
AWSToolkit for VS Code

Manual do usuário



AWSToolkit for VS Code: Manual do usuário

Copyright © 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 AWS Toolkit for Visual Studio Code ?	1
Informações relacionadas	1
Configuração	2
Instalação do Toolkit for VS Code	2
Prerequisites	2
Instalar o Toolkit for VS Code	3
Estabelecer credenciais	4
O uso doAWS-credenciais suportadas	4
Usando um processo de credencial externa	10
Como conectar-se ao AWS	10
Conectar-se aoAWSPor meio do Toolkit for VS Code	10
Usar váriasAWSContas da com o Toolkit	12
AlterandoAWSRegiões da	12
Adicionar uma Região ao AWS Explorer	13
Ocultar uma região do AWS Explorer	14
Configurar a cadeia de ferramentas	15
Configurar uma cadeia de ferramentas para .NET Core	15
Configurar uma cadeia de ferramentas para Node.js	16
Configurar uma cadeia de ferramentas para Python	16
Configurar uma cadeia de ferramentas para Java	16
Configurar uma cadeia de ferramentas para Go	16
Como usar a cadeia de ferramentas	17
Navegando pelo Toolkit for VS Code	18
Componentes fundamentais da IU	18
O AWS Explorer	18
O AWS CDK Explorer	19
Locais de Comando	19
Paleta de comandos de código do Visual Studio	19
AWSExplorerMenu	21
Como trabalhar com os serviços da AWS	22
Recursos experimentais	22
AWS Explorer	22
Amazon API Gateway	24
AWS App Runner	24
AWS CloudFormationPilhas do	30
Amazon CloudWatch Logs	31
Amazon EventBridge	33
Funções AWS Lambda	35
Amazon S3	41
AWS Systems Manager	45
AWS Step Functions	49
Recursos	56
Amazon ECS	59
Usando o IntelliSense para arquivos de definição de tarefas	60
AWS CDK Explorer	60
AWS CDKAplicativos do	61
Aplicação sem servidor da AWS	63
Pressupostos e pré-requisitos	63
Permissões do IAM para aplicativos sem servidor	64
Criando um novo aplicativo sem servidor (local)	64
Abrindo um aplicativo sem servidor (local)	65
Executando e depurando um aplicativo sem servidor a partir do modelo (local)	66
Implantar um aplicativo sem servidor noAWSNuvem	67

Excluir um aplicativo sem servidor doAWSNuvem	68
Executando e depurando funções do Lambda diretamente do código	68
Executando e depurando recursos locais do Amazon API Gateway	71
Opções de configuração para depurar aplicativos sem servidor	73
Solução de problemas	78
Segurança	80
Proteção de dados	80
Identity and Access Management	81
Registro em log e monitoramento	81
Validação de conformidade	82
Resiliência	82
Segurança da infraestrutura	83
Análise de configuração e vulnerabilidade	83
Histórico do documento	84
.....	lxxxvi

AWS Toolkit for Visual Studio Code

Este é o guia do usuário doAWSToolkit for 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 AWS Toolkit for Visual Studio Code ?

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

Topics

- [Configurar o AWS Toolkit for Visual Studio Code](#) (p. 2)
- [Navegação no AWS Toolkit for Visual Studio Code](#) (p. 18)
- [Como trabalhar com os serviços da AWS](#) (p. 22)

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, visite<https://code.visualstudio.com/>.

Configurar o AWS Toolkit for Visual Studio Code

Esta seção ajuda a configurar o AWS Toolkit for Visual Studio Code . Ela fornece informações sobre como instalar e configurar o toolkit, configurar suas credenciais e se conectar àAWS.

Tópicos

- [Instalar o AWS Toolkit for Visual Studio Code](#) (p. 2)
- [Estabelecer credenciais para o AWS Toolkit for Visual Studio Code](#) (p. 4)
- [Conectar-se à AWS por meio do AWS Toolkit for Visual Studio Code](#) (p. 10)
- [AlterandoAWSRegiões da](#) (p. 12)
- [Configurar a cadeia de ferramentas](#) (p. 15)

Instalar o AWS Toolkit for Visual Studio Code

Esta seção descreve como instalar o AWS Toolkit for Visual Studio Code .

Prerequisites

Required

Antes de instalar o Toolkit for VS Code, você deve ter o seguinte:

- Uma conta da Amazon Web Services— Para obter umAWS, acesse o[AWSPágina inicial do](#). SelecioneCriar umAWSConta, ouComplete Sign Up(se você já visitou o site antes). O cadastro permitirá usar todos os serviços oferecidos pela AWS.
- Um sistema operacional compatível— O Toolkit for VS Code é compatível com Windows, Linux e macOS.
- VS Code versão 1.42.0 ou posterior— tentamos manter o Toolkit for VS Code atualizado com a versão padrão disponível na[Página de download de código VS](#).

Optional

Antes de poder utilizar determinadas funcionalidades do Toolkit for VS Code, é preciso ter:

- Desenvolvimento de código— o SDK relevante para o idioma 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:<http://aws.amazon.com/corretto/>
 - Go SDK:<https://golang.org/doc/install>
- AWS SAMCLI— Este é umAWSCLI do 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 em seguida), pois ele é necessário para as funcionalidades

