



Manual do usuário

AWS Kit de ferramentas para VS Code



AWS Kit de ferramentas para VS Code: Manual do usuário

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

As marcas comerciais e imagens comerciais da Amazon não podem ser usadas no contexto de nenhum produto ou serviço que não seja da Amazon, nem de qualquer maneira que possa gerar confusão entre os clientes ou que deprecie ou desprestige a Amazon. Todas as outras marcas comerciais que não pertencem à Amazon pertencem a seus respectivos proprietários, que podem ou não ser afiliados, patrocinados pela Amazon ou ter conexão com ela.

Table of Contents

AWS Toolkit for Visual Studio Code	1
O que é o AWS Toolkit for Visual Studio Code	1
Informações relacionadas	1
Amazon Q Developer e Amazon CodeWhisperer	2
Baixar o kit de ferramentas	3
Baixar o kit de ferramentas para VS Code Marketplace	3
Outros kits de ferramentas de IDE da AWS	3
Conceitos básicos	4
Instalar o kit de ferramentas para VS Code	4
Pré-requisitos	4
Baixando e instalando o AWS Toolkit for Visual Studio Code	4
Pré-requisitos opcionais	5
Conectando-se a AWS	6
Pré-requisitos	6
Abrindo o painel de login	7
Conectando-se a AWS partir do kit de ferramentas	7
Autenticação para Amazon CodeCatalyst	8
AWS Regiões em mudança	9
Adicionando uma região ao AWS Explorer	9
Ocultar uma região do AWS Explorer	10
Configurar cadeia de ferramentas	10
Configurar uma cadeia de ferramentas para .NET Core	10
Configurar uma cadeia de ferramentas para Node.js	10
Configurar uma cadeia de ferramentas para Python	11
Configurar uma cadeia de ferramentas para Java	11
Configurar uma cadeia de ferramentas para Go	12
Usar a cadeia de ferramentas	12
Autenticação e acesso	13
IAM Identity Center	13
IAMcredenciais	13
Criando um IAM usuário	14
Criando um arquivo de credenciais compartilhado a partir do AWS Toolkit for Visual Studio Code	15
Adicionar outros perfis de credenciais	16

AWS ID do construtor	17
Usar um processo de credencial externo	17
Trabalhando com AWS	18
Atributos experimentais	19
AWS Explorador	19
Amazon CodeCatalyst	20
O que é a Amazon CodeCatalyst?	20
Começando com CodeCatalyst	21
Trabalhar com recursos da CodeCatalyst	21
Trabalhar com Ambientes de Desenvolvimento	25
Solução de problemas	28
Amazon API Gateway	29
AWS App Runner	30
Pré-requisitos	30
Definição de preço	34
Criar serviços do App Runner	34
Gerenciar serviços do App Runner	37
AWS Application Composer	40
Trabalhando com o AWS Application Composer	40
AWS CDK	42
AWS CDK aplicações	42
Pilhas do AWS CloudFormation	44
Excluir uma pilha do AWS CloudFormation	45
Crie um CloudFormation modelo	46
Amazon CloudWatch Logs	47
Exibir grupos e fluxos de log do CloudWatch	48
Trabalhar com eventos do CloudWatch Logs	49
Pesquisar grupos de logs	51
Amazon ECR	54
Pré-requisitos	54
Usar o Amazon ECR com o kit de ferramentas para VS Code	56
Amazon ECS	65
Usar o IntelliSense para arquivos de definição de tarefas	65
Amazon ECS Exec	66
Amazon EventBridge	69
Trabalhando com EventBridge esquemas da Amazon	69

Analizador de acesso do AWS IAM	71
Trabalhando com o AWS IAM Access Analyzer	72
AWS IoT	76
Pré-requisitos da AWS IoT	77
Coisas de AWS IoT	77
Certificados do AWS IoT	79
Políticas do AWS IoT	82
Funções AWS Lambda	85
Interagir de forma remota com as funções do Lambda	86
Amazon Redshift	92
Trabalhar com o Amazon Redshift	92
Amazon S3	97
Trabalhar com recursos do S3	98
Trabalhar com objetos do S3	99
AWS Serverless Application	104
Conceitos básicos	104
Executando e depurando funções do Lambda enquanto exclui recursos de modelo AWS SAM	112
Executar e depurar recursos locais do Amazon API Gateway	116
Opções de configuração para depurar aplicações sem servidor	120
Solução de problemas	128
AWS Systems Manager	130
Suposições e pré-requisitos	131
Permissões do IAM para documentos de automação do Systems Manager	131
Criar um documento do Systems Manager Automation	132
Abrir um documento existente do Systems Manager Automation	133
Editar um documento do Systems Manager Automation	133
Publicar um documento do Systems Manager Automation	134
Excluir um documento do Systems Manager Automation	135
Executar um documento do Systems Manager Automation	135
Solução de problemas	136
AWS Step Functions	136
AWS Step Functions e VS Code	137
Compositor de ameaças	147
Trabalhando com o Threat Composer	147
Recursos	148

IAMpermissões para acessar recursos	149
Como adicionar e interagir com recursos existentes	150
Como criar e editar recursos	151
Segurança	154
Proteção de dados	154
Histórico do documento	156
.....	clxii

AWS Toolkit for Visual Studio Code

Este é o guia do usuário do kit de ferramentas da AWS para VS Code. Se você está procurando pelo AWS Toolkit for Visual Studio, consulte o [Guia do usuário do AWS Toolkit for Visual Studio](#).

O que é o AWS Toolkit for Visual Studio Code

O kit de ferramentas para VS Code é uma extensão de código aberto para o editor Visual Studio Code (VS Code). Essa extensão facilita o desenvolvimento, a depuração local e a implantação de aplicações sem servidor que usam a Amazon Web Services (AWS).

Tópicos

- [Conceitos básicos do AWS Toolkit for Visual Studio Code](#)
- [Trabalhando com AWS serviços e ferramentas](#)

Informações relacionadas

Use os recursos a seguir para acessar o código fonte do toolkit ou visualizar as ocorrências abertas no momento.

- [Código-fonte](#)
- [Acompanhamento de ocorrências](#)

Para saber mais sobre o editor do Visual Studio Code, acesse <https://code.visualstudio.com/>.

Amazon Q Developer e Amazon CodeWhisperer

Em 30 de abril de 2024, a Amazon agora CodeWhisperer faz parte do Amazon Q Developer, isso inclui sugestões de código em linha e escaneamentos de segurança do Amazon Q Developer. Faça o download da [IDE extensão Amazon Q Developer no VS Code Marketplace](#) para começar.

Para obter detalhes sobre o serviço Amazon Q Developer, consulte o [Amazon Q Developer User Guide](#). Para obter informações detalhadas sobre planos e preços do Amazon Q, consulte o guia [Preços do Amazon Q](#).

Baixar o kit de ferramentas para VS Code

Você pode baixar, instalar e configurar o AWS Toolkit for Visual Studio Code por meio do VS Code Marketplace no IDE. Para obter instruções detalhadas, consulte a seção [Baixar e instalar](#) no tópico Conceitos básicos deste guia do usuário.

Baixar o kit de ferramentas para VS Code Marketplace

Como alternativa, você pode baixar os arquivos de instalação do AWS Toolkit for Visual Studio Code ao acessar o [VS Code Marketplace](#) no navegador.

Outros kits de ferramentas de IDE da AWS

Além do AWS Toolkit for Visual Studio Code, a AWS também oferece kits de ferramentas de IDE para JetBrains e Visual Studio.

Links do AWS Toolkit for JetBrains

- Siga este link para [Baixar o AWS Toolkit for JetBrains](#) no JetBrains Marketplace.
- Para saber mais sobre o AWS Toolkit for JetBrains, consulte o Guia do usuário do [AWS Toolkit for JetBrains](#).

Links do kit de ferramentas para Visual Studio

- Siga este link para [Baixar o kit de ferramentas para Visual Studio](#) no Visual Studio Marketplace.
- Para saber mais sobre o kit de ferramentas para Visual Studio, consulte o Guia do usuário do [kit de ferramentas para Visual Studio](#).

Conceitos básicos do AWS Toolkit for Visual Studio Code

O AWS Toolkit for Visual Studio Code disponibiliza serviços e recursos da AWS diretamente no ambiente de desenvolvimento integrado (IDE) do VS Code.

Para começar, os tópicos a seguir descrevem como configurar, instalar e configurar o AWS Toolkit for Visual Studio Code.

Tópicos

- [Instalando o AWS Toolkit for Visual Studio Code](#)
- [Conectando-se a AWS](#)
- [AWS Regiões em mudança](#)
- [Configurar cadeia de ferramentas](#)

Instalando o AWS Toolkit for Visual Studio Code

Pré-requisitos

Para começar a trabalhar com o VS AWS Toolkit for Visual Studio Code Code, os seguintes pré-requisitos devem ser atendidos. Para saber mais sobre como acessar todos os AWS serviços e recursos disponíveis no AWS Toolkit for Visual Studio Code, consulte a [the section called “Pré-requisitos opcionais”](#) seção deste guia.

- O VS Code requer o sistema operacional Windows, macOS ou Linux.
- AWS Toolkit for Visual Studio Code Isso requer que você trabalhe com o VS Code versão 1.73.0 ou posterior.

Para obter mais informações sobre o VS Code ou para baixar a versão mais recente do VS Code, consulte o site de [downloads do VS Code](#).

Baixando e instalando o AWS Toolkit for Visual Studio Code

Você pode baixar, instalar e configurar o AWS Toolkit for Visual Studio Code por meio do VS Code Marketplace em seu IDE. Como alternativa, você pode baixar os arquivos AWS Toolkit for Visual Studio Code de instalação navegando até o [VS Code Marketplace](#) a partir do seu navegador.

Instalando o a AWS Toolkit for Visual Studio Code partir do VS Code IDE Marketplace

1. Abra a AWS Toolkit for Visual Studio Code extensão em seu VS Code IDE com o seguinte link:
[Abra o VS Code Marketplace.](#)

Note

Se o VS Code ainda não estiver em execução na máquina, essa operação poderá demorar alguns instantes enquanto o VS Code estiver sendo carregado.

2. Na AWS Toolkit for Visual Studio Code extensão no VS Code Marketplace, escolha Instalar para iniciar o processo de instalação.
3. Quando solicitado, reinicie o VS Code para realizar o processo de instalação.

Pré-requisitos opcionais

Antes de usar determinados recursos do AWS Toolkit for Visual Studio Code, você deve ter o seguinte:

- Conta Amazon Web Services (AWS): Uma AWS conta não é um requisito para usar a AWS Toolkit for Visual Studio Code, mas a funcionalidade é significativamente limitada sem ela. Para obter uma AWS conta, acesse a [página AWS inicial](#). Escolha Criar uma AWS conta ou Concluir a inscrição (se você já visitou o site antes).
- Desenvolvimento de código — O relevante SDK para a linguagem que você deseja usar. Você pode fazer download a partir dos links a seguir ou usar seu gerenciador de pacotes favorito:
 - .NET SDK: <https://dotnet.microsoft.com/download>
 - Node.js SDK: <https://nodejs.org/en/download>
 - Python SDK: <https://www.python.org/downloads>
 - Java SDK: <https://aws.amazon.com/corretto/>
 - Vá SDK: <https://golang.org/doc/install>
- AWS SAM CLI— Essa é uma AWS CLI ferramenta que ajuda você a desenvolver, testar e analisar seus aplicativos sem servidor localmente. Isso não é necessário para a instalação do toolkit. No entanto, recomendamos que você o instale (e o Docker, descrito a seguir) porque ele é necessário para qualquer funcionalidade AWS Serverless Application Model (AWS SAM), como [Criar uma aplicação sem servidor \(local\)](#).

Para obter mais informações, consulte [Instalando o AWS SAM CLI](#) no [Guia do AWS Serverless Application Model desenvolvedor](#).

- Docker — O AWS SAM CLI requer essa plataforma de contêiner de software de código aberto. Para obter mais informações e instruções para fazer download, consulte [Docker](#).
- Gerenciador de pacotes: — um gerenciador de pacotes para baixar e compartilhar o código da aplicação.
 - .NET: [NuGet](#)
 - Node.js: [npm](#)
 - Python: [pip](#)
 - Java: [Gradle](#) ou [Maven](#)

Conectando-se a AWS

A maioria dos recursos da Amazon Web Services (AWS) é gerenciada por meio de uma AWS conta. Não é necessária uma AWS conta para usar o AWS Toolkit for Visual Studio Code, no entanto, as funções do Toolkit são limitadas sem uma conexão.

Se você já configurou uma AWS conta e uma autenticação por meio de outro AWS serviço (como o AWS Command Line Interface), o AWS Toolkit for Visual Studio Code detectará automaticamente suas credenciais.

Pré-requisitos

Se você é novo AWS ou não criou uma conta, há três etapas principais para conectá-la AWS Toolkit for Visual Studio Code à sua AWS conta:

1. Inscrevendo-se AWS em uma conta: Você pode se inscrever AWS em uma conta no [portal de AWS inscrição](#). Para obter informações detalhadas sobre como configurar uma nova AWS conta, consulte o tópico [Visão geral](#) no Guia do usuário de AWS configuração.
2. Configurando a autenticação: Existem 3 métodos principais para se autenticar com sua AWS conta a AWS Toolkit for Visual Studio Code partir do. Para saber mais sobre cada um desses métodos, consulte o tópico [Autenticação e acesso](#) neste guia do usuário.
3. Autenticação com o Kit AWS de Ferramentas: Você pode se conectar à sua AWS conta a partir do Kit de Ferramentas concluindo os procedimentos nas seções a seguir deste Guia do Usuário.

Abrindo o painel de login

Conclua um dos procedimentos a seguir para abrir o painel de login do AWS kit de ferramentas.

Para abrir o painel de login do AWS Toolkit a partir do AWS Explorer:

1. A partir do AWS Toolkit for Visual Studio Code, expanda EXPLORER.
2. Expanda mais ações... menu selecionando o... ícone.
3. Das mais ações... menu, escolha Conectar a AWS para abrir o painel Login do AWS kit de ferramentas.

Para abrir o painel de login do AWS Toolkit usando a paleta de comandos do VS Code:

1. Abra a paleta de comando pressionando **Shift+Command+P (Ctrl+Shift+P/Windows)**.
2. Entre **AWS: Add a New Connection** no campo de pesquisa.
3. Selecione **AWS: Add a New Connection** para abrir o painel de login do AWS Toolkit.

Conectando-se a AWS partir do kit de ferramentas

Autentique e conecte-se com SSO

Para autenticar e se conectar AWS usando AWS IAM Identity Center, conclua o procedimento a seguir.

Note

A autenticação com o AWS Builder ID ou o IAM Identity Center inicia o portal de AWS autorização em seu navegador padrão. Sempre que suas credenciais expirarem, esse processo deverá ser repetido para renovar a conexão entre sua AWS conta e o AWS Toolkit for Visual Studio Code

Autentique e conecte-se com o AWS IAM Identity Center

1. No painel Login do AWS kit de ferramentas, escolha a guia Força de trabalho e, em seguida, selecione o botão Continuar para continuar.

2. No painel Entrar com o IAM Identity Center, insira o Início URL para sua organização. Isso URL é fornecido a você por um administrador ou suporte técnico da sua empresa.
3. Selecione sua AWS região no menu suspenso. Essa é a AWS região que hospeda seu diretório de identidade.
4. Escolha o botão Continuar e confirme que você deseja abrir o site de solicitação de AWS autorização em seu navegador padrão.
5. Siga as instruções no navegador da web padrão. Você receberá a notificação de que é seguro fechar o navegador e retornar ao VS Code quando o processo for concluído.

Autentique e conecte-se com credenciais IAM

Para autenticar e se conectar AWS usando IAM Credenciais, conclua o procedimento a seguir.

Autentique e conecte-se com credenciais IAM

1. No painel Login do AWS kit de ferramentas, escolha IAMCredencial e selecione o botão Continuar para continuar.
2. Insira o **Profile Name** **Access Key**, e **Secret Key** da sua AWS conta nos campos fornecidos e, em seguida, escolha o botão Continuar para adicionar o perfil ao seu arquivo de configuração e conectar o Kit de ferramentas à sua AWS conta.
3. O AWS Explorer do kit de ferramentas é atualizado para exibir os serviços e recursos da AWS quando a autenticação é concluída e uma conexão é estabelecida.

Autenticação para Amazon CodeCatalyst

Para começar a trabalhar com CodeCatalyst o Toolkit, autentique-se e conecte-se com suas credenciais AWS Builder ID ou IAM Identity Center.

Os procedimentos a seguir descrevem como autenticar e conectar o kit de ferramentas com sua conta da AWS .

Autentique e conecte-se com um AWS Builder ID

1. No painel Login do AWS kit de ferramentas, escolha a guia Força de trabalho e, em seguida, selecione o botão Continuar para continuar.
2. Na parte superior do SSO painel Entrar com, escolha o link Ir para entrar.

3. Siga as instruções no navegador da web padrão. Você receberá a notificação de que é seguro fechar o navegador e retornar ao VS Code quando o processo for concluído.

Autentique e conecte-se com o IAM Identity Center

1. No painel Login do AWS kit de ferramentas, escolha a guia Força de trabalho e, em seguida, selecione o botão Continuar para continuar.
2. No painel Entrar com o IAM Identity Center, insira o Início URL para sua organização. Isso URL é fornecido a você por um administrador ou suporte técnico da sua empresa.
3. Selecione sua AWS região no menu suspenso. Essa é a AWS região que hospeda seu diretório de identidade.
4. Escolha o botão Continuar e confirme que você deseja abrir o site de solicitação de AWS autorização em seu navegador padrão.
5. Siga as instruções no navegador da web padrão. Você receberá a notificação de que é seguro fechar o navegador e retornar ao VS Code quando o processo for concluído.

AWS Regiões em mudança

Uma AWS região específica onde seus AWS recursos são gerenciados. Sua AWS região padrão é detectada quando você se conecta à sua AWS conta a partir do AWS Toolkit for Visual Studio Code, exibida automaticamente no AWS Explorer.

As seções a seguir descrevem como adicionar ou ocultar uma região no AWS Explorer.

Adicionando uma região ao AWS Explorer

Conclua o procedimento a seguir para adicionar uma região ao AWS Explorer.

1. No VS Code, abra a paleta de comandos, clique em Exibir no menu principal e selecione Paleta de comandos. Ou use as teclas de atalho a seguir:
 - Windows e Linux: pressione **Ctrl+Shift+P**.
 - macOS: pressione **Shift+Command+P**.
2. Na Paleta de comandos, pesquise **AWS: Show or Hide Regions** e escolha AWS: Mostrar ou ocultar regiões para exibir uma lista das regiões disponíveis.
3. Na lista, selecione as AWS regiões que você deseja adicionar ao AWS Explorer.

4. Escolha o botão OK para confirmar suas escolhas e atualizar o AWS Explorer.

Ocultar uma região do AWS Explorer

Para ocultar uma região da visualização do AWS Explorer, conclua o procedimento a seguir.

1. No AWS Explorer, localize a AWS região que você deseja ocultar.
2. Abra o menu de contexto (clique com o botão direito do mouse) da região que você deseja ocultar.
3. Escolha Mostrar ou ocultar regiões para abrir as opções AWS: Mostrar ou ocultar regiões no VS Code.
4. Desmarque as regiões que você deseja ocultar na visualização do AWS Explorer.

Configurar cadeia de ferramentas

O AWS Toolkit for Visual Studio Code é compatível com vários idiomas em todos os serviços da AWS. As seções a seguir descrevem como configurar a cadeia de ferramentas para diferentes idiomas.

Configurar uma cadeia de ferramentas para .NET Core

1. Verifique se o kit de ferramentas da AWS para VS Code está [instalado](#).
2. Instale a [extensão C#](#). Essa extensão permite que o VS Code depure aplicações .NET Core.
3. Abra uma aplicação AWS Serverless Application Model (AWS SAM) ou [crie uma](#).
4. Abra a pasta que contém `template.yaml`.

Configurar uma cadeia de ferramentas para Node.js

1. Verifique se o kit de ferramentas da AWS para VS Code está [instalado](#).
2. Abra uma aplicação AWS SAM ou [crie uma](#).
3. Abra a pasta que contém `template.yaml`.

Note

Ao depurar uma função do Lambda do TypeScript diretamente do código-fonte (a configuração de inicialização tem "target": "code"), o compilador TypeScript deve ser instalado globalmente ou no package.json do projeto.

Configurar uma cadeia de ferramentas para Python

1. Verifique se o kit de ferramentas da AWS para VS Code está [instalado](#).
2. Instale a [extensão Python para Visual Studio Code](#). Essa extensão permite que o VS Code depure aplicações Python.
3. Abra uma aplicação AWS SAM ou [crie uma](#).
4. Abra a pasta que contém template.yaml.
5. Abra um terminal na raiz da aplicação e configure virtualenv executando `python -m venv ./venv`.

Note

Basta configurar virtualenv uma vez por sistema.

6. Ative virtualenv executando um dos seguintes procedimentos:
 - Bash shell: `./venv/Scripts/activate`
 - PowerShell: `./venv/Scripts/Activate.ps1`

Configurar uma cadeia de ferramentas para Java

1. Verifique se o kit de ferramentas da AWS para VS Code está [instalado](#).
2. Instale [a extensão Java e o Java 11](#). Essa extensão permite que o VS Code reconheça funções Java.
3. Instale a [extensão do depurador Java](#). Essa extensão permite que o VS Code depure aplicações Java.
4. Abra uma aplicação AWS SAM ou [crie uma](#).

5. Abra a pasta que contém `template.yaml`.

Configurar uma cadeia de ferramentas para Go

1. Verifique se o kit de ferramentas da AWS para VS Code está [instalado](#).
2. O Go 1.14 ou posterior é necessário para depurar funções do Lambda Go.
3. Instale a [extensão do Go](#).

Note

A versão 0.25.0 ou posterior é necessária para depurar o runtime Go 1.15 e posterior.

4. Instale as ferramentas do Go usando a [paleta de comandos](#):
 - a. Na paleta de comandos, selecione Go: Install/Update Tools.
 - b. No conjunto de caixas de seleção, selecione `dlv` e `gopls`.
5. Abra uma aplicação AWS SAM ou [crie uma](#).
6. Abra a pasta que contém `template.yaml`.

Usar a cadeia de ferramentas

Depois de configurar a cadeia de ferramentas, é possível usá-la para [executar ou depurar](#) a aplicação AWS SAM.

Autenticação e acesso para o AWS Toolkit for Visual Studio Code

Você não precisa se autenticar com AWS para começar a trabalhar com o AWS Toolkit for Visual Studio Code. No entanto, a maioria dos AWS recursos é gerenciada por meio de uma AWS conta. Para acessar todos os AWS Toolkit for Visual Studio Code serviços e recursos, você precisará se autenticar com AWS IAM Identity Center as credenciais AWS Builder ID ou IAM.

Os tópicos a seguir contêm detalhes adicionais sobre cada tipo de credencial.

Para obter detalhes sobre como se conectar ao AWS Toolkit for Visual Studio Code com suas credenciais existentes, consulte o AWS tópico [Conectando-se a](#) neste Guia do Usuário.

Tópicos

- [AWS Central de identidade do IAM](#)
- [AWS IAMcredenciais](#)
- [AWS ID do construtor para desenvolvedores](#)
- [Usar um processo de credencial externo](#)

AWS Central de identidade do IAM

AWS IAM Identity Center é a melhor prática recomendada para gerenciar a autenticação AWS da sua conta.

Para obter instruções detalhadas sobre como configurar o Centro de Identidade do IAM para kits de desenvolvimento de software (SDKs), consulte a seção [Autenticação do IAM Identity Center](#) no Guia de referência de SDKs e ferramentas da AWS .

Para obter detalhes sobre como autenticar e conectar o AWS kit de ferramentas às suas credenciais atuais do IAM Identity Center, consulte o AWS tópico [Connect to](#) neste Guia do usuário.

AWS IAMcredenciais

AWS IAM autenticação de credenciais com sua AWS conta por meio de chaves de acesso armazenadas localmente.

Para obter detalhes sobre como autenticar e conectar o AWS kit de ferramentas com suas AWS IAM credenciais existentes, consulte o AWS tópico [Conectar a](#) neste Guia do usuário.

As seções a seguir descrevem como configurar IAM credenciais para se autenticar com sua AWS conta a partir do. AWS Toolkit for Visual Studio Code

Important

Antes de configurar IAM as credenciais para se autenticar com sua AWS conta, observe que:

- Se você já definiu IAM credenciais por meio de outro AWS serviço (como o AWS CLI), o detecta AWS Toolkit for Visual Studio Code automaticamente essas credenciais e as disponibiliza no VS Code.
- AWS recomenda o uso da autenticação do IAM Identity Center. Para obter informações adicionais sobre as AWS IAM melhores práticas, consulte as [melhores práticas de segurança na IAM](#) seção do Guia do usuário do AWS Identity and Access Management.
- Para evitar riscos de segurança, não use IAM usuários para autenticação ao desenvolver software específico ou trabalhar com dados reais. Em vez disso, use a federação com um provedor de identidade, como [O que é o IAM Identity Center?](#) no Guia do AWS IAM Identity Center usuário.

Criando um IAM usuário

Antes de configurar a autenticação com sua AWS conta, você precisa concluir AWS Toolkit for Visual Studio Code a Etapa 1: Criar seu IAM usuário e a Etapa 2: Obtenha suas chaves de acesso no tópico [Autenticar usando credenciais de longo prazo](#) no Guia de referência de ferramentas AWS SDKs e ferramentas.

Note

Etapa 3: Atualizar o arquivo de credenciais compartilhadas no Guia de Referência de Ferramentas AWS SDKs e Ferramentas é opcional.

Se você concluir a Etapa 3, ele AWS Toolkit for Visual Studio Code detectará automaticamente suas credenciais durante a etapa abaixo [the section called “Criando um arquivo de credenciais compartilhado a partir do AWS Toolkit for Visual Studio Code”](#).

Se você não concluiu a Etapa 3, ela AWS Toolkit for Visual Studio Code o guiará pelo processo de criação de um, `credentials` file conforme [the section called “Criando um](#)

[arquivo de credenciais compartilhado a partir do AWS Toolkit for Visual Studio Code](#)” descrito abaixo.

Criando um arquivo de credenciais compartilhado a partir do AWS Toolkit for Visual Studio Code

Seu arquivo de configuração compartilhado e o arquivo de credenciais compartilhadas armazenam informações de configuração e credenciais de suas contas. AWS Para obter mais informações sobre configuração e credenciais compartilhadas, consulte a seção [Onde as definições de configuração são armazenadas?](#) no Guia do usuário da AWS Command Line Interface .

Criando um arquivo de credenciais compartilhado por meio do AWS Toolkit for Visual Studio Code

1. Abra a paleta de comando pressionando **Shift+Command+P (Ctrl+Shift+P)**Windows).
2. Entre **AWS: Add a New Connection** no campo de pesquisa.
3. Selecione **AWS: Add a New Connection** para abrir o painel de login do AWS Toolkit.
4. No painel Login do AWS kit de ferramentas, escolha IAMCredencial e selecione o botão Continuar para continuar.
5. Insira o **Profile Name****Access Key**, e **Secret Key** da sua AWS conta nos campos fornecidos e, em seguida, escolha o botão Continuar para adicionar o perfil ao seu arquivo de configuração e conectar o Kit de ferramentas à sua AWS conta.
6. O AWS Explorer do kit de ferramentas é atualizado para exibir os serviços e recursos da AWS quando a autenticação é concluída e uma conexão é estabelecida.

Note

Neste exemplo, suponha que *[Profile_Name]* contém erros de sintaxe e faz com que a autenticação falhe.

```
[Profile_Name]  
xaws_access_key_id= AKIAI44QH8DHBEXAMPLE  
xaws_secret_access_key= wJalrXUtnFEMI/K7MDENG/bPxrFiCYEXAMPLEKEY
```

Veja a seguir um exemplo de mensagem de log gerada em resposta a uma tentativa de autenticação malsucedida.

```
2022-11-02 22:01:54 [ERROR]: Profile [Profile_Name] is not a valid Credential
Profile: not supported by the Toolkit
2022-11-02 22:01:54 [WARN]: Shared Credentials Profile [Profile_Name] is not
valid. It will not be used by the toolkit.
```

Adicionar outros perfis de credenciais

Você pode adicionar várias credenciais aos arquivos de configuração. Para fazer isso, abra a paleta de comandos e selecione Criar perfil de credenciais do kit de ferramentas da AWS . Isso abrirá o arquivo de credenciais. Nessa página, você pode adicionar um novo perfil abaixo do primeiro perfil, conforme mostrado no exemplo a seguir:

```
# Amazon Web Services Credentials File used by AWS CLI, SDKs, and tools
# This file was created by the AWS Toolkit for Visual Studio Code extension.
#
# Your AWS credentials are represented by access keys associated with IAM users.
# For information about how to create and manage AWS access keys for a user, see:
# https://docs.aws.amazon.com/IAM/latest/UserGuide/id_credentials_access-keys.html
#
# This credential file can store multiple access keys by placing each one in a
# named "profile". For information about how to change the access keys in a
# profile or to add a new profile with a different access key, see:
# https://docs.aws.amazon.com/cli/latest/userguide/cli-config-files.html
#
[Profile1_Name]
# The access key and secret key pair identify your account and grant access to AWS.
aws_access_key_id = AKIAIOSFODNN7EXAMPLE
# Treat your secret key like a password. Never share your secret key with anyone. Do
# not post it in online forums, or store it in a source control system. If your secret
# key is ever disclosed, immediately use IAM to delete the access key and secret key
# and create a new key pair. Then, update this file with the replacement key details.
aws_secret_access_key = wJalrXUtnFEMI/K7MDENG/bPxrFiCYEXAMPLEKEY
[Profile2_Name]
aws_access_key_id = AKIAI44QH8DHBEXAMPLE
aws_secret_access_key = je7MtGbClwBF/2Zp9Utk/h3yCo8nvbEXAMPLEKEY
```

AWS ID do construtor para desenvolvedores

Um AWS Builder ID é uma AWS conta adicional opcional ou necessária para determinados AWS serviços. Para obter informações detalhadas sobre o método de autenticação do AWS Builder ID, consulte o tópico [Entrar com o AWS Builder ID](#) no Guia do usuário AWS de login.

Para obter detalhes sobre como autenticar e conectar o AWS kit de ferramentas com seu AWS Builder ID existente, consulte o AWS tópico [Conectar a](#) neste Guia do usuário.

Usar um processo de credencial externo

Você pode configurar o AWS Toolkit for Visual Studio Code para processos de credencial que não são diretamente compatíveis com a AWS. Para isso, modifique o `shared config file`.

A modificação do `shared config file` para processos de credencial é o mesmo para o AWS Toolkit for Visual Studio Code e o AWS Command Line Interface. Para obter informações detalhadas sobre como configurar credenciais externas, consulte o tópico [Credenciais de origem com um processo externo](#) no Guia do usuário da AWS Command Line Interface.

Trabalhando com AWS serviços e ferramentas

AWS Toolkit for Visual Studio Code Isso disponibiliza AWS serviços, ferramentas e recursos para você, diretamente no VS Code. A seguir está uma lista de tópicos do guia que abrangem cada serviço do Toolkit for VS Code e seus recursos. Escolha um serviço ou ferramenta para obter mais informações sobre o que ele faz, como configurá-lo e como trabalhar com recursos básicos.

Tópicos

- [Como trabalhar com atributos experimentais](#)
- [Trabalhando com AWS serviços no AWS Explorer](#)
- [Amazon CodeCatalyst para VS Code](#)
- [Trabalhar com o Amazon API Gateway](#)
- [Usando AWS App Runner com AWS Toolkit for Visual Studio Code](#)
- [AWS Application Composer](#)
- [AWS CDK para VS Code](#)
- [Trabalhar com pilhas do AWS CloudFormation](#)
- [Como trabalhar com o CloudWatch Logs usando o AWS Toolkit for Visual Studio Code](#)
- [Trabalhar com o Amazon Elastic Container Registry](#)
- [Trabalhar com o Amazon Elastic Container Service](#)
- [Trabalhar com o Amazon EventBridge](#)
- [Analisador de acesso do AWS IAM](#)
- [Trabalhar com os AWS IoT no AWS Toolkit for Visual Studio Code](#)
- [Trabalhar com funções do AWS Lambda](#)
- [Amazon Redshift no kit de ferramentas para VS Code](#)
- [Trabalhar com o Amazon S3](#)
- [Trabalhar com aplicações sem servidor](#)
- [Trabalhar com documentos do Systems Manager Automation](#)
- [Trabalhando com AWS Step Functions](#)
- [Trabalhando com o Threat Composer](#)
- [Trabalhar com recursos da](#)

Como trabalhar com atributos experimentais

Os atributos experimentais oferecem acesso antecipado a atributos no AWS Toolkit for Visual Studio Code antes de serem oficialmente lançados.

Warning

Como os atributos experimentais continuam sendo testados e atualizados, eles podem ter problemas de usabilidade. Além disso, os atributos experimentais podem ser removidos do AWS Toolkit for Visual Studio Code sem aviso prévio.

Você pode habilitar recursos experimentais para serviços específicos da AWS na seção Kit de ferramentas da AWS do painel Configurações no IDE do VS Code.

1. Para editar as configurações da AWS no VS Code, selecione Arquivo, Preferências, Configurações.
2. No painel Configurações, expanda Extensões e selecione Kit de ferramentas da AWS.
3. Em AWS: experimentos, marque as caixas de seleção dos recursos experimentais que você deseja acessar antes do lançamento. Se você quiser desativar um atributo experimental, desmarque a caixa de seleção relevante.

Trabalhando com AWS serviços no AWS Explorer

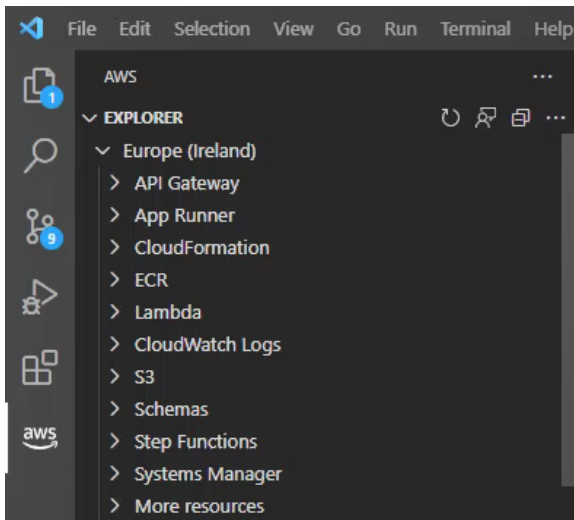
O AWS Explorer oferece uma visão de alguns dos AWS serviços com os quais você pode trabalhar ao usar AWS Toolkit for Visual Studio Code o.

Esta seção fornece informações sobre como acessar e usar o AWS Explorer no VS Code. Presume-se que você já [instalou e configurou](#) o kit de ferramentas para VS Code no sistema.

Alguns pontos importantes:

- Se o kit de ferramentas estiver instalado e configurado corretamente, você deverá ver itens no AWS Explorer. Para ver o AWS Explorer, escolha o ícone AWS na Barra de atividades.

Por exemplo: .



- Certos recursos exigem certas AWS permissões. Por exemplo, para ver as AWS Lambda funções em sua AWS conta, as credenciais nas quais você configurou [Autenticação e acesso](#) devem incluir pelo menos permissões do Lambda somente para leitura. Consulte os tópicos a seguir para obter mais informações sobre as permissões de que cada recurso precisa.
- Se você quiser interagir com AWS serviços que não estão imediatamente visíveis no AWS Explorer, acesse Mais recursos e escolha entre centenas de recursos que podem ser adicionados à interface.

Por exemplo, você pode escolher AWS Toolkit:CodeArtifact: :Repository na seleção dos tipos de recursos disponíveis. Depois que esse tipo de recurso for adicionado a Mais recursos, você poderá expandir a entrada para ver uma lista de recursos que criam CodeArtifact repositórios diferentes com suas próprias propriedades e atributos. Além disso, você pode descrever as propriedades e os atributos dos recursos em modelos formatados em JSON, que podem ser salvos para criar novos recursos na nuvem. AWS

Amazon CodeCatalyst para VS Code

O que é a Amazon CodeCatalyst?

