Exam Guide (DVA-C02)

AWS Certified Developer - Associate

AWS Certified Developer - Associate: Exam Guide (DVA-C02)

Table of Contents

AWS Certified Developer - Associate (DVA-C02)

The AWS Certified Developer - Associate (DVA-C02) exam is intended for individuals who perform a developer role. The exam validates a candidate's ability to demonstrate proficiency in developing, testing, deploying, and debugging AWS Cloud-based applications.

Note: AWS exam guides are periodically reviewed and revised to ensure that each certification exam tests skills and AWS services and features that are current and relevant for the job role(s) that the certification is designed to target. Exam guide revisions will be published at least one month before changes are reflected on your exam. Check the Revisions section for a summary of changes.

Topics

Introduction

The AWS Certified Developer - Associate (DVA-C02) exam is intended for individuals who perform a developer role. The exam validates a candidate's ability to demonstrate proficiency in developing, testing, deploying, and debugging AWS Cloud-based applications.

The exam also validates a candidate's ability to complete the following tasks:

- Develop and optimize applications on AWS.
- Package and deploy by using continuous integration and continuous delivery (CI/CD) workflows.
- Secure application code and data.
- Identify and resolve application issues.

Target Candidate Description

The target candidate should have 1 or more years of hands-on experience in developing and maintaining applications by using AWS services.

Recommended general IT knowledge

The target candidate should have the following general IT knowledge:

- Proficiency in at least one high-level programming language
- Understanding of application lifecycle management
- Basic understanding of cloud-focused applications to write code
- Ability to develop functional applications
- Experience in using development tools

Recommended AWS knowledge

The target candidate should be able to complete the following tasks:

- Develop and secure applications by using AWS service APIs, the AWS Command Line Interface (AWS CLI), and SDKs.
- Use a CI/CD pipeline to deploy applications on AWS.

Job tasks that are out of scope for the target candidate

The following list contains job tasks that the target candidate is not expected to be able to perform. This list is non-exhaustive. These tasks are out of scope for the exam:

- Design architectures (for example, distributed systems, microservices, database schemas and modeling).
- Design and create CI/CD pipelines.
- Administer IAM users and groups.
- Administer servers and operating systems.
- Design AWS networking infrastructure (for example, Amazon Virtual Private Cloud [Amazon VPC], AWS Direct Connect).

Exam Content

Response types

There are two types of questions on the exam:

- **Multiple choice:** Has one correct response and three incorrect responses (distractors)
- **Multiple response:** Has two or more correct responses out of five or more response options

Select one or more responses that best complete the statement or answer the question. Distractors, or incorrect answers, are response options that a candidate with incomplete knowledge or skill might choose. Distractors are generally plausible responses that match the content area.

Unanswered questions are scored as incorrect; there is no penalty for guessing. The exam includes 50 questions that affect your score.

Unscored content

The exam includes 15 unscored questions that do not affect your score. AWS collects information about performance on these unscored questions to evaluate these questions for future use as scored questions. These unscored questions are not identified on the exam.

Exam results

The AWS Certified Developer - Associate (DVA-C02) exam has a pass or fail designation. The exam is scored against a minimum standard established by AWS professionals who follow certification industry best practices and guidelines.

Your results for the exam are reported as a scaled score of 100–1,000. The minimum passing score is 720. Your score shows how you performed on the exam as a whole and whether you passed. Scaled scoring models help equate scores across multiple exam forms that might have slightly different difficulty levels.

Your score report could contain a table of classifications of your performance at each section level. The exam uses a compensatory scoring model, which means that you do not need to achieve a passing score in each section. You need to pass only the overall exam.

Each section of the exam has a specific weighting, so some sections have more questions than other sections have. The table of classifications contains general information that highlights your strengths and weaknesses. Use caution when you interpret section-level feedback.

Content outline

This exam guide includes weightings, content domains, and tasks for the exam. This guide does not provide a comprehensive list of the content on the exam. However, additional context for each task is available to help you prepare for the exam.