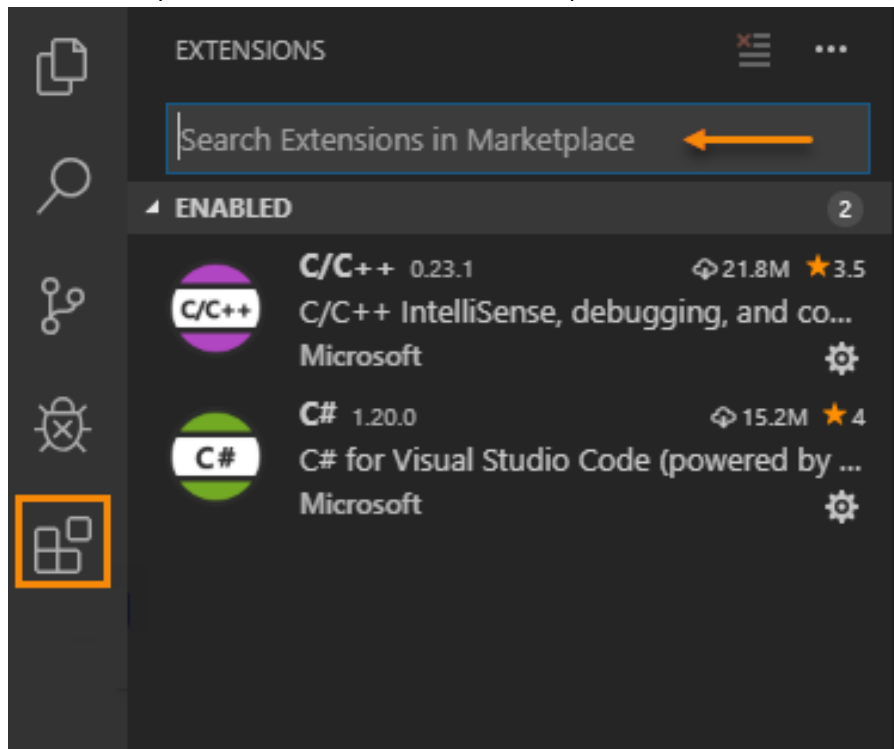
do AWS Serverless Application Model (AWS SAM), como [Criando um novo aplicativo sem servidor \(local\)](#) (p. 64).

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

- 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 pacote— Um gerenciador de pacotes para que você possa baixar e compartilhar o código do aplicativo.
 - .NET: [NuGet](#)
 - Node.js: [NPM](#)
 - Python: [pip](#)
 - Java: [Gradle](#) ou [Maven](#)

Instalar o Toolkit for VS Code

1. Inicie o editor de código VS.
2. Na barra lateral do editor de código VS, escolha o ícone de Extensões. Essa ação abre a Extensões, que a você acessar o VS Code Marketplace.



3. Na caixa de pesquisa de Extensions (Extensões), procure AWS Toolkit. Escolha a entrada para visualizar seus detalhes no painel à direita.
4. No painel à direita, escolha Install (Instalar).
5. Depois da instalação, se for solicitado que você reinicie o editor, escolha Reload Required (Necessário recarregar) para concluir a instalação.

Depois de instalar o Toolkit for VS Code, você deve configurar [as credenciais](#) (p. 4) para permitir que você acesse seus recursos AWS de dentro do VS Code.

Estabelecer credenciais para o AWS Toolkit for Visual Studio Code

Esta seção mostra os tipos de credenciais que é possível usar com o AWSToolkit for VS Code. Ela fornece informações sobre como obter e configurar essas credenciais.

Você pode obter credenciais por meio do AWS e fornecê-las ao toolkit usando arquivos de configuração. Você também pode obter credenciais por meio de um processo credencial externo que não é diretamente compatível com a AWS.

Tópicos

- [O uso do AWS-credenciais suportadas](#) (p. 4)
- [Usando um processo de credencial externa](#) (p. 10)

O uso do AWS-credenciais suportadas

AWS podem ser fornecidas para o AWSToolkit for VS Code de usando seu Compartilhado AWS arquivo confi gou seu Compartilhado AWS arquivo de credenciais. Os métodos para usar esses arquivos são os mesmos usados para a AWS CLI. Para obter informações gerais sobre como usar esses arquivos, consulte [Arquivos de configuração e credencial](#) no AWS Command Line Interface Guia do usuário do.

Tópicos

- [Obter o AWS Chaves de acesso](#) (p. 4)
- [Configuração do AWS credenciais](#) (p. 5)
- [Uso de credenciais do AWS SSO](#) (p. 8)

Obter o AWS Chaves de acesso

Para acessar a Amazon Web Services (AWS) com o AWS Toolkit for Visual Studio Code , você deve configurar o toolkit com o AWS credenciais de conta. Para fazer isso com credenciais compatíveis com a AWS, primeiro você deve obter as chaves de acesso apropriadas da AWS.

Para obter mais informações sobre os usuários e as credenciais que estão fora do escopo deste guia, consulte os seguintes recursos:

- [AWS Credenciais de segurança](#) da Amazon Web Services Referência geral
- [Visão geral do gerenciamento de identidades: Usuários](#) do IAM User Guide

O que é? AWS Chaves de acesso

As chaves de acesso são as credenciais que identificam você para a AWS e permitem acessar de forma programática os serviços e recursos da AWS. As chaves de acesso podem ser associadas à sua AWS (o “usuário raiz” da conta) ou com usuários criados com o AWS Identity and Access Management (IAM).

Warning

Como o usuário raiz é essencialmente um administrador com acesso total aos serviços e recursos, em vez disso, recomendamos que você crie um usuário do IAM somente com as permissões para executar as tarefas necessárias. Em seguida, você pode usar uma chave de acesso associada a esse usuário para suas credenciais. Para obter mais detalhes, consulte [Criação de um usuário do IAM no AWS](#) no IAM User Guide.

