션 작업의 최대 수.
플릿	각 지원되는 리전: 20	<a href="#">예</a>	현재 리전의 이 계정에서 생성할 수 있는 플릿의 최대 수.
분당 GPU 시뮬레이션 작업 생성률	지원되는 각 리전: 2	아니요	현재 리전의 이 계정에서 분당 생성할 수 있는 최대 GPU 시뮬레이션 작업 수입니다.
최소 배치 제한 시간	지원되는 각 리전: 5	아니요	시뮬레이션 작업 배치에 지정할 수 있는 최소 제한 시간(분).
최소 시뮬레이션 기간	지원되는 각 리전: 5	아니요	시뮬레이션 작업에 지정할 수 있는 최소 기간(분).
로봇 애플리케이션	지원되는 각 리전: 40	<a href="#">예</a>	현재 리전의 이 계정에서 생성할 수 있는 로봇 애플리케이션의 최대 수.
로봇	지원되는 각 리전: 100	<a href="#">예</a>	현재 리전의 이 계정에서 생성할 수 있는 로봇의 최대 수.

이름	기본값	조정 가능	설명
플릿당 로봇	지원되는 각 리전: 100	<a href="#">예</a>	플릿에 등록할 수 있는 로봇의 최대 수.
분당 시뮬레이션 작업 생성률	us-east-1: 10 us-west-2: 10 각각의 지원되는 다른 리전: 5	아니요	현재 리전의 이 계정에서 분당 생성할 수 있는 최대 시뮬레이션 작업 수입니다.
시뮬레이션 애플리케이션	지원되는 각 리전: 40	<a href="#">예</a>	현재 리전의 이 계정에서 생성할 수 있는 시뮬레이션 애플리케이션의 최대 수.
시뮬레이션 기간	지원되는 각 리전: 14	아니요	재시작을 포함하여 시뮬레이션 작업이 실행될 수 있는 최대 기간(일).
배치당 시뮬레이션 작업 요청 수	각 지원되는 리전: 20	<a href="#">예</a>	StartSimulationJobBatch 호출에서 제출할 수 있는 시뮬레이션 작업 요청의 최대 수
소스 크기	지원되는 각 리전: 5기가바이트	아니요	로봇 애플리케이션 또는 시뮬레이션 애플리케이션의 소스 최대 크기(GB).
로봇 애플리케이션별 버전	지원되는 각 리전: 40	<a href="#">예</a>	로봇 애플리케이션에 대해 생성할 수 있는 버전의 최대 수.

이름	기본값	조정 가능	설명
시뮬레이션 애플리케이션별 버전	지원되는 각 리전: 40	<a href="#">예</a>	시뮬레이션 애플리케이션에 대해 생성할 수 있는 버전의 최대 수.
계정별 월드 템플릿	지원되는 각 리전: 40	<a href="#">예</a>	이 리전의 이 계정에서 생성할 수 있는 최대 월드 템플릿 수입니다.
내보내기 작업별 월드	각 지원되는 리전: 1	아 니 요	월드 내보내기 작업 요청의 최대 월드 수입니다.
생성 작업별 월드	각 지원되는 리전: 50	아 니 요	월드 생성 작업 요청의 최대 월드 수입니다.

# AWS RoboMaker 문제 해결

다음 섹션에서는 AWS RoboMaker 시뮬레이션, IDE 또는 Simulation WorldForge를 사용할 때 발생할 수 있는 오류 및 문제에 대한 문제 해결 조언을 제공합니다. 여기에 나열되지 않은 문제를 발견하는 경우 이 페이지 하단의 피드백 제공 링크를 사용하여 해당 문제를 보고할 수 있습니다.

[CloudWatch Logs 콘솔](#)에서 시뮬레이션 작업에 대한 로그를 찾을 수 있습니다. 기본적으로 AWS RoboMaker는 애플리케이션에 대해 생성된 시뮬레이션 작업에 대한 로그를 업로드합니다.

streamOutputToCloudWatch가 True로 설정된 경우 도구에도 동일한 동작이 적용됩니다. 시뮬레이션 작업에서 [사용자 지정 업로드 구성 추가](#)로 업로드를 구성할 수도 있습니다.