A Amazon CodeCatalyst é um espaço de colaboração baseado em nuvem para equipes de desenvolvimento de software. Por meio do AWS Toolkit for Visual Studio Code, você pode visualizar e gerenciar seus CodeCatalyst recursos diretamente do VS Code. Você também pode trabalhar diretamente na nuvem iniciando o CodeCatalyst Dev Environment. Para obter mais informações sobre o CodeCatalyst serviço, consulte o Guia CodeCatalyst do usuário [da Amazon](#).

Os tópicos a seguir descrevem como conectar o VS Code Catalyst Code e como trabalhar com o Toolkit for VS Code Catalyst Code.

Tópicos

- [Introdução ao Toolkit for VS Code Catalyst e o Kit de ferramentas](#)
- [Trabalhando com Catalyst recursos da Amazon no VS Code](#)
- [Trabalhar com o kit de ferramentas em Ambientes de Desenvolvimento](#)
- [Solução de problemas da Amazon Catalyst e do VS Code](#)

Introdução ao Toolkit for VS Code Catalyst e o Kit de ferramentas

Para começar a trabalhar com Catalyst o VS Code, siga estes procedimentos.

Tópicos

- [Criando uma Catalyst conta](#)
- [Conectando o AWS kit de ferramentas com Catalyst](#)

Criando uma Catalyst conta

Você deve ter um ID ou AWS IAM Identity Center credenciais do AWS Builder ativos para se conectar a Catalyst partir do Toolkit for VS Code. Para saber mais sobre AWS Builder ID, IAM Identity Center e Catalyst credenciais, consulte a Catalyst seção [Configuração com](#) no Guia do Catalyst usuário.

Conectando o AWS kit de ferramentas com Catalyst

Para conectar o AWS kit de ferramentas à sua Catalyst conta, consulte a Catalyst seção [Autenticação para a Amazon](#) no AWS tópico Conectando-se a deste Guia do usuário.

Trabalhando com Catalyst recursos da Amazon no VS Code

As seções a seguir fornecem uma visão geral dos recursos de gerenciamento de Catalyst recursos da Amazon que estão disponíveis no Toolkit for VS Code.

Para obter mais informações sobre ambientes de desenvolvimento e como você pode acessá-los Catalyst, consulte a seção [Ambientes de desenvolvimento](#) no Guia Catalyst do usuário da Amazon.

As seções a seguir descrevem como criar, abrir e trabalhar no Ambiente de Desenvolvimento do VS Code.

Tópicos

- [Clonar um repositório](#)
- [Abrir um Ambiente de Desenvolvimento](#)
- [Criando um ambiente de CodeCatalyst desenvolvimento](#)
- [Como criar um ambiente de desenvolvimento em um repositório de terceiros](#)
- [CodeCatalyst comandos no VS Code](#)

Clonar um repositório

CodeCatalyst é um serviço baseado em nuvem que exige que você esteja conectado à nuvem para trabalhar em CodeCatalyst projetos. Se preferir trabalhar em um projeto localmente, você pode clonar seus CodeCatalyst repositórios em sua máquina local e sincronizá-los com seu CodeCatalyst projeto on-line, na próxima vez em que estiver conectado à nuvem.

Para clonar um repositório da sua CodeCatalyst conta para o VS Code com o AWS Toolkit, conclua as seguintes etapas:

Note

Se você estiver clonando um repositório de um serviço de terceiros, talvez precise se autenticar com as credenciais desse serviço.

Enquanto o repositório está sendo clonado, o VS Code exibe o andamento na janela de status Clonando repositório. Depois que o repositório for clonado, a mensagem Gostaria de abrir o repositório clonado? será exibida.

1. No Toolkit for VS Code, expanda DEVELOPERTOOLS no explorador.
2. Expanda CodeCatalyst, escolha Clone Repository.
3. Na caixa de diálogo Selecionar um CodeCatalyst repositório, pesquise o repositório que você deseja clonar e selecione-o para abrir a caixa de diálogo Escolher uma pasta para clonar.
4. Escolha Selecionar local do repositório para fechar o prompt e começar a clonar o repositório.
5. Na janela de diálogo, escolha uma das seguintes opções para concluir o processo de clonagem:

- Para abrir o repositório na janela atual do VS Code, selecione Abrir.
- Para abrir o repositório em uma nova janela do VS Code, selecione Abrir em uma nova janela.
- Para realizar o processo de clonagem sem abrir o repositório, feche a janela de diálogo.

Abrir um Ambiente de Desenvolvimento

Para abrir um Ambiente de Desenvolvimento existente no VS Code, conclua as etapas a seguir.

Note

Selecionar o Dev Environment inicia o processo de conexão com o VS Code CodeCatalyst abrindo seu Dev Environment. Durante esse processo, o VS Code exibe atualizações de progresso em uma janela de CodeCatalyst status. A janela de status será atualizada quando o processo for concluído.

- Se o Ambiente de Desenvolvimento não for aberto, o status será atualizado com informações sobre o motivo da falha do processo e um link para abrir os respectivos logs.
- Se o processo for bem-sucedido, o Ambiente de Desenvolvimento será aberto em uma nova janela, no VS Code.

1. No Toolkit for VS Code, expanda DEVELOPERTOOLS no explorador.
2. Expanda CodeCatalyst e escolha Abrir ambiente de desenvolvimento para abrir a caixa de diálogo Selecionar um ambiente de CodeCatalyst desenvolvimento no VS Code.
3. Na caixa de diálogo Selecionar um CodeCatalyst ambiente de desenvolvimento, escolha o ambiente de desenvolvimento que você deseja abrir.

Criando um ambiente de CodeCatalyst desenvolvimento

Para criar um Ambiente de Desenvolvimento, conclua as seguintes etapas:

Note

Ao criar um Ambiente de Desenvolvimento, observe o seguinte:

- AWS recomenda que você especifique um alias porque ele simplifica a organização e melhora os recursos de pesquisa para ambientes de desenvolvimento.

- Os Ambientes de Desenvolvimento salvam o trabalho constantemente. Isso significa que o Ambiente de Desenvolvimento pode ser interrompido sem perder o trabalho. A interrupção do Ambiente de Desenvolvimento reduz os custos necessários para mantê-lo em funcionamento.
- Armazenamento é o único ajuste que não pode ser alterado depois que o Ambiente de Desenvolvimento é criado.
- O VS Code exibe o andamento da criação do Ambiente de Desenvolvimento em uma janela de status. Depois que o Ambiente de Desenvolvimento é criado, o VS Code abre o Ambiente de Desenvolvimento em uma nova janela e exibe a mensagem Os autores dos arquivos desta pasta são confiáveis? Concorde com os termos e condições para continuar trabalhando no Ambiente de Desenvolvimento.

1. No Toolkit for VS Code, expanda DEVELOPERTOOLS no explorador.
2. CodeCatalyst expanda e escolha a opção Criar ambiente de desenvolvimento para abrir o menu Criar um ambiente de CodeCatalyst desenvolvimento no VS Code.
3. Na seção Código-fonte, escolha uma das opções a seguir:
 - Use um CodeCatalyst repositório existente: cria um ambiente de desenvolvimento a partir de um CodeCatalyst repositório existente. Você deve selecionar o CodeCatalyst Projeto e a Filial.
 - Criar um Ambiente de Desenvolvimento vazio: cria um Ambiente de Desenvolvimento vazio.
4. (Opcional) Na seção Alias, insira um nome alternativo para o Ambiente de Desenvolvimento.
5. (Opcional) Na seção Configuração de Ambientes de Desenvolvimento, altere as configurações a seguir para atender às suas necessidades específicas.
 - Computação: escolha Editar computação para alterar a quantidade de capacidade de processamento atribuída ao seu sistema. RAM
 - Tempo limite: selecione Editar tempo limite para alterar a quantidade de tempo ocioso do sistema permitido antes que o Ambiente de Desenvolvimento seja interrompido.
 - Armazenamento: escolha Editar tamanho de armazenamento para alterar a quantidade de espaço de armazenamento atribuída ao sistema.
6. Selecione Criar Ambiente de Desenvolvimento para criar o Ambiente de Desenvolvimento em nuvem.

Como criar um ambiente de desenvolvimento em um repositório de terceiros

Você pode criar ambientes de desenvolvimento em um repositório de terceiros vinculando-se ao repositório como fonte.

A vinculação a um repositório de terceiros como fonte é tratada no nível do projeto em CodeCatalyst. Para obter instruções e detalhes adicionais sobre como conectar um repositório de terceiros ao seu ambiente de desenvolvimento, consulte o tópico [Vinculação de um repositório de origem](#) no Guia do usuário da Amazon CodeCatalyst.

CodeCatalyst comandos no VS Code

Há comandos adicionais do VS Code atribuídos a recursos CodeCatalyst relacionados que não são exibidos diretamente no AWS Toolkit.

Para ver uma lista de comandos atribuídos a CodeCatalyst partir da paleta de comandos, conclua as seguintes etapas:

1. No Toolkit for VS Code, expanda DEVELOPERTOOLS no explorador.
2. Escolha Mostrar CodeCatalyst comandos para abrir a Paleta de comandos com uma pesquisa pré-preenchida por CodeCatalyst.
3. Selecione um CodeCatalyst comando na lista para ativá-lo.

Trabalhar com o kit de ferramentas em Ambientes de Desenvolvimento

Os ambientes de desenvolvimento são ambientes de computação virtual para a Amazon CodeCatalyst. As seções a seguir descrevem como criar, abrir e trabalhar em Ambientes de Desenvolvimento usando o AWS Toolkit for Visual Studio Code.

Para obter informações detalhadas sobre ambientes de desenvolvimento, consulte o tópico [Ambientes de desenvolvimento](#) no Guia CodeCatalyst do usuário da Amazon.

Configurar o Ambiente de Desenvolvimento com devfiles

A `devfile` especificação é um formato de padrão aberto YAML que pode ser usado para definir configurações para ambientes de desenvolvimento. Todo Ambiente de Desenvolvimento tem um devfile. Se você criar um Ambiente de Desenvolvimento sem um repositório ou usando um repositório que não contenha um devfile, um padrão será aplicado automaticamente à origem.

Devfiles podem ser atualizados de CodeCatalyst ou seu IDE. Os processos para atualizar um devfile em uma instância local ou remota do VS Code são idênticos, mas se você atualizar um devfile localmente, deverá enviar as atualizações para o repositório de origem antes que elas sejam aplicadas.

Para obter informações detalhadas sobre a configuração de ambientes de desenvolvimento com devfiles, consulte o tópico [Configurando seu ambiente de desenvolvimento](#) no Guia do usuário da Amazon CodeCatalyst.

O procedimento a seguir descreve como editar o devfile de uma instância remota do kit de ferramentas enquanto ele está sendo executado em um Ambiente de Desenvolvimento.

Important

Se você editar o `Devfile` do VS Code, esteja ciente do seguinte:

- Alterar o nome do devfile ou o nome do componente devfile substitui o conteúdo do diretório raiz. Todo o conteúdo anterior é perdido e irrecuperável.
- Se você criar um Ambiente de Desenvolvimento sem um devfile na pasta raiz ou um Ambiente de Desenvolvimento que não esteja associado a um repositório de origem, um devfile com configurações padrão será gerado para o Ambiente de Desenvolvimento quando você o criar.
- Para obter instruções sobre como definir e configurar o `Devfile`, consulte a documentação [Adding Commands](#) no site [Devfile.io](#).

1. No Toolkit for VS Code, expanda DEVELOPERTOOLS no explorador.
2. Expanda CodeCatalyst e escolha Abrir Devfile para abrir `devfile.yaml` em uma nova janela do editor, dentro do seu ambiente de desenvolvimento atual.
3. No editor do VS Code, atualize o devfile e salve as alterações.
4. Na próxima vez em que você iniciar o Ambiente de Desenvolvimento, a configuração será atualizada para corresponder às especificações definidas no `Devfile`.

Autenticando e conectando-se a AWS partir do seu ambiente de desenvolvimento

Para acessar todos os seus AWS recursos do seu ambiente de desenvolvimento, você deve autenticar e conectar sua instância remota do Toolkit à sua AWS conta. A instância remota do kit de

ferramentas é autenticada automaticamente com as credenciais herdadas da instância local do kit de ferramentas quando o Ambiente de Desenvolvimento é iniciado.

Os procedimentos para atualizar as credenciais para uma instância remota do kit de ferramentas são idênticos à experiência de autenticação na instância local do kit de ferramentas. Para obter instruções detalhadas sobre como atualizar credenciais, fazer a autenticação e conectar-se à AWS do kit de ferramentas, consulte a seção [Connecting to AWS](#) no tópico Getting started deste guia do usuário.

Para obter informações adicionais sobre cada um dos métodos de AWS autenticação compatíveis com o AWS Toolkit for Visual Studio Code, consulte o tópico [Autenticação e acesso](#) neste Guia do usuário.

Trabalhar com o kit de ferramentas para VS Code em Ambientes de Desenvolvimento

Depois de abrir ou criar um Ambiente de Desenvolvimento no VS Code, você poderá trabalhar usando o kit de ferramentas para VS Code, da mesma forma que faria em uma instância local do VS Code. Os ambientes de desenvolvimento que executam o VS Code são configurados para instalar automaticamente o AWS Toolkit e se conectar ao seu AWS Builder ID.

Interromper um ambiente de desenvolvimento

Para interromper o Ambiente de Desenvolvimento atual:

1. No Toolkit for VS Code, expanda DEVELOPERTOOLS no explorador.
2. Expanda CodeCatalyst e escolha Stop Dev Environment.
3. Quando solicitado pelo VS Code, confirme que você deseja interromper o Ambiente de Desenvolvimento.
4. O Ambiente de Desenvolvimento é interrompido com êxito quando o VS Code fecha a conexão remota e retorna para uma instância de desenvolvimento local.

Abrir as configurações do ambiente de desenvolvimento

Para abrir as configurações do Ambiente de Desenvolvimento atual, conclua as seguintes etapas:

Note

Não é possível alterar a quantidade de espaço de armazenamento atribuída ao Ambiente de Desenvolvimento depois que ele é criado.

1. No Toolkit for VS Code, expanda DEVELOPERTOOLSo explorador.
2. Expanda CodeCatalyste escolha Abrir configurações para abrir a visualização Configurações do ambiente de desenvolvimento para seu ambiente de desenvolvimento atual.
3. Na visualização Configurações do Ambiente de Desenvolvimento, as seções a seguir contêm opções para o Ambiente de Desenvolvimento:
 - Alias: visualize e altere o alias atribuído ao Ambiente de Desenvolvimento.
 - Status: veja o status atual do Ambiente de Desenvolvimento e o projeto ao qual ele está atribuído e interrompa o ambiente.
 - Devfile: veja o nome e a localização do Devfile para o Ambiente de Desenvolvimento. Para abrir o Devfile, selecione o botão Abrir no editor.
 - Configurações de computação: altere o tamanho e a duração do tempo limite padrão para o Ambiente de Desenvolvimento.

Solução de problemas da Amazon CodeCatalyst e do VS Code

Os tópicos a seguir abordam possíveis problemas técnicos ao trabalhar com a Amazon CodeCatalyst e o VS Code.

Tópicos

- [Versão do VS Code](#)
- [Permissões para a Amazon CodeCatalyst](#)
- [Conectar-se a um Ambiente de Desenvolvimento usando o kit de ferramentas para VS Code](#)

Versão do VS Code

Espera-se que sua versão do VS Code configure um manipulador para `vscode://` URIs seu sistema. Sem esse manipulador, você não pode acessar todos os CodeCatalyst recursos do AWS Toolkit. Por exemplo, você encontra um erro ao iniciar um Ambiente de Desenvolvimento do VS Code Insiders. Isso ocorre porque o VS Code Insiders manipula `vscode-insiders://` URIs e não `vscode://` URIs manipula.

Permissões para a Amazon CodeCatalyst

A seguir estão os requisitos de permissão CodeCatalyst de arquivo para trabalhar com o AWS Toolkit for Visual Studio Code:

- Defina suas próprias permissões de acesso ao arquivo `~/ .ssh/config` para `read` e `write`. Restrinja as permissões `write` para todos os outros usuários.
- Defina as permissões de acesso para os arquivos `~/ .ssh/id_dsa` e `~/ .ssh/id_rsa` somente para `read`. Restrinja as permissões `read`, `write` e `execute` para todos os outros usuários.
- O arquivo `globals.context.globalStorageUri.fsPath` deve estar em um local gravável.

Conectar-se a um Ambiente de Desenvolvimento usando o kit de ferramentas para VS Code

Se você receber o seguinte erro ao tentar se conectar ao Ambiente de Desenvolvimento do AWS Toolkit for Visual Studio Code:

O `~/ .ssh/config` tem uma seção `aws-devenv-*` que pode estar desatualizada.

- Selecione o botão `Abrir configuração...` para abrir o arquivo `~/ .ssh/config` no editor do VS Code.
- No Editor, selecione e exclua o conteúdo da seção `Host aws-devenv-*`.
- Salve as alterações que você fez no `Host aws-devenv-*` de `~/ .ssh/config`. Depois, feche o arquivo.
- Tente novamente se conectar a um Ambiente de Desenvolvimento do kit de ferramentas para VS Code.

Trabalhar com o Amazon API Gateway

Você pode pesquisar e executar recursos remotos do API Gateway na conta da AWS conectada usando o AWS Toolkit for Visual Studio Code.

Note

Esse recurso não comporta depuração.

Como pesquisar e executar recursos remotos do API Gateway

1. No AWS Explorer, selecione API Gateway para expandir o menu. Os recursos remotos do API Gateway são relacionados.

2. Localize o recurso do API Gateway que você deseja invocar, abra o menu de contexto (clique com o botão direito do mouse) e selecione Invocar na AWS.
3. No formulário de parâmetros, especifique os parâmetros de invocação.
4. Para executar o recurso remoto do API Gateway, selecione Invocar. Os resultados são exibidos na visualização Resultado do VS Code.

Usando AWS App Runner com AWS Toolkit for Visual Studio Code

[AWS App Runner](#) fornece uma maneira rápida, simples e econômica de implantar a partir do código-fonte ou de uma imagem de contêiner diretamente em um aplicativo web escalável e seguro na AWS nuvem. Ao usá-lo, você não precisa aprender novas tecnologias, decidir qual serviço de computação usar ou saber como provisionar e configurar AWS recursos.

Você pode usar AWS App Runner para criar e gerenciar serviços com base em uma imagem de origem ou código-fonte. Se você usar uma imagem-fonte, poderá escolher uma imagem de contêiner pública ou privada, armazenada em um repositório de imagens. O App Runner é compatível com os seguintes provedores de repositórios de imagens:

- Amazon Elastic Container Registry (Amazon ECR): armazena imagens privadas em AWS sua conta.
- Amazon Elastic Container Registry Public (Amazon ECR Public): armazena imagens que podem ser lidas pelo público.

Se você escolher a opção de código-fonte, poderá implantar a partir de um repositório de códigos-fonte mantido por um provedor de repositórios compatível. Atualmente, o App Runner oferece suporte [GitHub](#) como provedor de repositório de código-fonte.

Pré-requisitos

Para interagir com o App Runner usando o, é AWS Toolkit for Visual Studio Code necessário o seguinte:

- Uma AWS conta
- Uma versão AWS Toolkit for Visual Studio Code desses recursos AWS App Runner

Além desses requisitos principais, certifique-se de que todos os usuários relevantes do IAM tenham permissões para interagir com o serviço do App Runner. Além disso, você precisa obter informações específicas sobre sua fonte de serviço, como o URI da imagem do contêiner ou a conexão com o GitHub repositório. Você precisa dessas informações para criar o serviço do App Runner.

Configurar permissões do IAM para o App Runner

A maneira mais fácil de conceder as permissões necessárias para o App Runner é anexar uma política AWS gerenciada existente à entidade relevante AWS Identity and Access Management (IAM), especificamente um usuário ou grupo. O App Runner fornece duas políticas gerenciadas que podem ser anexadas aos usuários do IAM:

- `AWSAppRunnerFullAccess`: permite que os usuários realizem todas as ações do App Runner.
- `AWSAppRunnerReadOnlyAccess`: permite que os usuários listem e visualizem detalhes sobre os recursos do App Runner.

Além disso, se você escolher um repositório privado do Amazon Elastic Container Registry (Amazon ECR) como fonte do serviço, deverá criar a seguinte função de acesso para o serviço do App Runner:

- `AWSAppRunnerServicePolicyForECRAccess`: permite que o App Runner acesse imagens do Amazon Elastic Container Registry (Amazon ECR) na sua conta.

Você pode criar essa função automaticamente ao configurar sua instância de serviço com a paleta de comandos do VS Code.

Note

A função `AWSAppRunnerServiceRoleForAppRunner` vinculada ao serviço permite AWS App Runner concluir as seguintes tarefas:

- Envie os registros para os grupos de CloudWatch registros do Amazon Logs.
- Crie regras do Amazon CloudWatch Events para assinar o push de imagem do Amazon Elastic Container Registry (Amazon ECR).

Você não precisa criar manualmente o perfil vinculado ao serviço. Quando você cria um AWS App Runner no AWS Management Console ou usando operações de API chamadas por

AWS Toolkit for Visual Studio Code, AWS App Runner cria essa função vinculada ao serviço para você.

Para mais informações, consulte [Identity and access management for App Runner](#) (Gerenciamento de identidade e acesso para o App Runner) no Guia do desenvolvedor do AWS App Runner .


Obtendo fontes de serviço para o App Runner

Você pode usar o AWS App Runner para implantar serviços a partir de uma imagem ou código-fonte.

Source image

Se você estiver implantando a partir de uma imagem de origem, poderá obter um link para o repositório dessa imagem em um registro de AWS imagem público ou privado.

- Registro privado do Amazon ECR: copie o URI para um repositório privado que usa o console do Amazon ECR em <https://console.aws.amazon.com/ecr/repositories>.
- Registro público do Amazon ECR: copie o URI para um repositório público que usa a Amazon ECR Public Gallery em <https://gallery.ecr.aws/>.

 Note

Você também pode obter o URI de um repositório privado do Amazon ECR diretamente do AWS Explorer no kit de ferramentas para VS Code:

- Abra o AWS Explorer e expanda o nó ECR para ver a lista de repositórios dessa AWS região.
- Clique com o botão direito do mouse em um repositório e escolha Copy Repository URI (Copiar URI do repositório) para copiar o link para a área de transferência.

Você especifica o URI para o repositório de imagens ao configurar a instância de serviço com a paleta de comandos do VS Code

Para mais informações, consulte [App Runner service based on a source image](#) (Serviço do App Runner baseado em uma imagem-fonte) no Guia do desenvolvedor do AWS App Runner .

Source code

Para que seu código-fonte seja implantado em um AWS App Runner serviço, esse código deve ser armazenado em um repositório Git mantido por um provedor de repositório compatível. O App Runner oferece suporte a um provedor de repositório de código-fonte: [GitHub](#)

Para obter informações sobre como configurar um GitHub repositório, consulte a [documentação de introdução](#) em GitHub.

Para implantar seu código-fonte em um serviço do App Runner a partir de um GitHub repositório, o App Runner estabelece uma conexão com. GitHub Se seu repositório for privado (ou seja, não estiver acessível publicamente no GitHub), você deverá fornecer ao App Runner os detalhes da conexão.

Important

Para criar GitHub conexões, você deve usar o console do App Runner (<https://console.aws.amazon.com/apprunner>) para criar uma conexão vinculada GitHub a. AWS Você pode selecionar as conexões que estão disponíveis na página de GitHubconexões ao configurar sua instância de serviço com a Paleta de Comando do VS Code.

Para mais informações, consulte [Managing App Runner connections](#) (Gerenciar conexões do App Runner) no Guia do desenvolvedor do AWS App Runner .

A instância do serviço App Runner fornece um tempo de execução gerenciado que permite que seu código seja criado e executado. AWS App Runner atualmente suporta os seguintes tempos de execução:

- Runtime gerenciado pelo Python
- Tempo de execução gerenciado pelo Node.js

Como parte da configuração do serviço, você fornece informações sobre como o serviço do App Runner compila e inicia o serviço. Você pode inserir essas informações usando a paleta de comandos ou especificar o [arquivo de configuração do App Runner](#) no formato YAML. Os valores desse arquivo instruem o App Runner sobre como compilar e iniciar o serviço e fornecem o contexto do tempo de execução. Isso inclui configurações de rede e variáveis de ambiente relevantes. O arquivo de configuração é denominado `apprunner.yaml`. Ele é adicionado automaticamente ao diretório raiz do repositório da aplicação.

Definição de preço

A cobrança será efetuada pelos recursos de computação e de memória que sua aplicação usar. Além disso, se você automatizar as implantações, pagará também uma taxa mensal definida para cada aplicação, que cobrirá todas as implantações automatizadas para aquele mês. Se optar por implantar a partir do código-fonte, você também pagará uma taxa de compilação pela quantidade de tempo que o App Runner leva para criar um contêiner a partir do código-fonte.

Para obter mais informações, consulte [Preços do AWS App Runner](#).

Tópicos

- [Criar serviços do App Runner](#)
- [Gerenciar serviços do App Runner](#)

Criar serviços do App Runner

Você pode criar um serviço do App Runner no kit de ferramentas para VS Code usando o AWS Explorer e a Paleta de comandos do VS Code. Depois de escolher criar um serviço em uma AWS região específica, as etapas numeradas fornecidas pela Paleta de Comandos orientam você pelo processo de configuração da instância de serviço em que seu aplicativo é executado.

Antes de criar um serviço do App Runner, verifique se você atendeu aos [pré-requisitos](#). Isso inclui fornecer as permissões relevantes do IAM e confirmar o repositório-fonte específico que você deseja implantar.

Para criar um serviço do App Runner

1. Abra o AWS Explorer, se ele ainda não estiver aberto.
2. Clique com o botão direito do mouse no nó do App Runner e escolha Create Service (Criar serviço).

A paleta de comandos é exibida.

3. Para Select a source code location type (Selecionar um tipo de localização de código-fonte), escolha ECR ou Repository (Repositório).

Se escolher ECR, você especificará uma imagem de contêiner em um repositório mantido pelo Amazon Elastic Container Registry. Se escolher Repository (Repositório), você especificará um

repositório de códigos-fonte mantido por um provedor de repositórios compatível. Atualmente, o App Runner oferece suporte [GitHub](#) como provedor de repositório de código-fonte.

Implantar do ECR

1. Para Select or enter an image repository (Selecionar ou inserir um repositório de imagens), escolha ou insira a URL do repositório de imagens mantido pelo registro privado do Amazon ECR ou pela Amazon ECR Public Gallery.

Note

Se especificar um repositório da Amazon ECR Public Gallery, verifique se as implantações automáticas estão desativadas, porque o App Runner não dá suporte a implantações automáticas de imagens de um repositório público do ECR.

As implantações automáticas são desativadas por padrão, e isso é indicado quando o ícone no cabeçalho da paleta de comandos está cruzado por uma linha diagonal. Se você escolher ativar implantações automáticas, uma mensagem informará que essa opção pode gerar custos adicionais.

2. Se a etapa da paleta de comandos informar que Nenhuma tag foi encontrada, será necessário voltar uma etapa para selecionar um repositório que contenha uma imagem de contêiner marcada.
3. Se você estiver usando um registro privado do Amazon ECR, precisará da função de acesso ECR, AppRunnerECR AccessRole, que permite que o App Runner acesse imagens do Amazon Elastic Container Registry (Amazon ECR) em sua conta. Escolha o ícone “+” no cabeçalho da paleta de comandos para criar automaticamente essa função. (Uma função de acesso não será necessária se a imagem estiver armazenada no Amazon ECR público, onde as imagens estão disponíveis ao público.)
4. Para Port (Porta), insira a porta IP usada pelo serviço (porta 8000, por exemplo).
5. Para Configure environment variables, (Configurar variáveis de ambiente), você pode especificar um arquivo que contém as variáveis de ambiente usadas para personalizar o comportamento na instância do serviço. Ou pode ignorar essa etapa.
6. Para Name your service (Dar nome ao serviço), insira um nome exclusivo sem espaços e pressione Enter.
7. Para Select instance configuration (Selecionar a configuração da instância), escolha uma combinação de unidades de CPU e memória em GB para a instância do serviço.

Quando o serviço está sendo criado, o status é alterado de Creating (Sendo criado) para Running (Em execução).

8. Depois que o serviço começar a ser executado, clique com o botão direito do mouse e escolha Copy Service URL (Copiar URL do serviço).
9. Para acessar a aplicação implantada, cole a URL copiada na barra de endereços do navegador da Web.

Implantar de um repositório remoto

1. Em Selecionar uma conexão, escolha uma conexão GitHub vinculada AWS a. As conexões que estão disponíveis para seleção estão listadas na página de GitHub conexões no console do App Runner.
2. Em Selecionar um GitHub repositório remoto, escolha ou insira uma URL para o repositório remoto.

Repositórios remotos que já estão configurados com o gerenciamento de controle de origem (SCM) do Visual Studio Code estão disponíveis para seleção. Você também pode colar um link para o repositório se ele não estiver listado.

3. Para Select a branch (Selecionar uma ramificação), escolha qual ramificação Git do código-fonte você deseja implantar.
4. Para Choose configuration source (Escolher fonte de configuração), especifique como você deseja definir a configuração de tempo de execução.

Se escolher Use configuration file (Usar arquivo de configuração), a instância do serviço será definida pelas configurações do arquivo de configuração `apprunner.yaml`. Esse arquivo está no diretório raiz do repositório da aplicação.

Se você escolher Defina todas as configurações aqui, use a paleta de comandos para especificar o seguinte:

- Runtime (Tempo de execução): escolha Python 3 ou Nodejs 12.
 - Build command (Comando Build): insira o comando para compilar a aplicação no ambiente do tempo de execução da instância do serviço.
 - Start command (Comando Start): insira o comando para compilar a aplicação no ambiente do tempo de execução da instância do serviço.
5. Para Port (Porta), insira a porta IP usada pelo serviço (porta 8000, por exemplo).

6. Para Configure environment variables, (Configurar variáveis de ambiente), você pode especificar um arquivo que contém as variáveis de ambiente usadas para personalizar o comportamento na instância do serviço. Ou pode ignorar essa etapa.
7. Para Name your service (Dar nome ao serviço), insira um nome exclusivo sem espaços e pressione Enter.
8. Para Select instance configuration (Selecionar a configuração da instância), escolha uma combinação de unidades de CPU e memória em GB para a instância do serviço.

Quando o serviço está sendo criado, o status é alterado de Creating (Sendo criado) para Running (Em execução).

9. Depois que o serviço começar a ser executado, clique com o botão direito do mouse e escolha Copy Service URL (Copiar URL do serviço).
10. Para acessar a aplicação implantada, cole o URL copiado na barra de endereços do navegador da web.

Note

Se a tentativa de criar um serviço do App Runner falhar, o serviço exibirá um status de Create failed (Falha na criação) no AWS Explorer. Para obter dicas para a solução de problemas, consulte [When service creation fails](#) (Quando a criação do serviço falha) no App Runner Developer Guide (Guia do desenvolvedor do App Runner).

Gerenciar serviços do App Runner

Depois de criar um serviço App Runner, você pode gerenciá-lo usando o painel AWS Explorer para realizar as seguintes atividades:

- [Pausar e reiniciar os serviços do App Runner](#)
- [Implantar serviços do App Runner](#)
- [Visualizar transmissões de log para o App Runner](#)
- [Excluir serviços do App Runner](#)

Pausar e reiniciar os serviços do App Runner

Se precisar desativar temporariamente seu aplicativo web e interromper a execução do código, você pode pausar o serviço AWS App Runner. O App Runner reduzirá a capacidade computacional do serviço a zero. Quando quiser executar sua aplicação novamente, reinicie o serviço do App Runner. O App Runner provisiona nova capacidade computacional, implanta nela a aplicação e executa a aplicação.

Important

A cobrança pelo App Runner só é efetuada quando ele está em execução. Portanto, você pode pausar e reiniciar a aplicação conforme necessário para administrar os custos. Isso é particularmente útil em cenários de desenvolvimento e teste.

Para pausar o serviço do App Runner

1. Abra o AWS Explorer, se ele ainda não estiver aberto.
2. Expanda App Runner para visualizar a lista de serviços.
3. Clique com o botão direito do mouse no serviço e escolha Pause (Pausar).
4. Na caixa de diálogo exibida, escolha Confirm (Confirmar).

Enquanto o serviço está pausando, o status do serviço é alterado de Running (Em execução) para Pausing (Pausando) e depois para Paused (Pausado).

Para reiniciar o serviço do App Runner

1. Abra o AWS Explorer, se ele ainda não estiver aberto.
2. Expanda App Runner para visualizar a lista de serviços.
3. Clique com o botão direito do mouse no serviço e escolha Resume (Reiniciar).

Enquanto o serviço está sendo reiniciado, o status do serviço é alterado de Resuming (Reiniciando) para Running (Em execução).

Implantar serviços do App Runner

Se você escolher a opção de implantação manual para o serviço, precisará iniciar explicitamente cada implantação no serviço.

1. Abra o AWS Explorer, se ele ainda não estiver aberto.
2. Expanda App Runner para visualizar a lista de serviços.
3. Clique com o botão direito do mouse no serviço e escolha Start Deployment (Iniciar implantação).
4. Enquanto a aplicação está sendo implantada, o status do serviço é alterado de Deploying (Sendo implantado) para Running (Em execução).
5. Para confirmar que a aplicação foi implantada com sucesso, clique com o botão direito do mouse no mesmo serviço e escolha Copy Service URL (Copiar URL do serviço).
6. Para acessar a aplicação Web implantada, cole a URL copiado na barra de endereços do navegador da Web.

Visualizar transmissões de log para o App Runner

Use o CloudWatch Logs para monitorar, armazenar e acessar seus fluxos de registros para serviços como o App Runner. Uma transmissão de log é uma sequência de eventos de log que compartilham a mesma fonte.

1. Expanda App Runner para visualizar a lista de instâncias de serviço.
2. Expanda uma instância de serviço específica para visualizar a lista de grupos de logs. (Um grupo logs é um grupo de transmissões de log que compartilham as mesmas configurações de retenção, monitoramento e controle de acesso.)
3. Clique com o botão direito do mouse em um grupo de logs e escolha View Log Streams (Visualizar transmissões de log).
4. Na Paleta de comandos, escolha um fluxo de logs do grupo.

O editor do VS Code exibe a lista de eventos de log que compõem o fluxo. Você pode escolher carregar eventos mais antigos ou mais recentes no editor.

Excluir serviços do App Runner

Important

Se você excluir o serviço do App Runner, ele será removido permanentemente, e os dados armazenados serão excluídos. Se você precisar recriar o serviço, o App Runner precisará buscar a fonte novamente e compilá-la se for um repositório de códigos. A aplicação Web obtém um novo domínio do App Runner.

1. Abra o AWS Explorer, se ele ainda não estiver aberto.
2. Expanda App Runner para visualizar a lista de serviços.
3. Clique com o botão direito do mouse no serviço e escolha Delete Service (Excluir serviço).
4. Na Paleta de comandos, insira excluir e pressione Enter para confirmar.

O serviço excluído exibe o status Deleting (Sendo excluído) e, então, desaparece da lista.

AWS Application Composer

Você pode usar o AWS Toolkit for Visual Studio Code para trabalhar com o serviço AWS Application Composer. O AWS Application Composer é um criador visual para aplicações da AWS que ajuda a desenvolver a arquitetura de aplicações e visualizar a infraestrutura AWS CloudFormation.

Para obter informações detalhadas sobre o serviço AWS Application Composer, consulte o guia do usuário do [AWS Application Composer](#).

Os tópicos a seguir descrevem como trabalhar com o AWS Application Composer no AWS Toolkit for Visual Studio Code.

Tópicos

- [Trabalhando com o AWS Application Composer a partir do Toolkit](#)

Trabalhando com o AWS Application Composer a partir do Toolkit

O AWS Application Composer for the AWS Toolkit for Visual Studio Code permite que você crie aplicativos visualmente por meio de uma tela interativa. Você também pode usar o Application

Composer para visualizar e modificar modelos AWS CloudFormation e AWS Serverless Application Model (AWS SAM). Ao trabalhar com o Application Composer, suas alterações são armazenadas de forma persistente, permitindo que você alterne facilmente entre a edição de arquivos diretamente no editor do VS Code ou o uso da tela interativa.