Uma chave de acesso consiste em um ID de chave de acesso, que é semelhante a um nome de usuário, e uma chave de acesso secreta, que é semelhante a uma senha. Essa chave de acesso é usada para assinar solicitações programáticas que você cria para a AWS. Se você não tiver chaves de acesso, você pode criá-las usando AWS Management Console. [Recomendamos que você use as chaves de acesso para um usuário do IAM em vez das chaves do usuário raiz da conta.](#)

Note

Para criar chaves de acesso, você deve ter permissões para realizar as ações do IAM necessárias. Para obter mais informações, consulte [Conceder permissão ao usuário do IAM para gerenciar política de senha e credenciais](#) no IAM User Guide.

Obtenha seuAWSChaves de acesso

1. Abra o console do IAM em <https://console.aws.amazon.com/iam/>.
2. No menu de navegação, selecione Users (Usuários).
3. Selecione seu nome de usuário do IAM (não a caixa de seleção) para visualizar seus detalhes.
4. Na guia Security credentials (Credenciais de segurança), escolha Create access key (Criar chave de acesso).
5. Para ver a nova chave de acesso, escolha Mostrar. As credenciais são semelhantes ao seguinte:
 - ID de chave de acesso: AKIAIOSFODNN7EXAMPLE
 - Chave de acesso secreta: wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY
6. Para baixar o par de chaves, escolha Baixar arquivo .csv. Armazene as chaves em um lugar seguro.

Important

- Mantenha a confidencialidade das chaves para proteger sua conta da AWS e nunca as envie por e-mail. Não compartilhe as chaves fora da sua organização, mesmo se uma pesquisa parecer vir da AWS ou do site Amazon.com. Alguém que legitimamente represente a Amazon jamais pedirá a você sua chave secreta.
- Você não pode recuperar a chave secreta se perdê-la. Em vez disso, por motivos de segurança, você deve criar um novo par de chaves e excluir o par antigo.

Depois de ter obtido o seuAWS, você pode usar as teclasAWSToolkit for VS Code de para armazená-los em seuAWSarquivo de configuração ou seu arquivo compartilhadoAWSarquivo de credenciais. Para saber como, consulte [Adicione seuAWSChaves de acesso ao seu ambiente](#) (p. 6).

Configuração doAWSCredenciais

Para acessar a Amazon Web Services (AWS) com o AWS Toolkit for Visual Studio Code , você deve fazer com que seuAWSdisponíveis para o kit de ferramentas. Para usar credenciais compatíveis com a AWS, continue lendo este tópico. Para usar um processo de credencial externa, consulte [Usando um processo de credencial externa](#) (p. 10).

Note

Alguns recursos do Toolkit for VS Code, como criar um aplicativo sem servidor, não exigemAWSCredenciais da .

Obtenha seuAWSChaves de acesso

Se ainda não tiver as chaves de acesso adequadas da AWS para armazenar no seu arquivo de configuração compartilhado da AWS ou no arquivo de credenciais compartilhado da AWS, você deve obtê-las agora.

Para fazer isto, consulte [Obter oAWSChaves de acesso \(p. 4\)](#).

Sobre o CompartilhadoAWSfiles

O arquivo de configuração compartilhado da AWS e o arquivo de credenciais compartilhado da AWS são arquivos que você pode usar para armazenar informações de configuração e credenciais da AWS. Por padrão, esses arquivos estão localizados no diretório `.aws`, dentro do diretório inicial, e são nomeados `config` e `credentials`, respectivamente. Para obter mais informações, consulte [Onde as definições de configuração ficam armazenadas?](#) no [AWS Command Line Interface Guia do usuário do](#).

O Toolkit for VS Code localiza e usaAWSChaves de acesso através do seuAWSarquivo de configuração e seu arquivo compartilhadoAWSarquivo de credenciais. Este é o método usado pelo AWS CLI e SDKs da AWS. As chaves de acesso inseridas no Toolkit for VS Code são salvas em um desses arquivos.

Esses arquivos compartilhados podem conter as credenciais de mais de uma conta da AWS, armazenadas como perfis. Várias contas podem ser úteis, por exemplo, para fornecer a desenvolvedores e administradores recursos separados para desenvolvimento e liberação/publicação.

Adicione seuAWSChaves de acesso ao seu ambiente

Se você já definiu seuAWS(por exemplo, usando a propriedadeAWS CLI), o Toolkit for VS Code detectará automaticamente essas credenciais e as disponibilizará para o toolkit. Se você ainda não tiver definido seuAWScredenciais, ou se você deseja incluirAWSEm seu ambiente ou atualizar um perfil de credenciais existente, você pode fazê-lo por meio do Toolkit for VS Code, como mostrado aqui.

Note

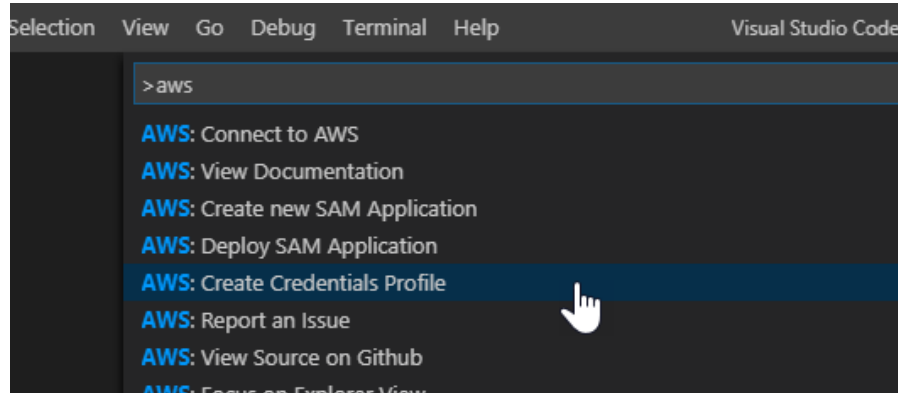
Como alternativa a esses procedimentos, você pode usar o comando [aws configure](#) da AWS CLI para adicionar as credenciais da AWS ao seu ambiente. Você também pode usar `aws configure` para definir o padrão doAWSRegião, necessário para determinadas operações, como a criação de um aplicativo sem servidor.