The exam has the following content domains and weightings:

- Content Domain 1: Development with AWS Services (32% of scored content)
- Content Domain 2: Security (26% of scored content)
- Content Domain 3: Deployment (24% of scored content)
- Content Domain 4: Troubleshooting and Optimization (18% of scored content)

Service References

The following sections provide detailed information about AWS services, technologies, and concepts relevant to this certification exam:

- Mentions of AWS Services on the Exam
- In-Scope AWS Services
- Out-of-Scope AWS Services
- Technologies and Concepts

Content Domain 1: Development with AWS Services

Tasks

Task 1: Develop code for applications hosted on AWS

- Skill 1.1.1: Describe architectural patterns (for example, event-driven, microservices, monolithic, choreography, orchestration, fanout)
- Skill 1.1.2: Describe differences between stateful and stateless concepts
- Skill 1.1.3: Describe differences between tightly coupled and loosely coupled components
- Skill 1.1.4: Describe differences between synchronous and asynchronous patterns
- Skill 1.1.5: Create fault-tolerant and resilient applications in a programming language (for example, Java, C#, Python, JavaScript, TypeScript, Go)
- Skill 1.1.6: Create, extend, and maintain APIs (for example, response/request transformations, enforcing validation rules, overriding status codes)
- Skill 1.1.7: Write and run unit tests in development environments (for example, using AWS SAM)
- Skill 1.1.8: Write code to use messaging services
- Skill 1.1.9: Write code that interacts with AWS services by using APIs and AWS SDKs
- Skill 1.1.10: Handle streaming data using AWS services
- Skill 1.1.11: Use Amazon Q Developer to assist with development
- Skill 1.1.12: Use Amazon EventBridge to implement event-driven patterns
- Skill 1.1.13: Implement resilient application code for third-party service integrations (for example, retry logic, circuit breakers, error handling patterns)

Task 2: Develop code for AWS Lambda

- Skill 1.2.1: Describe the access of private resources in VPCs from Lambda code

- Skill 1.2.2: Configure Lambda functions by defining environment variables and parameters (for example, memory, concurrency, timeout, runtime, handler, layers, extensions, triggers, destinations)
- Skill 1.2.3: Handle the event lifecycle and errors by using code (for example, Lambda Destinations, dead-letter queues)
- Skill 1.2.4: Write and run test code by using AWS services and tools
- Skill 1.2.5: Integrate Lambda functions with AWS services
- Skill 1.2.6: Tune Lambda functions for optimal performance
- Skill 1.2.7: Use Lambda functions to process and transform data in near real time

Task 3: Use data stores in application development

- Skill 1.3.1: Describe high-cardinality partition keys for balanced partition access
- Skill 1.3.2: Describe database consistency models (for example, strongly consistent, eventually consistent)
- Skill 1.3.3: Describe differences between query and scan operations
- Skill 1.3.4: Define Amazon DynamoDB keys and indexing
- Skill 1.3.5: Serialize and deserialize data to provide persistence to a data store
- Skill 1.3.6: Use, manage, and maintain data stores
- Skill 1.3.7: Manage data lifecycles
- Skill 1.3.8: Use data caching services
- Skill 1.3.9: Use specialized data stores based on access patterns (for example, Amazon OpenSearch Service)

Content Domain 2: Security

Tasks

Task 1: Implement authentication and/or authorization for applications and AWS services

- Skill 2.1.1: Use an identity provider to implement federated access (for example, Amazon Cognito, IAM)
- Skill 2.1.2: Secure applications by using bearer tokens
- Skill 2.1.3: Configure programmatic access to AWS
- Skill 2.1.4: Make authenticated calls to AWS services
- Skill 2.1.5: Assume an IAM role
- Skill 2.1.6: Define permissions for IAM principals
- Skill 2.1.7: Implement application-level authorization for fine-grained access control
- Skill 2.1.8: Handle cross-service authentication in microservice architectures

Task 2: Implement encryption by using AWS services

- Skill 2.2.1: Define encryption at rest and in transit
- Skill 2.2.2: Describe certificate management (for example, AWS Private CA)
- Skill 2.2.3: Describe differences between client-side encryption and server-side encryption
- Skill 2.2.4: Use encryption keys to encrypt or decrypt data
- Skill 2.2.5: Generate certificates and SSH keys for development purposes
- Skill 2.2.6: Use encryption across account boundaries
- Skill 2.2.7: Enable and disable key rotation