Para obter informações detalhadas sobre o serviço AWS Application Composer, informações básicas e tutoriais, consulte o Guia do usuário do serviço [AWS Application Composer](#).

As seções a seguir descrevem como acessar o serviço AWS Application Composer a AWS Toolkit for Visual Studio Code partir do.

Acessando o AWS Application Composer a partir do kit de ferramentas

Existem três maneiras principais de acessar o AWS Application Composer a partir do Toolkit.

Acessando o AWS Application Composer a partir de um modelo existente

1. No VS Code, abra um arquivo de modelo existente no editor do VS Code.
2. Na janela do editor, clique no botão AWS Application Composer localizado no canto superior direito da janela do editor.
3. AWS O Application Composer abre e visualiza seu arquivo de modelo na janela do editor do VS Code.

Acessando o AWS Application Composer a partir do menu de contexto (clique com o botão direito do mouse)

1. No VS Code, clique com o botão direito do mouse no arquivo de modelo que você deseja abrir com o AWS Application Composer.
2. No menu de contexto, escolha a opção Abrir com o App Composer.
3. AWS O Application Composer abre e visualiza seu arquivo de modelo em uma nova janela do editor VS Code.

Acessando o AWS Application Composer a partir da Paleta de Comandos

1. No VS Code, abra a paleta de comandos pressionando **Cmd + Shift + P** ou **Ctrl + Shift + P** (Windows)
2. No campo de pesquisa, insira **AWS Application Composer** e escolha AWS Application Composer quando ele for preenchido nos resultados.

3. Escolha o arquivo de modelo que você deseja abrir, o AWS Application Composer abre e visualiza seu arquivo de modelo em uma nova janela do editor do VS Code.

AWS CDK para VS Code

Esta é uma documentação de pré-lançamento de um recurso em versão de pré-visualização. Está sujeita a alteração.

O serviço AWS CDK permite que você trabalhe com aplicações do [AWS Cloud Development Kit \(AWS CDK\)](#) ou aplicativos. É possível encontrar informações detalhadas sobre o AWS CDK no [Guia do desenvolvedor do AWS Cloud Development Kit \(AWS CDK\)](#).

As aplicações do AWS CDK são compostas de blocos de criação conhecidos como [estruturas](#), que incluem definições para as pilhas do AWS CloudFormation e recursos da AWS. Com o AWS CDK Explorer, é possível visualizar as [pilhas](#) e os [recursos](#) definidos nos constructos do AWS CDK. Essa visualização é fornecida em uma visualização em árvore no painel Ferramentas do desenvolvedor no editor do Visual Studio Code (VS Code).

Esta seção traz informações sobre como acessar e usar o AWS CDK no editor do VS Code. Presume-se que você já [instalou e configurou](#) o kit de ferramentas para VS Code para o IDE local.

Tópicos

- [Trabalhando com AWS CDK aplicativos](#)

Trabalhando com AWS CDK aplicativos

Esta é uma documentação de pré-lançamento de um recurso em versão de pré-visualização. Está sujeita a alteração.

Use o AWS CDK Explorer no AWS Toolkit for VS Code para visualizar e trabalhar AWS CDK com aplicativos.

Pré-requisitos

- Verifique se o sistema atende aos pré-requisitos especificados em [Instalar o kit de ferramentas para VS Code](#).
- Instale a interface da linha de AWS CDK comando, conforme descrito nas primeiras seções de [Introdução ao AWS CDK](#) no Guia do AWS Cloud Development Kit (AWS CDK) desenvolvedor.

Important

A AWS CDK versão deve ser 1.17.0 ou posterior. Use **cdk --version** na linha de comando para ver qual versão está sendo executada.

Visualize um aplicativo AWS CDK

Usando o AWS Toolkit for VS AWS CDK Code Explorer, você pode gerenciar [as](#) pilhas [e](#) os recursos armazenados nas construções CDK dos seus aplicativos. O AWS CDK Explorer exibe seus recursos em uma visualização em árvore usando as informações definidas no `tree.json` arquivo, que é criado quando você executa o **cdk synth** comando. O arquivo `tree.json` está localizado no diretório `cdk.out` de uma aplicação, por padrão.

Para começar a usar o Toolkit AWS CDK Explorer, você precisará criar um CDK aplicativo.

1. Realize as primeiras etapas do [Hello World Tutorial](#), localizado no [Guia do desenvolvedor do AWS CDK](#).

Important

Ao chegar à etapa do tutorial Implantando a pilha, pare e retorne a este guia.

Note

Você pode executar os comandos fornecidos no tutorial (por exemplo, **mkdir** e **cdk init**) em uma linha de comando do sistema operacional ou em uma janela do Terminal dentro do editor do VS Code.

2. Depois de concluir as etapas necessárias do CDK tutorial, abra o CDK conteúdo que você criou no editor do VS Code.
3. No painel de AWS navegação, expanda o título CDK(Visualizar). Seus CDK aplicativos e seus recursos associados agora são exibidos na visualização em árvore do CDK Explorer.

Observações importantes

- Ao carregar CDK aplicativos no editor do VS Code, você pode carregar várias pastas ao mesmo tempo. Cada pasta pode conter vários CDK aplicativos, conforme mostrado na imagem anterior. O AWS CDK Explorer encontra aplicativos no diretório raiz do projeto e em seus subdiretórios diretos.
- Ao executar as primeiras etapas do tutorial, você observará que o último comando executado é **cdk synth**, que gera o arquivo `tree.json`. Se você alterar aspectos de um CDK aplicativo, por exemplo, adicionar mais recursos, precisará executar esse comando novamente para ver as alterações refletidas na visualização em árvore.

Execute outras operações em um AWS CDK aplicativo

Você pode usar o editor do VS Code para realizar outras operações em um CDK aplicativo, da mesma forma que faria usando a linha de comando do seu sistema operacional ou outras ferramentas. Por exemplo, você pode atualizar os arquivos de código no editor e implantar a aplicação usando uma janela do Terminal do VS Code.

Para testar esses tipos de ações, use o editor do VS Code para prosseguir com o [Tutorial Hello World](#) no Guia do desenvolvedor do AWS CDK . Certifique-se de executar a última etapa, Destruindo os recursos do aplicativo, para não incorrer em custos inesperados em sua AWS conta.

Trabalhar com pilhas do AWS CloudFormation

O AWS Toolkit for Visual Studio Code oferece suporte para pilhas do [AWS CloudFormation](#). Ao usar o kit de ferramentas para VS Code, você poderá executar determinadas tarefas com pilhas do AWS CloudFormation; por exemplo, excluí-las.

Tópicos

- [Excluir uma pilha do AWS CloudFormation](#)

- [Crie um AWS CloudFormation modelo usando o AWS Toolkit for Visual Studio Code](#)

Excluir uma pilha do AWS CloudFormation

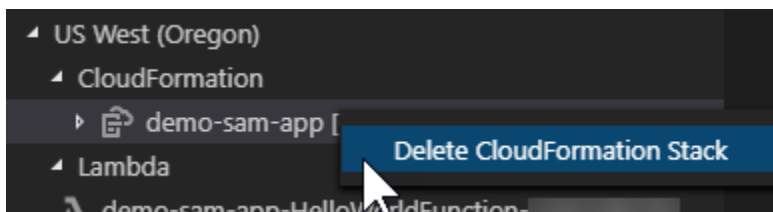
Você pode usar o AWS Toolkit for Visual Studio Code para excluir pilhas do AWS CloudFormation.

Pré-requisitos

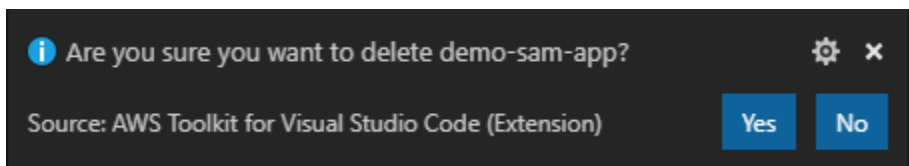
- Verifique se o sistema atende aos pré-requisitos especificados em [Instalar o kit de ferramenta para VS Code](#).
- Certifique-se de que as credenciais que você configurou em [Autenticação e acesso](#) incluem acesso adequado de leitura/gravação ao serviço do AWS CloudFormation. Se no AWS Explorer, em CloudFormation, você vir uma mensagem semelhante a "Error loading CloudFormation resources", (Erro ao carregar os recursos do CloudFormation), verifique as permissões anexadas a essas credenciais. As alterações feitas nas permissões levarão alguns minutos para afetar o AWS Explorer no VS Code.

Excluir uma pilha do CloudFormation

1. No AWS Explorer, abra o menu de contexto da pilha do AWS CloudFormation que você deseja excluir.



2. Escolha Delete CloudFormation Stack (Excluir pilha do CloudFormation).
3. Na mensagem exibida, escolha Yes (Sim) para confirmar a exclusão.



Depois de eliminada, a pilha não será mais listada no AWS Explorer.

Crie um AWS CloudFormation modelo usando o AWS Toolkit for Visual Studio Code

Eles AWS Toolkit for Visual Studio Code podem ajudá-lo a escrever AWS CloudFormation e a criar modelos de SAM.

Pré-requisitos

Kit de ferramentas para VS Code e pré-requisitos de credenciais

- Antes de acessar o CloudFormation serviço a partir do Toolkit for VS Code, você precisa atender aos requisitos descritos no [guia do usuário Instalando o Toolkit for VS Code](#).
- As credenciais que você criou [Autenticação e acesso](#) devem incluir acesso adequado de leitura/gravação ao serviço. AWS CloudFormation

Note

Se o CloudFormation serviço exibir uma mensagem de erro ao carregar CloudFormation recursos, verifique as permissões que você anexou a essas credenciais. Observe também que as alterações feitas nas permissões podem levar alguns minutos para serem atualizadas no AWS Explorer.

CloudFormation pré-requisitos do modelo

- Instale e habilite a extensão [Redhat Developer YAML VS Code](#).
- Você precisa de conexão com a internet ao usar a extensão Redhat Developer YAML VS Code, porque ela é usada para baixar e registrar esquemas JSON na máquina.

Escrevendo um CloudFormation modelo com o YAML Schema Support

O kit de ferramentas usa suporte à linguagem YAML e esquemas JSON para agilizar o processo de escrita e modelos SAM. CloudFormation Recursos como validação de sintaxe e preenchimento automático não apenas agilizam o processo, mas também ajudam a melhorar a qualidade do modelo. Ao selecionar um esquema para o modelo, recomendamos seguir as práticas recomendadas.

CloudFormation modelo

- O arquivo tem uma extensão `.yaml` ou `.yml`.
- O arquivo tem um nó `AWSTemplateFormatVersion` ou Recursos de nível superior.

Modelo do SAM

- Todos os critérios já descritos para CloudFormation
- O arquivo tem um nó Transformação de nível superior, contendo um valor que começa com `AWS::Serverless`.

O esquema será aplicado após a modificação do arquivo. Por exemplo, um esquema de modelo do SAM será aplicado após adicionar uma transformação sem servidor a um CloudFormation modelo e salvar o arquivo.

Validação de sintaxe

A extensão YAML aplicará automaticamente a validação de tipo ao modelo. Isso destaca entradas com tipos inválidos para determinada propriedade. Se você passar o mouse sobre uma entrada destacada, as extensões exibirão ações corretivas.

Preenchimento automático

Ao adicionar novos campos, valores enumerados ou outros [tipos de recurso](#), você pode iniciar o recurso de preenchimento automático da extensão YAML digitando `Ctrl + space`.

Como trabalhar com o CloudWatch Logs usando o AWS Toolkit for Visual Studio Code

O Amazon CloudWatch Logs permite centralizar os logs de todos os sistemas, aplicações e produtos da AWS que você usa em um único serviço altamente escalável. Você pode visualizá-los facilmente, pesquisá-los por códigos de erro ou padrões específicos, filtrá-los com base em campos específicos ou arquivá-los com segurança para análise futura. Para obter mais informações, consulte [O que é o Amazon CloudWatch Logs?](#) no Manual do usuário do Amazon CloudWatch.

Os tópicos a seguir descrevem como usar o AWS Toolkit for Visual Studio Code para trabalhar com o CloudWatch Logs em uma conta da AWS.

Tópicos

- [Como visualizar grupos de logs e fluxos de logs do CloudWatch usando o AWS Toolkit for Visual Studio Code](#)
- [Como trabalhar com eventos de logs do CloudWatch em fluxos de logs usando o AWS Toolkit for Visual Studio Code](#)
- [Pesquisar grupos de logs do CloudWatch](#)

Como visualizar grupos de logs e fluxos de logs do CloudWatch usando o AWS Toolkit for Visual Studio Code

Stream de log é uma sequência de eventos de log que compartilham a mesma origem. Cada origem separada de logs no CloudWatch Logs compõe um fluxo de logs separado.

Um grupo de logs é um grupo de fluxos de log que compartilham as mesmas configurações de retenção, monitoramento e controle de acesso. Você pode definir grupos de logs e especificar quais fluxos colocar em cada grupo. Não há limite para o número de streams de log que podem pertencer a um grupo de logs.

Para obter mais informações, consulte [Trabalhar com grupos e streams de log](#) no Manual do usuário do Amazon CloudWatch.

Tópicos

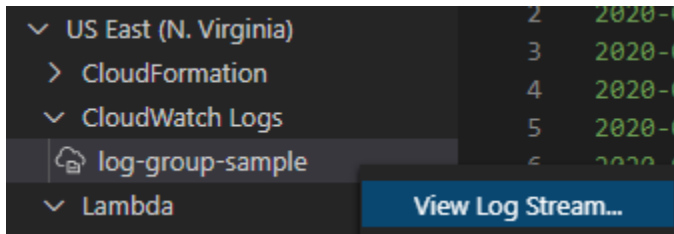
- [Exibir grupos e fluxos de log com o nó do CloudWatch Logs](#)

Exibir grupos e fluxos de log com o nó do CloudWatch Logs

1. No VS Code, selecione Exibir, Explorer para abrir o AWS Explorer.
2. Clique no nó CloudWatch Logs para expandir a lista de grupos de logs.

Os grupos de logs para a região da AWS são exibidos sob o nó CloudWatch Logs.

3. Para exibir os fluxos de logs em um grupo de logs, clique com o botão direito do mouse no nome do grupo de logs e selecione Visualizar fluxo de logs.



4. Na Paleta de comandos, selecione um fluxo de logs do grupo para visualizar.

Note

A paleta de comandos exibe uma marcação de data e hora para o último evento em cada fluxo.

O [editor de fluxo de logs](#) é iniciado para exibir os eventos de log do fluxo.

Como trabalhar com eventos de logs do CloudWatch em fluxos de logs usando o AWS Toolkit for Visual Studio Code

Depois de abrir o editor de fluxo de logs, você poderá acessar os eventos de log em cada fluxo. Eventos de log são registros de atividades registradas pela aplicação ou recurso que estiver sendo monitorado.

Tópicos

- [Exibir e copiar informações do fluxo de log](#)
- [Salve o conteúdo do editor de fluxo de log em um arquivo local](#)

Exibir e copiar informações do fluxo de log

Ao abrir um fluxo de logs, o editor de fluxo de logs exibe a sequência de eventos de log do fluxo.

1. Para localizar e exibir um fluxo de logs, abra o editor de fluxo de logs (consulte [Exibir grupos e fluxos de log do CloudWatch](#)).

Cada linha que lista um evento tem um carimbo de data/hora para mostrar quando ele foi registrado.

2. É possível visualizar e copiar informações sobre os eventos do stream usando as seguintes opções:

- Exibir eventos por tempo: exiba os eventos de log mais recentes e antigos escolhendo Load newer events (Carregar eventos mais recentes) ou Load older events (Carregar eventos mais antigos).

Note

O editor Log Stream (Fluxo de dados de log) inicialmente carrega um lote das 10.000 linhas mais recentes dos eventos de log ou 1 MB de dados de log (o que for menor). Se você escolher Load newer events (Carregar eventos mais recentes), o editor exibirá os eventos que foram registrados após o último lote ter sido carregado. Se você escolher Load older events (Carregar eventos mais antigos), o editor exibirá um lote de eventos que ocorreram antes daqueles exibidos atualmente.

- Copiar eventos de log: selecione os eventos a serem copiados, clique com o botão direito e selecione Copy (Copiar) no menu.
- Copie o nome do fluxo de logs: clique com o botão direito do mouse na guia Fluxo de logs e selecione Copiar nome do fluxo de logs.

Note

Você também pode usar a paleta de comandos para executar o comando Copiar nome do fluxo de logs do kit de ferramentas da AWS.

Salve o conteúdo do editor de fluxo de log em um arquivo local

Você pode fazer o download do conteúdo do editor de stream de log do CloudWatch para um log na sua máquina local.

Note

Com essa opção, você salva no arquivo apenas os eventos de log exibidos no momento no editor de fluxo de logs. Por exemplo, se o tamanho total de um fluxo de log for 5MB e somente 2MB for carregado no editor, o arquivo salvo também conterá apenas 2MB de

dados de log. Para exibir mais dados a serem salvos, escolha Load newer events (Carregar eventos mais recentes) ou Load older events (Carregar eventos mais antigos) no editor.

1. Para localizar e copiar o fluxo de logs, abra o editor de fluxo de logs (consulte [Exibir grupos e fluxos de log do CloudWatch](#)).
2. Selecione o ícone Salvar ao lado da guia que exibe o nome do fluxo de logs.

Note

Você também pode usar a paleta de comandos para executar a opção Salvar o conteúdo atual do fluxo de logs do kit de ferramentas da AWS.

3. Use a caixa de diálogo para selecionar ou criar uma pasta de download para o arquivo de log e clique em Save (Salvar).

Pesquisar grupos de logs do CloudWatch

Você pode usar Pesquisar grupo de logs para pesquisar todos os fluxos de logs em um grupo de logs.

Para obter informações detalhadas sobre o serviço Amazon CloudWatch Logs, consulte o tópico [Trabalhar com grupos de logs e fluxos de logs](#) no Guia do usuário do Amazon CloudWatch.

Pesquisar grupos de logs na paleta de comandos do VS Code

Para pesquisar grupos de logs usando a paleta de comandos do VS Code, siga as etapas abaixo.

Para obter informações detalhadas sobre filtros e padrões do Amazon CloudWatch Logs, consulte a seção [Sintaxe de padrões de filtros](#) no Guia do usuário do Amazon CloudWatch.

1. No VS Code, abra a paleta de comandos pressionando **cmd+shift+p** (Windows: **ctrl+shift+p**).
2. Na paleta de comandos, insira o comando **AWS: Search Log Group** e selecione-o para abrir a caixa de diálogo de pesquisa de grupo de logs no VS Code e siga as instruções para continuar.

Note

Na primeira solicitação, você tem a opção de mudar de região da AWS antes de passar para as próximas etapas.

3. Na solicitação Seleccionar grupo de logs (1/3), escolha o grupo de logs que você deseja pesquisar.
4. Na solicitação Seleccionar filtro de tempo (2/3), escolha o filtro de tempo a ser aplicado à pesquisa.
5. Na solicitação Pesquisar grupo de logs... (3/3), insira o padrão de pesquisa no campo fornecido e pressione a tecla **Enter** para continuar ou a tecla **ESC** para cancelar a pesquisa.
6. Os resultados da pesquisa serão abertos no editor do VS Code quando a pesquisa for concluída.


Pesquisar grupo de logs no AWS Explorer

Para pesquisar grupos de logs no AWS Toolkit for Visual Studio Code Explorer, siga as etapas abaixo.

1. No AWS Toolkit for Visual Studio Code Explorer, expanda o CloudWatch.
2. Abra o menu de contexto (clique com o botão direito do mouse) do grupo de logs que deseja pesquisar e selecione Pesquisar grupo de logs para abrir o prompt de pesquisa.
3. Siga as instruções selecionando um período para continuar.
4. Quando solicitado, insira o padrão de pesquisa no campo fornecido e pressione a tecla **Enter** para continuar ou a tecla **ESC** para cancelar a pesquisa.
5. Os resultados da pesquisa serão abertos no editor do VS Code quando a pesquisa for concluída.

Trabalhar com resultados de pesquisa de log

Ao concluir uma pesquisa bem-sucedida de grupo de logs do CloudWatch, os resultados são abertos no editor do VS Code. Os procedimentos a seguir descrevem como trabalhar com resultados de pesquisa de log.

 Note

Ao visualizar um único fluxo de logs, os recursos a seguir são limitados aos resultados do fluxo de logs ativo no momento.

Salvar os resultados da pesquisa de grupo de logs

Para salvar localmente os resultados da pesquisa de grupo de logs, siga as etapas abaixo.

1. Nos resultados da pesquisa do grupo de logs, selecione o botão do ícone Salvar log em arquivo, localizado no canto superior direito do editor do VS Code.
2. Na solicitação Salvar como, especifique o nome e o local em que você deseja salvar o arquivo.
3. Selecione OK para salvar o arquivo na máquina local.

Alterar o intervalo de tempo: período

Para alterar o período ativo nos resultados da pesquisa do grupo de logs, siga as etapas abaixo.

1. Nos resultados de pesquisa do grupo de logs, escolha o botão do ícone Pesquisar por data..., localizado no canto superior direito do editor do VS Code.
2. Na solicitação Selecionar filtro de tempo, escolha um novo período para os resultados da pesquisa de log.
3. Os resultados são atualizados quando a solicitação Selecionar filtro de tempo é fechada.

Alterar o padrão de pesquisa

Para alterar o padrão de pesquisa que está ativo nos resultados da pesquisa do grupo de logs, siga as etapas abaixo.

1. Nos resultados da pesquisa do grupo de logs, selecione o botão do ícone Pesquisar por padrão..., localizado no canto superior direito do editor do VS Code.
2. Na solicitação Pesquisar grupo de logs, insira o novo padrão de pesquisa no campo fornecido.
3. Pressione a tecla **Enter** para fechar o prompt e atualizar os resultados com o novo padrão de pesquisa.

Trabalhar com o Amazon Elastic Container Registry

O Amazon Elastic Container Registry (Amazon ECR) é um serviço de registro de contêiner gerenciado pela AWS que é seguro e escalável. Várias funções de serviço do Amazon ECR podem ser acessadas no Explorer do kit de ferramentas para VS Code.

- Criar um repositório.
- Criar um serviço do AWS App Runner para seu repositório ou imagem marcada.
- Acesso à etiqueta de imagem e aos URIs ou ARNs do repositório.
- Exclusão de etiquetas e repositórios de imagens.

Também é possível acessar a variedade completa de funções do Amazon ECR por meio do console do VS Code integrando a AWS CLI e outras plataformas ao VS Code.

Para obter mais informações sobre o Amazon ECR, consulte [What is Amazon ECR?](#) (O que é o Amazon ECR?) no Guia do usuário do Amazon Elastic Container Registry.

Pré-requisitos

Para acessar o serviço Amazon ECR no Explorer do VS Code, você precisa seguir as etapas abaixo.

Criar um usuário do IAM

Para poder acessar um serviço da AWS, como o Amazon ECR, você deve fornecer credenciais. Isso ocorre para que o serviço possa determinar se você tem permissão para acessar os recursos. Não recomendamos que você acesse a AWS diretamente por meio das credenciais da conta raiz da AWS. Em vez disso, use o AWS Identity and Access Management (IAM) para criar um usuário do IAM e, em seguida, adicione esse usuário a um grupo do IAM com permissões administrativas. Em seguida, você pode acessar a AWS usando um URL especial e as credenciais do usuário do IAM.

Se você se cadastrou na AWS, mas não criou um usuário do IAM, poderá criar um usando o console do IAM.

Para criar um usuário administrador, selecione uma das opções a seguir.

Selecionar uma forma de gerenciar o administrador	Para	Por	Você também pode
Centro de Identidade do IAM (Recomendado)	Use credenciais de curto prazo para acessar a AWS. Isso está de acordo com as práticas recomendadas de segurança. Para obter informações sobre as práticas recomendadas, consulte Práticas recomendadas de segurança no IAM no Guia do usuário do IAM.	Seguindo as instruções em Conceitos básicos no Guia do usuário do AWS IAM Identity Center.	Para configurar o acesso programático, consulte Configurar a AWS CLI para usar o AWS IAM Identity Center no Guia do usuário da AWS Command Line Interface.
No IAM (Não recomendado)	Use credenciais de curto prazo para acessar a AWS.	Seguindo as instruções em Criar o seu primeiro usuário administrador e um grupo de usuários do IAM no Guia do usuário do IAM para obter instruções.	Para configurar o acesso programático, consulte Gerenciamento de chaves de acesso de usuários do IAM no Guia do usuário do IAM.

Para fazer login como esse novo usuário do IAM, saia do Console da AWS e use o URL a seguir. No URL a seguir, `your_aws_account_id` é o número da conta da AWS sem os hifens (por exemplo, se o número da sua conta da AWS for 1234-5678-9012, o ID da sua conta da AWS será 123456789012):

```
https://your_aws_account_id.signin.aws.amazon.com/console/
```

Insira o nome e a senha de usuário do IAM que você acabou de criar. Quando você está conectado, a barra de navegação exibe "your_user_name @ your_aws_account_id".

Se você não quiser que a URL da página de cadastro contenha o ID da sua conta da AWS, crie um alias da conta. No painel do IAM, escolha Personalizar e insira um Alias da conta. O alias pode ser o nome da sua empresa. Para obter mais informações, consulte [O ID da sua conta da AWS e seu alias](#) no Guia do usuário do IAM.

Para fazer o login depois de criar o alias de uma conta, use o seguinte URL:

```
https://your_account_alias.signin.aws.amazon.com/console/
```

Para verificar o link de cadastro para usuários do IAM para a conta, abra o console do IAM e marque IAM users sign-in link no painel.

Para obter mais informações sobre o IAM, consulte o [Manual do usuário do AWS Identity and Access Management](#).

Instalar e configurar o Docker

Você pode instalar e configurar o Docker selecionando o sistema operacional da sua preferência no guia do usuário [Install Docker Engine](#) e seguindo as instruções.

Instalar e configurar a AWS CLI versão 2

Instale e configure a AWS CLI versão 2 selecionando o sistema operacional de sua preferência em [Instalar, atualizar e desinstalar a AWS CLI](#) no Guia do usuário da versão 2.

Tópicos

- [Trabalhar com o serviço Amazon Elastic Container Registry no VS Code](#)

Trabalhar com o serviço Amazon Elastic Container Registry no VS Code

Você pode acessar o serviço Amazon Elastic Container Registry (Amazon ECR) diretamente do AWS Explorer no VS Code e usá-lo para enviar uma imagem de programa para um repositório do Amazon ECR. Para começar, faça o seguinte:

1. Crie um Dockerfile que contenha as informações necessárias para criar uma imagem.
2. Crie uma imagem com base nesse Dockerfile e marque-a para processamento.
3. Crie um repositório dentro da instância do Amazon ECR.
4. Envie a imagem marcada ao repositório.

Seções

- [Pré-requisitos](#)
- [1. Criar um Dockerfile](#)
- [2. Criar uma imagem com base no Dockerfile](#)
- [3. Criar um repositório](#)
- [4. Enviar, extrair e excluir imagens](#)

Pré-requisitos

Para poder usar o recurso do serviço Amazon ECR do kit de ferramentas para VS Code, você deve atender a estes [pré-requisitos](#).

1. Criar um Dockerfile

O Docker usa um arquivo chamado Dockerfile para definir uma imagem que pode ser enviada e armazenada em um repositório remoto. Para carregar uma imagem em um repositório do ECR, primeiro você deve criar um Dockerfile e uma imagem com base nesse Dockerfile.

Criar um Dockerfile

1. Use o Explorer do kit de ferramentas para VS Code para acessar o diretório em que você deseja armazenar o Dockerfile.
2. Crie um arquivo chamado Dockerfile.

Note

O VS Code pode solicitar que você selecione um tipo ou extensão de arquivo. Se isso ocorrer, selecione texto sem formatação. O VS Code tem uma extensão “dockerfile”. No entanto, não recomendamos usá-la. Isso porque ela pode causar conflitos com determinadas versões do Docker ou outras aplicações associadas.

Editar o Dockerfile usando o VS Code

Se o Dockerfile tiver uma extensão de arquivo, abra o menu de contexto (clique com o botão direito do mouse) do arquivo e remova a extensão.

Depois que a extensão do arquivo for removida do Dockerfile:

1. Abra o Dockerfile vazio diretamente no VS Code.
2. Copie o conteúdo do exemplo a seguir no Dockerfile:

Exemplo Modelo de imagem do Dockerfile

```
FROM ubuntu:18.04

# Install dependencies
RUN apt-get update && \
    apt-get -y install apache2

# Install apache and write hello world message
RUN echo 'Hello World!' > /var/www/html/index.html

# Configure apache
RUN echo '. /etc/apache2/envvars' > /root/run_apache.sh && \
    echo 'mkdir -p /var/run/apache2' >> /root/run_apache.sh && \
    echo 'mkdir -p /var/lock/apache2' >> /root/run_apache.sh && \
    echo '/usr/sbin/apache2 -D FOREGROUND' >> /root/run_apache.sh && \
    chmod 755 /root/run_apache.sh

EXPOSE 80

CMD /root/run_apache.sh
```

Este é um Dockerfile que usa uma imagem do Ubuntu 18.04. As instruções RUN atualizam os caches do pacote. Instale os pacotes de software para o servidor Web e, depois, escreva o conteúdo “Hello World!” na raiz do documento do servidor Web. A instrução EXPOSE expõe a porta 80 do contêiner e a instrução CMD inicia o servidor Web.

3. Salve o Dockerfile.

⚠ Important

O Dockerfile não deve ter uma extensão anexada ao nome. Um Dockerfile com extensões pode causar conflitos com determinadas versões do Docker ou outras aplicações associadas.

2. Criar uma imagem com base no Dockerfile

O Dockerfile criado contém as informações necessárias para criar uma imagem para um programa. Para enviar essa imagem à instância do Amazon ECR, primeiro você deve criá-la.

Criar uma imagem com base no Dockerfile

1. Use a CLI do Docker ou uma CLI integrada à instância do Docker para acessar o diretório que contém o Dockerfile.
2. Execute o comando Docker build para criar a imagem definida no Dockerfile.

```
docker build -t hello-world .
```

3. Execute o comando Docker images para verificar se a imagem foi criada corretamente.

```
docker images --filter reference=hello-world
```

Example resultado do exemplo:

REPOSITORY	TAG	IMAGE ID	CREATED
hello-world	latest	e9ffedc8c286	4 minutes ago
SIZE			
241MB			

4.

Note

Esta etapa não é necessária para criar ou enviar a imagem, mas você pode ver como a imagem do programa funciona quando é executada.

Para executar a imagem que acabou de ser criada, use o comando Docker run.

```
docker run -t -i -p 80:80 hello-world
```

A opção `-p` especificada no exemplo anterior mapeia a porta 80 exposta no contêiner para a porta 80 do sistema host. Se você estiver executando o Docker localmente, acesse <http://localhost:80> pelo navegador. Se o programa for executado corretamente, aparecerá a mensagem "Hello World!".

Para obter mais informações sobre o comando Docker run, consulte [Docker Run reference](#) (Referência de execução do Docker) no site do Docker.

3. Criar um repositório

Para fazer upload de sua imagem na instância do Amazon ECR, crie um repositório onde ela possa ser armazenada.

Criar um repositório do Amazon ECR

1. Na Barra de atividades do VS Code, selecione o ícone do kit de ferramentas da AWS.
2. Expanda o menu do AWS Explorer.
3. Localize a região padrão da AWS associada à conta da AWS. Em seguida, selecione para ver uma lista dos serviços disponíveis por meio do kit de ferramentas para VS Code.
4. Escolha a opção ECR + para iniciar o processo Criar repositório.
5. Para realizar o processo, siga as instruções.
6. Após a conclusão, você poderá acessar o novo repositório na seção ECR do menu do AWS Explorer.

4. Enviar, extrair e excluir imagens

Depois de criar uma imagem do Dockerfile e criar um repositório, você poderá enviar a imagem ao repositório do Amazon ECR. Além disso, usando o AWS Explorer com o Docker e a AWS CLI, você pode fazer o seguinte:

- Enviar uma imagem pelo repositório.
- Excluir uma imagem que esteja armazenada no repositório.
- Excluir o repositório.

Autenticar o Docker com o registro padrão

A autenticação é necessária para trocar dados entre instâncias do Amazon ECR e do Docker. Para autenticar o Docker com o registro:

1. Abra um sistema operacional de linha de comando conectado à instância da AWS CLI.
2. Use o método `get-login-password` para se autenticar no registro privado do ECR.

```
aws ecr get-login-password --region region | docker login --username AWS --password-stdin AWS_account_id.dkr.ecr.region.amazonaws.com
```

Important

No comando anterior, você deve atualizar a **region** e o **AWS_account_id** com as informações específicas da sua conta da AWS.

Marcar e enviar uma imagem para o repositório

Depois de autenticar o Docker com a instância da AWS, envie uma imagem ao repositório.

1. Use o comando `Docker images` para visualizar as imagens armazenadas localmente e identificar aquela que você gostaria de marcar.

```
docker images
```

Example resultado do exemplo:

REPOSITORY SIZE	TAG	IMAGE ID	CREATED
hello-world 241MB	latest	e9ffedc8c286	4 minutes ago

2. Marque a imagem com o comando Docker tag.

```
docker tag hello-world:latest AWS_account_id.dkr.ecr.region.amazonaws.com/hello-world:latest
```

3. Envie a imagem marcada para o repositório com o comando Docker tag.

```
docker push AWS_account_id.dkr.ecr.region.amazonaws.com/hello-world:latest
```

Example resultado do exemplo:

```
The push refers to a repository [AWS_account_id.dkr.ecr.region.amazonaws.com/hello-world] (len: 1)
e9ae3c220b23: Pushed
a6785352b25c: Pushed
0998bf8fb9e9: Pushed
0a85502c06c9: Pushed
latest: digest:
  sha256:215d7e4121b30157d8839e81c4e0912606fca105775bb0636b95aed25f52c89b size: 6774
```

Após o upload bem-sucedido da imagem marcada para o repositório, ela ficará visível no menu do AWS Explorer.

Extrair uma imagem do Amazon ECR

- Você pode extrair uma imagem para a instância local do comando Docker tag.

```
docker pull AWS_account_id.dkr.ecr.region.amazonaws.com/hello-world:latest
```

Exemplo resultado do exemplo:

```
The push refers to a repository [AWS_account_id.dkr.ecr.region.amazonaws.com/hello-world] (len: 1)
e9ae3c220b23: Pushed
a6785352b25c: Pushed
0998bf8fb9e9: Pushed
0a85502c06c9: Pushed
latest: digest:
  sha256:215d7e4121b30157d8839e81c4e0912606fca105775bb0636b95aed25f52c89b size: 6774
```

Excluir uma imagem do repositório do Amazon ECR

Existem dois métodos para excluir uma imagem do VS Code. O primeiro método é usar o AWS Explorer.

1. No AWS Explorer, expanda o menu ECR.
2. Expanda o repositório do qual você deseja excluir uma imagem.
3. Selecione a tag da imagem associada à imagem que deseja excluir ao abrir o menu de contexto (clique com o botão direito do mouse).
4. Selecione a opção Excluir tag... para excluir todas as imagens armazenadas associadas a essa tag.

Excluir uma imagem usando a AWS CLI

- Também é possível excluir uma imagem do repositório com o comando `AWSecr batch-delete-image`.

```
AWS ecr batch-delete-image \  
  --repository-name hello-world \  
  --image-ids imageTag=latest
```

Exemplo resultado do exemplo:

```
{
  "failures": [],
  "imageIds": [
    {
      "imageTag": "latest",
      "imageDigest":
"sha256:215d7e4121b30157d8839e81c4e0912606fca105775bb0636b95aed25f52c89b"
    }
  ]
}
```

Excluir um repositório da instância do Amazon ECR

Existem dois métodos para excluir um repositório do VS Code. O primeiro método é usar o AWS Explorer.

1. No AWS Explorer, expanda o menu ECR.
2. Selecione o repositório que deseja excluir abrindo o menu de contexto (clique com o botão direito do mouse).
3. Selecione a opção Excluir repositório... para o repositório escolhido.