Crie oAWSarquivo de credenciais

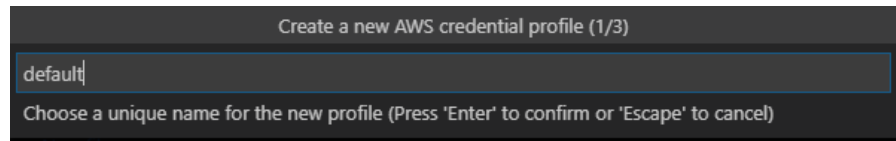
- Se você já tiver um arquivo de credenciais compartilhado da AWS, pode pular para o [próximo procedimento \(p. 7\)](#).
- Se você já tem umAWS Confige quiser usá-lo, você poderá pular para o [próximo procedimento \(p. 7\)](#).
- Se você tiver apenas umAWS ConfigMas NÃO quiser usá-lo, você deve primeiro criar umaAWSusando técnicas que são normais para o seu sistema operacional. Depois disso, você pode pular para o [próximo procedimento \(p. 7\)](#).

Siga estas etapas para criar o arquivo de credenciais compartilhado da AWS.

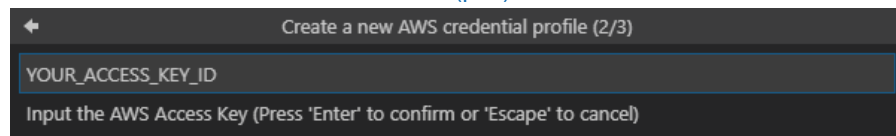
1. Abra VS Code.
2. Para abrir a Command Palette (Paleta de comandos), na barra de menus, escolha View (Exibir), Command Palette (Paleta de comandos). Ou use as teclas de atalho a seguir:
 - Windows e Linux: pressioneCtrl+Shift+P.
 - macOS: pressioneShift+Command+P.
3. Search ByAWSe escolhaAWS: Criar Perfil de Credenciais.



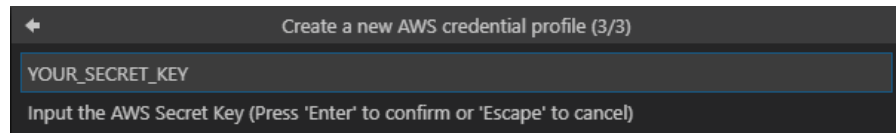
4. Atribua um nome ao perfil inicial.



5. Insira o ID da chave de acesso para as credenciais. Se você não tiver um ID de chave de acesso, consulte [Obter oAWSChaves de acesso \(p. 4\)](#).



6. Insira a chave de acesso secreta para as credenciais. Se você não tiver uma chave de acesso secreta, consulte [Obter oAWSChaves de acesso \(p. 4\)](#).

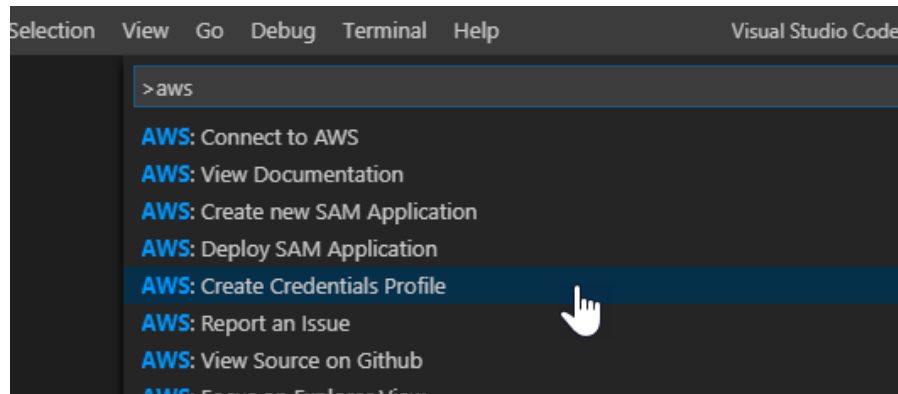


Depois de concluir esse procedimento, você pode verificar o arquivo de credenciais compartilhado da AWS abrindo-o diretamente ou executando as três primeiras etapas do próximo procedimento (sem fazer nenhuma atualização).

[Atualizar um de seus arquivos compartilhados](#)

Siga estas etapas para adicionar um novo perfil ao arquivo de configuração compartilhado existente da AWS ou ao arquivo de credenciais compartilhado da AWS. Você também pode atualizar um perfil existente.

1. Código VS Aberto.
2. Para abrir a Command Palette (Paleta de comandos), na barra de menus, escolha View (Exibir), Command Palette (Paleta de comandos). Ou use as teclas de atalho a seguir:
 - Windows e Linux: pressioneCtrl+Shift+P.
 - macOS: pressioneShift+Command+P.
3. Search ByAWS e escolhaAWS: Criar Perfil de Credenciais.



4. Quando um ou ambos os arquivos compartilhados forem abertos no editor de código do VS, adicione ou atualize um perfil.
5. Ao finalizar a atualização, salve o arquivo.