Task 3: Manage sensitive data in application code

- Skill 2.3.1: Describe data classification (for example, personally identifiable information [PII], protected health information [PHI])
- Skill 2.3.2: Encrypt environment variables that contain sensitive data
- Skill 2.3.3: Use secret management services to secure sensitive data
- Skill 2.3.4: Sanitize sensitive data
- Skill 2.3.5: Implement application-level data masking and sanitization
- Skill 2.3.6: Implement data access patterns for multi-tenant applications

Content Domain 3: Deployment

Tasks

Task 1: Prepare application artifacts to be deployed to AWS

- Skill 3.1.1: Manage the dependencies of the code module (for example, environment variables, configuration files, container images) within the package
- Skill 3.1.2: Organize files and a directory structure for application deployment
- Skill 3.1.3: Use code repositories in deployment environments
- Skill 3.1.4: Apply application requirements for resources (for example, memory, cores)
- Skill 3.1.5: Prepare application configurations for specific environments (for example, by using AWS AppConfig)

Task 2: Test applications in development environments

- Skill 3.2.1: Test deployed code by using AWS services and tools
- Skill 3.2.2: Write integration tests and mock APIs for external dependencies
- Skill 3.2.3: Test applications by using development endpoints (for example, configuring stages in Amazon API Gateway)
- Skill 3.2.4: Deploy application stack updates to existing environments (for example, deploying an AWS SAM template to a different staging environment)
- Skill 3.2.5: Test event-driven applications

Task 3: Automate deployment testing

- Skill 3.3.1: Create application test events (for example, JSON payloads for testing AWS Lambda, API Gateway, AWS SAM resources)

- Skill 3.3.2: Deploy API resources to various environments
- Skill 3.3.3: Create application environments that use approved versions for integration testing (for example, Lambda aliases, container image tags, AWS Amplify branches, AWS Copilot environments)
- Skill 3.3.4: Implement and deploy infrastructure as code (IaC) templates (for example, AWS SAM templates, AWS CloudFormation templates)
- Skill 3.3.5: Manage environments in individual AWS services (for example, differentiating between development, test, and production in API Gateway)
- Skill 3.3.6: Use Amazon Q Developer to generate automated tests

Task 4: Deploy code by using AWS Continuous Integration and Continuous Delivery (CI/CD) services

- Skill 3.4.1: Describe Lambda deployment packaging options
- Skill 3.4.2: Describe API Gateway stages and custom domains
- Skill 3.4.3: Update existing IaC templates (for example, AWS SAM templates, CloudFormation templates)
- Skill 3.4.4: Manage application environments by using AWS services
- Skill 3.4.5: Deploy an application version by using deployment strategies
- Skill 3.4.6: Commit code to a repository to invoke build, test, and deployment actions
- Skill 3.4.7: Use orchestrated workflows to deploy code to different environments
- Skill 3.4.8: Perform application rollbacks by using existing deployment strategies
- Skill 3.4.9: Use labels and branches for version and release management
- Skill 3.4.10: Use existing runtime configurations to create dynamic deployments (for example, using staging variables from API Gateway in Lambda functions)
- Skill 3.4.11: Configure deployment strategies (for example, blue/green, canary, rolling) for application releases

Content Domain 4: Troubleshooting and Optimization

Tasks

Task 1: Assist in a root cause analysis

- Skill 4.1.1: Debug code to identify defects
- Skill 4.1.2: Interpret application metrics, logs, and traces
- Skill 4.1.3: Query logs to find relevant data
- Skill 4.1.4: Implement custom metrics (for example, Amazon CloudWatch embedded metric format [EMF])
- Skill 4.1.5: Review application health by using dashboards and insights
- Skill 4.1.6: Troubleshoot deployment failures by using service output logs
- Skill 4.1.7: Debug service integration issues in applications

Task 2: Instrument code for observability

- Skill 4.2.1: Describe differences between logging, monitoring, and observability
- Skill 4.2.2: Implement an effective logging strategy to record application behavior and state
- Skill 4.2.3: Implement code that emits custom metrics
- Skill 4.2.4: Add annotations for tracing services
- Skill 4.2.5: Implement notification alerts for specific actions (for example, notifications about quota limits or deployment completions)
- Skill 4.2.6: Implement tracing by using AWS services and tools
- Skill 4.2.7: Implement structured logging for application events and user actions
- Skill 4.2.8: Configure application health checks and readiness probes