Excluir um repositório do Amazon ECR da AWS CLI

- É possível excluir um repositório com o comando AWS `ecr delete-repository`.

Note

Por padrão, não é possível excluir um repositório que contenha imagens. No entanto, o sinalizador `--force` permite a exclusão.

```
AWS ecr delete-repository \  
--repository-name hello-world \  
--force
```

Exemplo resultado do exemplo:

```
{
  "failures": [],
  "imageIds": [
    {
      "imageTag": "latest",
      "imageDigest":
"sha256:215d7e4121b30157d8839e81c4e0912606fca105775bb0636b95aed25f52c89b"
    }
  ]
}
```

Trabalhar com o Amazon Elastic Container Service

O AWS Toolkit for Visual Studio Code é compatível com o [Amazon Elastic Container Service \(Amazon ECS\)](#). O kit de ferramentas para VS Code auxilia em determinados trabalhos relacionados ao Amazon ECS, como a criação de definições de tarefas.

Tópicos

- [Usar arquivos de definição de tarefa do IntelliSense para Amazon ECS](#)
- [Executivo do Amazon Elastic Container Service em AWS Toolkit for Visual Studio Code](#)

Usar arquivos de definição de tarefa do IntelliSense para Amazon ECS

Uma das coisas que você pode fazer ao trabalhar com o Amazon Elastic Container Service (Amazon ECS) é criar definições de tarefa, conforme descrito em [Criação de uma definição de tarefa usando o console](#) no Guia do desenvolvedor do Amazon Elastic Container Service. Ao instalar o AWS Toolkit for Visual Studio Code, a instalação inclui a funcionalidade do IntelliSense para arquivos de definição de tarefas do Amazon ECS.

Pré-requisitos

- Verifique se o sistema atende aos pré-requisitos especificados em [Instalar o kit de ferramentas para VS Code](#).

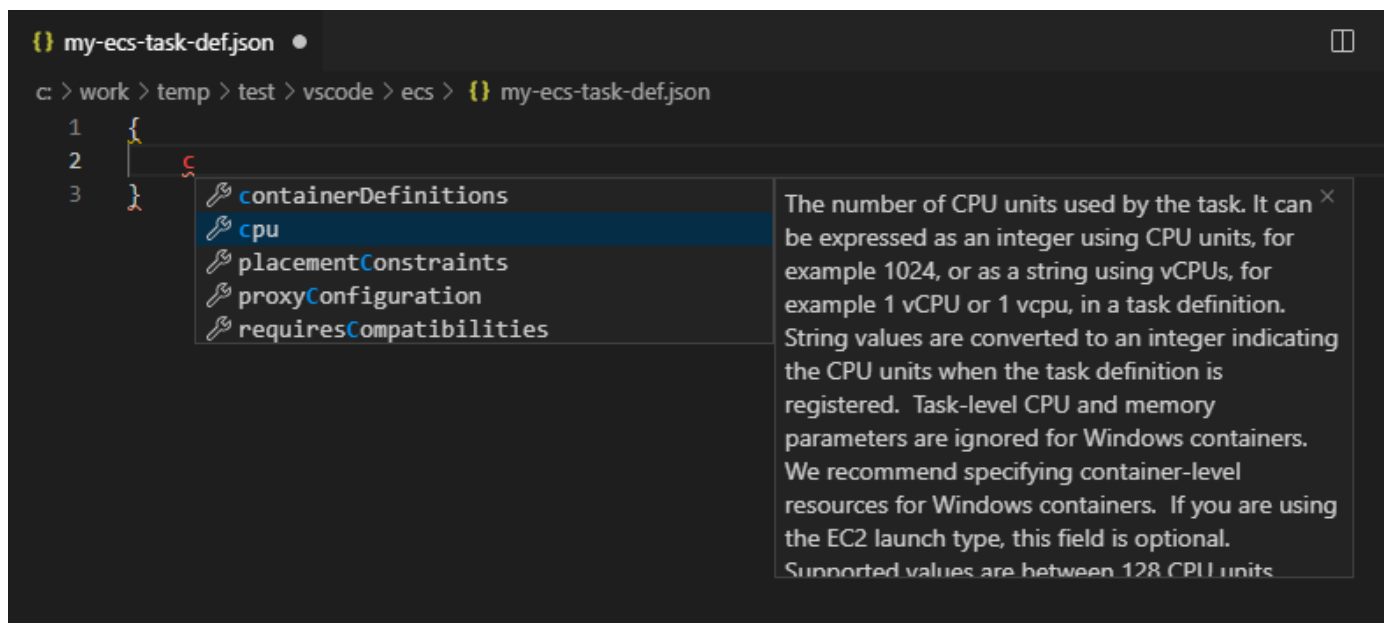
Usar o IntelliSense em arquivos de definição de tarefas do Amazon ECS

O exemplo a seguir mostra como você pode aproveitar as vantagens do IntelliSense em arquivos de definição de tarefas do Amazon ECS.

1. Crie um arquivo JSON para a definição de tarefas do Amazon ECS. O nome do arquivo deve conter `ecs-task-def.json` no final, mas pode ter caracteres adicionais no início.

Para este exemplo, crie um arquivo chamado `my-ecs-task-def.json`

2. Abra o arquivo em um editor do VS Code e insira as chaves iniciais.
3. Digite a letra “c” como se quisesse adicionar `cpu` à definição. Observe que a caixa de diálogo do IntelliSense é exibida e é semelhante ao que segue.



Executivo do Amazon Elastic Container Service em AWS Toolkit for Visual Studio Code

Você pode emitir comandos únicos em um contêiner do Amazon Elastic Container Service (Amazon ECS) com AWS Toolkit for Visual Studio Code o, usando o recurso Amazon ECS Exec.

⚠ Important

Ativar e desativar o Amazon ECS Exec altera o estado dos recursos em sua conta. AWS As alterações incluem interromper e reiniciar o serviço. Alterar o estado dos recursos enquanto o Amazon ECS Exec está habilitado pode levar a resultados imprevisíveis. Para obter mais

informações sobre o Amazon ECS Exec, consulte [Usar o Amazon ECS Exec para depuração](#) no Guia do desenvolvedor.

Pré-requisitos do Amazon ECS Exec

Para usar o recurso Amazon ECS Exec, você precisa atender a determinados pré-requisitos.

Requisitos do Amazon ECS

Dependendo se suas tarefas estão hospedadas no Amazon EC2 ou, o AWS Fargate (Fargate) Amazon ECS Exec tem requisitos de versão diferentes.

- Se você estiver usando o Amazon EC2, deverá usar uma AMI otimizada para Amazon ECS que tenha sido lançada após 20 de janeiro de 2021, com a versão 1.50.2 ou superior do agente. Informações adicionais estão disponíveis para você no guia do desenvolvedor das [AMIs otimizadas do Amazon ECS](#).
- Se você estiver usando AWS Fargate, deverá usar a versão 1.4.0 ou superior da plataforma. Informações adicionais sobre os requisitos do Fargate estão disponíveis no guia do desenvolvedor das [versões da plataforma do AWS Fargate](#).

AWS configuração da conta e permissões do IAM

Para usar o recurso Amazon ECS Exec, você precisa ter um cluster Amazon ECS existente associado à sua conta. AWS O Amazon ECS Exec usa o Systems Manager para estabelecer uma conexão com os contêineres no cluster e exige permissões específicas de tarefa do perfil do IAM.

Você pode encontrar informações sobre a política e o perfil do IAM, específicas do Amazon ECS Exec, no guia do desenvolvedor das [permissões do IAM necessárias para o ECS Exec](#).


Trabalhar com o Amazon ECS Exec

Você pode ativar ou desativar o Amazon ECS Exec diretamente do AWS Explorer no Toolkit for VS Code. Ao habilitar o Amazon ECS Exec, você poderá escolher contêineres no menu do Amazon ECS e executar comandos neles.

Habilitar o Amazon ECS Exec

1. No AWS Explorer, localize e expanda o menu Amazon ECS.


2. Expanda o cluster com o serviço que você deseja modificar.
3. Abra o menu de contexto (clique com o botão direito do mouse) do serviço e escolha Enable Command Execution (Habilitar execução de comandos).

 Important

Essa etapa inicia uma nova implantação do serviço e pode levar alguns minutos. Para obter mais informações, consulte a nota no início desta seção.

Desabilitar o Amazon ECS Exec

1. No AWS Explorer, localize e expanda o menu Amazon ECS.
2. Expanda o cluster que contém o serviço que você deseja.
3. Abra o menu de contexto (clique com o botão direito do mouse) do serviço e escolha Disable Command Execution (Desabilitar execução de comandos).

 Important

Essa etapa inicia uma nova implantação do serviço e pode levar alguns minutos. Para obter mais informações, consulte a nota no início desta seção.

Executar comandos referentes a um contêiner

Para executar comandos em um contêiner usando o AWS Explorer, o Amazon ECS Exec deve estar habilitado. Se não estiver habilitado, consulte o procedimento Habilitar o ECS Exec nesta seção.

1. No AWS Explorer, localize e expanda o menu Amazon ECS.
2. Expanda o cluster que contém o serviço que você deseja.
3. Expanda o serviço para listar os contêineres associados.
4. Abra o menu de contexto do contêiner (clique com o botão direito do mouse) e escolha Run Command in Container (Executar comando no contêiner).
5. Um prompt será aberto com uma lista de tarefas em execução. Selecione o ARN da tarefa que você deseja.

Note

Se apenas uma tarefa estiver em execução para esse serviço, ela será selecionada automaticamente e esta etapa será ignorada.

- Quando solicitado, insira o comando que você deseja executar e pressione Enter para continuar.

Trabalhar com o Amazon EventBridge

O AWS Toolkit for Visual Studio Code (VS Code) é compatível com o [Amazon EventBridge](#). Ao usar o kit de ferramentas para VS Code, você poderá trabalhar com determinados aspectos do EventBridge, como esquemas.

Tópicos

- [Trabalhando com EventBridge esquemas da Amazon](#)

Trabalhando com EventBridge esquemas da Amazon

Você pode usar o AWS Toolkit for Visual Studio Code (VS Code) para realizar várias operações nos [EventBridge esquemas da Amazon](#).

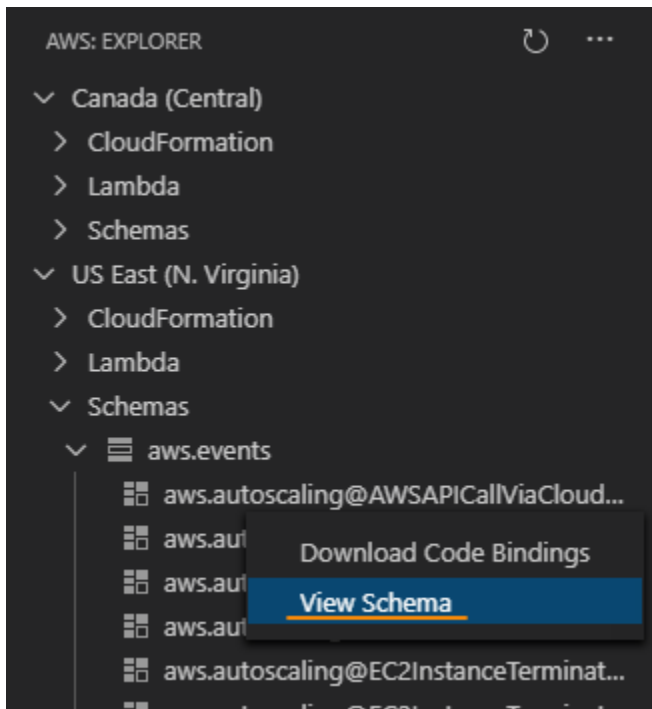
Pré-requisitos

- Verifique se o sistema atende aos pré-requisitos especificados em [Instalar o kit de ferramentas para VS Code](#).
- O EventBridge esquema com o qual você deseja trabalhar deve estar disponível em sua AWS conta. Se não estiver, crie um ou carregue-o. Consulte [EventBridge Esquemas da Amazon](#) no [Guia do EventBridge usuário da Amazon](#).

Visualizar um esquema disponível

- No AWS Explorer, expanda Schemas (Esquemas).
- Expanda o nome do registro que contém o esquema que deseja visualizar. Por exemplo, muitos dos esquemas fornecidos estão AWS no registro aws.events.

3. Para visualizar um esquema no editor, abra o menu de contexto do esquema e escolha View Schema (Visualizar esquema).

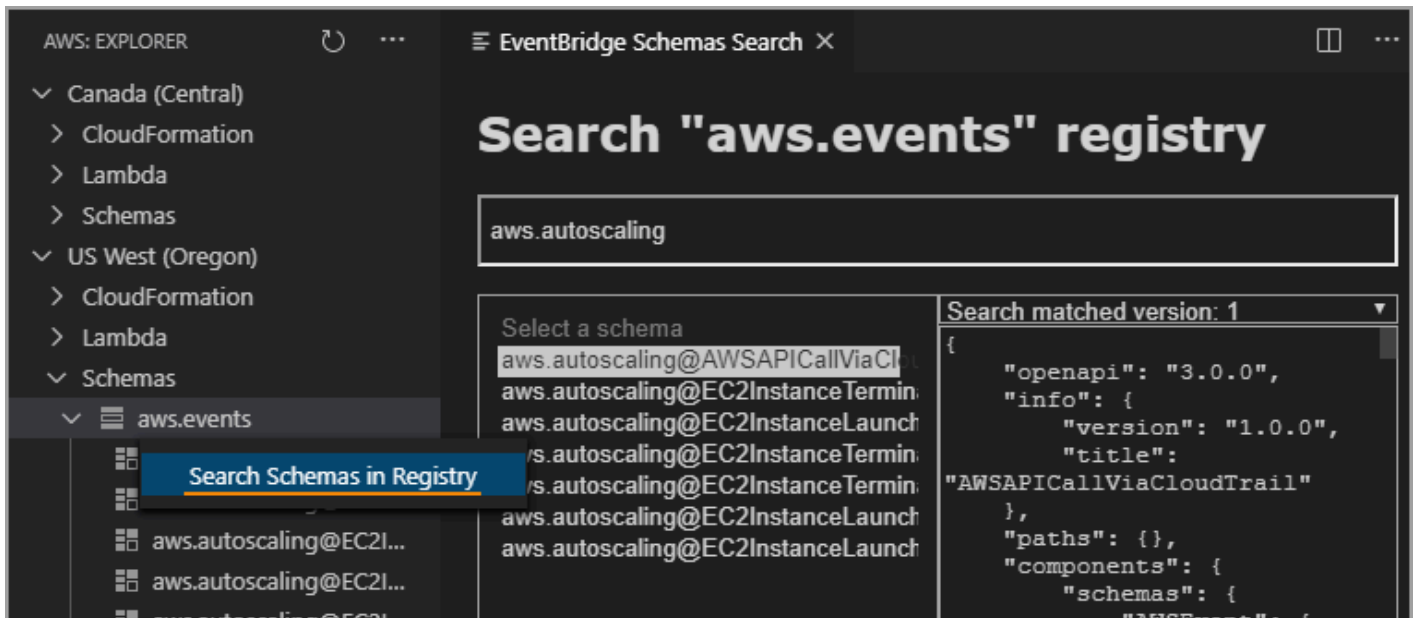


Localizar um esquema disponível

No AWS Explorer, execute uma ou mais das ações a seguir:

- Comece a digitar o título do esquema que deseja encontrar. O AWS Explorer destaca os títulos de esquemas que contêm uma correspondência. (Um registro deve ser expandido para ver os títulos destacados.)
- Abra o menu de contexto de Schemas (Esquemas) e escolha Search Schemas (Pesquisar esquemas). Ou expanda Schemas (Esquemas), abra o menu de contexto do registro que contém o esquema que deseja encontrar e escolha Search Schemas in Registry (Pesquisar esquemas no registro). Na caixa de diálogo Pesquisa de EventBridge esquemas, comece digitando o título do esquema que você deseja encontrar. A caixa de diálogo exibe os títulos do esquema que contêm uma correspondência.

Para exibir o esquema na caixa de diálogo, selecione o título do esquema.



Gerar código para um esquema disponível

1. No AWS Explorer, expanda Schemas (Esquemas).
2. Expanda o nome do registro que contém o esquema para o qual deseja gerar o código.
3. Clique com o botão direito no título do esquema e escolha Download code bindings (Fazer download de vinculações de códigos).
4. Nas páginas do assistente resultantes, escolha:
 - A Version (Versão) do esquema
 - A linguagem da vinculação do código
 - A pasta do workspace na qual deseja armazenar o código gerado na máquina de desenvolvimento local

Analizador de acesso do AWS IAM

Você pode executar verificações de política do [AWS Identity and Access Management \(IAM\) Access Analyzer](#) em suas IAM políticas criadas em AWS CloudFormation modelos, planos do Terraform e documentos de JSON políticas, usando o IAM Access Analyzer no. AWS Toolkit for Visual Studio Code

IAMAs verificações de política do Access Analyzer incluem validação de políticas e verificações de políticas personalizadas. A validação de políticas ajuda a validar suas IAM políticas de acordo com os padrões detalhados na [Gramática da linguagem da IAM JSON política](#) e nas [melhores práticas de AWS segurança nos IAM](#) tópicos, localizados no Guia do AWS Identity and Access Managementusuário. Suas descobertas de validação de políticas incluem avisos de segurança, erros, avisos gerais e sugestões de políticas.

Você também pode executar verificações de políticas personalizadas para novos acessos, com base em seus padrões de segurança. Uma cobrança é associada a cada verificação de política personalizada para novo acesso. Para obter informações detalhadas sobre preços, consulte o site de [preços do AWS IAM Access Analyzer](#). Para obter detalhes sobre as verificações de políticas do IAM Access Analyzer, consulte o tópico [Verificações de validação de políticas](#) no Guia do AWS Identity and Access ManagementUsuário.

Os tópicos a seguir descrevem como trabalhar com as verificações de política do IAM Access Analyzer no AWS Toolkit for Visual Studio Code.

Tópicos

- [Trabalhando com o AWS IAM Access Analyzer](#)

Trabalhando com o AWS IAM Access Analyzer

As seções a seguir descrevem como realizar a validação de IAM políticas e verificações personalizadas de políticas no AWS Toolkit for Visual Studio Code. Para obter detalhes adicionais, consulte os seguintes tópicos no Guia do AWS Identity and Access Management Usuário: [Validação da política do IAM Access Analyzer](#) e verificações de [políticas personalizadas do IAM Access Analyzer](#).

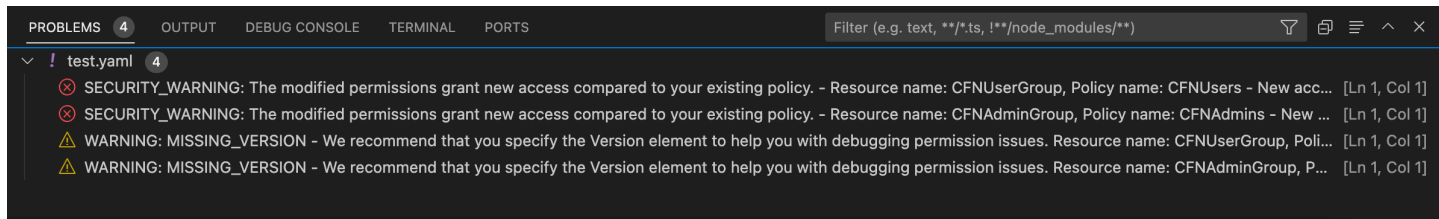
Pré-requisitos

Os pré-requisitos a seguir devem ser atendidos antes que você possa trabalhar com as verificações de políticas do IAM Access Analyzer do Toolkit.

- Instale o Python versão 3.6 ou posterior.
- Instale o [IAMPolicy Validator AWS CloudFormation](#) ou o [IAMPolicy Validator for Terraform](#), exigido pelas CLI ferramentas do Python e especificado na janela Policy Checks. IAM
- Configure suas credenciais de AWS função.

IAMVerificações de políticas do Access Analyzer

Você pode realizar verificações de políticas para AWS CloudFormation modelos, planos do Terraform e documentos JSON de política, usando o. AWS Toolkit for Visual Studio Code Suas descobertas de verificação podem ser visualizadas no Painel de problemas do VS Code. A imagem a seguir mostra o painel de problemas do VS Code.



IAM Access Analyzer fornece 4 tipos de verificações:

- Validar política
- CheckAccessNotGranted
- CheckNoNewAccess
- CheckNoPublicAccess

As seções a seguir descrevem como executar cada tipo de verificação.

Note

Configure suas credenciais de AWS função antes de executar qualquer tipo de verificação. Os arquivos compatíveis incluem os seguintes tipos de documentos: AWS CloudFormation modelos, planos do Terraform e documentos JSON de políticas

As referências de caminho de arquivo geralmente são fornecidas pelo administrador ou pela equipe de segurança e podem ser um caminho de arquivo do sistema ou um bucket do Amazon S3. URI Para usar um bucket do Amazon S3URI, sua função atual deve ter acesso ao bucket do Amazon S3.

Uma cobrança é associada a cada verificação de política personalizada. Para obter detalhes sobre preços de verificação de políticas personalizadas, consulte o guia de [preços do AWS IAM Access Analyzer](#).

Executando a política de validação

A verificação Validate Policy, também conhecida como validação de política, valida sua política em relação à gramática IAM política e AWS às melhores práticas. Para obter informações adicionais, consulte a [gramática da linguagem da IAM JSON política](#) e [as melhores práticas de AWS segurança nos IAM](#) tópicos, localizados no Guia do AWS Identity and Access Management usuário.

1. No VS Code, abra um arquivo compatível que contenha AWS IAM Políticas, no editor do VS Code.
2. Para abrir as verificações de política do IAM Access Analyzer, abra a Paleta de Comandos do VS Code pressionando **CRTL+Shift+P**, pesquise **IAM Policy Checks** e clique para abrir o painel Verificações de IAM políticas no editor do VS Code.
3. No painel Verificações de IAM políticas, selecione o tipo de documento no menu suspenso.
4. Na seção Validar políticas, escolha o botão Executar validação de política para executar a verificação Validar política.
5. No Painel de problemas do VS Code, analise suas descobertas de verificação de políticas.
6. Atualize sua política e repita esse procedimento, executando novamente a verificação Validar política até que as descobertas da verificação de política não exibam mais avisos ou erros de segurança.

Correndo CheckAccessNotGranted

CheckAccessNotGranted é uma verificação de política personalizada para verificar se IAM ações específicas não são permitidas pela sua política.

Note

As referências de caminho de arquivo geralmente são fornecidas pelo administrador ou pela equipe de segurança e podem ser um caminho de arquivo do sistema ou um bucket do Amazon S3. URI Para usar um bucket do Amazon S3URI, sua função atual deve ter acesso ao bucket do Amazon S3. Pelo menos uma ação ou recurso deve ser especificado e o arquivo deve ser estruturado de acordo com o exemplo a seguir:

```
    {"actions": ["action1", "action2", "action3"], "resources":  
    ["resource1", "resource2", "resource3"]}
```


1. No VS Code, abra um arquivo compatível que contenha AWS IAM Políticas, no editor do VS Code.
2. Para abrir as verificações de política do IAM Access Analyzer, abra a Paleta de Comandos do VS Code pressionando **CRTL+Shift+P**, pesquise **IAM Policy Checks** e clique para abrir o painel Verificações de IAM políticas no editor do VS Code.
3. No painel Verificações de IAM políticas, selecione o tipo de documento no menu suspenso.
4. Na seção Verificações de políticas personalizadas, selecione **CheckAccessNotGranted**.
5. No campo de entrada de texto, você pode inserir uma lista separada por vírgulas que contém ações e recursos. ARNs Pelo menos uma ação ou recurso deve ser fornecido.
6. Escolha o botão Executar verificação de política personalizada.
7. No Painel de problemas do VS Code, analise suas descobertas de verificação de políticas. As verificações de políticas personalizadas retornam um FAIL resultado PASS ou.
8. Atualize sua política e repita esse procedimento, executando novamente a **CheckAccessNotGranted** verificação até que ela retorne **PASS**.

Correndo CheckNoNewAccess

CheckNoNewAccess é uma verificação de política personalizada para verificar se sua política concede novo acesso em comparação com uma política de referência.

1. No VS Code, abra um arquivo compatível que contenha AWS IAM Políticas, no editor do VS Code.
2. Para abrir as verificações de política do IAM Access Analyzer, abra a Paleta de Comandos do VS Code pressionando **CRTL+Shift+P**, pesquise **IAM Policy Checks** e clique para abrir o painel Verificações de IAM políticas no editor do VS Code.
3. No painel Verificações de IAM políticas, selecione o tipo de documento no menu suspenso.
4. Na seção Verificações de políticas personalizadas, selecione **CheckNoNewAccess**.
5. Insira um documento de JSON política de referência. Como alternativa, você pode fornecer um caminho de arquivo que faça referência a um documento JSON de política.
6. Selecione o tipo de política de referência que corresponda ao tipo do seu documento de referência.

7. Escolha o botão Executar verificação de política personalizada.
8. No Painel de problemas do VS Code, analise suas descobertas de verificação de políticas. As verificações de políticas personalizadas retornam um FAIL resultado PASS ou.
9. Atualize sua política e repita esse procedimento, executando novamente a CheckNoNewAccess verificação até que ela retornePASS.

Correndo CheckNoPublicAccess

CheckNoPublicAccess é uma verificação de política personalizada para verificar se sua política concede acesso público aos tipos de recursos compatíveis em seu modelo.

Para obter informações específicas sobre os tipos de recursos compatíveis, consulte os [cloudformation-iam-policy-validator](#) e [terraform-iam-policy-validator](#) GitHub repositórios e.

1. No VS Code, abra um arquivo compatível que contenha AWS IAM Políticas, no editor do VS Code.
2. Para abrir as verificações de política do IAM Access Analyzer, abra a Paleta de Comandos do VS Code pressionando **CTRL+Shift+P**, pesquise **IAM Policy Checks** e clique para abrir o painel Verificações de IAM políticas no editor do VS Code.
3. No painel Verificações de IAM políticas, selecione o tipo de documento no menu suspenso.
4. Na seção Verificações de políticas personalizadas, selecione CheckNoPublicAccess.
5. Escolha o botão Executar verificação de política personalizada.
6. No Painel de problemas do VS Code, analise suas descobertas de verificação de políticas. As verificações de políticas personalizadas retornam um FAIL resultado PASS ou.
7. Atualize sua política e repita esse procedimento, executando novamente a CheckNoNewAccess verificação até que ela retornePASS.

Trabalhar com os AWS IoT no AWS Toolkit for Visual Studio Code

Com a AWS IoT no AWS Toolkit for Visual Studio Code, você pode interagir com o serviço AWS IoT e, ao mesmo tempo, reduzir as interrupções no fluxo de trabalho no VS Code. Este guia do usuário tem como objetivo ajudar a começar a usar os recursos do serviço AWS IoT disponíveis no AWS Toolkit for Visual Studio Code. Para obter mais informações sobre o serviço AWS IoT, consulte [O que é o AWS IoT?](#) no Guia do desenvolvedor

Pré-requisitos da AWS IoT

Para começar a usar o AWS IoT do kit de ferramentas para VS Code, verifique se a conta da AWS e o VS Code atendem aos requisitos destes guias:

- Para ver os requisitos de conta da AWS e as permissões de usuário da AWS específicas para o serviço AWS IoT, consulte [Conceitos básicos do AWS IoT Core](#) no Guia do desenvolvedor.
- Para ver os requisitos específicos do kit de ferramentas para VS Code, consulte [Setting up the Toolkit for VS Code](#) no Guia do usuário.

Coisas de AWS IoT

A AWS IoT conecta dispositivos a serviços e recursos de nuvem da AWS. Você pode conectar seus dispositivos ao AWS IoT usando objetos denominados coisas. Coisa é uma representação de um dispositivo específico ou entidade lógica. Ela pode ser um dispositivo físico ou sensor (por exemplo, uma lâmpada ou um interruptor em uma parede). Para obter mais informações sobre AWS IoT, consulte [Gerenciamento de dispositivos com o AWS IoT](#) no Guia do desenvolvedor.

Gerenciar coisas de AWS IoT

O kit de ferramentas para VS Code tem vários recursos que tornam o gerenciamento de coisas da AWS IoT mais eficiente. Estas são maneiras de usar o kit de ferramentas para VS Code para gerenciar coisas da AWS IoT:

- [Create a thing](#)
- [Attach a certificate to a thing](#)
- [Detach a certificate from a thing](#)
- [Delete a thing](#)

Como criar uma coisa

1. No AWS Explorer, expanda o título do serviço IoT e selecione o contexto (clique com o botão direito do mouse) em Coisas.
2. Para abrir uma caixa de diálogo, selecione Criar coisa no menu de contexto.
3. Siga a solicitação inserindo um nome para a coisa de IoT no campo Nome da coisa.

4. Quando essa etapa for concluída, um ícone de coisa seguido do nome que você especificou ficará visível na seção Coisa.

Como anexar um certificado a uma coisa

1. No AWS Explorer, expanda a seção do serviço IoT.
2. Na subseção Coisas, localize a coisa à qual você está anexando o certificado.
3. Selecione o contexto (clique com o botão direito do mouse) em coisa e escolha Anexar certificado no menu de contexto, para abrir um seletor de entrada com uma lista de certificados.
4. Na lista, selecione o ID do certificado correspondente ao certificado que você deseja anexar à coisa.
5. Quando essa etapa for concluída, o certificado ficará acessível no AWS Explorer como um item daquilo ao qual você o anexou.

Como desanexar um certificado de uma coisa

1. No AWS Explorer, expanda a seção do serviço IoT.
2. Na subseção Things (Coisas), localize a coisa da qual você deseja desanexar um certificado.
3. Selecione o contexto (clique com o botão direito do mouse) em coisa e selecione Desanexar certificado no menu de contexto.
4. Quando essa etapa for concluída, o certificado desvinculado não será mais exibido sob esse item no AWS Explorer, mas ainda estará acessível na subseção Certificados.

Como excluir uma coisa

1. No AWS Explorer, expanda a seção do serviço IoT.
2. Na subseção Coisas, localize a coisa que você deseja excluir.
3. Selecione o contexto (clique com o botão direito do mouse) na coisa e escolha Excluir coisa no menu de contexto para excluí-la.
4. Quando essa etapa for concluída, o item excluído não estará mais disponível na subseção Coisas.

Note

Observação: você só pode excluir uma coisa que não tenha um certificado anexado.

Certificados do AWS IoT

Os certificados são uma forma comum de criar uma conexão segura entre seus serviços e dispositivos de AWS IoT. Os certificados X.509 são certificados digitais que usam a infraestrutura de chave pública X.509 padrão para associar uma chave pública a uma identidade contida em um certificado. Para obter mais informações sobre certificados AWS IoT, consulte [Autenticação](#) no Guia do desenvolvedor.

Gerenciar certificados

O kit de ferramentas para VS Code oferece diversas maneiras de gerenciar certificados de AWS IoT, diretamente no AWS Explorer.

- [Create a certificate](#)
- [Change a certificate status](#)
- [Attach a policy to a certificate](#)
- [Delete a certificate](#)

Como criar um certificado de AWS IoT

Um certificado X.509 pode ser usado para realizar a conexão com a instância de AWS IoT.

1. No AWS Explorer, expanda a seção do serviço IoT e selecione o contexto (clique com o botão direito do mouse) em Certificados.
2. Selecione Criar certificado no menu de contexto para abrir uma caixa de diálogo.
3. Selecione um diretório no sistema de arquivos local para salvar o par de chaves RSA e o certificado X.509.

Note

- Os nomes de arquivo padrão contêm o ID do certificado como prefixo.

- Apenas o certificado X.509 é armazenado na conta da AWS, por meio do serviço AWS IoT.
- Seu par de chaves RSA só pode ser emitido uma vez, salve-o em um local seguro em seu sistema de arquivos quando solicitado.
- Se o certificado ou o par de chaves não puderem ser salvos no sistema de arquivos no momento, o kit de ferramentas da AWS excluirá o certificado da conta da AWS.

Como modificar o status de um certificado

O status de um certificado individual é exibido próximo ao ID no AWS Explorer e pode ser definido como ativo, inativo ou revogado.

Note

- Seu certificado precisa de um status ativo para poder usá-lo a fim de conectar seu dispositivo ao seu serviço AWS IoT.
- Um certificado inativo pode ser ativado, independentemente de ter sido desativado anteriormente ou estar inativo por padrão.
- Um certificado que foi revogado não pode ser reativado.

1. No AWS Explorer, expanda a seção do serviço IoT.
 2. Na subseção Certificados, localize o certificado que você deseja modificar.
 3. Selecione o contexto (clique com o botão direito do mouse) no certificado para abrir um menu de contexto que exibe as opções de alteração de status disponíveis para o certificado.
- Se um certificado tiver o status inativo, selecione activate (ativar) para alterar o status para ativo.
 - Se um certificado tiver o status ativo, selecione deactivate (desativar) para alterar o status para inativo.
 - Se um certificado tiver o status ativo ou inativo, selecione revoke (revogar) para alterar o status para revogado.

Note

Cada uma dessas ações de alteração de status estará disponível se você selecionar um certificado anexado a uma coisa enquanto estiver exibida na subseção Coisas.

Como anexar uma política de IoT a um certificado

1. No AWS Explorer, expanda a seção do serviço IoT.
2. Na subseção Certificados, localize o certificado que você deseja modificar.
3. Selecione o contexto (clique com o botão direito do mouse) no certificado para abrir o menu de contexto e selecione Anexar política para abrir um seletor de entrada com uma lista de políticas disponíveis.
4. Escolha a política que deseja anexar ao certificado.
5. Quando essa etapa for concluída, a política selecionada será adicionada ao certificado como um item de submenu.

Como anexar uma política de IoT de um certificado

1. No AWS Explorer, expanda a seção do serviço IoT.
2. Na subseção Certificados, localize o certificado que você deseja modificar.
3. Expanda o certificado e localize a política que você deseja desanexar.
4. Selecione o contexto (clique com o botão direito do mouse) na política e escolha Desanexar no menu de contexto.
5. Quando essa etapa for concluída, a política não estará mais acessível no certificado; ela estará disponível na subseção Política.

Para excluir um certificado

1. No AWS Explorer, expanda o título do serviço IoT.
2. Na subseção Certificados, localize o certificado que você deseja excluir.
3. Selecione o contexto (clique com o botão direito do mouse) no certificado e selecione Excluir certificado no menu de contexto.

Note

Não será possível excluir um certificado se ele estiver anexado a uma coisa ou tiver um status ativo. Você pode excluir um certificado que tenha políticas anexadas.

Políticas do AWS IoT

As políticas do AWS IoT Core são definidas por meio de documentos JSON, cada um contendo uma ou mais declarações de política. As políticas definem como AWS IoT, a AWS e seu dispositivo podem interagir entre si. Para obter mais informações sobre como criar um documento de política, consulte [Políticas do IoT](#) no Guia do desenvolvedor.

Note

As políticas nomeadas têm controle de versão para que você possa revertê-las. No AWS Explorer, as políticas do IoT estão relacionadas na subseção Políticas, no serviço IoT. Você pode visualizar as versões da política expandindo uma política. A versão padrão é indicada por um asterisco.

Gerenciamento de políticas

O kit de ferramentas para VS Code oferece várias maneiras de gerenciar as políticas do serviço AWS IoT. Essas são maneiras de gerenciar ou modificar suas políticas diretamente no AWS Explorer no VS Code:

- [Create a policy](#)
- [Upload a new policy version](#)
- [Edit a policy version](#)
- [Change the policy version default](#)
- [Change the policy version default](#)

Como criar uma política do AWS IoT

Note

Você pode criar uma política no AWS Explorer, mas o documento JSON que define a política já deve existir no sistema de arquivos.

1. No AWS Explorer, expanda a seção do serviço IoT.
2. Selecione o contexto (clique com o botão direito do mouse) na subseção Políticas e selecione Criar política usando o documento para abrir o campo de entrada Nome da política.
3. Digite um nome e siga as instruções para abrir uma caixa de diálogo solicitando que você selecione um documento JSON do seu sistema de arquivos.
4. Escolha o arquivo JSON que contém as definições de política. A política estará disponível no AWS Explorer após a conclusão.

Como fazer upload de uma nova versão da política de AWS IoT

Uma nova versão de uma política pode ser criada carregando um documento JSON para a política.