Adicionar perfis de credencial adicionais

É possível adicionar outros perfis e credenciais. Para isso, abra oPaleta de comandos doe escolhaAWS: Criar Perfil de Credenciais. Isso abrirá o arquivo de credenciais. Nesta página, você pode adicionar um novo perfil abaixo do seu primeiro perfil, como no exemplo abaixo:

```
# 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/bPxrFcIYEXAMPLEKEY
[Profile2_Name]
aws_access_key_id = AKIAI44QH8DHBEXAMPLE
aws_secret_access_key = je7MtGbClwBF/2Zp9Utk/h3yCo8nvbEXAMPLEKEY
```

Uso de credenciais do AWS SSO

Para se conectar com oAWS Single Sign-On(AWS SSO), conclua os pré-requisitos a seguir:

1. Habilitar oAWS SSO— Isso inclui escolher sua fonte de identidade e configurarAWS SSOAcesso aoAWScontas. Para obter mais informações, consulteConceitos básicosnoAWS Single Sign-OnGuia do usuário do.
2. Adicionar umAWS SSOPerfil do— ComAWS SSO, você define um perfil nomeado no arquivo de credenciais ou configuração que você usa para recuperar credenciais temporárias para o seuAWSConta.

A definição de perfil especifica o AWS SSO [portal de usuários](#), bem como o AWS Conta e função do IAM associados ao usuário que solicita acesso.

Como adicionar um AWS SSO Perfil do

O procedimento a seguir descreve como adicionar um AWS SSO para o seu `credentials` ou `config` file.

Adicionar um AWS SSO para seu arquivo de credenciais no VS Code

1. Abra VS Code.
2. Para abrir a Command Palette (Paleta de comandos), na barra de menus, escolha View (Exibir), Command Palette (Paleta de comandos). Ou use as teclas de atalho a seguir:
 - Windows e Linux: pressione `Ctrl+Shift+P`.
 - macOS: pressione `Shift+Command+P`.
3. Search By AWS e escolha AWS: Criar Perfil de Credenciais. Isso abrirá o arquivo de credenciais.
4. No `credentials` ou `config` arquivo, em `[default]`, adicione um modelo para um AWS SSO Perfil do. Siga-se um perfil de exemplo:

```
... Named profile in credentials file ...

[profile sso-user-1]
sso_start_url = https://example.com/start
sso_region = us-east-2
sso_account_id = 123456789011
sso_role_name = readOnly
region = us-west-2
```

Important

Não use a palavra `Perfil do` Ao criar uma entrada no arquivo de credenciais. Isso ocorre porque o arquivo de credenciais usa um formato de nomeação diferente do que o arquivo de configuração. Inclua a palavra de prefixo `profile` Somente ao configurar um perfil nomeado no arquivo de configuração.

Ao atribuir valores ao seu perfil, lembre-se do seguinte:

- **sso_start_url**— O URL que aponta para o AWS SSO portal de usuários.
- **sso_region**— O AWS Região que contém seu AWS SSO host de portal. Isso pode ser diferente do AWS Região especificada posteriormente no padrão `region` parâmetro.
- **sso_account_id**— O AWS ID da conta que contém a função do IAM com a permissão que você deseja conceder a este AWS SSO usuário.
- **sso_role_name**— O nome da função do IAM que define as permissões do usuário ao usar esse perfil para obter credenciais por meio do AWS SSO.
- **region**— O padrão do AWS Região que este AWS SSO usuário entrará em.

Fazer login com AWS SSO

Ao iniciar sessão com um AWS SSO, o navegador padrão é iniciado no portal especificado. Você deve verificar seu AWS SSO efetue login antes de poder acessar seu AWS recursos em VS Code. Observe que, se suas credenciais expirarem, você precisará repetir o processo de conexão para obter novas credenciais temporárias.

Usando um processo de credencial externa

Se você tiver um processo para gerar ou procurar credenciais não compatíveis diretamente com aAWS, configure oAWSToolkit for VS Code de usar esse processo em vez de qualquer[armazenadoAWSCredenciais](#) (p. 4).

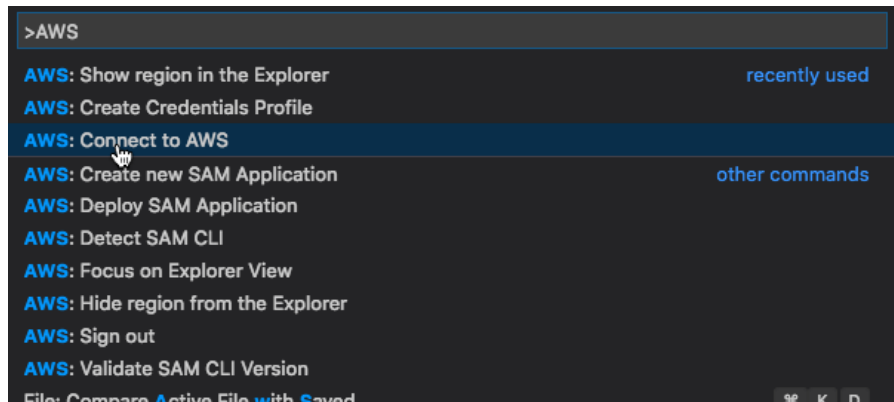
O método para especificar esse processo credencial externo é o mesmo usado para a AWS CLI e consiste em adicionar uma definição `credential_process` ao arquivo de configuração compartilhado da AWS. Para obter informações detalhadas sobre como fazer isso, consulte [Credenciais de fornecimento com um processo externo](#) noAWS Command Line InterfaceGuia do usuário do.

Conectar-se à AWS por meio do AWS Toolkit for Visual Studio Code

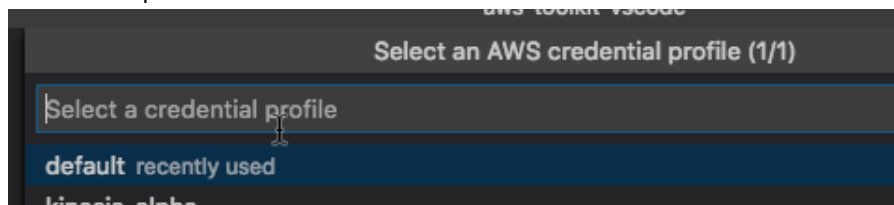
Para interagir com a Amazon Web Services (AWS) por meio do AWS Toolkit for Visual Studio Code , você deve estabelecer uma conexão com aAWS.