Task 3: Optimize applications by using AWS services and features

- Skill 4.3.1: Define concurrency
- Skill 4.3.2: Profile application performance
- Skill 4.3.3: Determine minimum memory and compute power for an application

- Skill 4.3.4: Use subscription filter policies to optimize messaging
- Skill 4.3.5: Cache content based on request headers
- Skill 4.3.6: Implement application-level caching to improve performance
- Skill 4.3.7: Optimize application resource usage
- Skill 4.3.8: Analyze application performance issues
- Skill 4.3.9: Use application logs to identify performance bottlenecks

Technologies and Concepts

The following list contains technologies and concepts that might appear on the exam. This list is non-exhaustive and is subject to change. The order and placement of the items in this list is no indication of their relative weight or importance on the exam:

- Analytics
- Application integration
- Compute
- Containers
- Cost and capacity management
- Database
- Developer tools
- Management and governance
- Networking and content delivery
- Security, identity, and compliance
- Storage

Mentions of AWS Services on the Exam

AWS Certification is reducing the reading load on this exam by using official short names for well-known AWS service names that contain abbreviations or parenthetical information. For example, Amazon Simple Notification Service (Amazon SNS) appears on the exam as Amazon SNS.

- The Help feature in the exam (available for every question) contains the list of the short AWS service names and their corresponding full names.

- You can consult AWS Service Names on the AWS Certification website for the list of services that appear as their short names on the exam. Any services that are on the list but that are out of scope for the exam will not appear on the exam.

Note: Not every abbreviation is fully spelled out on the exam or available in the Help feature. The official full name for some AWS services includes an abbreviation that is never expanded (for example, Amazon API Gateway, Amazon EMR). The exam also might contain other abbreviations that the target audience is expected to know.

In-Scope AWS Services

The following list contains AWS services and features that are in scope for the AWS Certified Developer - Associate (DVA-C02) exam. This list is non-exhaustive and is subject to change. AWS offerings appear in categories that align with the offerings' primary functions.

Topics

Analytics

- Amazon Athena
- Amazon Kinesis
- Amazon OpenSearch Service

Application Integration

- AWS AppSync
- Amazon EventBridge
- Amazon Simple Notification Service (Amazon SNS)
- Amazon Simple Queue Service (Amazon SQS)
- AWS Step Functions

Compute

- Amazon EC2
- AWS Elastic Beanstalk
- AWS Lambda

Containers

- Amazon Elastic Container Registry (Amazon ECR)
- Amazon Elastic Container Service (Amazon ECS)
- Amazon Elastic Kubernetes Service (Amazon EKS)

Database

- Amazon Aurora
- Amazon DynamoDB
- Amazon ElastiCache
- Amazon RDS

Developer Tools

- AWS Amplify
- AWS CloudShell
- AWS CodeArtifact

- AWS CodeBuild
- AWS CodeDeploy
- AWS CodePipeline
- AWS X-Ray
- Amazon Q Developer

Management and Governance

- AWS AppConfig
- AWS Cloud Development Kit (AWS CDK)
- AWS CloudFormation
- AWS CloudTrail
- Amazon CloudWatch
- AWS Command Line Interface (AWS CLI)
- AWS Systems Manager

Networking and Content Delivery

- Amazon API Gateway
- Amazon CloudFront
- Elastic Load Balancing
- Amazon Route 53
- Amazon VPC

Security, Identity, and Compliance

- Amazon Cognito
- AWS Identity and Access Management (IAM)
- AWS Key Management Service (AWS KMS)
- AWS Secrets Manager
- AWS Security Token Service (AWS STS)
- AWS WAF

Storage

- Amazon Elastic Block Store (Amazon EBS)
- Amazon Elastic File System (Amazon EFS)
- Amazon S3

Out-of-Scope AWS Services

The following list contains AWS services and features that are out of scope for the AWS Certified Developer - Associate (DVA-C02) exam. This list is non-exhaustive and is subject to change.