Note

O novo documento JSON deve estar presente em seu sistema de arquivos para criar uma versão com o uso do AWS Explorer.

1. No AWS Explorer, expanda a seção do serviço IoT.
2. Expandir a subseção Políticas para ver as políticas de AWS IoT
3. Selecione o contexto (clique com o botão direito do mouse) na política que você deseja atualizar e selecione Criar versão do documento.
4. Quando a caixa de diálogo for aberta, selecione o arquivo JSON que contém as atualizações de suas definições de política.
5. A nova versão pode ser acessada pela política no AWS Explorer.

Como editar uma versão de política de AWS IoT

Um documento de política pode ser aberto e editado usando o VS Code. Quando terminar de editar o documento, você poderá salvá-lo no sistema de arquivos. Em seguida, você pode carregá-lo no serviço AWS IoT por meio do AWSExplorer.

1. No AWS Explorer, expanda a seção do serviço IoT.
2. Expanda a subseção Políticas e localize a política que deseja atualizar. Crie uma política usando o documento para abrir o campo de entrada Nome da política.
3. Expanda a política que você deseja atualizar e selecione o contexto (clique com o botão direito do mouse) na versão da política que deseja editar.
4. Selecione Exibir no menu de contexto para abrir a versão da política no VS Code
5. Quando o documento de política for aberto, edite e salve as alterações desejadas.

Note

Nesse momento, as alterações feitas na política serão salvas apenas no sistema de arquivos local. Para atualizar a versão e rastreá-la com o AWS Explorer, repita as etapas descritas no procedimento [Upload a new policy version](#).

Como selecionar uma nova versão da política como padrão

1. No AWS Explorer, expanda a seção do serviço IoT.
2. Expanda a subseção Políticas (Políticas) e localize a política que você deseja atualizar.
3. Expanda a política que você deseja atualizar, selecione o contexto (clique com o botão direito do mouse) na versão da política que você deseja definir e escolha Definir como padrão.
4. Quando essa etapa for concluída, a nova versão padrão que você selecionou terá uma estrela ao lado.

Para excluir políticas do

Note

Antes de excluir uma política ou uma versão da política, as condições a seguir devem ser atendidas.

- Não será possível excluir uma política se ela estiver anexada a um certificado.
- Também não será possível excluir uma política se ela tiver alguma versão não padrão.
- Você só poderá excluir a versão padrão de uma política se uma nova versão padrão for selecionada ou se toda a política for excluída.
- Antes de excluir uma política inteira, você deve excluir toda a versão não padrão dessa mesma política.

1. No AWS Explorer, expanda a seção do serviço IoT.
2. Expanda a subseção Políticas (Políticas) e localize a política que você deseja atualizar.
3. Expanda a política que você deseja atualizar, selecione o contexto (clique com o botão direito do mouse) na versão da política que você deseja excluir e escolha Excluir.
4. Quando uma versão for excluída, ela não ficará mais visível no Explorer.
5. Se apenas a versão padrão de uma política for mantida, selecione o contexto (clique com o botão direito do mouse) na política principal e escolha Excluir.

Trabalhar com funções do AWS Lambda

O AWS Toolkit for Visual Studio Code oferece suporte para funções do [AWS Lambda](#). Com o kit de ferramentas para VS Code, você pode criar código para funções do Lambda que fazem parte de [aplicações sem servidor](#). Além disso, você pode invocar funções do Lambda localmente ou na AWS.

O Lambda é um serviço de computação totalmente gerenciado que executa código em resposta a eventos gerados por código personalizado ou em diversos serviços da AWS, como Amazon Simple Storage Service (Amazon S3), Amazon DynamoDB, Amazon Kinesis, Amazon Simple Notification Service (Amazon SNS) e Amazon Cognito.

Tópicos

- [Interagir de forma remota com as funções do Lambda](#)

Interagir de forma remota com as funções do Lambda

Ao usar o kit de ferramentas para VS Code, você poderá interagir com funções do [AWS Lambda](#) de várias maneiras, conforme descrito posteriormente neste tópico.

Para obter mais informações sobre o Lambda, consulte o [Manual do desenvolvedor do AWS Lambda](#).

Note

Se você já criou funções do Lambda usando o AWS Management Console ou de alguma outra forma, será possível invocá-las no Toolkit. Para criar uma função (usando o VS Code) que possa ser implantada no AWS Lambda, você deve primeiro [criar uma aplicação sem servidor](#).

Tópicos

- [Pré-requisitos](#)
- [Invocar uma função do Lambda](#)
- [Excluir uma função do Lambda](#)
- [Importar uma função do Lambda](#)
- [Carregar uma função do Lambda](#)

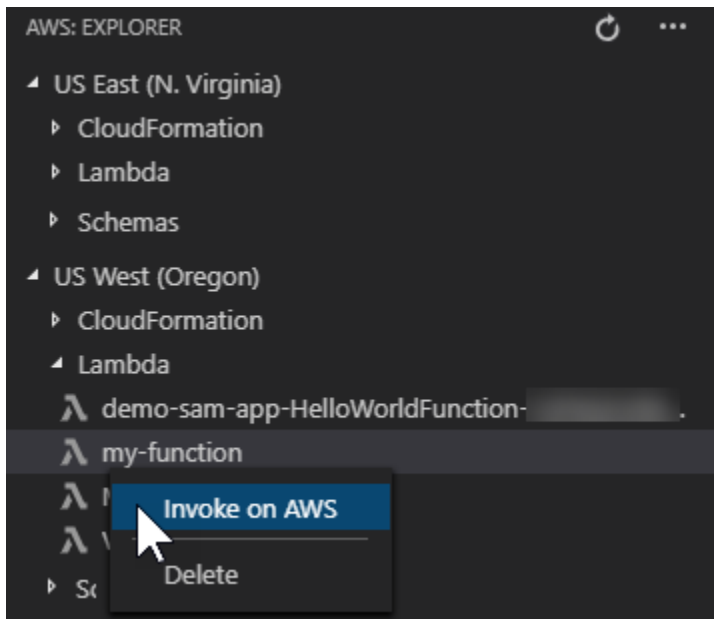
Pré-requisitos

- Verifique se o sistema atende aos pré-requisitos especificados em [Instalar o kit de ferramenta para VS Code](#).
- Certifique-se de que as credenciais que você configurou em [Autenticação e acesso](#) incluem acesso adequado de leitura/gravação ao serviço do AWS Lambda. Se no AWS Explorer, em Lambda, você vir uma mensagem semelhante a "Error loading Lambda resources" (Erro ao carregar os recursos do Lambda), verifique as permissões anexadas a essas credenciais. As alterações feitas nas permissões levarão alguns minutos para afetar o AWS Explorer no VS Code.

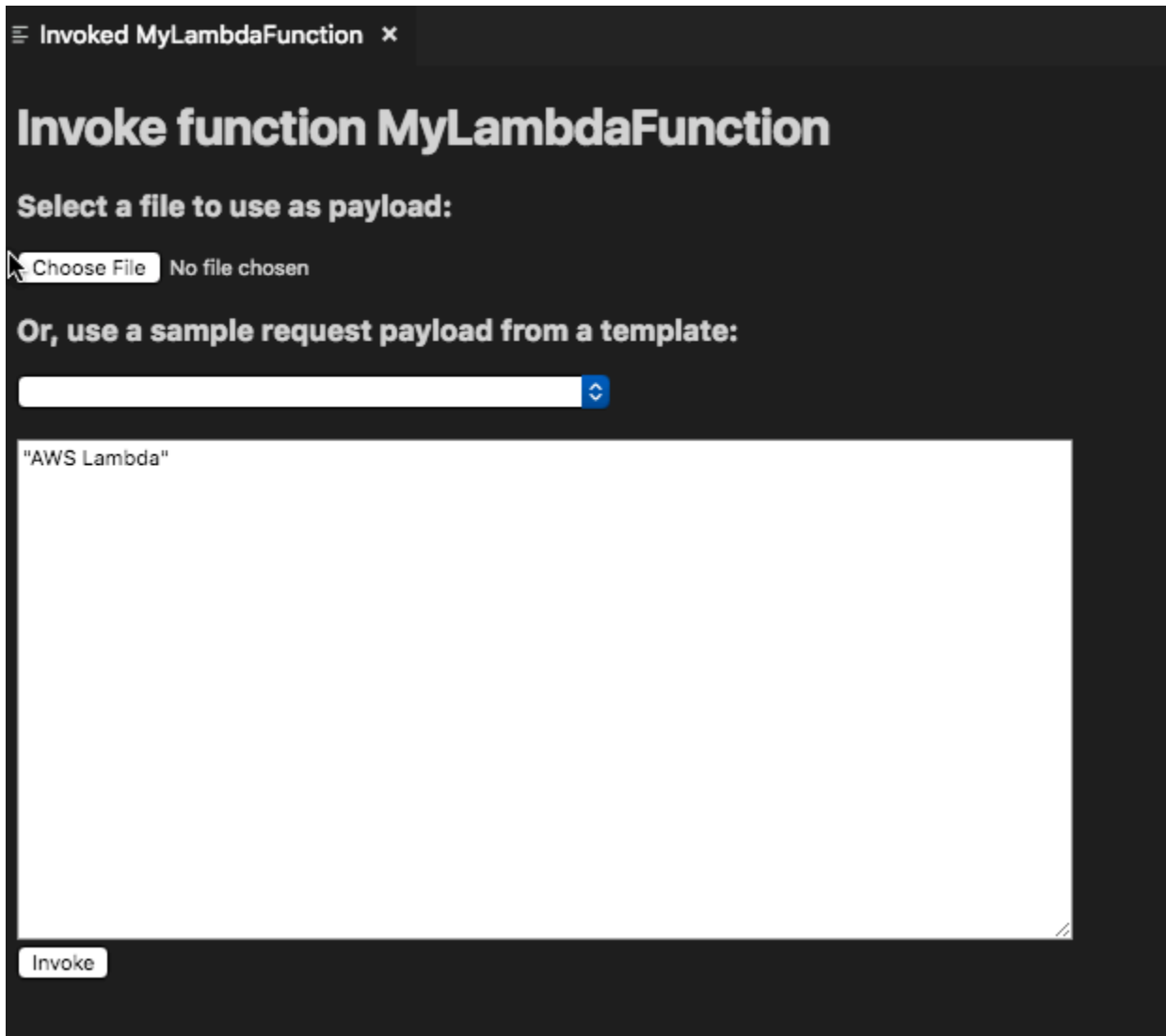
Invocar uma função do Lambda

Você pode invocar uma função do Lambda na AWS usando o kit de ferramentas para VS Code.

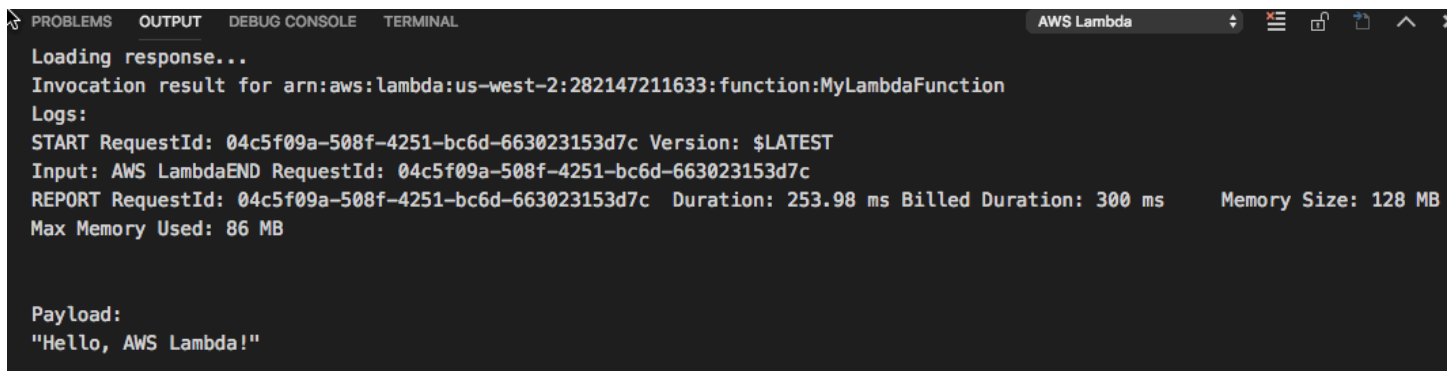
1. No AWS Explorer, escolha o nome da função do Lambda que você deseja invocar e abra o menu de contexto.



2. Selecione Invocar na AWS.
3. Na janela de invocação, insira a entrada necessária para a função do Lambda. A função do Lambda pode, por exemplo, exigir uma string como entrada, conforme mostrado na caixa de texto.



Você verá a saída da função do Lambda da mesma forma que veria em qualquer outro projeto usando o VS Code.



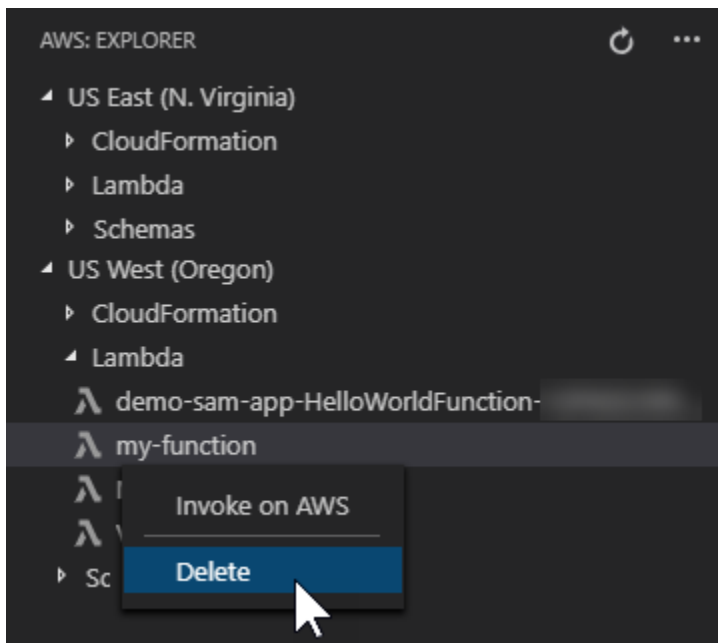
Excluir uma função do Lambda

Você também pode excluir uma função do Lambda usando o mesmo menu de contexto.

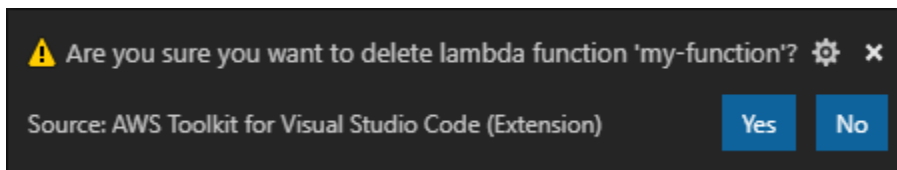
Warning

Não use este procedimento para excluir as funções do Lambda associadas ao [AWS CloudFormation](#) (por exemplo, a função do Lambda que foi criada ao [criar uma aplicação sem servidor](#) anteriormente neste guia). Essas funções devem ser excluídas por meio da pilha do AWS CloudFormation.

1. No AWS Explorer, escolha o nome da função do Lambda que você deseja excluir e abra o menu de contexto.



2. Escolha Delete (Excluir).
3. Na mensagem exibida, escolha Yes (Sim) para confirmar a exclusão.



Depois de excluída, a função deixa de ser listada no AWS Explorer.

Importar uma função do Lambda

Você pode importar remotamente o código de uma função do Lambda para o espaço de trabalho do VS Code para edição e depuração.

Note

O kit de ferramentas só é compatível com a importação de funções do Lambda usando runtimes Node.js e Python compatíveis.

1. No AWS Explorer, escolha o nome da função do Lambda que você deseja importar e abra o menu de contexto.
2. Escolha Importar...
3. Selecione uma pasta para importar o código do Lambda. Pastas fora do espaço de trabalho atual serão adicionadas ao espaço de trabalho atual.
4. Após o download, o kit de ferramentas adiciona o código ao espaço de trabalho e abre o arquivo que contém o código do manipulador do Lambda. O kit de ferramentas também cria uma configuração de inicialização, que aparece no painel de execução do VS Code para que você possa executar e depurar localmente a função do Lambda usando o AWS Serverless Application Model. Para obter mais informações sobre o uso de AWS SAM, consulte [the section called “Executar e depurar uma aplicação sem servidor usando um modelo \(local\)”](#).

Carregar uma função do Lambda

Você pode atualizar funções existentes do Lambda com código local. A atualização do código dessa forma não usa a função AWS SAM CLI para implantação e não cria uma pilha do AWS CloudFormation. Essa funcionalidade pode carregar uma função do Lambda com qualquer tempo de execução compatível com o Lambda.

Warning

O kit de ferramentas não pode verificar se o código funciona. Verifique se o código funciona antes de atualizar a produção de funções do Lambda.

1. No AWS Explorer, escolha o nome da função do Lambda que você deseja importar e abra o menu de contexto.
2. Selecione Upload Lambda... (Fazer upload do Lambda...)
3. Escolha entre as três opções para fazer o upload da função do Lambda. As opções incluem:

Carregar um arquivo .zip pré-criado

- Selecione Arquivo ZIP no menu.
- Escolha um arquivo .zip do sistema de arquivos e confirme o upload na caixa de diálogo modal. Isso carrega o arquivo .zip no estado em que se encontra e atualiza imediatamente o Lambda após a implantação.

Carregar um diretório como está

- Selecione Diretório no menu Seleção rápida.
- Selecione um diretório no sistema de arquivos.
- Selecione Não na solicitação para criar o diretório e confirme o upload com a caixa de diálogo modal. Isso carrega o diretório como está e atualiza imediatamente o Lambda após a implantação.

Criar e carregar um diretório

Note

Isso requer a AWS SAM CLI.

- Selecione Diretório no menu Seleção rápida.
- Selecione um diretório no sistema de arquivos.
- Selecione Sim na solicitação para criar o diretório e confirme o upload com a caixa de diálogo modal. Isso constrói o código no diretório usando o comando `AWS SAMCLIsam build` e atualiza imediatamente o Lambda após a implantação.

Note

O kit de ferramentas avisará se não conseguir detectar um manipulador correspondente antes do upload. Se desejar alterar o manipulador vinculado à função do Lambda, você poderá fazer isso por meio do AWS Management Console ou da AWS CLI.

Amazon Redshift no kit de ferramentas para VS Code

O Amazon Redshift é um serviço de data warehouse totalmente gerenciado e em escala de petabytes na Nuvem . Para obter informações detalhadas sobre o serviço Amazon Redshift, consulte o sumário dos guias do usuário do [Amazon Redshift](#).

Os tópicos a seguir descrevem como trabalhar com o Amazon Redshift no AWS Toolkit for Visual Studio Code.

Tópicos

- [Trabalhar com o Amazon Redshift usando o kit de ferramentas para VS Code](#)

Trabalhar com o Amazon Redshift usando o kit de ferramentas para VS Code

As seções a seguir descrevem como começar a trabalhar com o Amazon Redshift usando o AWS Toolkit for Visual Studio Code.

Para obter informações detalhadas sobre o serviço Amazon Redshift, consulte os tópicos do Guia do usuário do [Amazon Redshift](#).

Conceitos básicos

Antes de começar a trabalhar com o Amazon Redshift usando o AWS Toolkit for Visual Studio Code, os requisitos a seguir devem ser atendidos.

1. Você se conectou a uma ou mais contas da AWS pelo kit de ferramentas. Para obter mais informações sobre como conectar-se à conta da AWS pelo kit de ferramentas, consulte o tópico [Conectar-se à AWS](#) neste guia do usuário.

2. Você criou um data warehouse provisionado ou sem servidor.

Se você ainda não criou um cluster do Amazon Redshift sem servidor ou um cluster provisionado do Amazon Redshift, os procedimentos a seguir descrevem como criar um data warehouse com um conjunto de dados de amostra no Console da AWS.

Criar um data warehouse provisionado

Para obter mais informações sobre a criação de um data warehouse de cluster provisionado do Amazon Redshift, consulte o tópico [Criar um cluster de amostra do Amazon Redshift](#) no Guia de conceitos básicos do Amazon Redshift.

1. No navegador da internet de sua preferência, faça login no Console de Gerenciamento da AWS e abra o console do Amazon Redshift em <https://console.aws.amazon.com/redshift/>.
2. No console do Amazon Redshift, selecione Painel de clusters provisionados.
3. Em Painel de clusters provisionados, selecione o botão Criar cluster para abrir o painel Criar cluster.
4. Preencha os campos obrigatórios na seção Configuração do cluster.
5. Na seção Dados de exemplo, selecione a caixa Carregar dados de exemplo para carregar o conjunto de dados de amostra **Tickit** no banco de dados padrão **Dev** com o esquema **public**.
6. Na seção Configurações do banco de dados, insira valores nos campos Nome de usuário administrador e Senha do usuário administrador.
7. Selecione Criar cluster para criar o data warehouse provisionado.

Criar um data warehouse sem servidor

Para obter mais informações sobre a criação de um data warehouse do Amazon Redshift sem servidor, consulte a seção [Criar um data warehouse com o Amazon Redshift sem servidor](#) em Guia de conceitos básicos do Amazon Redshift.

1. No navegador da internet de sua preferência, faça login no Console de Gerenciamento da AWS e abra o console do Amazon Redshift em <https://console.aws.amazon.com/redshift/>.
2. No console do Amazon Redshift, selecione o botão Experimente o Amazon Redshift sem servidor para abrir o painel Conceitos básicos do Amazon Redshift sem servidor.
3. Na seção Configurações, selecione o radial Usar configurações padrão.

4. Na parte inferior do painel Conceitos básicos do Amazon Redshift sem servidor, selecione Salvar configuração para criar um data warehouse sem servidor com configurações padrão de grupo de trabalho, namespace, credencial e criptografia.

Conectar-se a um data warehouse pelo kit de ferramentas

Existem três métodos para conexão com um banco de dados por meio do kit de ferramentas:

- Nome de usuário e senha do banco de dados
- AWS Secrets Manager
- Credenciais temporárias

Para se conectar a um banco de dados localizado em um cluster provisionado existente ou data warehouse sem servidor usando o kit de ferramentas, siga as etapas abaixo.

Important

Se você seguiu as etapas na seção Pré-requisitos deste tópico do guia do usuário e o data warehouse não estiver visível no Explorer do kit de ferramentas, verifique se você está trabalhando na região correta da AWS no Explorer.

Conectar-se ao data warehouse com o método de Nome de usuário e senha do banco de dados

1. No Explorer do kit de ferramentas, expanda a Região da AWS em que o data warehouse está localizado.
2. Expanda o Redshift e selecione o data warehouse para abrir a caixa de diálogo Selecionar um tipo de conexão no VS Code.
3. Na caixa de diálogo Selecionar um tipo de conexão, escolha Nome de usuário e senha do banco de dados e forneça as informações exigidas por cada um dos prompts.
4. Os bancos de dados, tabelas e esquemas disponíveis ficam visíveis no Explorer do kit de ferramentas quando o kit de ferramentas se conecta ao data warehouse e o procedimento é concluído.

Conectar-se ao data warehouse com o AWS Secrets Manager

Note

Este procedimento requer um segredo de banco de dados do AWS Secrets Manager para ser concluído. Para obter instruções sobre como configurar um segredo de banco de dados, consulte [Criação de um segredo de banco de dados do AWS Secrets Manager](#) no Guia do usuário do AWS Secrets Manager.

1. No Explorer do kit de ferramentas, expanda a Região da AWS em que o data warehouse está localizado.
2. Expanda o Redshift e selecione o data warehouse para abrir a caixa de diálogo Selecionar um tipo de conexão no VS Code.
3. Na caixa de diálogo Selecionar um tipo de conexão, escolha Secrets Manager e forneça as informações exigidas por cada um dos prompts.
4. Os bancos de dados, tabelas e esquemas disponíveis ficam visíveis no Explorer do kit de ferramentas quando o kit de ferramentas se conecta ao data warehouse e o procedimento é concluído.

Conectar-se ao data warehouse usando credenciais temporárias

1. No Explorer do kit de ferramentas, expanda a região da AWS em que o data warehouse está localizado.
2. Expanda o Redshift e selecione o data warehouse para abrir a caixa de diálogo Selecionar um tipo de conexão no VS Code.
3. Na caixa de diálogo Selecionar um tipo de conexão, selecione Credenciais temporárias e forneça as informações exigidas por cada um dos prompts.
4. Os bancos de dados, tabelas e esquemas disponíveis ficam visíveis no Explorer do kit de ferramentas quando o kit de ferramentas se conecta ao data warehouse e o procedimento é concluído.

Editar a conexão usando o data warehouse

Você pode editar a conexão usando o data warehouse para alterar com qual banco de dados estabelecer conexão.

1. No Explorer do kit de ferramentas, expanda a Região da AWS em que o data warehouse está localizado.
2. Expanda o Redshift, clique com o botão direito no data warehouse ao qual você estabeleceu conexão, selecione Editar conexão e forneça o nome do banco de dados ao qual deseja se conectar.
3. Os bancos de dados, tabelas e esquemas disponíveis ficam visíveis no Explorer do kit de ferramentas quando o kit de ferramentas se conecta ao data warehouse e o procedimento é concluído.

Excluir a conexão usando o data warehouse

1. No Explorer do kit de ferramentas, expanda a Região da AWS em que o data warehouse está localizado.
2. Expanda o Redshift, clique com o botão direito no data warehouse usando a conexão que deseja excluir e selecione Excluir conexão. Essa ação remove os bancos de dados, tabelas e esquemas disponíveis do Explorer do kit de ferramentas.
3. Para se reconectar ao data warehouse, selecione Clique para se conectar e forneça as informações exigidas por cada um dos prompts. Por padrão, a reconexão usa o método anterior de autenticação para estabelecer a conexão com o data warehouse. Para usar outro método, selecione a seta para voltar na caixa de diálogo até chegar ao prompt de autenticação.

Executar instruções SQL

Os procedimentos a seguir descrevem como criar e executar instruções SQL no banco de dados usando o AWS Toolkit for Visual Studio Code.

Note

Para seguir as etapas em cada um dos procedimentos, primeiro você deve concluir a seção Conectar-se a um data warehouse pelo kit de ferramentas, localizada neste tópico do guia do usuário.

1. No Explorer do kit de ferramentas, expanda o Redshift e, em seguida, expanda o data warehouse que contém o banco de dados que você deseja consultar.

2. Selecione Create-Notebook para especificar um nome de arquivo e um local para armazenar o bloco de anotações localmente e, em seguida, escolha OK para abrir o bloco de anotações no editor do VS Code.
3. No editor do VS Code, insira as instruções SQL que deseja armazenar nesse bloco de anotações.
4. Selecione o botão Executar tudo para executar as instruções SQL que você inseriu.
5. A saída para as instruções SQL é exibida abaixo das instruções que você inseriu.

Adicionar Markdown no bloco de anotações

1. No bloco de anotações no editor do VS Code, selecione o botão Markdown para adicionar uma célula Markdown ao bloco de anotações.
2. Insira o Markdown na célula fornecida.
3. A célula Markdown pode ser editada usando as ferramentas do editor localizadas no canto superior direito da célula Markdown.

Adicionar código ao bloco de anotações

1. No bloco de anotações no editor do VS Code, selecione o botão Código para adicionar uma célula de código ao bloco de anotações.
2. Insira o código na célula fornecida.
3. Você pode executar o código acima ou abaixo da célula Código selecionando o botão apropriado nas ferramentas do editor de célula, localizado no canto superior direito da célula Código.

Trabalhar com o Amazon S3

Amazon Simple Storage Service (Amazon S3) é um serviço escalável de armazenamento de dados. O AWS Toolkit for Visual Studio Code permite gerenciar objetos e recursos do Amazon S3 diretamente no VS Code.

Para obter informações detalhadas sobre o serviço Amazon S3, consulte o Guia do usuário do [Amazon S3](#).

Os tópicos a seguir descrevem como trabalhar com objetos e recursos do Amazon S3 por meio do AWS Toolkit for Visual Studio Code.

Tópicos

- [Trabalhar com recursos do Amazon S3](#)
- [Trabalhar com objetos do Amazon S3](#)

Trabalhar com recursos do Amazon S3

Você pode usar o Amazon S3 a partir do AWS Toolkit for Visual Studio Code para visualizar, gerenciar e editar seus buckets e outros recursos do Amazon S3.

As seções a seguir descrevem como trabalhar com recursos do Amazon S3 no AWS Toolkit for Visual Studio Code. Para obter informações sobre como trabalhar com objetos do Amazon S3, como pastas e arquivos, a partir do AWS Toolkit for Visual Studio Code, consulte o tópico [Trabalhando com objetos do S3](#) neste Guia do usuário.

Como criar um bucket do Amazon S3

1. No Explorer do kit de ferramentas, abra o menu de contexto (clique com o botão direito do mouse) do serviço S3 e selecione Criar bucket.... Ou escolha o ícone Criar bucket para abrir a caixa de diálogo Criar bucket.
2. No campo Bucket name (Nome do bucket), insira um nome para o bucket.

Pressione Enter para criar o bucket e fechar a caixa de diálogo. O novo bucket é exibido no serviço S3 no kit de ferramentas.

Note

Como o Amazon S3 permite que seu bucket seja usado como um URL que pode ser acessado publicamente, o nome do bucket que você escolher deve ser globalmente exclusivo. Se outra conta já tiver criado um bucket com o nome que você deseja usar, você deverá usar outro nome.

Se você não conseguir criar um bucket, verifique os Logs do kit de ferramentas da AWS na guia Saída. Se você tentar usar um nome de bucket inválido, ocorrerá o erro `BucketAlreadyExists`.

Para obter mais informações, consulte [Restrições e limitações de bucket](#) no Manual do usuário do Amazon Simple Storage Service.

Adição de uma pasta a um bucket do Amazon S3

Você pode organizar o conteúdo de um bucket S3 agrupando os objetos em pastas. Você também pode criar pastas dentro de pastas.

1. No Explorer do kit de ferramentas, expanda o serviço S3 para ver uma lista de recursos do S3.
2. Escolha o ícone Criar pasta para abrir a caixa de diálogo Criar pasta. Abra o menu de contexto (clikando com o botão direito do mouse) de um bucket ou uma pasta e selecione Criar pasta.
3. Insira um valor no campo Nome da pasta e pressione Enter para criar a pasta e fechar a caixa de diálogo. A nova pasta é exibida no recurso do S3 correspondente no menu do kit de ferramentas.

Excluir um bucket do Amazon S3

Ao excluir um bucket do S3, você também exclui as pastas e os objetos contidos nele. Portanto, ao tentar excluir um bucket, o sistema pede para confirmar se você deseja excluí-lo.

1. No menu principal do kit de ferramentas, expanda o serviço Amazon S3 para ver uma lista dos recursos do S3.
2. Abra o menu de contexto (clikando com o botão direito do mouse) de um bucket ou uma pasta e selecione Excluir bucket do S3.
3. Quando receber a solicitação, insira o nome do bucket no campo de texto e pressione Enter para excluir o bucket e fechar a solicitação de confirmação.

Note

Se o bucket contiver objetos, ele será esvaziado antes de ser excluído. Se você tentar excluir um grande número de recursos ou objetos de uma vez, poderá levar algum tempo para que sejam excluídos. Após a exclusão, você receberá uma notificação informando que eles foram excluídos com sucesso.

Trabalhar com objetos do Amazon S3

Os arquivos, pastas e outros dados armazenados no bucket de recursos do S3 são conhecidos como objetos do S3.

As seções a seguir descrevem como trabalhar com objetos do Amazon S3 por meio do AWS Toolkit for Visual Studio Code. Para obter mais informações sobre como trabalhar com recursos do Amazon S3, como buckets do S3, pelo AWS Toolkit for Visual Studio Code, consulte o tópico [Trabalhar com recursos do S3](#) neste guia do usuário.

Paginação de objetos

Se você estiver trabalhando com um grande número de objetos e pastas do Amazon S3, a paginação permitirá especificar o número de itens que deseja exibir em uma página.

1. Acesse a Barra de atividades do VS Code e selecione Extensões.
2. Na extensão do kit de ferramentas da AWS, escolha o ícone de configurações e selecione Configurações de extensão.
3. Na página Configurações, role para baixo até a configuração AWS > S3: máximo de itens por página.
4. Altere o valor padrão do número de itens do S3 que você deseja exibir antes que a mensagem “carregar mais” seja exibida.

Note

Os valores válidos incluem qualquer número entre 3 e 1.000. Essa configuração se aplica somente ao número de objetos ou pastas exibidos ao mesmo tempo. Todos os buckets que você criou são exibidos de uma vez. Por padrão, você pode criar até 100 buckets em cada conta da AWS.

5. Feche a página Configurações para confirmar as alterações.

Você também pode atualizar as configurações em um arquivo formatado em JSON escolhendo o ícone Abrir configurações (JSON) no canto superior direito da página Configurações.


Carregar e baixar objetos do Amazon S3

Você pode carregar arquivos armazenados localmente para buckets do Amazon S3 ou baixar objetos remotos do Amazon S3 no sistema local usando o AWS Toolkit for Visual Studio Code.

Carregar um arquivo usando o kit de ferramentas

1. No Explorer do kit de ferramentas, expanda o serviço Amazon S3 para ver uma lista dos recursos do S3.

2. Selecione o ícone Carregar arquivo localizado próximo a um bucket ou pasta para abrir a caixa de diálogo Carregar arquivo. Ou você pode abrir o menu de contexto (clique com o botão direito do mouse) e selecionar Carregar arquivo.

 Note

Para carregar um arquivo na pasta ou no recurso principal do objeto, abra o menu de contexto (clique com o botão direito do mouse) de qualquer objeto do S3 e selecione Carregar principal.

3. Use o gerenciador de arquivos do sistema para selecionar um arquivo e escolha Carregar arquivo para fechar a caixa de diálogo e carregar o arquivo.

Carregar um arquivo usando a paleta de comandos

Você pode usar a interface do kit de ferramentas ou a paleta de comandos para carregar um arquivo em um bucket.

1. Para selecionar um arquivo para carregar, selecione a guia do arquivo no VS Code.
2. Pressione Ctrl + Shift + P para exibir o painel Paleta de comandos.
3. Na Paleta de comandos, insira a frase `upload file` para exibir uma lista dos comandos recomendados.
4. Escolha o comando `AWS:carregar arquivo` para abrir a caixa de diálogo `AWS: carregar arquivo`.
5. Quando solicitado, escolha o arquivo que deseja carregar e escolha o bucket no qual deseja carregá-lo.
6. Confirme o upload para fechar a caixa de diálogo e iniciar o processo de upload. Quando o upload for concluído, o objeto será exibido no menu do kit de ferramentas com metadados que incluem tamanho do objeto, data da última modificação e caminho.

Baixar um objeto do Amazon S3

1. No Explorer do kit de ferramentas, expanda o serviço S3.
2. Em um bucket ou pasta, abra o menu de contexto (clique com o botão direito do mouse) de um objeto que você deseja baixar. Em seguida, selecione Fazer download como para abrir a caixa de diálogo Fazer download como. Ou escolha o ícone Fazer download como próximo ao objeto.

3. Ao usar o gerenciador de arquivos do sistema, selecione uma pasta de destino, insira um nome de arquivo e escolha Fazer download para fechar a caixa de diálogo e iniciar o download.

Editar objetos remotos

Você pode usar o AWS Toolkit for Visual Studio Code para editar os objetos do Amazon S3 que estão armazenados nos recursos remotos do Amazon S3.

1. No Explorer do kit de ferramentas, expanda o serviço S3.
2. Expanda o recurso do S3 que contém o arquivo que você deseja editar.
3. Para editar o arquivo, escolha o ícone de lápis (Editar arquivo).
4. Para editar um arquivo aberto no modo somente leitura, visualize o arquivo no editor do VS Code e escolha o ícone de lápis localizado no canto superior direito da interface de usuário.

Note

- Se você reiniciar ou sair do VS Code, o IDE se desconectará dos recursos do S3. Se algum arquivo remoto do S3 estiver sendo editado quando você se desconectar, a edição será interrompida. Você deverá reiniciar o VS Code e reabrir a guia de edição para retomar a edição.
- O botão Editar arquivo fica no canto superior direito da interface de usuário. Ele só fica visível quando você está visualizando ativamente um arquivo somente leitura no editor do VS Code.
- Arquivos que não são de texto não podem ser abertos no modo somente leitura. Eles sempre abrem no modo de edição.
- Não é possível voltar para o modo somente leitura do modo somente edição, apenas o contrário.