자세한 내용은 [AWS RoboMaker의 로깅 및 모니터링](#) 섹션을 참조하세요.

## 섹션

- [시뮬레이션 작업](#)
- [Simulation WorldForge](#)

## 시뮬레이션 작업

문제: 시뮬레이션 작업이 실패했습니다.

아래 질문을 사용하여 근본 원인을 파악한 다음 권장 조치를 취하세요.

Amazon S3 리소스가 AWS RoboMaker와 동일한 리전에 있습니까?

로봇 애플리케이션, 시뮬레이션 애플리케이션 및 출력 위치가 AWS RoboMaker와 동일한 리전에 있어야 합니다. 애플리케이션 소스와 시뮬레이션 작업 출력 위치를 확인합니다.

로봇 애플리케이션이 비정상적으로 종료되었습니까?

시뮬레이션을 위한 로봇 애플리케이션 설정에 문제가 있습니다. Amazon CloudWatch에서 해당 시뮬레이션 작업에 대한 로봇 애플리케이션 로그를 살펴봅니다.

로그에는 시뮬레이션 작업 세부 정보 화면에서 액세스할 수 있습니다. 로그를 선택한 다음 로그 스트림을 선택합니다. 특정 문제를 살펴보고 싶다면 필터를 사용합니다. 예: WARNING 또는 ERROR.

애플리케이션에 **.so** 파일이 있습니까?

애플리케이션이 충돌한 경우 종속 공유 객체(.so) 파일이 없을 수 있습니다. 환경에서 애플리케이션 번들을 추출하고, 필요한 공유 객체 라이브러리가 /usr/lib 또는 /usr/local/lib에 있는지 확인합니다. 패키지 .xml 파일에 종속성이 추가되었는지 확인합니다.

AWS CLI에서 역할의 ARN을 사용했습니까?

AWS CLI에서 create-simulation-job을 호출하는 경우 역할 이름만 사용하는 것이 아니라 역할의 전체 Amazon 리소스 이름(ARN)을 사용합니다.

역할에 AWS RoboMaker에 대한 신뢰 정책이 있습니까?

AWS CLI에서 create-simulation-job을 호출할 때 IAM 역할의 전체 Amazon 리소스 이름(ARN)을 전달하는 경우 신뢰 정책에 충분한 권한이 있어야 합니다. 역할을 확인하여 다음과 같이 robomaker.amazonaws.com과 신뢰 관계가 있는지 확인합니다.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "robomaker.amazonaws.com"
      },
      "Action": "sts:AssumeRole",
      "Condition": {
        "StringEquals": {
          "aws:SourceAccount": "account#" // Account where
          the simulation job resource is created
        },
        "StringEquals": {
          "aws:SourceArn":
            "arn:aws:robomaker:region:account#:simulation-job/*"
        }
      }
    }
  ]
}
```

조건 키는 AWS 서비스가 서비스간 트랜잭션 중에 [혼동된 대리자](#)로 사용되는 것을 방지합니다. 조건 키에 대한 추가 정보는 [SourceAccount](#) 및 [SourceARN](#)을 참조하세요.

역할 액세스를 확인하고 IAM 역할에 신뢰 정책을 추가하는 방법에 대한 자세한 내용은 [역할 수정](#)을 참조하세요.

역할에 Amazon S3에 게시할 수 있는 권한이 있습니까?

시뮬레이션 작업에 대한 출력 Amazon S3 버킷을 지정한 경우 역할에는 버킷에 대한 쓰기 권한이 있어야 합니다. 쓰기 권한을 포함하도록 신뢰 정책을 업데이트합니다. 다음은 Amazon S3 버킷에 대한 읽기, 나열 및 쓰기 권한을 추가하는 신뢰 정책 예시입니다.

```
{
  "Action": "s3:ListBucket",
```

```

    "Resource": [
      "my-bucket/*"
    ],
    "Effect": "Allow"
  }, {"Action": [
    "s3:Get*",
    "s3:List*"
  ],
    "Resource": [
      "my-bucket/*"
    ],
    "Effect": "Allow"
  }, {"Action": "s3:Put*",
    "Resource": [
      "my-bucket/*"
    ],
    "Effect": "Allow"
  }
}

```

역할에 CloudWatch에 게시할 수 있는 권한이 있습니까?

CloudWatch 액세스 권한으로 IAM 역할의 권한 정책을 업데이트합니다.