Topics

Analytics

- Amazon EMR
- AWS Glue
- Amazon Redshift

Business Applications

- Amazon Connect
- Amazon SES

Compute

- AWS Batch
- Amazon Lightsail
- AWS Outposts

Database

- Amazon DocumentDB
- Amazon Neptune
- Amazon Quantum Ledger Database (Amazon QLDB)

End User Computing

- Amazon AppStream 2.0
- Amazon WorkSpaces

Internet of Things (IoT)

- AWS IoT Core
- AWS IoT Greengrass

Machine Learning

- Amazon Comprehend
- Amazon Forecast
- Amazon Lex
- Amazon Polly

- Amazon Rekognition
- Amazon SageMaker
- Amazon Textract
- Amazon Transcribe
- Amazon Translate

Management and Governance

- AWS Config
- AWS Control Tower
- AWS License Manager
- AWS Organizations
- AWS Service Catalog
- AWS Trusted Advisor

Media Services

- Amazon Elastic Transcoder
- Amazon Kinesis Video Streams

Migration and Transfer

- AWS Database Migration Service (AWS DMS)
- AWS DataSync
- AWS Migration Hub
- AWS Snow Family
- AWS Transfer Family

Networking and Content Delivery

- AWS App Mesh
- AWS Cloud Map

- AWS Direct Connect
- AWS Global Accelerator
- AWS PrivateLink
- AWS Transit Gateway

Robotics

- AWS RoboMaker

Satellite

- AWS Ground Station

Storage

- AWS Backup
- Amazon FSx
- AWS Storage Gateway

Revisions

AWS exam guides are periodically reviewed and revised to ensure that each certification exam tests skills and AWS services and features that are current and relevant for the job role(s) that the certification is designed to target. Exam guide revisions will be published at least one month before changes are reflected on your exam.

Version	Publication Date
2.1	December 12, 2024
2.0	

Changes with Version 2.1

The separate knowledge and skills in Version 2.0 of the exam guide were consolidated into one list of skills under each task. Knowledge items in Version 2.0 that overlapped with existing skills were removed in Version 2.1.

Changes to knowledge and skills

Version 2.1	Version 2.0
Skill 3.2.2: Write integration tests and mock APIs for external dependencies	Skills in: Performing mock integration for APIs and resolving integration dependencies
Skill 3.4.11: Configure deployment strategie s (blue/green, canary, rolling) for application releases	Knowledge of: Deployment strategies (for example, canary, blue/green, rolling)
Skill 4.2.7: Implement structured logging for application events and user actions	Knowledge of: Structured logging

New skills added

- Skill 1.1.11: Use Amazon Q Developer for development assistance
- Skill 1.1.12: Implement event-driven patterns using Amazon EventBridge
- Skill 1.1.13: Implement resilient application code for third-party service integrations (including retry logic, circuit breakers, and error handling patterns)
- Skill 1.2.7: Implement Lambda functions for real-time data processing and transformation
- Skill 1.3.9: Use specialized data stores based on access patterns (for example, Amazon OpenSearch Service)
- Skill 2.1.7: Implement application-level authorization for fine-grained access control
- Skill 2.1.8: Handle cross-service authentication in microservices architectures
- Skill 2.3.5: Implement application-level data masking and sanitization
- Skill 2.3.6: Implement data access patterns for multi-tenant applications
- Skill 3.1.5: Prepare application configurations for different environments (for example, AWS AppConfig)

- Skill 3.2.5: Test event-driven applications
- Skill 3.3.6: Use Amazon Q Developer to generate automated tests
- Skill 4.1.7: Debug service integration issues in applications
- Skill 4.2.8: Create application health checks and readiness probes
- Skill 4.3.6: Implement application-level caching for improved performance
- Skill 4.3.7: Optimize application resource usage
- Skill 4.3.8: Analyze application performance issues
- Skill 4.3.9: Use application logs to identify performance bottlenecks

Skills removed

There are no knowledge and skills removed on Version 2 other than knowledge items that were already covered by existing skills.

Changes to in- and out-of-scope services

Services added to the in-scope list

- Amazon Q Developer

Services removed from the in-scope list

- AWS Copilot
- Amazon CodeGuru

Services added to the out-of-scope list

- No services were added to the out-of-scope list

Services removed from the out-of-scope list

- AWS Device Farm
- Amazon Lex
- AWS Service Catalog
- AWS Database Migration Service (AWS DMS)

Survey

How useful was this exam guide? Let us know by taking our survey.