Copiar o caminho de um objeto do Amazon S3

O procedimento a seguir descreve como copiar o caminho de um objeto do Amazon S3 no AWS Toolkit for Visual Studio Code.

1. No Explorer do kit de ferramentas, expanda o serviço S3.

2. Expanda o bucket de recursos que contém o objeto cujo caminho você deseja copiar.
3. Abra o menu de contexto (clique com o botão direito do mouse) do objeto para o qual você deseja copiar o caminho e selecione Copiar caminho para copiar o caminho do objeto na área de transferência local.

Gerar um presigned URL para um objeto do Amazon S3

Você pode compartilhar objetos privados do Amazon S3 com outras pessoas concedendo permissões por tempo limitado para downloads por meio do recurso de URL pré-assinado. Para obter mais informações, consulte [Compartilhar objetos com URLs pré-assinados](#).

1. No Explorer do kit de ferramentas, expanda o serviço S3.
2. Em um bucket ou pasta, abra o menu de contexto (clique com o botão direito do mouse) de um objeto que você deseja compartilhar. Em seguida, selecione Gerar URL pré-assinado para abrir a paleta de comandos.
3. Na paleta de comandos, insira o número de minutos em que o URL pode ser usado para acessar o objeto. Em seguida, selecione Enter para confirmar e fechar a caixa de diálogo.
4. Depois que o URL pré-assinado é gerado, a Barra de status do VS Code exibe o URL pré-assinado do objeto que foi copiado para a área de transferência local.

Excluir um objeto do Amazon S3

Se um objeto estiver em um bucket sem versão, você poderá excluí-lo permanentemente. No caso de buckets com versionamento habilitado, uma solicitação de exclusão não exclui permanentemente esse objeto. Em vez disso, o Amazon S3 insere um marcador de exclusão no bucket. Para obter mais informações, consulte [Excluir versões de objeto](#).

1. No Explorer do kit de ferramentas, expanda o serviço S3 para ver uma lista de recursos do S3.
2. Abra o menu de contexto (clique com o botão direito do mouse) de um objeto que você deseja excluir e selecione Excluir para abrir a caixa de diálogo de confirmação.
3. Selecione Excluir. . . para confirmar que deseja excluir o objeto S3. Em seguida, feche a caixa de diálogo.

Trabalhar com aplicações sem servidor

O AWS Toolkit for Visual Studio Code oferece suporte para [AWS Serverless Application](#). Os tópicos a seguir descrevem como começar a criar e trabalhar com aplicações AWS Serverless Application Model (AWS SAM), com base no AWS Toolkit for Visual Studio Code.

Tópicos

- [Conceitos básicos do uso de aplicações sem servidor](#)
- [Executando e depurando funções do Lambda enquanto exclui recursos de modelo AWS SAM](#)
- [Executar e depurar recursos locais do Amazon API Gateway](#)
- [Opções de configuração para depurar aplicações sem servidor](#)
- [Solucionar problemas de aplicações sem servidor](#)

Conceitos básicos do uso de aplicações sem servidor

As seções a seguir descrevem como começar a criar um AWS Serverless Application por meio do AWS Toolkit for Visual Studio Code usando o AWS Serverless Application Model (AWS SAM) e pilhas do AWS CloudFormation.

Pré-requisitos

Antes de poder criar ou trabalhar com um AWS Serverless Application, os pré-requisitos a seguir devem ser atendidos.

Note

As operações a seguir podem exigir que você saia ou reinicie o VS Code antes que as alterações sejam concluídas.

- Instalar a interface de linha de comandos (CLI) do AWS SAM. Para obter mais informações e instruções sobre como instalar a CLI do AWS SAM, consulte o tópico [Instalar a CLI do AWS SAM](#) neste Guia do usuário do AWS Serverless Application Model.
- No arquivo de configuração AWS, identifique a região da AWS padrão. Para obter mais informações sobre o arquivo de configuração, consulte o tópico [Configurações de arquivos de configuração e credenciais](#) no Guia do usuário da versão 2 da AWS Command Line Interface.

- Instale o SDK do seu idioma e configure a cadeia de ferramentas. Para obter mais informações sobre como configurar a cadeia de ferramentas usando o AWS Toolkit for Visual Studio Code, consulte o tópico [Configurar uma cadeia de ferramentas](#) neste guia do usuário.
- Instale a [extensão de suporte à linguagem YAML](#) no VS Code Marketplace. Isso é necessário para que o recurso CodeLens de arquivos de modelo do AWS SAM esteja acessível. Para obter mais informações sobre o CodeLens, consulte a seção [CodeLens](#) na documentação do VS Code

Permissões do IAM para aplicações sem servidor

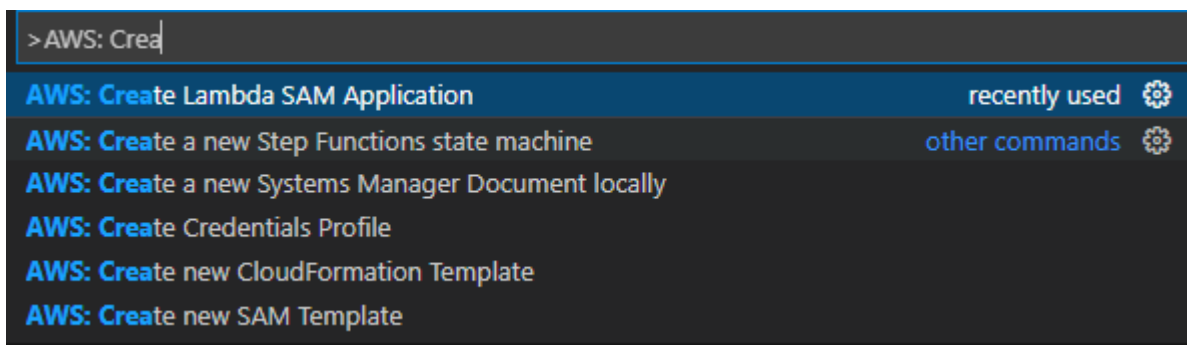
No kit de ferramentas para VS Code, você deve ter um perfil de credenciais que contenha as permissões do AWS Identity and Access Management (IAM) necessárias para implementar e executar aplicações sem servidor. Você deve ter acesso adequado de leitura/gravação aos seguintes produtos: AWS CloudFormation, IAM, Lambda, Amazon API Gateway, Amazon Simple Storage Service (Amazon S3) e Amazon Elastic Container Registry (Amazon ECR).

Para obter mais informações sobre como configurar a autenticação necessária para implantar e executar aplicações sem servidor, consulte [Managing resource access and permissions](#) no Guia do desenvolvedor do AWS Serverless Application Model. Para obter informações sobre como configurar credenciais, consulte [AWS IAMcredenciais](#) neste guia do usuário.

Criar uma aplicação sem servidor (local)

Este procedimento mostra como criar uma aplicação sem servidor com o kit de ferramentas para VS Code usando o AWS SAM. A saída deste procedimento é um diretório local no host de desenvolvimento contendo um exemplo de aplicação sem servidor, que você pode criar, testar localmente, modificar e implantar na Nuvem AWS.

1. Para abrir a paleta de comandos, selecione Exibir, Paleta de comandos e digite AWS.
2. Escolha Kit de ferramentas da AWS: criar aplicação Lambda SAM.



Note

Se a CLI do AWS SAM não estiver instalada, você verá um erro no canto inferior direito do editor do VS Code. Se isso acontecer, verifique se todos os [pré-requisitos e suposições](#) foram atendidos.

3. Selecione o runtime da aplicação do AWS SAM.

Note

Se você selecionar um dos tempos de execução com “(Image)”, o tipo de pacote da aplicação será Image. Se você selecionar um dos tempos de execução sem “(Image)”, o tipo de pacote da aplicação será Zip. Para obter mais informações sobre a diferença entre o Image e os tipos de pacote de Zip, consulte [Pacotes de implantação do Lambda](#) no Manual do desenvolvedor do AWS Lambda.

4. Dependendo do runtime selecionado, você talvez precise selecionar um gerenciador de dependências e uma arquitetura de runtime para a aplicação do SAM.

Dependency Manager

Escolha entre Gradle ou Maven.

Note

Essa opção de ferramentas de automação de compilação está disponível apenas para o runtime Java.


Architecture

Escolha entre x86_64 ou arm64.

A opção de executar a aplicação sem servidor em um ambiente emulado baseado em ARM64 em vez do ambiente padrão baseado em x86_64 está disponível para os seguintes runtimes:

- nodejs12.x (ZIP e imagem)

- nodejs14.x (ZIP e imagem)
- python3.8 (ZIP e imagem)
- python3.9 (ZIP e imagem)
- python3.10 (ZIP e imagem)
- python3.11 (ZIP e imagem)
- java8.al2 com Gradle (ZIP e imagem)
- java8.al2 com Maven (somente ZIP)
- java11 com Gradle (ZIP e imagem)
- java11 com Maven (somente ZIP)

 Important

Você deve instalar a versão 1.33.0 ou posterior da AWS CLI para que as aplicações sejam executadas em ambientes baseados em ARM64. Para obter mais informações, consulte [Pré-requisitos](#).

5. Escolha um local para o seu novo projeto. Você pode usar uma pasta workspace existente se uma estiver aberta, Select a different folder (Selecionar uma pasta diferente) já existente, ou criar uma nova pasta e selecioná-la. Neste exemplo, escolha There are no workspace folders open (Não há pastas workspace abertas) para criar uma pasta chamada MY-SAM-APP.
6. Insira um nome para seu novo projeto. Neste exemplo, use my-sam-app-nodejs. Depois de pressionar Enter, o kit de ferramentas para VS Code levará alguns instantes para criar o projeto.

Quando o projeto é criado, sua aplicação é adicionada ao espaço de trabalho atual. Você o verá listado na janela do Explorer.

Abrir uma aplicação sem servidor (local)

Para abrir uma aplicação sem servidor no host de desenvolvimento local, abra a pasta que contém o arquivo de modelo da aplicação.

1. No menu Arquivo, selecione Abrir.
2. Na caixa de diálogo Abrir pasta, acesse a pasta da aplicação sem servidor que você deseja abrir.
3. Selecione o botão Selecionar pasta.

Ao abrir a pasta de uma aplicação, ela é adicionada à janela do Explorer.

Executar e depurar uma aplicação sem servidor usando um modelo (local)

Você pode usar o kit de ferramentas para VS Code para configurar como depurar aplicações sem servidor e executá-las localmente no ambiente de desenvolvimento.

Você começa a configurar o comportamento de depuração usando o recurso [CodeLens](#) do VS Code para identificar uma função do Lambda qualificada. O CodeLens permite interações sensíveis ao conteúdo com o código-fonte. Para obter informações sobre como garantir que você possa acessar o recurso CodeLens, consulte a seção [Pré-requisitos](#) anterior neste tópico.

Note

Neste exemplo, você depura uma aplicação que usa JavaScript. No entanto, é possível usar os recursos de depuração do kit de ferramentas para VS Code com os seguintes runtimes e linguagens:

- C#: .NET Core 2.1, 3.1; .NET 5.0
- JavaScript/TypeScript: Node.js 12. x, 14. x
- Python: 3.6, 3.7, 3.8, 3.9, 3.10, 3.11
- Java: 8, 8.al2, 11
- Go: 1.x

A escolha de idioma também afeta como o CodeLens detecta manipuladores qualificados do Lambda. Para obter mais informações, consulte [Executando e depurando funções do Lambda enquanto exclui recursos de modelo AWS SAM](#).

Neste procedimento, você usa a aplicação de exemplo criada anteriormente na seção [Criar uma aplicação sem servidor \(local\)](#) neste tópico.

1. Para visualizar os arquivos da aplicação no explorador de arquivos do VS Code, selecione Exibir, Explorador.
2. Na pasta da aplicação (por exemplo, my-sample-app), abra o arquivo `template.yaml`.

Note

Se você usar um modelo com um nome que não seja `template.yaml`, o indicador do CodeLens não ficará automaticamente disponível no arquivo YAML. Isso significa que você deve adicionar manualmente uma configuração de depuração.

3. No editor de `template.yaml`, acesse a seção `Resources` do modelo que define os recursos sem servidor. Nesse caso, esse é o recurso `HelloWorldFunction` do tipo `AWS::Serverless::Function`.

No indicador do CodeLens desse recurso, escolha `Adicionar configuração de depuração`.

4. Na paleta de comandos, selecione o runtime em que a aplicação do AWS SAM será executada.
5. No editor para o arquivo `launch.json`, edite ou confirme valores para as seguintes propriedades de configuração:
 - `"name"` – Insira um nome de leitura fácil para aparecer na caixa `Configuration (Configuração)` no campo suspenso da exibição `Run (Executar)`.
 - `"target"`: verifique se o valor é `"template"` para que o modelo AWS SAM seja o ponto de entrada para a sessão de depuração.
 - `"templatePath"` – Insira um caminho relativo ou absoluto para o arquivo `template.yaml`.
 - `"logicalId"`: verifique se o nome corresponde ao especificado na seção `Recursos` do modelo AWS SAM. Neste caso, é o `HelloWorldFunction` do tipo `AWS::Serverless::Function`.

Para obter mais informações sobre essas e outras entradas no arquivo `launch.json`, consulte [Opções de configuração para depurar aplicações sem servidor](#).

6. Se você estiver satisfeito com sua configuração de depuração, salve `launch.json`. Em seguida, para iniciar a depuração, selecione o botão verde “reproduzir” na visualização EXECUTAR.

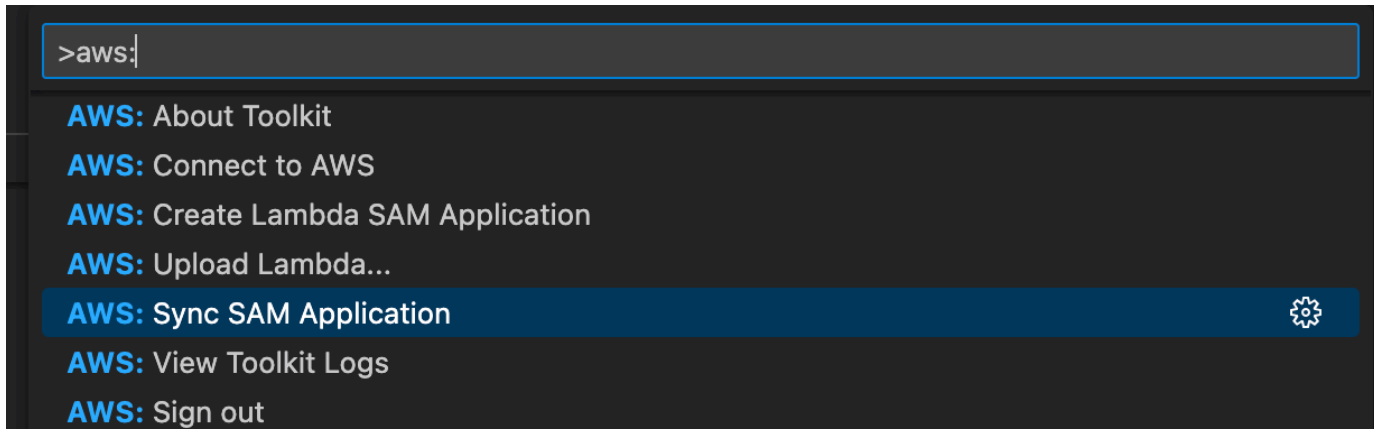
Quando as sessões de depuração forem iniciadas, o painel `DEBUG CONSOLE (CONSOLE DE DEPURAÇÃO)` mostra a saída de depuração e exibe todos os valores retornados pela função Lambda. (Ao depurar aplicações AWS SAM, o kit de ferramentas da AWS é selecionado como canal de Saída no painel Saída.)

Sincronizar aplicações AWS SAM

O AWS Toolkit for Visual Studio Code executa o comando `aws sam sync` da CLI do AWS SAM para implantar as aplicações sem servidor na Nuvem AWS. Para obter mais informações sobre a sincronização do AWS SAM, consulte o tópico [Referência de comandos da AWS SAMCLI](#) no Guia do desenvolvedor do AWS Serverless Application Model

O procedimento a seguir descreve como implantar aplicações sem servidor usando o kit de ferramentas da Nuvem AWS para VS Code `aws sam sync`.

1. No menu principal do VS Code, abra a paleta de comandos, expanda Exibir e selecione Paleta de comandos.
2. Na Paleta de comandos, busque AWS e escolha Sincronizar aplicação do SAM para começar a configurar a sincronização.




3. Selecione a região da AWS para sincronizar a aplicação sem servidor.
4. Escolha o arquivo `template.yaml` a ser usado para a implantação.
5. Selecione um bucket do Amazon S3 existente ou insira um novo nome de bucket do Amazon S3 para implantar a aplicação.

Important

O bucket do Amazon S3 deve atender aos seguintes requisitos:

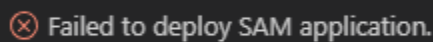
- O bucket deve estar na região com a qual você está sincronizando.
- Esse nome de bucket do Amazon S3 deve ser exclusivo globalmente com relação a todos os nomes de buckets existentes no Amazon S3.


6. Se a aplicação sem servidor incluir uma função com o tipo de pacote Image, insira o nome de um repositório do Amazon ECR que essa implantação pode usar. O repositório deve estar na região na qual você estiver implantando.
7. Selecione uma pilha de implantação na lista de implantações anteriores ou crie uma pilha de implantação inserindo um novo nome de pilha. Depois, prossiga para iniciar o processo de sincronização.

 Note

As pilhas usadas em implantações anteriores são recuperadas por espaço de trabalho e região.

8. Durante o processo de sincronização, o status da implantação é capturado na guia Terminal do VS Code. Verifique se a sincronização foi bem-sucedida na guia Terminal. Se ocorrer um erro, você receberá uma notificação.



 Note

Para obter mais informações sobre a sincronização, os logs do AWS Toolkit for Visual Studio Code podem ser acessados na paleta de comandos.

Para acessar os logs do AWS Toolkit for Visual Studio Code na paleta de comandos, expanda Exibir, selecione Paleta de comandos, busque por **AWS: View AWS Toolkits Logs** e escolha-o quando ele preencher a lista.

Após a conclusão da implantação, a aplicação é mostrada na janela do AWS Explorer. Para obter mais informações sobre como invocar a função do Lambda criada como parte da aplicação, consulte o tópico [Interagir de forma remota com as funções do Lambda](#) neste guia do usuário.

Excluir uma aplicação sem servidor da Nuvem AWS

A exclusão de uma aplicação sem servidor requer a exclusão do AWS CloudFormation que você implantou anteriormente na Nuvem AWS. Observe que esse procedimento não exclui o diretório da aplicação do host local.

1. Abra a [AWS Explorador](#).
2. Na janela do AWS Explorer, expanda a região que contém a aplicação implantada que você deseja excluir e expanda o AWS CloudFormation.
3. Abra o menu de contexto (clique com o botão direito do mouse) do nome da pilha do AWS CloudFormation que corresponde à aplicação sem servidor que você deseja excluir e selecione Excluir pilha do AWS CloudFormation.
4. Para confirmar que deseja excluir a pilha selecionada, escolha Excluir.

Se a exclusão da pilha for bem-sucedida, o kit de ferramentas para VS Code removerá o nome da pilha da lista do AWS CloudFormation no AWS Explorer.

Executando e depurando funções do Lambda enquanto exclui recursos de modelo AWS SAM

Ao testar o AWS SAM aplicativo, você pode optar por executar e depurar apenas a função Lambda e excluir outros recursos definidos AWS SAM pelo modelo. Essa abordagem envolve o uso do [CodeLens](#) recurso para identificar manipuladores de funções Lambda no código-fonte que você pode invocar diretamente.

Os manipuladores Lambda detectados por CodeLens dependem da linguagem e do tempo de execução que você está usando para seu aplicativo.

Idioma/tempo de execução	Critérios para que as funções Lambda sejam identificadas por indicadores CodeLens
C# (dotnetcore2.1, 3.1;. NET5,0)	<p>A função de pesquisa também tem os seguintes recursos:</p> <ul style="list-style-type: none"> • É uma função pública de uma classe pública. • Tem um ou dois parâmetros. Com dois parâmetros, o segundo parâmetro deve implementar a interface <code>ILambdaContext</code>. • Um arquivo <code>*.csproj</code> está na pasta principal dentro da pasta do espaço de trabalho do VS Code.

Idioma/tempo de execução	Critérios para que as funções Lambda sejam identificadas por indicadores CodeLens
	A extensão ms-dotnettools.csharp (ou qualquer extensão que forneça símbolos de linguagem para C#) está instalada e habilitada.
JavaScript/TypeScript (Node.js 12.x, 14.x)	A função de pesquisa também tem os seguintes recursos: <ul style="list-style-type: none">• É uma função exportada com até três parâmetros.• Um arquivo <code>package.json</code> está na pasta principal dentro da pasta do espaço de trabalho do VS Code.
Python (3.7, 3.8, 3.9, 3.10, 3.11, 3.12)	A função de pesquisa também tem os seguintes recursos: <ul style="list-style-type: none">• É uma função de nível superior.• Um arquivo <code>requirements.txt</code> está na pasta principal dentro da pasta do espaço de trabalho do VS Code. A extensão ms-python.python (ou qualquer extensão que forneça símbolos de linguagem para Python) está instalada e habilitada.

Idioma/tempo de execução	Critérios para que as funções Lambda sejam identificadas por indicadores CodeLens
Java (8, 8.al2, 11)	<p>A função de pesquisa também tem os seguintes recursos:</p> <ul style="list-style-type: none">• É uma função pública de uma classe pública e não abstrata.• Tem um, dois ou três parâmetros:<ul style="list-style-type: none">• Um parâmetro: o parâmetro pode ser qualquer coisa.• Dois parâmetros: os parâmetros devem ser <code>java.io.InputStream</code> e <code>java.io.OutputStream</code> OU o último parâmetro deve ser <code>com.amazonaws.services.lambda.runtime.Context</code>.• Três parâmetros: Os parâmetros devem ser <code>java.io.InputStream</code> e <code>java.io.OutputStream</code> AND a o último parâmetro deve ser <code>com.amazonaws.services.lambda.runtime.Context</code>.• Um arquivo <code>build.gradle</code> (Gradle) ou <code>pom.xml</code> (Maven) está em sua pasta principal, na pasta do espaço de trabalho do VS Code. <p>A extensão redhat.java (ou qualquer extensão que forneça símbolos de linguagem para Java) está instalada e habilitada. Essa extensão requer o Java 11, independentemente de qual runtime Java você estiver usando.</p>

Idioma/tempo de execução	Critérios para que as funções Lambda sejam identificadas por indicadores CodeLens
	O vscjava. vscode-java-debug extensão (ou qualquer extensão que forneça um depurador Java) está instalada e habilitada.
Go (1.x)	<p>A função de pesquisa também tem os seguintes recursos:</p> <ul style="list-style-type: none"> • É uma função de nível superior. • Requer entre 0 e 2 argumentos. Se houver dois argumentos, o primeiro argumento deverá implementar <code>context.Context</code>. • Retorna entre 0 e 2 argumentos. Se houver mais de 0 argumentos, o último argumento deverá implementar <code>error</code>. • Ele tem um arquivo <code>go.mod</code> na pasta do espaço de trabalho do VS Code. <p>A extensão golang.go está instalada, configurada e habilitada.</p>

Como executar e depurar uma aplicação sem servidor diretamente do código da aplicação

1. Para visualizar os arquivos da aplicação no explorador de arquivos do VS Code, selecione Exibir, Explorer.
2. Na pasta do aplicativo (por exemplo, my-sample-app), expanda a pasta de funções (nesse caso, hello-world) e abra o `app.js` arquivo.
3. No CodeLens indicador que identifica um manipulador de funções Lambda elegível, escolha `Add Debug Configuration`
4. Na paleta de comandos, selecione o runtime em que a aplicação do AWS SAM será executada.
5. No editor para o arquivo `launch.json`, edite ou confirme valores para as seguintes propriedades de configuração:

- "name": insira um nome de leitura fácil para aparecer no campo suspenso Configuration (Configuração) na exibição Run (Executar).
- "target" – Verifique se o valor é "code" para que um manipulador de função do Lambda seja diretamente chamado.
- "lambdaHandler" – Insira nome do método no código que o Lambda chama para executar a função. Por exemplo, para aplicativos em JavaScript, o padrão é `app.lambdaHandler`.
- "projectRoot" – Insira o caminho para o arquivo da aplicação que contém a função do Lambda.
- "runtime" – Insira ou confirme um tempo de execução válido para o ambiente de execução do Lambda, por exemplo, "nodejs.12x".
- "payload" – Escolha uma das seguintes opções para definir a carga útil do evento que você quer fornecer para a função Lambda como entrada:
 - "json": pares JSON de valores-chave formatados que definem a carga útil do evento.
 - "path": Um caminho para o arquivo que é usado como a carga útil do evento.

No exemplo abaixo, a opção "json" define a carga útil.

Para obter mais informações sobre essas e outras entradas no arquivo `launch.json`, consulte [Opções de configuração para depurar aplicações sem servidor](#).

6.

Se você estiver satisfeito com a configuração de depuração, para iniciar a depuração, escolha a seta de reprodução verde ao lado de. RUN

Quando as sessões de depuração começam, o DEBUGCONSOLEpainel mostra a saída da depuração e exibe todos os valores retornados pela função Lambda. (Ao depurar AWS SAM aplicativos, o AWS Toolkit é selecionado como o canal de saída no painel Saída.)

Executar e depurar recursos locais do Amazon API Gateway

Você pode executar ou depurar recursos locais do AWS SAM API Gateway, especificados em `template.yaml`. Para isso, execute a configuração de inicialização `type=aws-sam` do VS Code com o `invokeTarget.target=api`.

Note

O API Gateway suporta dois tipos de APIs: REST e HTTP. No entanto, o recurso do API Gateway com o AWS Toolkit for Visual Studio Code é compatível apenas com APIs REST. Às vezes, as APIs HTTP são chamadas de “API Gateway V2 APIs”.

Para executar e depurar recursos locais do API Gateway

1. Selecione uma das seguintes abordagens para criar uma configuração de execução para um recurso do API Gateway do AWS SAM:
 - Opção 1: acesse o código-fonte do manipulador (arquivo .js, .cs ou .py) no projeto AWS SAM, passe o mouse sobre o manipulador do Lambda e selecione a opção Adicionar configuração de depuração do CodeLens. Em seguida, no menu, escolha o item marcado como Evento de API.
 - Opção 2: edite `launch.json` e crie uma configuração de inicialização usando a sintaxe a seguir.

```
{
  "type": "aws-sam",
  "request": "direct-invoke",
  "name": "myConfig",
  "invokeTarget": {
    "target": "api",
    "templatePath": "n12/template.yaml",
    "logicalId": "HelloWorldFunction"
  },
  "api": {
    "path": "/hello",
    "httpMethod": "post",
    "payload": {
      "json": {}
    }
  },
  "sam": {},
  "aws": {}
}
```

2. No painel Executar do VS Code, escolha a configuração de inicialização (denominada `myConfig` no exemplo acima).
3. (Opcional) Adicione pontos de interrupção ao código do projeto do Lambda.
4. Digite F5 ou escolha Reproduzir no painel Executar.
5. No painel de saída, exiba os resultados.

Configuração

Quando você usa o Valor da propriedade do `invokeTarget.target`, `api`, o Toolkit altera a validação da configuração de execução e o comportamento para suportar um campo da `api`.

```
{
  "type": "aws-sam",
  "request": "direct-invoke",
  "name": "myConfig",
  "invokeTarget": {
    "target": "api",
    "templatePath": "n12/template.yaml",
    "logicalId": "HelloWorldFunction"
  },
  "api": {
    "path": "/hello",
    "httpMethod": "post",
    "payload": {
      "json": {}
    },
    "queryString": "abc=def&qrs=tuv",
    "headers": {
      "cookie": "name=value; name2=value2; name3=value3"
    }
  },
  "sam": {},
  "aws": {}
}
```

Substitua os valores no exemplo da seguinte forma:

`invokeTarget.logicalId`

Um recurso da API.

`path`

O caminho da API que a configuração de inicialização solicita, por exemplo, `"path": "/hello"`.

Deve ser um caminho de API válido resolvido a partir do `template.yaml` especificado por `invokeTarget.templatePath`.

`httpMethod`

Pode ser um dos seguintes verbos: `delete`, `get`, `head`, `options`, `patch`, `post`, `put`.

`payload`

A carga do JSON (corpo HTTP) para enviar na solicitação, com a mesma estrutura e regras que o campo [lambda.payload](#).

`payload.path` aponta para um arquivo que contém a carga útil JSON.

`payload.json` especifica uma carga útil JSON integrada.

`headers`

Mapa opcional de pares de nome/valor, que você usa para especificar cabeçalhos HTTP a serem incluídos na solicitação, conforme o exemplo abaixo.

```
"headers": {
  "accept-encoding": "deflate, gzip;q=1.0, *;q=0.5",
  "accept-language": "fr-CH, fr;q=0.9, en;q=0.8, de;q=0.7, *;q=0.5",
  "cookie": "name=value; name2=value2; name3=value3",
  "user-agent": "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_14_6)
AppleWebKit/537.36 (KHTML, like Gecko) Chrome/86.0.4240.198 Safari/537.36",
}
```

`querystring`

String opcional que define a `querystring` da solicitação, por exemplo, `"querystring": "abc=def&ghi=jkl"`.

AWS

Como as informações de conexão da AWS são fornecidas. Para obter mais informações, consulte a tabela AWS connection ("aws") properties na seção [Opções de configuração para depurar aplicações sem servidor](#).

sam

Como a CLI do AWS SAM compila a aplicação. Para obter mais informações, consulte a tabela AWS SAM CLI ("sam") properties na seção [Opções de configuração para depurar aplicações sem servidor](#).

Opções de configuração para depurar aplicações sem servidor

Ao abrir o arquivo `launch.json` para editar configurações de depuração, você pode usar o recurso [IntelliSense](#) do VS Code para ver e preencher automaticamente as propriedades válidas. Para acionar o IntelliSense no editor, pressione `Ctrl + barra de espaço`.

```
"lambda": {  
  "runtime": "nodejs12.x",  
  "event": {  
    "json": {}  
  }  
}
```

O IntelliSense permite encontrar e definir propriedades para invocar funções do Lambda diretamente ou com o modelo do AWS SAM. Defina também propriedades para o "lambda" (como a função é executada), "sam" (como o a CLI do AWS SAM cria a aplicação) e "aws" (como as informações de conexão da AWS são fornecidas).

AWS SAM: invocação direta do manipulador do Lambda/invocação do Lambda baseada em modelo

Propriedade	Descrição
type	Especifica qual extensão gerencia a configuração de inicialização. Sempre definido como <code>aws-sam</code> para usar a CLI do AWS SAM para criar e depurar localmente.

Propriedade	Descrição
<code>name</code>	Especifica um nome compatível com o leitor a ser exibido na lista Debug launch configuration (Configuração de execução da depuração).
<code>request</code>	Especifica o tipo de configuração a ser executada pela extensão designada (<code>aws-sam</code>). Sempre definido como <code>direct-invoke</code> para iniciar a função do Lambda.

Propriedade	Descrição
<code>invokeTarget</code>	<p>Especifica o ponto de entrada para invocar o recurso.</p> <p>Para invocar a função Lambda diretamente, defina valores para os seguintes campos de <code>invokeTarget</code> :</p> <ul style="list-style-type: none">• <code>target</code>: defina como <code>code</code>.• <code>lambdaHandler</code> – O nome do manipulador da função do Lambda a ser chamado..• <code>projectRoot</code> : o caminho para o arquivo da aplicação que contém o manipulador de função do Lambda.• <code>architecture</code> : arquitetura do processador do ambiente emulado no qual a aplicação SAM do Lambda local é executada. Para determinados runtimes, você pode escolher <code>arm64</code> em vez da arquitetura <code>x86_64</code> padrão. Para obter mais informações, consulte Criar uma aplicação sem servidor (local). <p>Para invocar os recursos do Lambda com o modelo do AWS SAM, defina valores para os seguintes campos <code>invokeTarget</code> :</p> <ul style="list-style-type: none">• <code>target</code>: defina como <code>template</code>.• <code>templatePath</code> : o caminho para o arquivo de modelo do AWS SAM.• <code>logicalId</code> – O nome do recurso da AWS::Lambda::Function ou AWS::Serverless::Function a ser chamado. Você pode encontrar o nome do recurso no modelo do AWS SAM formatado em YAML. Observe que o AWS Toolkit reconhece implicitamente funções definidas com <code>PackageType: Image</code> no modelo do AWS SAM como funções do Lambda baseadas em imagens. Para obter mais informações, consulte Pacotes de implantação do Lambda no Guia do desenvolvedor do AWS Lambda.

Propriedades do Lambda ("**lambda**")

Propriedade	Descrição
<code>environmentVariables</code>	<p>Transmite parâmetros operacionais para a função do Lambda. Por exemplo, se você estiver gravando em um bucket do Amazon S3, em vez de codificação rígida no nome do bucket em que estiver gravando, configure o nome do bucket como uma variável de ambiente.</p> <div data-bbox="620 604 657 640"></div> Note <p>Ao especificar variáveis de ambiente para uma aplicação sem servidor, você deve adicionar configurações ao modelo do AWS SAM (<code>template.yaml</code>) e ao arquivo <code>launch.json</code>.</p> <p>Exemplo de formatação para uma variável de ambiente no modelo do AWS SAM:</p> <pre>Resources: HelloWorldFunction: Type: AWS::Serverless::Function Properties: CodeUri: hello-world/ Handler: app.lambdaHandlerN10 Runtime: nodejs10.x Environment: Variables: SAMPLE1: Default Sample 1 Value</pre> <p>Exemplo de formatação para uma variável de ambiente no arquivo <code>launch.json</code> :</p> <pre>"environmentVariables": { "SAMPLE1": "My sample 1 value" }</pre>

Propriedade	Descrição
	<ul style="list-style-type: none">"json": Pares de chave/valor formatados em JSON que definem a carga útil do evento."path": Um caminho para o arquivo que é usado como a carga útil do evento.
memoryMB	Especifica megabytes (MB) de memória fornecidos para executar uma função do Lambda invocada.
runtime	O runtime usado pela função do Lambda. Para obter mais informações, consulte Tempos de execução do AWS Lambda .
timeoutSec	Define o tempo permitido, em segundos, antes que a sessão de depuração seja encerrada.

Propriedade	Descrição
pathMappings	<p>Especifica onde o código local está em relação ao local onde ele é executado no contêiner.</p> <p>Por padrão, o kit de ferramentas para VS Code define <code>localRoot</code> como a raiz do código da função do Lambda no espaço de trabalho local e <code>remoteRoot</code> como <code>/var/task</code>, que é o diretório de trabalho padrão para código em execução no Lambda. Se o diretório de trabalho for alterado no Dockerfile ou com o parâmetro <code>WorkingDirectory</code> no arquivo de modelo do AWS CloudFormation, pelo menos uma entrada <code>pathMapping</code> deverá ser especificada para que o depurador possa mapear com êxito os pontos de interrupção definidos localmente para o código em execução no contêiner do Lambda.</p> <p>Exemplo de formatação para <code>pathMappings</code> no arquivo <code>launch.json</code> :</p> <pre data-bbox="597 1033 1507 1352">"pathMappings": [{ "localRoot": " \${workspaceFolder}/sam-app/ HelloWorldFunction ", "remoteRoot": " /var/task " }]</pre> <p>Advertências:</p> <ul style="list-style-type: none">• Para funções do Lambda baseadas em imagem .NET, a entrada <code>remoteRoot</code> deve ser o diretório de compilação.• Para funções do Lambda baseadas em Node.js, você pode especificar somente uma única entrada de mapeamento de caminho.

O kit de ferramentas para VS Code usa a AWS SAM CLI para criar e depurar aplicações sem servidor localmente. Você pode configurar o comportamento dos comandos da CLI do AWS SAM usando propriedades da configuração do "sam" no arquivo `launch.json`.