```

{"Effect": "Allow",
 "Action": [
   "logs:CreateLogGroup",
   "logs:CreateLogStream",
   "logs:PutLogEvents",
   "logs:DescribeLogStreams"
 ],
 "Resource": "*"
}

```

애플리케이션에 일치하지 않는 엔터티 태그가 있습니까?

엔터티 태그(ETag)는 시뮬레이션 생성 중 제공되는 Amazon S3 객체의 해시입니다. ETag는 객체의 콘텐츠에 대한 변경 사항만 반영하고 메타데이터에 대한 변경을 반영하지 않습니다. AWS RoboMaker에서 사용하기 전에 Amazon S3에서 로봇 애플리케이션 또는 시뮬레이션 번들의 내용을 변경하면 버전 불일치가 발생합니다.

이 문제를 해결하려면 새 로봇 애플리케이션 또는 시뮬레이션 애플리케이션 버전을 생성하고 업데이트된 애플리케이션 번들에 대해 키 위치를 제공합니다. 자세한 내용은 [로봇 애플리케이션 버전 생성](#) 또는 [시뮬레이션 애플리케이션 버전 생성](#) 섹션을 참조하세요.

서브넷 탄력적 네트워크 인터페이스(ENI)에 대한 제한이 초과되었습니까?

AWS RoboMaker는 시뮬레이션 작업이 실행되는 서브넷에서 동시 시뮬레이션 작업당 ENI를 하나 사용합니다. 이들 각 작업에는 IP 주소가 할당되어 있어야 합니다. 이 문제는 다음과 같은 방법으로 해결할 수 있습니다.

사용하지 않는 ENI를 삭제해 서브넷에서 IP 주소를 확보합니다. 사용하지 않는 ENI를 삭제하려면 [네트워크 인터페이스 삭제](#)를 참조하세요.

AWS Management Console을 사용하여 특정 AWS 리전의 ENI에 대한 [서비스 한도 증가](#)를 요청합니다.

시작 명령이 적절하게 구성되어 있습니까?

시뮬레이션이 복잡하거나 컨테이너 이미지가 큰 경우 시뮬레이션 작업을 시작하는 데 몇 분 정도 걸릴 수 있습니다. AWS RoboMaker에서 시뮬레이션 작업을 준비하는 데 25분 이상 걸리는 경우 시작 명령과 관련된 문제가 있을 수 있습니다. 작업을 취소한 다음 새로운 시뮬레이션 작업을 생성하세요. 문제가 지속될 경우 AWS 지원에 문의하세요.

CloudWatch Logs를 사용하여 시뮬레이션 및 로봇 애플리케이션 실행 로그에 오류가 있는지 확인할 수 있습니다. 또한 사용자 지정된 터미널 도구를 추가하여 실행 중인 시뮬레이션 작업을 연결하고 문제를 해결할 수 있습니다.

서브넷이 AWS RoboMaker에서 지원하는 영역 내에 있습니까?

AWS RoboMaker에서 지원하는 AWS 가용 영역 두 곳에 서브넷을 제공합니다. API 응답에는 지원되는 AWS 가용 영역 목록이 포함됩니다.

월드 파일 모델 참조가 올바릅니까?

CloudWatch Log를 사용하여 월드 파일의 모든 모델이 올바른지 확인합니다. 모델을 찾을 수 없는 경우 다음과 같은 오류가 나타납니다.

```
[Wrn] [ModelDatabase.cc:340] Getting models from[http://models.gazebosim.org/]. This
may take a few seconds.
[Wrn] [ModelDatabase.cc:212] Unable to connect to model database using [http://
models.gazebosim.org//database.config]. Only locally installed models will be
available.
```