Conectar-se aoAWSPor meio do Toolkit for VS Code

1. Abra VS Code.
2. Para abrir a Command Palette (Paleta de comandos), na barra de menus, escolha View (Exibir), Command Palette (Paleta de comandos). Ou use as teclas de atalho a seguir:
 - Windows e Linux: pressioneCtrl+Shift+P.
 - macOS: pressioneShift+Command+P.
3. SearchAWS e escolhaAWS: Conectar-se aoAWS.



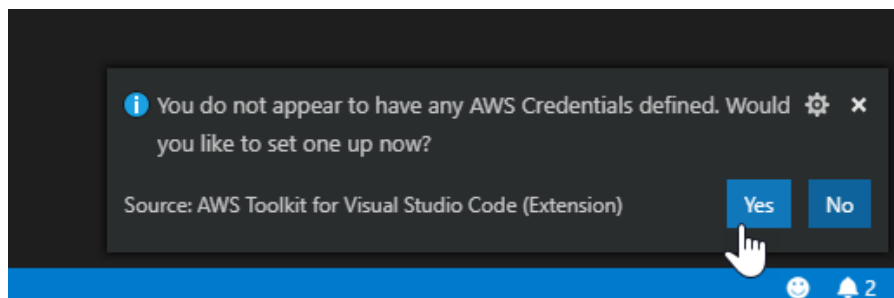
4. Escolha um perfil na lista.



Se não tiver um perfil de credenciais configurado, será solicitado que você configure um. Procure um pop-up no canto inferior direito do editor. Escolha Yes (Sim) e siga o assistente de configuração para inserir um nome de perfil, seu ID da chave de acesso e sua chave de acesso secreta. Para obter mais detalhes, consulte [Configuração doAWSCredenciais](#) (p. 5).

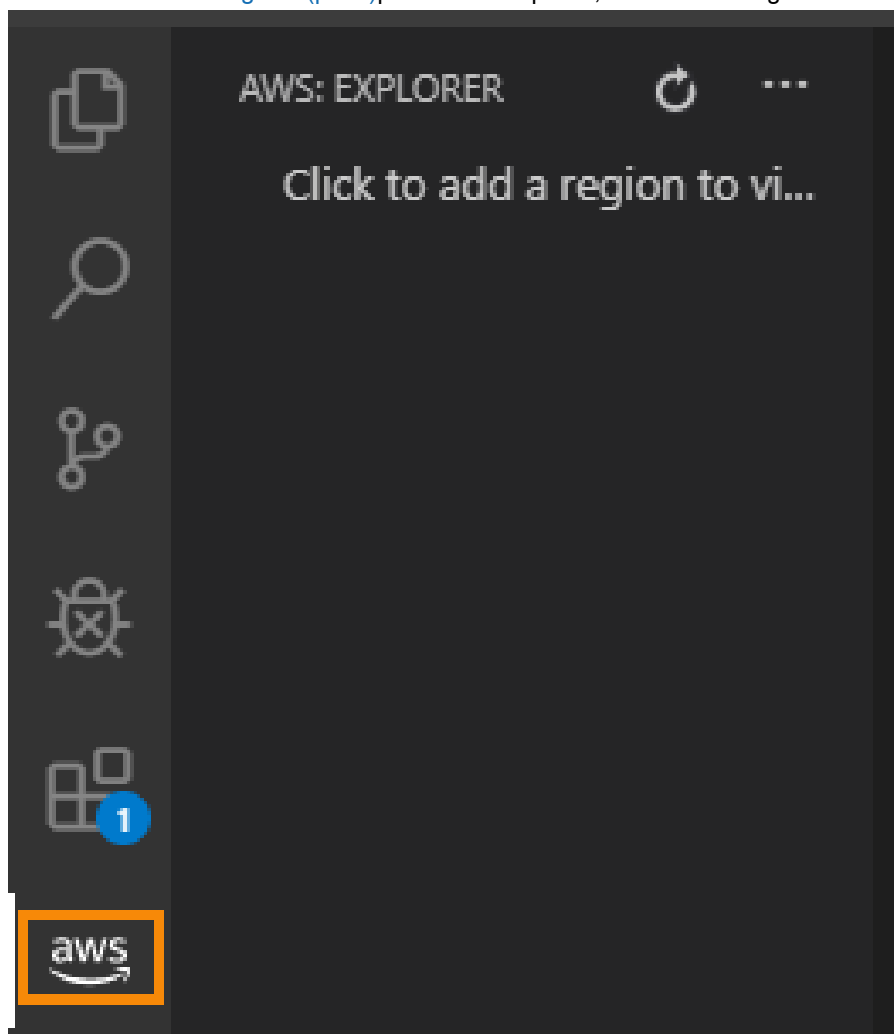
Note

Se você quiser fornecer um processo de credencial externa em vez de usar credenciais compatíveis com a AWS, escolha No (Não) e consulte [Usando um processo de credencial externa \(p. 10\)](#) como alternativa.