Propriedades da AWS SAM CLI ("sam")

Propriedade	Descrição	Valor padrão
<code>buildArguments</code>	Configura como o <code>sam build</code> constrói o código-fonte do Lambda. Para visualizar opções de compilação, consulte sam build no Manual do desenvolvedor do AWS Serverless Application Model.	String vazia
<code>containerBuild</code>	Indica se a sua função deve ser construída dentro de um contêiner do Docker semelhante ao Lambda.	false
<code>dockerNetwork</code>	Especifica o nome ou ID de uma rede Docker existente à qual os contêineres do Docker do Lambda devem se conectar, juntamente com a rede de ponte padrão. Se não for especificado, os contêineres do Lambda se conectarão somente à rede Docker da ponte padrão.	String vazia
<code>localArguments</code>	Especifica argumentos adicionais de invocação local.	String vazia
<code>skipNewImageCheck</code>	Especifica se o comando deve ignorar a extração da imagem mais recente do Docker para	false

Propriedade	Descrição	Valor padrão
	o tempo de execução do Lambda.	
template	Personaliza o modelo do AWS SAM usando parâmetros para inserir valores do cliente. Para obter mais informações, consulte Parâmetros no Guia do usuário do AWS CloudFormation.	"parameters": {}

Propriedades ("aws") da conexão da AWS

Propriedade	Descrição	Valor padrão
credentials	Seleciona um perfil específico (por exemplo, profile:default) do arquivo de credenciais para obter credenciais da AWS.	As credenciais da AWS fornecidas pelo arquivo de configuração compartilhado da AWS ou pelo arquivo de credenciais compartilhado da AWS ao kit de ferramentas do VS Code.
region	Define a região da AWS do serviço (por exemplo, us-east-1).	A região da AWS associada ao perfil de credenciais ativas.

Exemplo: configuração de inicialização de modelo

Veja um exemplo de arquivo de configuração de inicialização para um destino de modelo do AWS SAM:

```
{
  "configurations": [
    {
      "type": "aws-sam",
```

```
    "request": "direct-invoke",
    "name": "my-example:HelloWorldFunction",
    "invokeTarget": {
      "target": "template",
      "templatePath": "template.yaml",
      "logicalId": "HelloWorldFunction"
    },
    "lambda": {
      "payload": {},
      "environmentVariables": {}
    }
  }
]
}
```

Exemplo: configuração de inicialização de código

Veja um exemplo de arquivo de configuração de inicialização para um destino de função do Lambda:

```
{
  "configurations": [
    {
      "type": "aws-sam",
      "request": "direct-invoke",
      "name": "my-example:app.lambda_handler (python3.7)",
      "invokeTarget": {
        "target": "code",
        "projectRoot": "hello_world",
        "lambdaHandler": "app.lambda_handler"
      },
      "lambda": {
        "runtime": "python3.7",
        "payload": {},
        "environmentVariables": {}
      }
    }
  ]
}
```

Solucionar problemas de aplicações sem servidor

Este tópico detalha erros comuns que você pode encontrar ao criar aplicações sem servidor com o kit de ferramentas para VS Code e como resolvê-los.

Tópicos

- [Como posso usar um `samconfig.toml` com uma configuração de inicialização do SAM?](#)
- [Erro: "RuntimeError: Container does not exist"](#)
- [Erro: "docker.errors.APIError: 500 Server Error ... You have reached your pull rate limit."](#)
- [Erro: "500 Server Error: Mounting C:\Users\..."](#)
- [Ao usar o WSL, as visualizações da web \(por exemplo, o formulário "Invocar na AWS"\) estarão corrompidas](#)
- [Depurar uma aplicação TypeScript, exceto os pontos de interrupção que não funcionam](#)

Como posso usar um `samconfig.toml` com uma configuração de inicialização do SAM?

Especifique o local do [samconfig.toml](#) da CLI do SAM configurando o argumento `--config-file` na propriedade `localArguments` de configuração de ativação. Por exemplo, se o arquivo `samconfig.toml` estiver localizado no nível superior do espaço de trabalho:

```
"sam": {  
  "localArguments": ["--config-file", "${workspaceFolder}/samconfig.toml"],  
}
```

Erro: "RuntimeError: Container does not exist"

O comando `sam build` poderá mostrar esse erro se o sistema não tiver espaço em disco suficiente para o contêiner do Docker. Se o armazenamento do sistema tiver apenas de 1 GB a 2 GB de espaço disponível, o `sam build` poderá falhar durante o processamento, mesmo que o armazenamento do sistema não esteja completamente cheio antes do início da compilação. Para obter mais informações, consulte [este problema do GitHub](#).

Erro: "docker.errors.APIError: 500 Server Error ... You have reached your pull rate limit."

O Docker Hub limita as solicitações que usuários anônimos podem fazer. Se o sistema atingir o limite, o Docker apresentará falha e esse erro aparecerá na visualização SAÍDA do VS Code:

```
docker.errors.APIError: 500 Server Error: Internal Server Error ("toomanyrequests: You  
have  
reached your pull rate limit. You may increase the limit by authenticating and  
upgrading:
```

```
https://www.docker.com/increase-rate-limit")
```

Verifique se o serviço Docker do sistema foi autenticado com as credenciais do Docker Hub.

Erro: "500 Server Error: Mounting C:\Users\..."

Os usuários do Windows podem ver esse erro de instalação do Docker ao depurar aplicações AWS SAM:

```
Fetching lambci/lambda:nodejs10.x Docker container image.....
2019-07-12 13:36:58 Mounting C:\Users\\AppData\Local\Temp\ ... as /var/
task:ro,delegated inside runtime container
Traceback (most recent call last):
...
requests.exceptions.HTTPError: 500 Server Error: Internal Server Error ...
```

Atualize as credenciais das unidades compartilhadas (nas configurações do Docker).

Ao usar o WSL, as visualizações da web (por exemplo, o formulário “Invocar na AWS”) estarão corrompidas

Esse é um problema conhecido do VS Code para usuários do Cisco VPN. Para obter mais informações, consulte [este problema do GitHub](#).

Uma solução alternativa é sugerida [neste problema de rastreamento do WSL](#).

Depurar uma aplicação TypeScript, exceto os pontos de interrupção que não funcionam

Isso acontecerá se não houver um mapa de origem para vincular o arquivo JavaScript compilado ao arquivo TypeScript de origem. Para corrigir esse problema, abra o arquivo `tsconfig.json` e verifique se a opção e o valor a seguir estão definidos: `"inlineSourceMap": true`.

Trabalhar com documentos do Systems Manager Automation

O AWS Systems Manager oferece visibilidade e controle de infraestrutura na AWS. O Systems Manager oferece uma interface de usuário unificada para que você possa visualizar dados operacionais de diversos serviços da AWS e automatizar tarefas operacionais em seus recursos da AWS.

Um [documento do Systems Manager](#) define as ações que o Systems Manager realiza nas suas instâncias gerenciadas. No Automation, documento é um tipo de documento do Systems Manager usado para realizar tarefas comuns de manutenção e implantação, como criar ou atualizar uma imagem de máquina da Amazon (AMI). Este tópico descreve como criar, editar, publicar e excluir documentos de automação usando o AWS Toolkit for Visual Studio Code.

Tópicos

- [Suposições e pré-requisitos](#)
- [Permissões do IAM para documentos de automação do Systems Manager](#)
- [Criar um documento do Systems Manager Automation](#)
- [Abrir um documento existente do Systems Manager Automation](#)
- [Editar um documento do Systems Manager Automation](#)
- [Publicar um documento do Systems Manager Automation](#)
- [Excluir um documento do Systems Manager Automation](#)
- [Executar um documento do Systems Manager Automation](#)
- [Solucionar problemas de documentos do Systems Manager Automation no kit de ferramentas para VS Code](#)

Suposições e pré-requisitos

Antes de começar, certifique-se de que:

- Você instalou o Visual Studio Code e a versão mais recente do AWS Toolkit for Visual Studio Code. Para obter mais informações, consulte [Instalando o AWS Toolkit for Visual Studio Code](#).
- Você esteja familiarizado com o Systems Manager. Para obter mais informações, consulte o [Guia do usuário do AWS Systems Manager](#).
- Você já conhece os casos de uso do Systems Manager Automation. Para obter mais informações, consulte [Automação do AWS Systems Manager](#) no Guia do usuário do usuário do AWS Systems Manager.

Permissões do IAM para documentos de automação do Systems Manager

No kit de ferramentas para VS Code, você deve ter um perfil de credenciais que contenha as permissões do AWS Identity and Access Management (IAM) necessárias para criar, editar, publicar

e excluir documentos do Systems Manager Automation. O seguinte documento de política define as permissões do IAM necessárias que podem ser usadas em uma política de entidade principal:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "ssm:ListDocuments",
        "ssm:ListDocumentVersions",
        "ssm:DescribeDocument",
        "ssm:GetDocument",
        "ssm:CreateDocument",
        "ssm:UpdateDocument",
        "ssm:UpdateDocumentDefaultVersion",
        "ssm>DeleteDocument"
      ],
      "Resource": "*"
    }
  ]
}
```

Para obter informações sobre como atualizar uma política do IAM, consulte [Criar políticas do IAM](#) no Guia do usuário do IAM. Para obter informações sobre como configurar o perfil de credenciais, consulte [AWS IAMcredenciais](#).

Criar um documento do Systems Manager Automation

Você pode criar um documento de automação em JSON ou YAML usando o Visual Studio Code. Ao criar um documento de automação, ele é apresentado em um arquivo sem título. Você pode nomear o arquivo e salvá-lo no VS Code, porém o nome do arquivo não ficará visível na AWS.

Para criar um documento de automação

1. Abra o VS Code.
2. No menu Exibir, selecione Paleta de comandos para abrir a paleta de comandos.
3. Na paleta de comandos, insira Criar documento do Systems Manager localmente do kit de ferramentas da AWS.
4. Escolha um dos modelos iniciais para um exemplo de Hello World (Olá mundo).

5. Escolha JSON ou YAML.

Um documento de automação é criado.

Note

O novo documento de automação no VS Code não aparece automaticamente na AWS. Você deve publicá-lo na AWS para que possa executá-lo.

Abrir um documento existente do Systems Manager Automation

Você usa o AWS Explorer para encontrar documentos existentes do Systems Manager Automation. Ao abrir um documento de automação existente, ele aparece como um arquivo sem título no VS Code.

Como abrir um documento de automação

1. Abra o VS Code.
2. No painel de navegação à esquerda, selecione AWS para abrir o AWS Explorer.
3. No AWS Explorer, em Systems Manager, selecione o ícone de download no documento que você deseja abrir e escolha a respectiva versão. O arquivo será aberto no formato dessa versão. Caso contrário, selecione Fazer download usando JSON ou Baixar como YAML.

Note

Salvar localmente um documento de automação como um arquivo no VS Code não faz com que ele apareça na AWS. Para ser executado, primeiro ele precisa ser publicado na AWS.

Editar um documento do Systems Manager Automation

Se você tiver algum documento de automação, ele aparecerá na categoria De minha propriedade dos documentos do Systems Manager no AWS Explorer. Você pode ter documentos de automação que já existem na AWS e ter documentos novos ou atualizados publicados anteriormente na AWS por meio do VS Code.

Ao abrir um documento de automação para edição no VS Code, você tem mais opções do que no AWS Management Console. Por exemplo:

- A validação de esquema está disponível nos formatos JSON e YAML.
- Existem trechos disponíveis no editor de documentos para criar tipos de etapa de automação da sua escolha.
- O preenchimento automático está disponível em várias opções tanto em JSON quanto em YAML.

Como trabalhar com versões

Os documentos do Systems Manager Automation usam versões para gerenciamento de alterações. Você pode escolher a versão padrão para um documento de automação no VS Code.

Para definir uma versão padrão

- No AWS Explorer, procure o documento para o qual deseja definir a versão padrão, abra o menu de contexto (clique com o botão direito do mouse) do documento e selecione Set default version (Definir versão padrão).

Note

Se o documento escolhido possuir apenas uma versão, não será possível alterar o padrão.

Publicar um documento do Systems Manager Automation

Depois de editar o documento de automação no VS Code, você poderá publicá-lo na AWS.

Como publicar um documento de automação

1. Abra o documento de automação que deseja publicar usando o procedimento descrito em [Abrir um documento existente do Systems Manager Automation](#).
2. Faça as alterações desejadas para a publicação. Para obter mais informações, consulte [Editar um documento do Systems Manager Automation](#).
3. No canto superior direito do arquivo aberto, escolha o ícone de upload.
4. Na caixa de diálogo do fluxo de trabalho de publicação, escolha a região da AWS na qual você deseja publicar o documento de automação.

5. Se você estiver publicando um novo documento, selecione Criação rápida. Caso contrário, selecione Atualização rápida para atualizar um documento de automação existente na região da AWS.
6. Insira o nome desse documento de automação.

Ao publicar uma atualização em um documento de automação existente na AWS, uma nova versão é adicionada a ele.

Excluir um documento do Systems Manager Automation

Você pode excluir documentos de automação no VS Code. A exclusão de um documento de automação exclui o documento e todas as versões do documento.

Important

- A exclusão é uma ação destrutiva que não pode ser desfeita.
- A exclusão de um documento de automação que já foi executado não exclui os recursos da AWS que foram criados ou modificados quando ele foi iniciado.

Como excluir um documento de automação

1. Abra o VS Code.
2. No painel de navegação à esquerda, selecione AWS para abrir o AWS Explorer.
3. No AWS Explorer, em Systems Manager, abra o menu de contexto (clique com o botão direito do mouse) do documento que você deseja excluir e selecione Excluir documento.

Executar um documento do Systems Manager Automation

Depois que o documento de automação for publicado na AWS, você poderá executá-lo para realizar tarefas em seu nome na conta da AWS. Para executar o documento de automação, você usa o AWS Management Console, as APIs do Systems Manager, a AWS CLI ou o AWS Tools for PowerShell. Para obter instruções sobre como executar um documento de automação, consulte [Executar uma automação simples](#) no Guia do usuário do AWS Systems Manager.

Como alternativa, se quiser usar um dos AWS SDKs com as APIs do Systems Manager para executar o documento de automação, consulte [Referências do AWS SDK](#).

Note

A execução de um documento de automação pode criar recursos na AWS e gerar custos no faturamento. É altamente recomendável entender o que o documento de automação criará na conta antes de iniciá-lo.

Solucionar problemas de documentos do Systems Manager Automation no kit de ferramentas para VS Code

Salvei o documento de automação no VS Code, mas não o vejo no AWS Management Console.

Quando você salva um documento de automação no VS Code, ele não é publicado na AWS. Para mais informações sobre como publicar seu documento de automação, consulte [Publicar um documento do Systems Manager Automation](#).

A publicação do documento de automação foi malsucedida devido Automation um erro de permissão.

Certifique-se de que seu perfil de credenciais da AWS tenha as permissões necessárias para publicar documentos de automação. Para um exemplo de política de permissões, consulte [Permissões do IAM para documentos de automação do Systems Manager](#).

Publiquei o documento de automação na AWS, mas não o vejo no AWS Management Console.

Você deve publicar o documento na mesma região da AWS na qual está navegando no AWS Management Console.

Excluí meu documento de automação, mas ainda estou recebendo cobrança pelos recursos que ele criou.

A exclusão de um documento de automação não exclui os recursos que ele criou ou modificou. É possível identificar os recursos da AWS que você criou no [AWS Billing Management Console](#), explore suas cobranças e escolha quais recursos serão excluídos de lá.

Trabalhando com AWS Step Functions

O AWS Toolkit for Visual Studio Code (VS Code) fornece suporte para [AWS Step Functions](#). Use kit de ferramentas para VS Code para criar, atualizar e executar máquinas de estado do Step Functions.

Tópicos

- [Trabalhando com AWS Step Functions](#)

Trabalhando com AWS Step Functions

Você pode usar o AWS Toolkit for Visual Studio Code (VS Code) para realizar várias operações com [máquinas de estado](#).

Tópicos

- [Pré-requisitos](#)
- [Trabalhar com máquinas de estado no VS Code](#)
- [Modelos de máquina de estado](#)
- [Visualização gráfica de máquina de estado](#)
- [Trechos de código](#)
- [Preenchimento e validação de código](#)

Pré-requisitos

- Verifique se o sistema atende aos pré-requisitos especificados em [Instalar o kit de ferramentas para VS Code](#) e instale o kit de ferramentas.
- Antes de abrir o AWS Explorer, verifique se você configurou as credenciais.

Trabalhar com máquinas de estado no VS Code

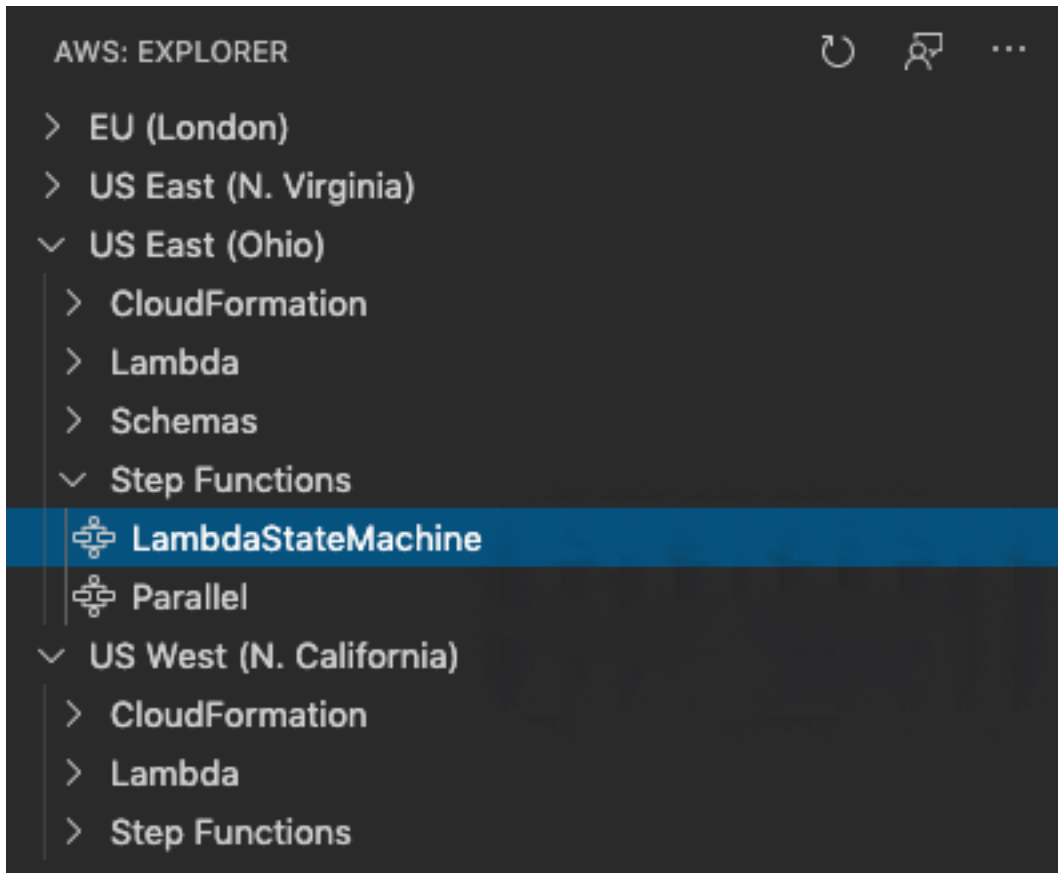
É possível usar o VS Code para interagir com máquinas de estado remotas e desenvolver máquinas de estado localmente no formato JSON ou YAML. É possível criar ou atualizar máquinas de estado, identificar máquinas de estado existentes, executá-las e baixá-las. O VS Code também permite que você crie máquinas de estado com base em modelos e veja uma visualização da máquina de estado, além de fornecer trechos de código, preenchimento de código e validação de código.

Listar máquinas de estado existentes

Se já tiver criado máquinas de estado, você poderá visualizar uma lista delas:

1. Abra o AWS Explorer.

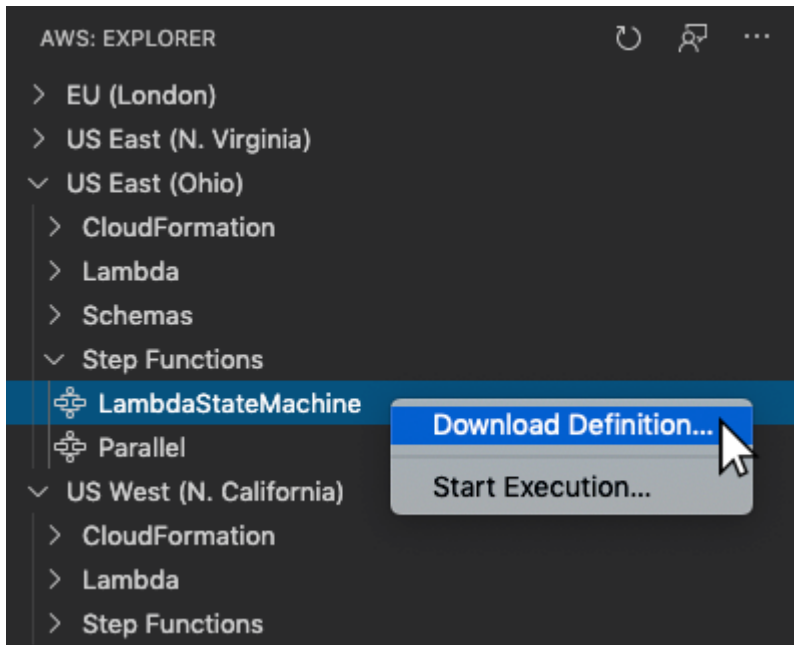
2. Selecione o Step Functions.
3. Verifique se ele lista todas as máquinas de estado em sua conta.



Baixar uma máquina de estado

Para baixar uma máquina de estado:

1. No AWS Explorer, clique com o botão direito do mouse na máquina de estado que deseja baixar.
2. Selecione Download (Baixar) e escolha o local onde deseja baixar a máquina de estado.
3. Verifique se ela baixou corretamente.



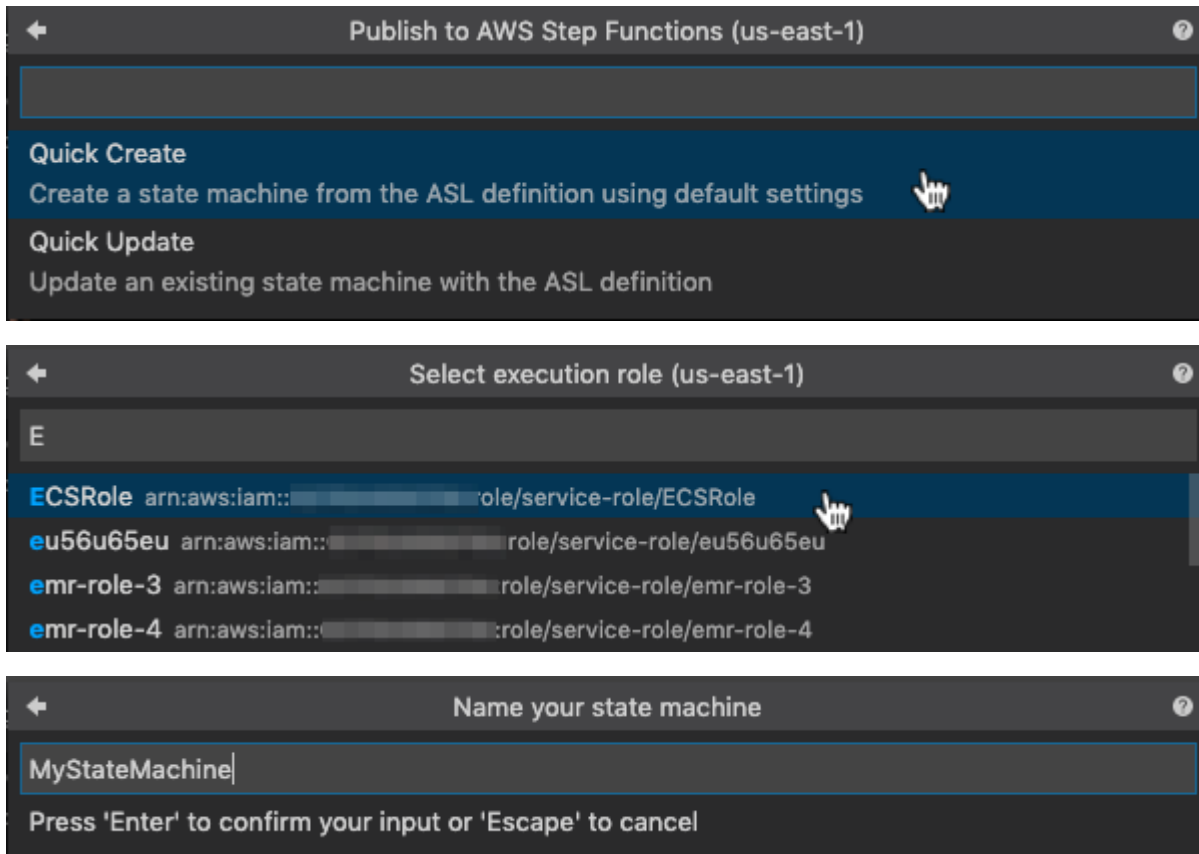
Criar uma máquina de estado

Você mesmo pode criar uma nova máquina de estado ou pode usar um modelo. Para obter mais informações sobre como criar uma máquina de estado a partir de um modelo, consulte a seção [State Machine Templates \(Modelos de máquina de estado\)](#). Para criar uma nova máquina de estado:

1. Crie um novo arquivo [Amazon States Language](#) (ASL) com sua definição da máquina de estado. Use o menu no canto inferior direito para defini-lo como Amazon States Language.
2. Selecione Publish to Step Functions (Publicar no Step Functions).