```
[Err] [ModelDatabase.cc:414] Unable to download model[model://model_name]
[Err] [SystemPaths.cc:429] File or path does not exist[""]
Error [parser.cc:581] Unable to find uri[model://model_name]
```

## Simulation WorldForge

문제: 월드 생성 작업이 실패했습니다.

월드 생성 작업이 완료되지 않았다면, 월드 수(floorplanCount \* interiorCountPerFloorplan)가 1보다 크고 50보다 작은지 확인합니다.

문제: 월드 내보내기 작업이 실패한 이유는 무엇입니까?

아래 질문을 사용하여 근본 원인을 파악한 다음 권장 조치를 취하세요.

AWS RoboMaker에 대한 신뢰 정책이 있습니까?

AWS CLI에서 `create-world-export-job`을 호출할 때 IAM 역할의 전체 Amazon 리소스 이름 (ARN)을 전달하는 경우 신뢰 정책에 충분한 권한이 있어야 합니다. 역할을 확인하여 다음과 같이 `robomaker.amazonaws.com`과 신뢰 관계가 있는지 확인합니다.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "robomaker.amazonaws.com"
      },
      "Action": "sts:AssumeRole",
      "Condition": {
        "StringEquals": {
          "aws:SourceAccount": "account#" // Account where
          the simulation job resource is created
        },
        "StringEquals": {
          "aws:SourceArn":
            "arn:aws:robomaker:region:account#:simulation-job/*"
        }
      }
    }
  ]
}
```

조건 키는 AWS 서비스가 서비스간 트랜잭션 중에 [혼동된 대리자](#)로 사용되는 것을 방지합니다. 조건 키에 대한 추가 정보는 [SourceAccount](#) 및 [SourceARN](#)을 참조하세요.

역할에 Amazon S3에 게시할 수 있는 권한이 있습니까?

내보내기 작업에 대한 출력 Amazon S3 버킷을 지정한 경우 역할에는 버킷에 대한 권한이 있어야 합니다. 다음 권한을 포함하도록 신뢰 정책을 업데이트합니다.

```

{"Effect": "Allow",
 "Action": [
   "s3:AbortMultipartUpload",
   "s3:GetObject",
   "s3:PutObject"
 ],
 "Resource": "my-bucket"
}

```

내보내기 작업에 지정된 버킷을 수정하거나 제거했습니까?

내보내기 작업 중에 버킷을 업데이트하면 내보내기 작업에서 ResourceNotFound 오류가 발생할 수 있습니다.

문제: 월드 이미지에 문제가 있습니다.

아래 질문을 사용하여 근본 원인을 파악한 다음 권장 조치를 취하세요.

출입구에 문이 없는 이유는 무엇입니까?

버전 2 이상의 템플릿을 사용해서만 문을 추가할 수 있습니다. 버전 1 템플릿을 최신 버전으로 업데이트할 수 있습니다. 자세한 내용은 [시뮬레이션 월드 템플릿 버전, 기능 및 변경 사항](#) 섹션을 참조하세요.

AWS RoboMaker Simulation WorldForge는 고유하고 무작위적인 월드를 생성하기 때문에 사용자가 지정한 문 구성은 월드 생성 시 월드에 존재하지 않을 수도 있습니다. 예를 들어 템플릿에서 거실과 주방 사이에 문을 지정해도 해당 방 사이에 개방된 벽이 있을 수 있습니다. 출입구 대신 개방된 벽이 있기 때문에 문을 추가할 수 없습니다.

왜 문이 방 입구를 막습니까?

문이 방 입구를 막고 있는 경우 로봇을 테스트하는 데에 사용할 수 있습니다. 로봇이 이러한 문제를 겪지 않는 월드를 만들려면 다음 중 하나를 수행할 수 있습니다.

월드 템플릿으로 다른 월드를 생성합니다. 새로운 월드에서 생성된 문이 입구를 막지 않을 수도 있습니다.

월드 템플릿에서 문의 개방 비율을 변경합니다.

월드 이미지의 벽이 시뮬레이션 작업이나 내보내기된 월드의 벽보다 짧은 이유는 무엇입니까?

AWS RoboMaker를 사용하면 월드 이미지에서 벽이 잘려 벽에 가려지지 않고 Simulation WorldForge 월드를 볼 수 있습니다. 벽의 높이는 사용자가 만드는 월드의 월드 템플릿에서 지정한 높이와 같습니다.

버전 2 이상의 템플릿으로 생성된 월드의 경우 월드 이미지에서 문 모델이 잘리지 않습니다. 월드 이미지의 문 높이는 사용자가 만든 월드의 문 높이와 같습니다.