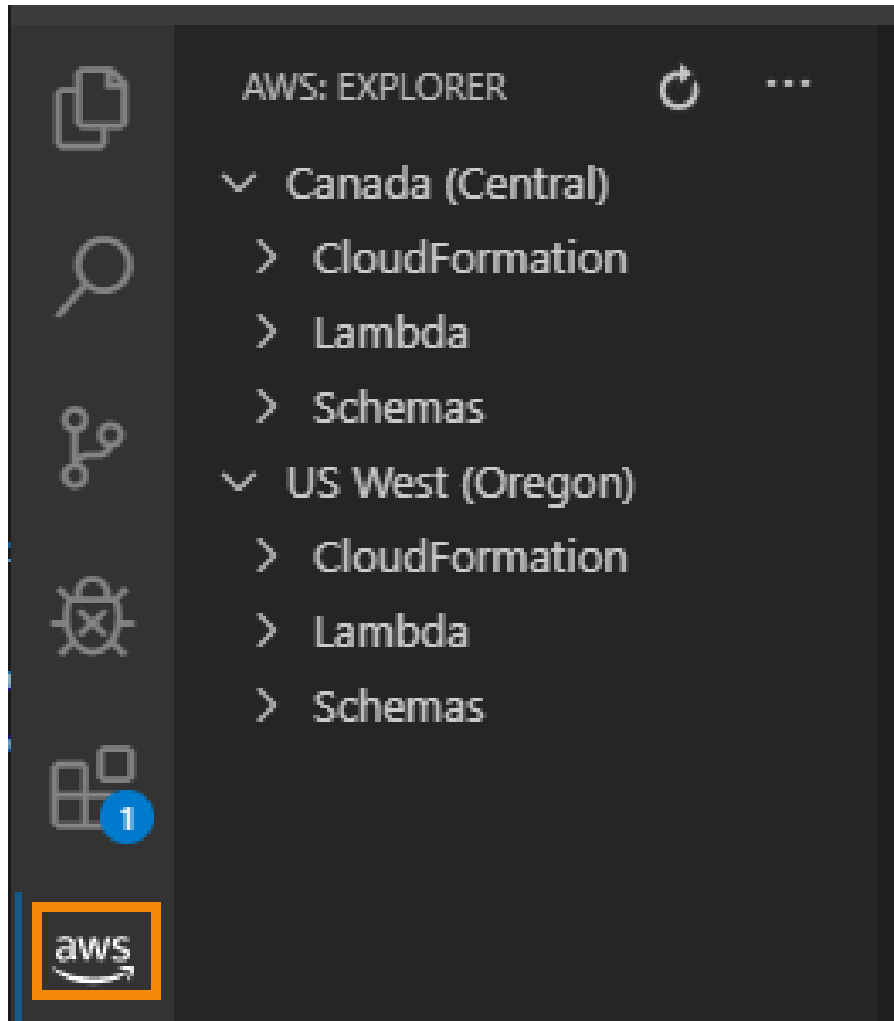


5. Abrir o AWS: Explorer Barra lateral, que chamamos de AWSExplorer, para verificar a conexão. Você verá uma lista de AWS Regiões (se você tiver [tornou todas as regiões visíveis \(p. 13\)](#) no AWSExplorer) ou uma mensagem para adicionar Regiões ao AWSExplorer.

Antes [adicionando Regiões \(p. 12\)](#) para o AWSExplorer, você verá o seguinte.



Depois de adicionar Regiões aoAWSExplorer, você verá algo parecido com o exemplo a seguir.



Usar váriasAWSContas da com o Toolkit

Você pode ter váriosAWSContas que você deseja acessar do Toolkit for VS Code. Várias contas podem ser úteis, por exemplo, para fornecer a desenvolvedores e administradores recursos separados para desenvolvimento e liberação/publicação.

Os conjuntos separados de credenciais da AWS são armazenados como perfis dentro do arquivo de configuração compartilhado da AWS ou do arquivo de credenciais compartilhado da AWS. Para escolher um conjunto diferente de credenciais, siga as etapas no procedimento anterior e escolha um perfil diferente.

AlterandoAWSRegiões da

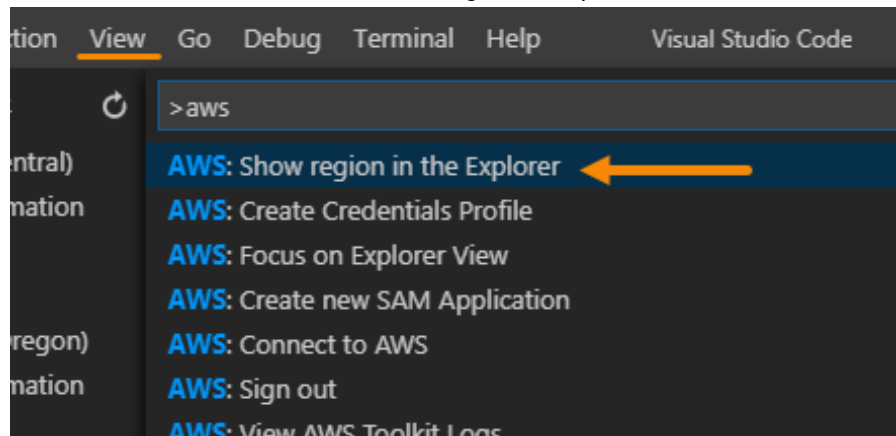
Quando você configura suas credenciais, o AWS Toolkit for Visual Studio Code seleciona automaticamente e mostra o padrãoAWSRegião para essas credenciais noAWSExplorer. Esse tópico descreve como alterar a lista de regiões exibidas no AWS Explorer.