```
Publish to Step Functions | Render graph
1 {
2   "StartAt": "FirstState",
3   "States": {
4     "FirstState": {
5       "Type": "Task",
6       "Resource": "arn:aws:lambda:us-east-1:123456789012:lambda:function:Function",
7       "Next": "ChoiceState"
8     },
9     "ChoiceState": {
10      "Type": "Choice",
```

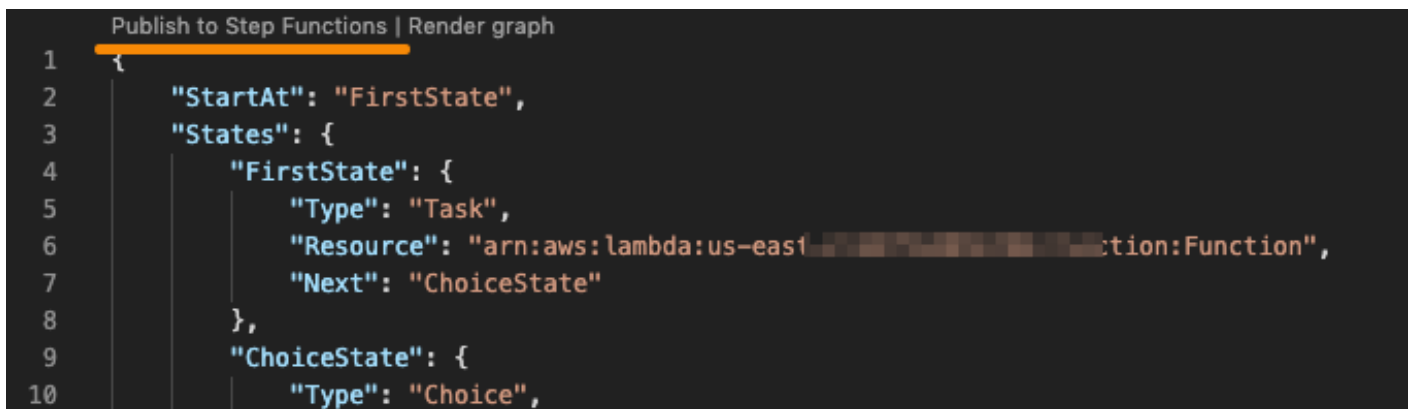
3. Selecione Quick Create (Criação rápida), escolha uma função e nomeie sua máquina de estado.



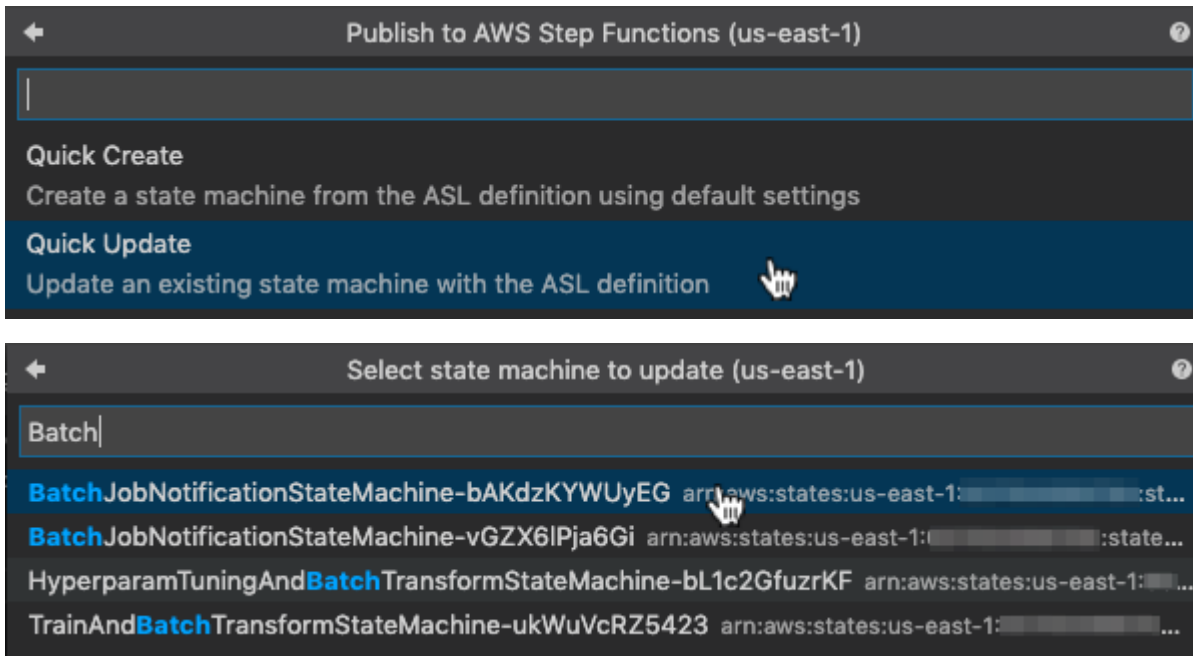
Atualizar uma máquina de estado

Para atualizar uma máquina de estado:

1. Edite o arquivo ASL com sua definição de máquina de estado.
2. Selecione Publish to Step Functions (Publicar no Step Functions).



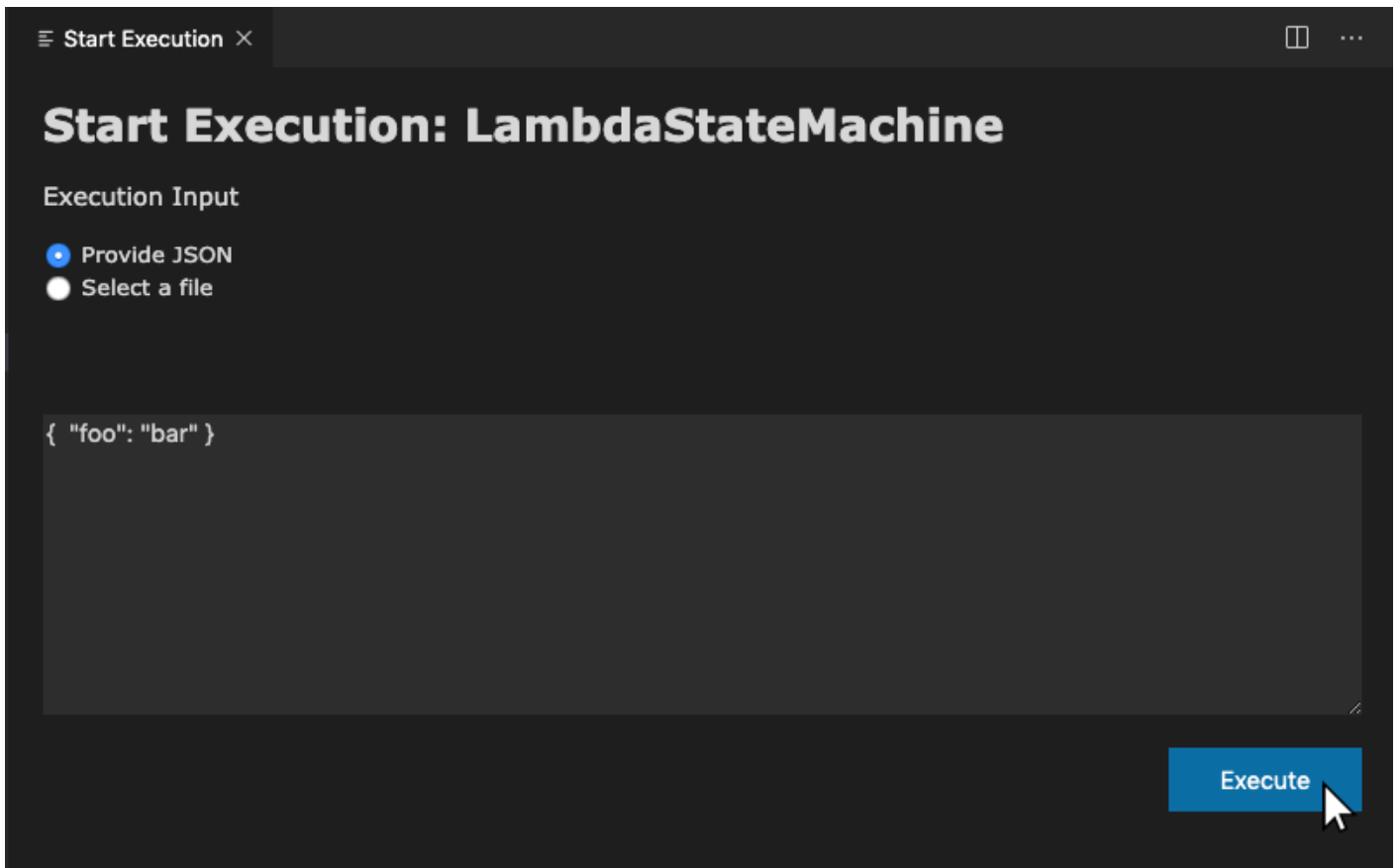
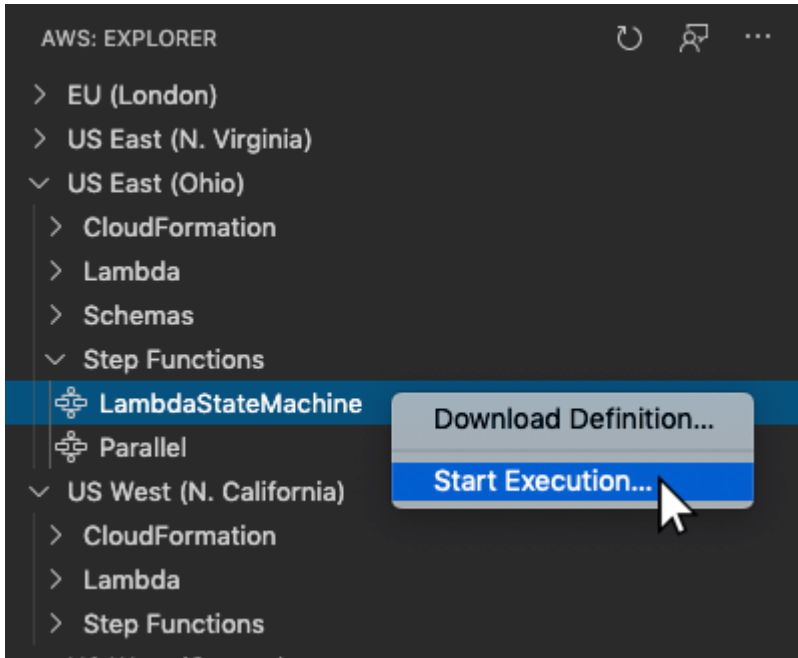
3. Selecione Quick Update (Atualização rápida) e escolha a máquina de estado que deseja atualizar.



Executar uma máquina de estado

Como executar uma máquina de estado:

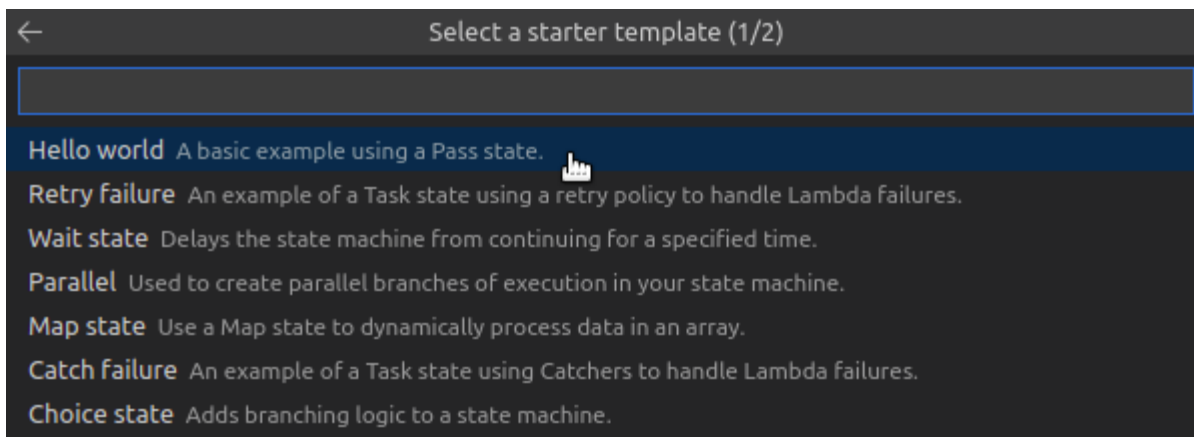
1. No AWS Explorer, clique com o botão direito do mouse na máquina de estado que deseja executar.
2. Forneça dados de entrada para a máquina de estado. Você pode tentar inseri-los de um arquivo ou inseri-los em uma caixa de texto.
3. Inicie a máquina de estado e verifique se ela é executada com êxito.



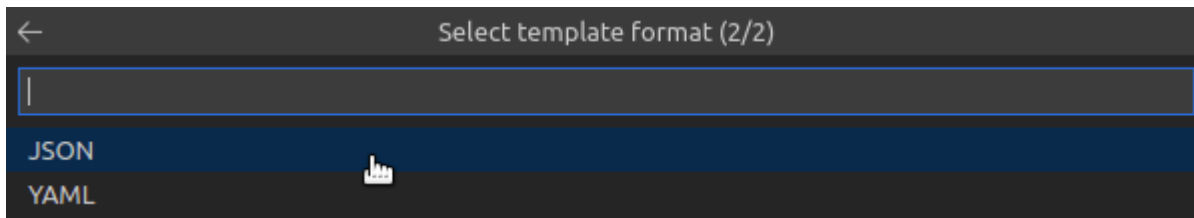
Modelos de máquina de estado

Ao criar uma máquina de estado, você tem a opção de criá-la a partir de um modelo. O modelo contém uma definição de máquina de estado de exemplo com vários estados normalmente usados e fornece um ponto de partida. Para usar modelos de máquina de estado:

1. Abra a paleta de comandos no VS Code.
2. Selecione Criar uma máquina de estado do Step Functions do kit de ferramentas da AWS .
3. Escolha o modelo que deseja usar.



4. Escolha se deseja usar o formato de modelo JSON ou YAML.



Visualização gráfica de máquina de estado

As visualizações de gráfico permitem que você veja a aparência da sua máquina de estado no formato gráfico. Ao criar uma visualização gráfica, outra guia será aberta e exibirá uma visualização da máquina de estado JSON ou YAML. Depois você poderá comparar a definição de máquina de estado que está gravando simultaneamente com sua visualização. À medida que você alterar a definição da máquina de estado, a visualização será atualizada.

Note

Para criar uma visualização de uma definição de máquina de estado, a definição deve estar aberta no editor ativo. Se você fechar ou renomear o arquivo de definição, a visualização será fechada.

Para criar uma visualização de gráfico de máquina de estado:

1. Defina sua máquina de estado.
2. Abra a paleta de comandos no VS Code.
3. Para criar uma visualização, use o botão de visualização no canto superior direito ou escolha Gráfico renderizado da AWS .

The screenshot displays the VS Code interface with two panels. The left panel shows the source ASL JSON file, and the right panel shows the rendered state machine graph.

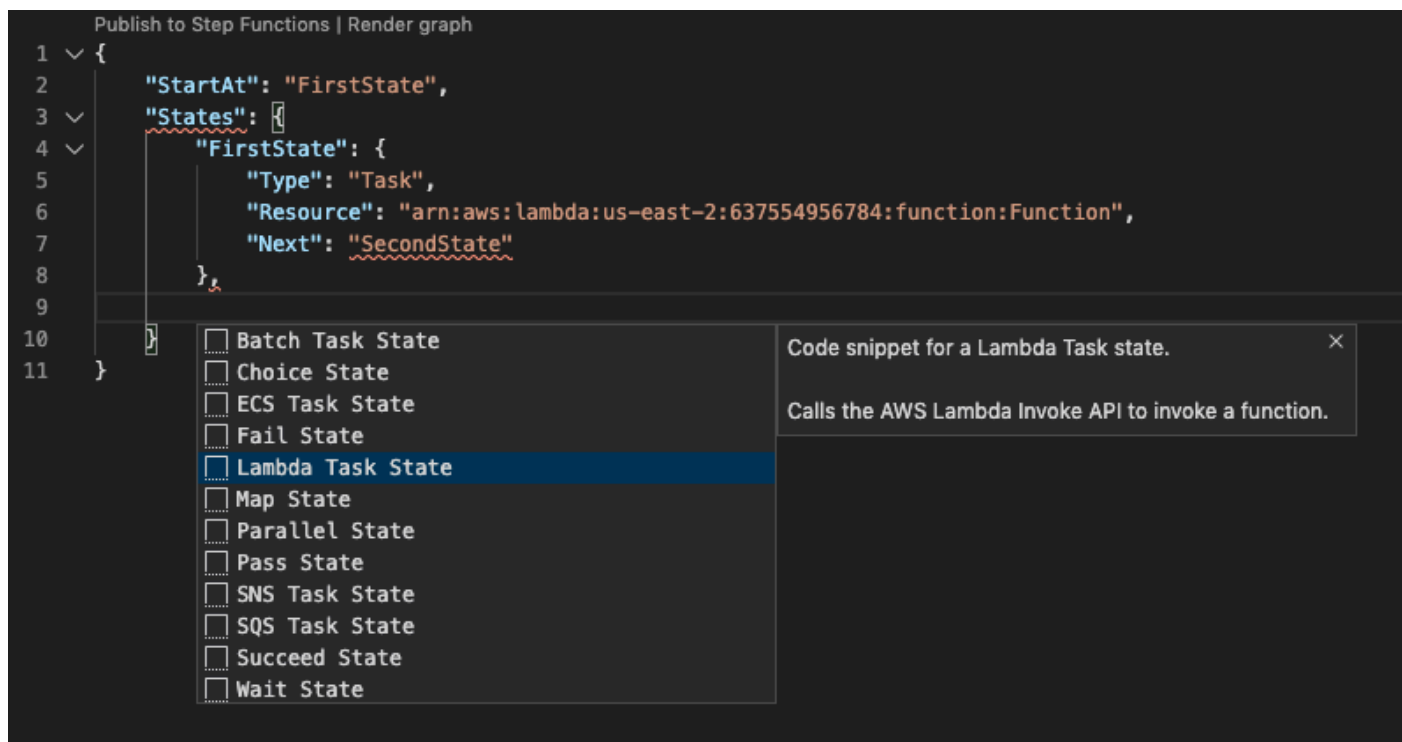
```
1  {
2    "Comment": "An example of the Amazon States Language using a choice
3    state.",
4    "StartAt": "FirstState",
5    "States": {
6      "FirstState": {
7        "Type": "Task",
8        "Resource":
9        "arn:aws:lambda:us-east-2:637554956784:function:Function",
10       "Next": "ChoiceState"
11     },
12     "ChoiceState": {
13       "Type": "Choice",
14       "Choices": [
15         {
16           "Variable": "$.foo",
17           "NumericEquals": 1,
18           "Next": "FirstMatchState"
19         },
20         {
21           "Variable": "$.foo",
22           "NumericEquals": 2,
23           "Next": "SecondMatchState"
24         }
25       ],
26       "Default": "DefaultState"
27     },
28     "FirstMatchState": {
29       "Type": "Task",
30       "Resource":
31       "arn:aws:lambda:us-east-2:637554956784:function:Function",
32       "Next": "NextState"
33     },
34     "SecondMatchState": {
35       "Type": "Task",
36       "Resource":
37       "arn:aws:lambda:us-east-2:637554956784:function:Function",
38       "Next": "NextState"
39     },
40     "NextState": {
41       "Type": "Task",
42       "Resource":
43       "arn:aws:lambda:us-east-2:637554956784:function:Function",
44       "Next": "End"
45     },
46     "DefaultState": {
47       "Type": "Task",
48       "Resource":
49       "arn:aws:lambda:us-east-2:637554956784:function:Function",
50       "Next": "End"
51     },
52     "End": {
53       "Type": "End"
54     }
55   }
56 }
```

The rendered graph on the right shows the following flow: Start (circle) → FirstState (rectangle) → ChoiceState (rectangle) → FirstMatchState (rectangle) and SecondMatchState (rectangle) → NextState (rectangle) → End (circle). There is also a path from ChoiceState to DefaultState (rectangle) → End (circle).

Trechos de código

Os trechos de código permitem inserir seções curtas do código. Para usar trechos de código:

1. Abra um arquivo e salve-o com a extensão `.asl.json` para o formato JSON ou `.asl.yaml` para o formato YAML.
2. Crie uma nova máquina de estado com a propriedade States (Estados).
3. Coloque o cursor dentro de States (Estados).
4. Use a combinação de teclas `Control + Space` e selecione seu trecho de código preferido.
5. Use `Tab` para percorrer a variável e os parâmetros no trecho de código.
6. Teste os trechos de código `Retry` (Repetir) e `Catch` (Capturar) colocando o cursor dentro do estado relacionado.



Preenchimento e validação de código

Para ver como funciona o preenchimento de código:

1. Crie vários estados.
2. Coloque o cursor após uma propriedade Avançar ou Padrão. `StartAt`
3. Use a combinação de teclas `Control + Space` para listar os preenchimentos disponíveis. É possível acessar propriedades adicionais usando `Control + Space` novamente, e será com base no valor `Type` de State.
4. À medida que você trabalhar, a validação de código ocorrerá para:

- Propriedades ausentes
- Valores incorretos
- Nenhum estado terminal
- Estados inexistentes que forem indicados

```

"FirstMatchState": {
  "Type": "Task",
  "Resource": "arn:aws:lambda:us-east-2:637554956784:function:Function",
  "Next": ""
},
"SecondMatchState": {
  "Type": "Task",
  "Resource": "arn:aws:lambda:us-east-2:637554956784:function:Function",
  "Next": "SecondMatchState"
},
"DefaultState": {
  "Type": "Fail",
  "Error": "DefaultStateError",
  "Cause": "No Matches!"
},
"NextState": {
  "Type": "Task",
  "Resource": "arn:aws:lambda:us-east-2:637554956784:function:Function",
  "End": true
}

```

```

"FirstMatchState": {
  "Type": "Task",
  "Resource": "arn:aws:lambda:us-east-2:637554956784:function:Function",
  "Catch": [
    {
      "Error": "DefaultStateError",
      "Cause": "No Matches!"
    }
  ]
},

```

An array of objects, called Catchers, that define a fallback state. This state is executed if the state encounters runtime errors and its retry policy is exhausted or isn't defined.

Trabalhando com o Threat Composer

Você pode usar o AWS Toolkit for Visual Studio Code para trabalhar com a ferramenta Threat Composer. O Threat Composer é uma ferramenta de modelagem de ameaças que pode simplificar seu processo de modelagem de ameaças.

Para obter informações detalhadas sobre a ferramenta Threat Composer, consulte o repositório [Threat Composer. GitHub](#)

Os tópicos a seguir descrevem como trabalhar com o Threat Composer no AWS Toolkit for Visual Studio Code.

Tópicos

- [Trabalhando com o Threat Composer a partir do kit de ferramentas](#)

Trabalhando com o Threat Composer a partir do kit de ferramentas

Com o Threat Composer, você pode criar, visualizar e editar modelos de ameaças do Threat Composer diretamente no VS Code. Para obter informações detalhadas sobre a ferramenta Threat Composer, consulte o repositório [Threat Composer. GitHub](#)

As seções a seguir descrevem como acessar as ferramentas do Threat Composer no AWS Toolkit for Visual Studio Code.

Acessando o Threat Composer a partir do kit de ferramentas

Há três maneiras principais de acessar o Threat Composer a partir do Toolkit.

Acessando o Threat Composer por meio de um modelo de ameaça existente

Para abrir o Threat Composer, abra um arquivo de modelo de ameaça existente (extensão .tc.json) no VS Code. O Threat Composer abre e renderiza automaticamente uma visualização do seu arquivo de modelo de ameaça na janela do editor do VS Code.

Criação de um novo modelo de ameaças do Threat Composer

1. No menu principal do VS Code, expanda Arquivo e escolha Novo Arquivo.
2. Na caixa de diálogo Novo arquivo, escolha Threat Composer File... .

3. Quando solicitado, digite a e pressione a **enter** tecla para abrir o Threat Composer e criar uma visualização do seu arquivo de modelo de ameaça vazio em uma nova janela do editor do VS Code. file name

Criação de um novo modelo de ameaça do Threat Composer a partir da Paleta de Comandos

1. No VS Code, abra a Paleta de Comandos pressionando **Cmd + Shift + P** ou **Ctrl + Shift + P** (Windows).
2. No campo de pesquisa, digite **Threat Composer** e escolha Criar novo arquivo Threat Composer quando ele for preenchido nos resultados.
3. Quando solicitado, digite a e pressione a **enter** tecla para abrir o Threat Composer e criar uma visualização do seu arquivo de modelo de ameaça vazio em uma nova janela do editor do VS Code. file name

Trabalhar com recursos da

Além de acessar AWS os serviços listados por padrão no AWS Explorer, você também pode acessar Recursos e escolher entre centenas de recursos para adicionar à interface. Em AWS, um recurso é uma entidade com a qual você pode trabalhar. Alguns dos recursos que podem ser adicionados incluem Amazon AppFlow, Amazon Kinesis Data AWS IAM Streams, funções, VPC Amazon e CloudFront distribuições da Amazon.

Depois de fazer sua seleção, você pode ir para Recursos e expandir o tipo de recurso para listar os recursos disponíveis para esse tipo. Por exemplo, se selecionar o tipo de recurso `AWS Toolkit::Lambda::Function`, você pode acessar os recursos que definem as diferentes funções, suas propriedades e seus atributos.

Depois de adicionar um tipo de recurso em Resources (Recursos), você pode interagir com ele e com seus recursos das seguintes maneiras:

- Veja uma lista dos recursos existentes que estão disponíveis na AWS região atual para esse tipo de recurso.
- Visualize uma versão somente para leitura do JSON arquivo que descreve um recurso.
- Copiar o identificador do recurso para o recurso.
- Veja a AWS documentação que explica a finalidade do tipo de recurso e o esquema (em JSON e em YAML formatos) para modelar um recurso.

- Crie um novo recurso editando e salvando um modelo JSON formatado que esteja em conformidade com um esquema. *
- Atualize ou exclua um recurso existente.*

Important

* Na versão atual do, AWS Toolkit for Visual Studio Code a opção de criar, editar e excluir recursos é um recurso experimental. Como os atributos experimentais continuam sendo testados e atualizados, eles podem ter problemas de usabilidade. E os recursos experimentais podem ser removidos do AWS Toolkit for Visual Studio Code sem aviso prévio.

Para permitir o uso de recursos experimentais para recursos, abra o painel Configurações no seu VS Code IDE, expanda Extensões e escolha Kit de AWS ferramentas.

Em AWS Toolkit Experiments, selecione `jsonResourceModification` para permitir que você crie, atualize e exclua recursos.

Para obter mais informações, consulte [Como trabalhar com atributos experimentais](#).

IAM permissões para acessar recursos

Você precisa de AWS Identity and Access Management permissões específicas para acessar os recursos associados aos AWS serviços. Por exemplo, uma IAM entidade, como um usuário ou uma função, exige permissões do `Lambda::Function` recursos.

Além das permissões para recursos de serviço, uma IAM entidade exige permissões para permitir que o Toolkit for VS Code AWS chame as operações do Cloud API Control em seu nome. APIs operações do Cloud Control permitem que o IAM usuário ou a função acesse e atualize os recursos remotos.

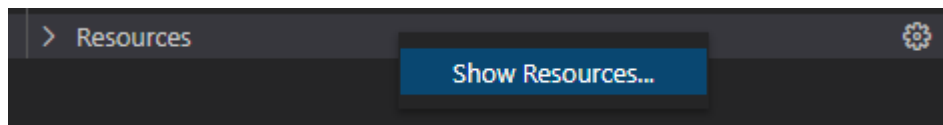
A maneira mais fácil de conceder permissões é anexar a política AWS gerenciada, `PowerUserAccess`, à IAM entidade que está chamando essas API operações usando a interface do Toolkit. Essa [política gerenciada](#) concede uma variedade de permissões para realizar tarefas de desenvolvimento de aplicativos, incluindo API operações de chamada.

Para obter permissões específicas que definem API operações permitidas em recursos remotos, consulte o [Guia do API usuário do AWS Cloud Control](#).

Como adicionar e interagir com recursos existentes

1. No AWS Explorer, clique com o botão direito em Recursos e selecione Mostrar recursos.

Um painel mostra uma lista de tipos de recurso disponíveis para seleção.



2. No painel de seleção, selecione os tipos de recurso a serem adicionados ao AWS Explorer e pressione Retornar ou selecione OK para confirmar.

Os tipos de recursos que você selecionou estão listados em Recursos.

Note

Se você já adicionou um tipo de recurso ao Explorador da AWS e desmarcou a caixa de seleção para esse tipo, ele não estará mais listado em Recursos depois de escolher OK. Somente os tipos de recursos atualmente selecionados ficam visíveis no Explorador da AWS.

3. Para exibir os recursos que já existem para um tipo de recurso, expanda a entrada para esse tipo de recurso.

Uma lista dos recursos disponíveis é exibida no tipo de recurso.

4. Para interagir com um recurso específico, clique com o botão direito do mouse no nome do recurso e selecione uma das seguintes opções:

- Copiar identificador de recurso: copie o identificador do recurso específico para a área de transferência. (Por exemplo, o recurso `AWS Toolkit:DynamoDB::Table` pode ser identificado usando a propriedade `TableName`.)
- Visualização: visualize uma versão somente para leitura do modelo JSON formatado que descreve o recurso.

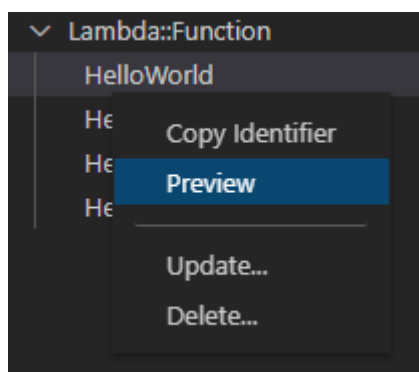
Depois que o modelo de recurso for exibido, você poderá modificá-lo escolhendo o ícone Atualizar à direita da guia do editor. Para atualizar um recurso, você deve estar com o [???](#) necessário habilitado.

- Atualização: edite o modelo JSON formatado para o recurso em um editor do VS Code. Para obter mais informações, consulte [Como criar e editar recursos](#).

- Excluir: exclua o recurso confirmando a exclusão em uma caixa de diálogo exibida. (Atualmente, a exclusão de recursos está [???](#) incluída nesta versão do AWS Toolkit for Visual Studio Code.)

Warning

Se você excluir um recurso, qualquer AWS CloudFormation pilha que use esse recurso falhará na atualização. Para corrigir essa falha de atualização, você precisa recriar o recurso ou remover a referência a ele no modelo da AWS CloudFormation pilha. Para obter mais informações, consulte este [artigo do Centro de Conhecimentos](#).



Como criar e editar recursos

Important

A criação e atualização de recursos é, atualmente, uma [???](#) neste versionamento do AWS Toolkit for Visual Studio Code.

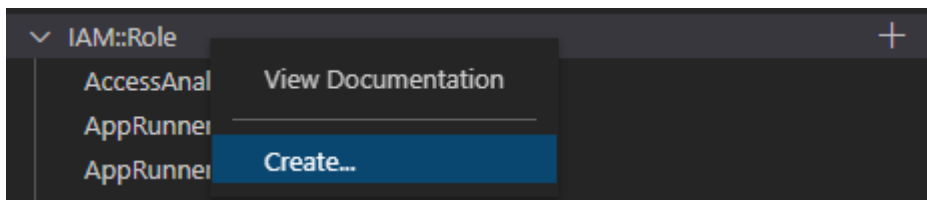
A criação de um novo recurso envolve adicionar um tipo de recurso à lista Recursos e, em seguida, editar um modelo JSON formatado que define o recurso, suas propriedades e seus atributos.

Por exemplo, um recurso que pertence ao tipo de `AWS Toolkit:SageMaker::UserProfile` recurso é definido com um modelo que cria um perfil de usuário para o Amazon SageMaker Studio. O modelo que define esse recurso do perfil de usuário deve estar em conformidade com o esquema do tipo de recurso para `AWS Toolkit:SageMaker::UserProfile`. Se o modelo não estiver em

conformidade com o esquema devido a propriedades ausentes ou incorretas, por exemplo, o recurso não poderá ser criado ou atualizado.

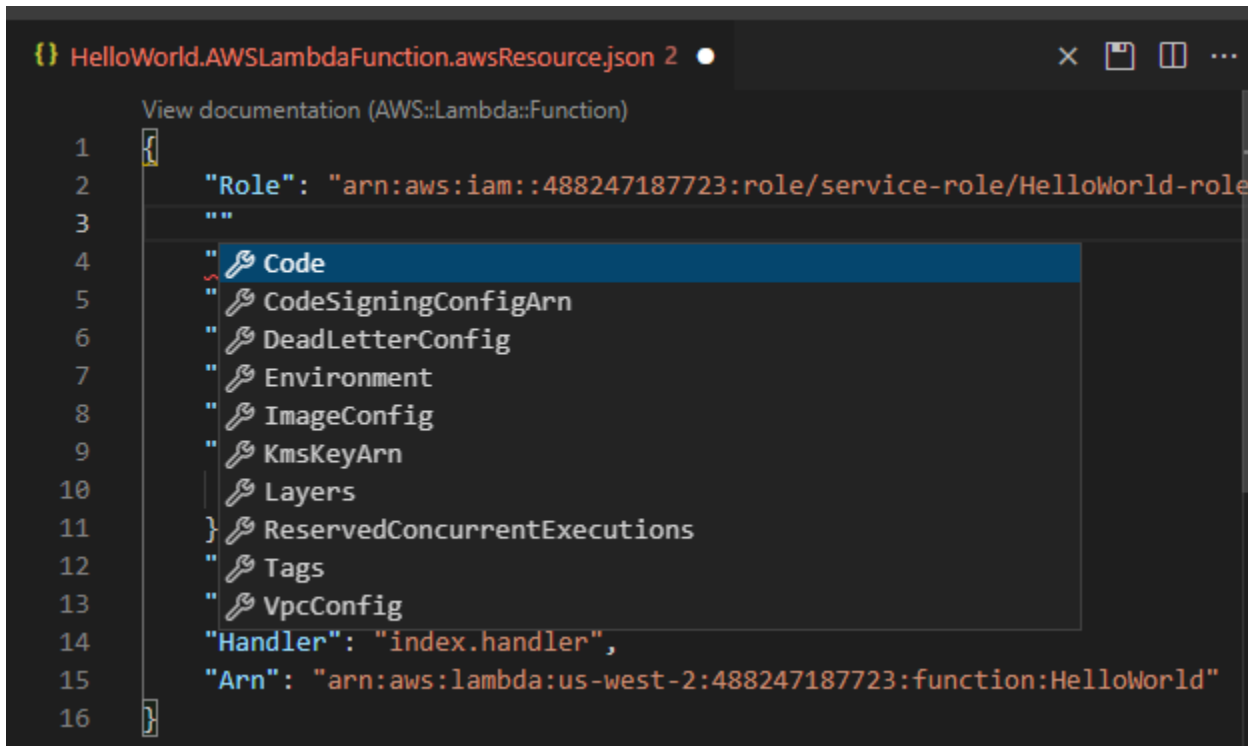
1. Adicione o tipo de recurso que você deseja criar clicando com o botão direito em Recursos e escolhendo Mostrar recursos.
2. Depois que o tipo de recurso for adicionado em Recursos, escolha o ícone de adição ("+") para abrir o arquivo de modelo no novo editor.

Ou você pode clicar com o botão direito do mouse no nome do tipo de recurso e selecionar Criar. Você também pode acessar informações sobre modelar o recurso ao selecionar Visualizar documentação.



3. No editor, comece a definir as propriedades que compõem o modelo do recurso. O atributo de preenchimento automático sugere nomes de propriedades que estejam de acordo com o esquema do seu modelo. Quando você passa o mouse sobre um tipo de propriedade, o painel exibe uma descrição da finalidade. Para obter informações detalhadas sobre o esquema, selecione Visualizar documentação.

O texto que não estiver em conformidade com o esquema de recursos será indicado por um sublinhado vermelho ondulado.



```
1 {
2   "Role": "arn:aws:iam::488247187723:role/service-role/HelloWorld-role",
3   ""
4   "Code"
5   "CodeSigningConfigArn"
6   "DeadLetterConfig"
7   "Environment"
8   "ImageConfig"
9   "KmsKeyArn"
10  "Layers"
11  "ReservedConcurrentExecutions"
12  "Tags"
13  "VpcConfig"
14  "Handler": "index.handler",
15  "Arn": "arn:aws:lambda:us-west-2:488247187723:function:HelloWorld"
16 }
```

4. Depois de terminar de declarar seu recurso, escolha o ícone Salvar para validar seu modelo e salvar o recurso na nuvem remota AWS .

Se seu modelo definir o recurso de acordo com seu esquema, uma mensagem será exibida para confirmar que o recurso foi criado. (Se o recurso já existir, a mensagem confirmará que o recurso foi atualizado.)

Depois que o recurso é criado, ele é adicionado à lista sob o título do tipo de recurso.

5. Se o arquivo contiver erros, será exibida uma mensagem explicando que o recurso não pôde ser criado ou atualizado. Selecione Visualizar logs para identificar os elementos do modelo que você precisa corrigir.

Segurança do AWS Toolkit for Visual Studio Code

Tópicos

- [Proteção de dados em AWS Toolkit for Visual Studio Code](#)

Proteção de dados em AWS Toolkit for Visual Studio Code

O [modelo de responsabilidade AWS compartilhada](#) se aplica à proteção de dados no AWS Toolkit for Visual Studio Code. Conforme descrito neste modelo, AWS é responsável por proteger a infraestrutura global que executa todos os Nuvem AWS. Você é responsável por manter o controle sobre seu conteúdo hospedado nessa infraestrutura. Você também é responsável pelas tarefas de configuração e gerenciamento de segurança dos serviços da AWS que usa. Para obter mais informações sobre privacidade de dados, consulte [Privacidade de dados FAQ](#). Para obter informações sobre proteção de dados na Europa, consulte o [Modelo de Responsabilidade AWS Compartilhada e GDPR](#) a postagem no blog AWS de segurança.

Para fins de proteção de dados, recomendamos que você proteja Conta da AWS as credenciais e configure usuários individuais com AWS IAM Identity Center ou AWS Identity and Access Management (IAM). Dessa maneira, cada usuário receberá apenas as permissões necessárias para cumprir suas obrigações de trabalho. Recomendamos também que você proteja seus dados das seguintes formas:

- Use a autenticação multifator (MFA) com cada conta.
- Use SSL/TLS para se comunicar com AWS os recursos. Exigimos TLS 1,2 e recomendamos TLS 1,3.
- Configure API e registre as atividades do usuário com AWS CloudTrail.
- Use soluções de AWS criptografia, juntamente com todos os controles de segurança padrão serviços da AWS.
- Use serviços gerenciados de segurança avançada, como o Amazon Macie, que ajuda a localizar e proteger dados sigilosos armazenados no Amazon S3.
- Se você precisar de FIPS 140-3 módulos criptográficos validados ao acessar AWS por meio de uma interface de linha de comando ou uma API, use um endpoint. FIPS Para obter mais informações sobre os FIPS endpoints disponíveis, consulte [Federal Information Processing Standard \(FIPS\) 140-3](#).

É altamente recomendável que nunca sejam colocadas informações de identificação confidenciais, como endereços de e-mail dos seus clientes, em marcações ou campos de formato livre, como um campo Nome. Isso inclui quando você trabalha com o AWS Toolkit for Visual Studio Code ou serviços da AWS outro usando o consoleAPI,, AWS CLI, AWS SDKs ou. Quaisquer dados inseridos em tags ou campos de texto de formato livre usados para nomes podem ser usados para logs de faturamento ou de diagnóstico. Se você fornecer um URL para um servidor externo, é altamente recomendável que você não inclua informações de credenciais no URL para validar sua solicitação para esse servidor.

Histórico de documentos para o Guia AWS Toolkit for Visual Studio Code do usuário

A tabela a seguir descreve as alterações importantes em cada versão do AWS Toolkit for Visual Studio Code. Para receber notificações sobre atualizações desta documentação, você pode assinar um [RSSfeed](#).

Alteração	Descrição	Data
AWS Atualizações do Identity and Access Management (IAM) Access Analyzer	Conteúdo atualizado do IAM Access Analyzer para incluir novas API referências.	10 de julho de 2024
AWS Identity and Access Management (IAM) Analisador de acesso	Foi adicionado um novo tópico do guia do usuário para o IAM Access Analyzer.	23 de maio de 2024
Conecte-se ao fluxo de AWS autorização atualizado	O fluxo de autorização foi atualizado para refletir as mudanças no processo de autenticação e a separação do Amazon Q do AWS Toolkit for Visual Studio Code.	30 de abril de 2024
Extensão Amazon Q para VS Code	Em 30 de abril de 2024, agora CodeWhisperer faz parte do Amazon Q e o Amazon Q está disponível como uma extensão para o VS Code.	30 de abril de 2024
Support para nuvem privada virtual em ambientes de desenvolvimento	Conteúdo atualizado que abrange mudanças na interface do usuário para suporte VPC em ambientes de desenvolvimento.	21 de janeiro de 2024

Compositor de aplicativos	Foi adicionado um novo tópico do Application Composer ao Guia AWS Toolkit for Visual Studio Code do usuário.	28 de novembro de 2023
SSOsuporte para CodeCatalyst	Conteúdo atualizado para cobrir o suporte do IAM Identity Center CodeCatalyst e os ambientes de desenvolvimento.	17 de novembro de 2023
Links de download do VS Code e do Toolkit adicionados	Conteúdo atualizado com links para download do VS Code e AWS Toolkit for Visual Studio Code o.	1.º de novembro de 2023
Tópico do Amazon Redshift	Foi adicionado um novo tópico do Amazon Redshift ao Guia do AWS Toolkit for Visual Studio Code usuário.	17 de outubro de 2023
Conecte-se ao fluxo de AWS autorização atualizado	Fluxo de autorização atualizado para se concentrar nos métodos de autenticação específicos do serviço.	29 de setembro de 2023
Guia do usuário criado: Crie um modelo CloudFormation	Criou um novo guia do usuário descrevendo como criar um CloudFormation modelo usando o Toolkit for VS Code	17 de dezembro de 2021
Atualização secundária da interface de usuário	Texto existente atualizado sobre “Visualizar estado da máquina” para “Renderizar gráfico” e melhorar a correspondência com a interface de usuário.	14 de dezembro de 2021

Criação do guia do usuário do Amazon Elastic Container Service Exec	Esta é uma visão geral do Amazon ECS Exec.	13 de dezembro de 2021
Guia do usuário criado para o serviço AWS IoT Toolkit for VS Code	Este guia do usuário tem como objetivo ajudá-lo a começar a usar o AWS IoT serviço do Toolkit for VS Code.	22 de novembro de 2021
Suporte para atributos experimentais	Foi adicionado suporte para ativar recursos experimentais para AWS serviços.	14 de outubro de 2021
Support for AWS resources	Suporte adicional para acessar tipos de recursos, junto com opções de interface , para criar, editar e excluir os recursos.	14 de outubro de 2021
Visão geral do ECR serviço da Amazon para AWS Toolkit for Visual Studio Code	Foi adicionada uma visão geral e um passo a passo dos recursos e funções do ECR serviço Amazon que podem ser acessados no VS Code	14 de outubro de 2021
Support para ARM64 ambientes	Agora você pode executar aplicativos sem servidor em ambientes emulados ARM64 baseados, bem como em ambientes baseados em x86_64.	1.º de outubro de 2021
AWS Serverless Application	Suporte adicional para execução de AWS SAM aplicativos na ARM64 plataforma	30 de setembro de 2021

Seção de atualização de formato do Node.js	De acordo com o feedback do cliente, formatação atualizada para TypeScript Node.js/.	12 de agosto de 2021
Suporte ao App Runner	Foi adicionado suporte AWS App Runner para AWS Toolkit for Visual Studio Code o.	11 de agosto de 2021
Depurar funções do Go	Compatibilidade adicionada para depuração de funções locais do Go.	10 de maio de 2021
Depurar funções do Java	Compatibilidade adicionada para depuração de funções locais do Java.	22 de abril de 2021
YAMLSuporte para AWS Step Functions	Foi adicionado YAML suporte para AWS Step Functions.	4 de março de 2021
Depuração de recursos do Amazon Gateway API	Foi adicionado suporte para depuração de recursos locais do Amazon API Gateway.	1º de dezembro de 2020
Amazon API Gateway	Foi adicionado suporte para o Amazon API Gateway.	1º de dezembro de 2020
AWS Serverless Application	Compatibilidade adicionada para imagens de contêiner do Lambda com aplicações sem servidor.	1º de dezembro de 2020
AWS Systems Manager apoio	Compatibilidade adicionada para documentos do Systems Manager Automation.	30 de setembro de 2020
CloudWatch Registros	Foi adicionado suporte para CloudWatch Logs.	24 de agosto de 2020

Amazon S3	Compatibilidade adicionada para o Amazon S3.	30 de julho de 2020
AWS Step Functions apoio	Foi adicionado suporte para AWS Step Functions.	31 de março de 2020
Conteúdo de segurança	Conteúdo de segurança adicionado.	6 de fevereiro de 2020
Trabalhando com EventBridge esquemas da Amazon	Suporte adicional para EventBridge esquemas da Amazon	1.º de dezembro de 2019
AWS CDK	Versão prévia do AWS CDK serviço.	25 de novembro de 2019
Usar um processo de credencial externo	Informações adicionadas sobre o uso de um processo de credencial externo para obter credenciais da AWS .	25 de setembro de 2019
Usando IntelliSense para arquivos de definição de tarefas	IntelliSense foi adicionado suporte para trabalhar com arquivos de definição de ECS tarefas da Amazon.	24 de setembro de 2019
Guia do usuário para o AWS Toolkit for Visual Studio Code	Versão de disponibilidade geral.	11 de julho de 2019
Guia do usuário para o AWS Toolkit for Visual Studio Code	Atualizamos a estrutura do documento por motivo de clareza e facilidade de uso.	3 de julho de 2019
Instalando o AWS Toolkit for Visual Studio Code	Foram adicionadas informações sobre a instalação do idioma SDKs para oferecer suporte a vários conjuntos de ferramentas.	12 de junho de 2019

[Configurar uma cadeia de ferramentas](#)

Foram adicionadas informações sobre como configurar várias cadeias de ferramentas.

12 de junho de 2019

[Versão inicial](#)

Versão inicial do guia do usuário do AWS Toolkit for Visual Studio Code.

28 de março de 2019

As traduções são geradas por tradução automática. Em caso de conflito entre o conteúdo da tradução e da versão original em inglês, a versão em inglês prevalecerá.