## 지원 정책

다음 섹션에서는 에 대한 지원 변경 사항에 대해 설명합니다 AWS RoboMaker.

### 지원 변경 사항: 2022년 12월 15일

2022년 6월 27일에 개발 환경 기능을 개발 환경 기능으로 AWS Cloud9 마이그레이션하고 지원을 종료했습니다. AWS RoboMaker 2022년 12월 15일부터 이전에 만든 개발 환경은 에서 AWS RoboMaker 액세스할 수 없습니다.

지원이 종료된 이유는 무엇입니까?

AWS Cloud9 새로운 기능, 유연성, 확장된 지역 지원을 통해 개선된 개발 경험을 제공합니다. 에서는 Amazon Linux 및 Ubuntu 플랫폼 옵션, 비용 절감 설정, 원하는 로봇 및 시뮬레이션 소프트웨어를 사용하고 구성할 수 있는 AWS Cloud9 유연성이 제공됩니다. AWS Cloud9 시작하기에 대한 자세한 내용은 [AWS Cloud9 사용 설명서](#)를 참조하세요.

의 새로운 개발 환경 AWS Cloud9

AWS Cloud9 사용하던 것과 동일한 개발 환경 기능에 액세스할 수 AWS RoboMaker 있습니다. AWS Cloud9 콘솔을 사용하여 개발 환경을 만들고 새 기능을 활용하십시오. 로봇 및 시뮬레이션 애플리케이션을 구축하고 시뮬레이션하도록 AWS Cloud9 개발 환경을 구성하는 방법을 알아보려면 AWS Robotics [블로그에서 AWS Cloud9로보틱스 애플리케이션 구축 및 시뮬레이션을](#) 참조하십시오.

AWS RoboMaker 기존 개발 환경

액세스 환경은 AWS Cloud9 콘솔을 통해 2022년 6월 27일 이전에 시작되었습니다. NICE DCV 기능을 유지하려면 AWS Cloud9에서 로봇 애플리케이션 구축 및 시뮬레이션의 [NICE DCV 구성](#) 섹션을 참조하세요.

### 지원 사항 변경: 2022년 5월 2일

2022년 5월 2일에 계정에서 기존 robots, fleets 및 deployments 작업 리소스가 제거되었습니다. 이러한 AWS RoboMaker 애플리케이션 배포 리소스를 삭제해도 물리적 하드웨어는 영향을 받지 않습니다. AWS IoT Greengrass Version 2 등의 다른 방법을 통해 물리적 로봇에 애플리케이션을 계속 배포할 수 있습니다.

다음 API 작업은 더 이상 사용되지 않습니다.

- CancelDeploymentJob

- DeleteFleet
- DeleteRobot
- DeregisterRobot
- DescribeDeploymentJob
- DescribeFleet
- DescribeRobot
- ListDeploymentJobs
- ListFleets
- ListRobots
- SyncDeploymentJob

## 권장 조치

다음 작업을 수행하는 것이 좋습니다.

1. 아직 수행하지 않았다면 로봇 및 시뮬레이션 애플리케이션을 지원되는 컨테이너 이미지로 마이그레이션합니다. 로봇 애플리케이션, 시뮬레이션 애플리케이션 및 시뮬레이션 작업을 도커 기반 워크플로로 이동하는 방법에 대한 자세한 내용은 [ROS 애플리케이션을 컨테이너로 마이그레이션](#) 섹션을 참조하세요.
2. 로 AWS IoT Greengrass Version 2 마이그레이션하십시오. 추가 정보는 [AWS IoT Greengrass Version 2 개발자 안내서](#)를 참조하세요. AWS IoT Greengrass Version 2 배포에 대해 자세히 알아보려면 다음 AWS Robotics 블로그 게시물을 참조하십시오.
  - [Docker를 사용하여 ROS 로봇을 배포하고 관리합니다. AWS IoT Greengrass Version 2](#)
  - [를 사용하여 ROS 애플리케이션을 스냅으로 배포 AWS IoT Greengrass Version 2](#)

## 지원 변경 사항: 2022년 3월 15일

2022년 3월 15일에 시뮬레이션 작업에 영향을 미칠 수 있는 AWS RoboMaker 시뮬레이션을 두 가지 변경했습니다.

1. AWS RoboMaker 시뮬레이션 작업을 컨테이너 이미지로 마이그레이션했습니다. 즉, 로봇 및 시뮬레이션 애플리케이션을 지원되는 컨테이너 이미지로 마이그레이션해야 합니다.
2. 시뮬레이션에서 사전 설치된 로봇 운영 소프트웨어 (ROS), Gazebo 및 Ubuntu 기반 이미지 판매를 중단했습니다. AWS RoboMaker ROS 및 Gazebo 기반 시뮬레이션을 계속 실행할 수 있지만 실행

메커니즘이 변경되었습니다. 로봇 애플리케이션용 일반 소프트웨어 제품군과 시뮬레이션 애플리케이션용 시뮬레이션 런타임 소프트웨어 제품군으로 업데이트해야 합니다.

이러한 변경을 한 이유는 무엇입니까?

AWS RoboMaker 이제 모든 로봇 및 시뮬레이션 소프트웨어에 대한 확장된 구성을 지원하므로 시뮬레이션을 실행하는 동안 원하는 로봇 및 시뮬레이션 소프트웨어를 사용하고 구성할 수 있습니다. ROS Kinetic과 같은 이전 버전이나 ROS2 Galactic과 같은 최신 버전을 비롯한 로봇 애플리케이션에서 여전히 ROS를 사용할 수 있지만, ROS를 사용하지 않고도 사용자 지정 로봇 애플리케이션을 실행할 수도 있습니다. 또한 이제 AWS RoboMaker에서 시뮬레이션을 실행하는 동안 원하는 시뮬레이션 소프트웨어를 사용할 수 있습니다.

이제 어떻게 됩니까?

마이그레이션되지 않은 기존 로봇 및 시뮬레이션 애플리케이션은 사용이 제한되지만 지원되는 소프트웨어 제품군 및 컨테이너 이미지로 마이그레이션할 수 있습니다. 2022년 3월 15일 이전에 시작되고 기간이 2022년 3월 15일 이후인 모든 시뮬레이션 작업 및 시뮬레이션 작업 배치는 완료될 때까지 계속 실행됩니다.

## 지원 종료일: 2022년 1월 31일

2022년 1월 31일에 배포 기능을 AWS RoboMaker 애플리케이션 배포 기능으로 AWS IoT Greengrass Version 2 마이그레이션하고 지원을 종료했습니다. AWS IoT Greengrass Version 2 기존 애플리케이션 배포 기능을 지원하고 새로운 기능과 개선된 배포 환경을 제공합니다. 2022년 1월 31일부터 더 이상 AWS RoboMaker에서 새 애플리케이션 배포 리소스(robots, fleets 및 deployments)를 생성할 수 없습니다.

다음 API 작업은 더 이상 사용되지 않습니다.

- CreateDeploymentJob
- CreateFleet
- CreateRobot
- RegisterRobot

## 지원 종료일: 2021년 4월 30일

2021년 4월 30일부터 더 이상 AWS RoboMaker에서 ROS Kinetic, Gazebo 7.1, ROS Dashing 또는 Ubuntu 16.04 리소스를 새로 생성할 수 없습니다. 하지만 기존 AWS RoboMaker 리소스는 모두 계정에 남아 있습니다. 업그레이드하지 않으면 기능 내에 있는 ROS 키네틱, 가제보 7.1, ROS 대싱 및 우분투 16.04 리소스의 기능이 변경되거나 중단될 수도 있습니다. AWS RoboMaker

다음 소프트웨어 제품군 조합은 더 이상 사용되지 않습니다.

- ROS Kinetic, Gazebo 7.1, Ubuntu 16.04
- ROS Kinetic, Gazebo 9, Ubuntu 16.04
- ROS Dashing, Gazebo 9, Ubuntu 16.04

다음 영역이 사용 중단의 영향을 받았습니다.

- AWS Cloud9 통합 개발 환경 (IDE)
  - 기존의 모든 ROS Kinetic 및 ROS Dashing 기반 IDE에 액세스할 수 있습니다. IDE 내에서 계속 작업할 수 있습니다. 빌드 및 번들 프로세스의 성공적인 실행은 보장되지 않습니다.
  - 새로운 ROS Kinetic 및 ROS Dashing 기반 IDE는 생성할 수 없습니다.
- 로봇 및 시뮬레이션 애플리케이션
  - ROS Kinetic 및 ROS Dashing 기반 로봇 애플리케이션은 새로 생성할 수 없습니다.
  - Gazebo 7.1을 사용하는 ROS Kinetic, Gazebo 9를 사용하는 ROS Kinetic 또는 Gazebo 9를 사용하는 ROS Dashing을 사용하여 새 시뮬레이션 애플리케이션을 생성할 수 없습니다.
  - 더 이상 사용되지 않는 ROS 및 Gazebo 버전을 사용하여 기존 로봇 또는 시뮬레이션 애플리케이션의 새 버전을 생성할 수 없습니다.
- 시뮬레이션 작업 및 시뮬레이션 배치
  - Kinetic, Dashing 또는 Gazebo 7.1을 사용하는 로봇 애플리케이션 및 시뮬레이션 애플리케이션으로 새 시뮬레이션 작업을 생성할 수 없습니다.

지원 종료일 이전에 시작되고 기간이 지원 종료일 이후인 시뮬레이션 작업은 완료될 때까지 계속 정상적으로 실행됩니다. 시뮬레이션 작업의 최대 기간은 14일이며, 이러한 작업은 지원 종료 후 최대 14일 동안 실행될 수 있습니다.

- 배포 작업
  - Kinetic 또는 Dashing 기반 로봇 애플리케이션에 대한 배포 작업은 생성할 수 없습니다.
- 샘플 애플리케이션 및 클라우드 확장

- ROS Kinetic 및 ROS Dashing 기반 애플리케이션에서는 더 이상 클라우드 확장이 지원되지 않습니다. ROS Kinetic 및 ROS Dashing 작업 영역에 클라우드 확장 프로그램을 설치할 수는 있지만 작동 여부를 보장할 수 없습니다.
- 더 이상 샘플 애플리케이션을 시작하기 위한 ROS 배포로 ROS Kinetic 또는 ROS Dashing을 선택할 수 없습니다. 샘플 애플리케이션은 기존 ROS Kinetic 및 ROS Dashing IDE에 계속 다운로드할 수 있습니다. 하지만 더 이상 지원되지 않으며 작동이 중단될 수 있습니다.

## 문서 기록

다음 표에는 AWS RoboMaker 서비스 및 문서에 기능 및 지원 중단이 적용된 시기가 나와 있습니다.

변경 사항	설명	날짜
AWS RoboMaker Simulation을 위한 퍼블릭 ECR	퍼블릭 ECR 이미지 사용에 대한 지원이 추가되었습니다. 전용 ECR 리포지토리를 만들지 않고도 AWS RoboMaker 로봇 및 시뮬레이션 애플리케이션을 만들 수 있습니다.	2023년 1월 26일
IDE 사용 중단	AWS RoboMaker IDE 사용 중단	2022년 12월 15일
사전 설치된 RUG 사용 중단	사전 설치된 로봇 운영 소프트웨어(ROS), Ubuntu 및 Gazebo 기본 이미지를 더 이상 사용하지 않고 AWS RoboMaker 시뮬레이션 작업을 컨테이너 이미지로 마이그레이션했습니다.	2022년 3월 15일
애플리케이션 배포 중단	AWS RoboMaker용 애플리케이션 배포가 중단되었습니다.	2022년 1월 31일
클라우드 확장 사용 중단	AWS RoboMaker용 클라우드 확장 프로그램이 더 이상 사용되지 않습니다.	2022년 1월 31일
샘플 사용 중단	AWS RoboMaker의 자율 주행 강화, 내비게이션, 사람 감지 및 음성 명령 샘플이 더 이상 사용되지 않습니다.	2020년 5월 15일

변경 사항	설명	날짜
태그 지원	많은 AWS RoboMaker 리소스에 태그 지원이 추가되었습니다.	2019년 1월 24일
새로운 서비스 및 가이드	AWS RoboMaker 및 AWS RoboMaker 개발자 설명서 첫 릴리스입니다.	11/27/2018

기계 번역으로 제공되는 번역입니다. 제공된 번역과 원본 영어의 내용이 상충하는 경우에는 영어 버전이 우선합니다.