Adicionar uma Região ao AWS Explorer

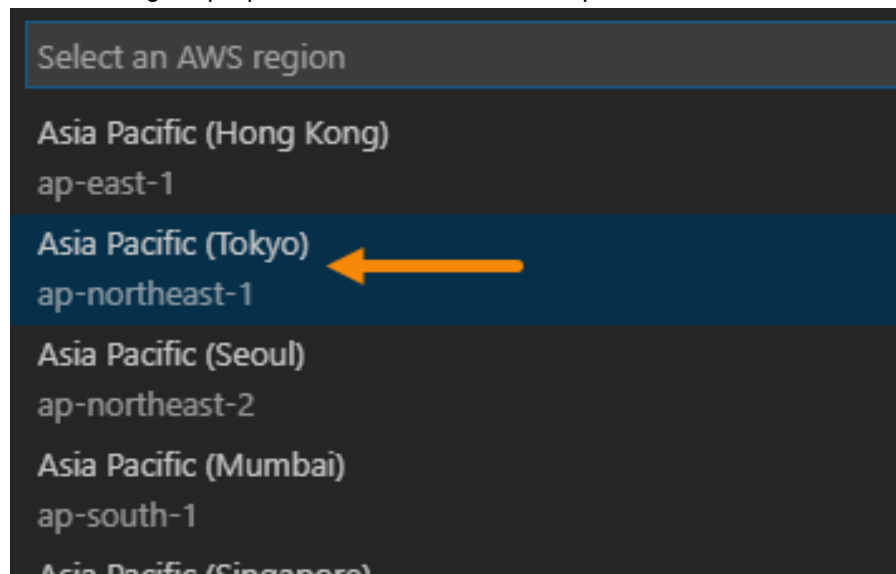
1. Para abrir a Command Palette (Paleta de comandos), na barra de menus, escolha View (Exibir), Command Palette (Paleta de comandos). Ou use as teclas de atalho a seguir:

- Windows e Linux: pressioneCtrl+Shift+P.
- macOS: pressioneShift+Command+P.

2. Search forAWS e escolhaAWS: Mostrar região no Explorador.



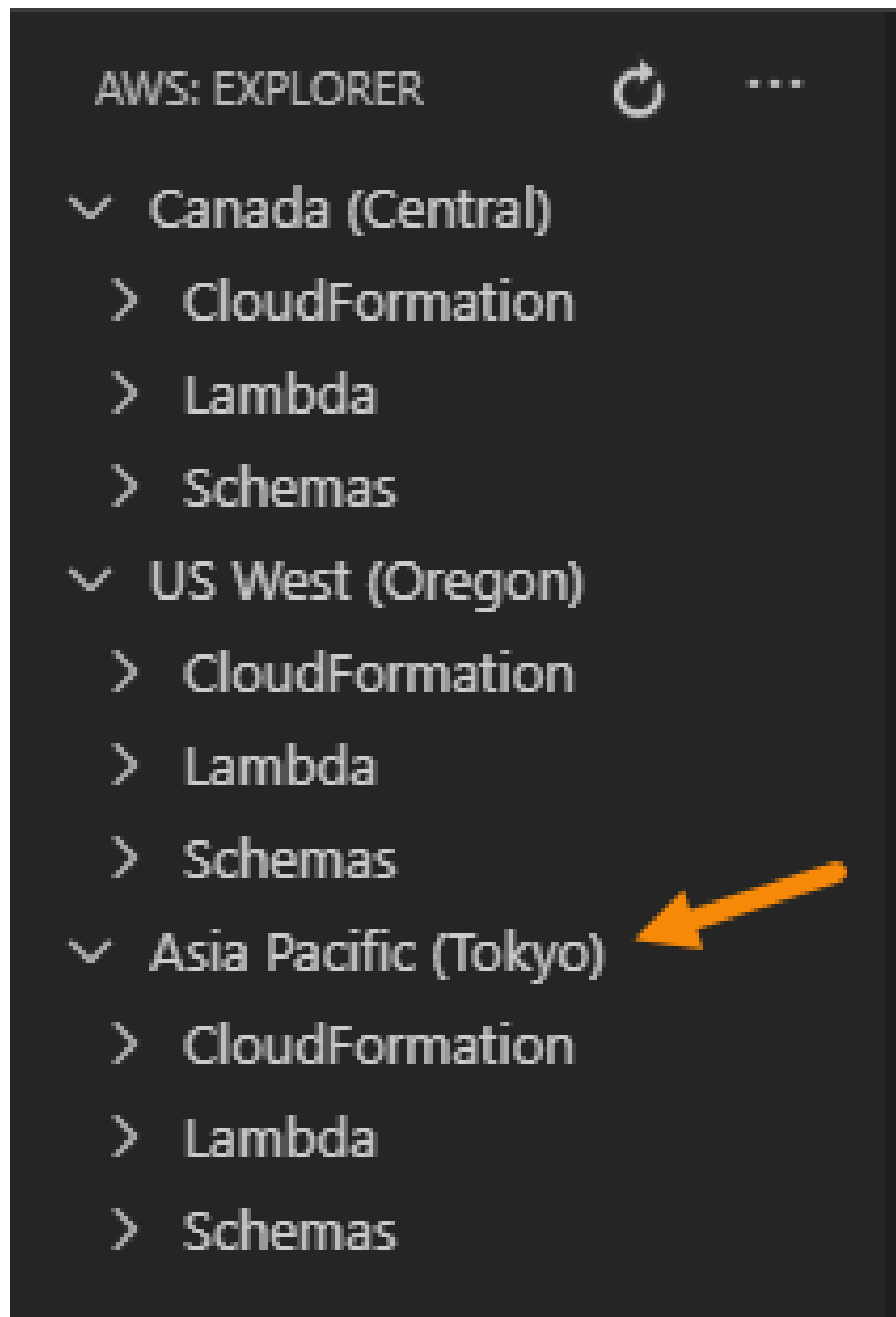
3. Escolha a região que pretende adicionar ao AWS Explorer.



Note

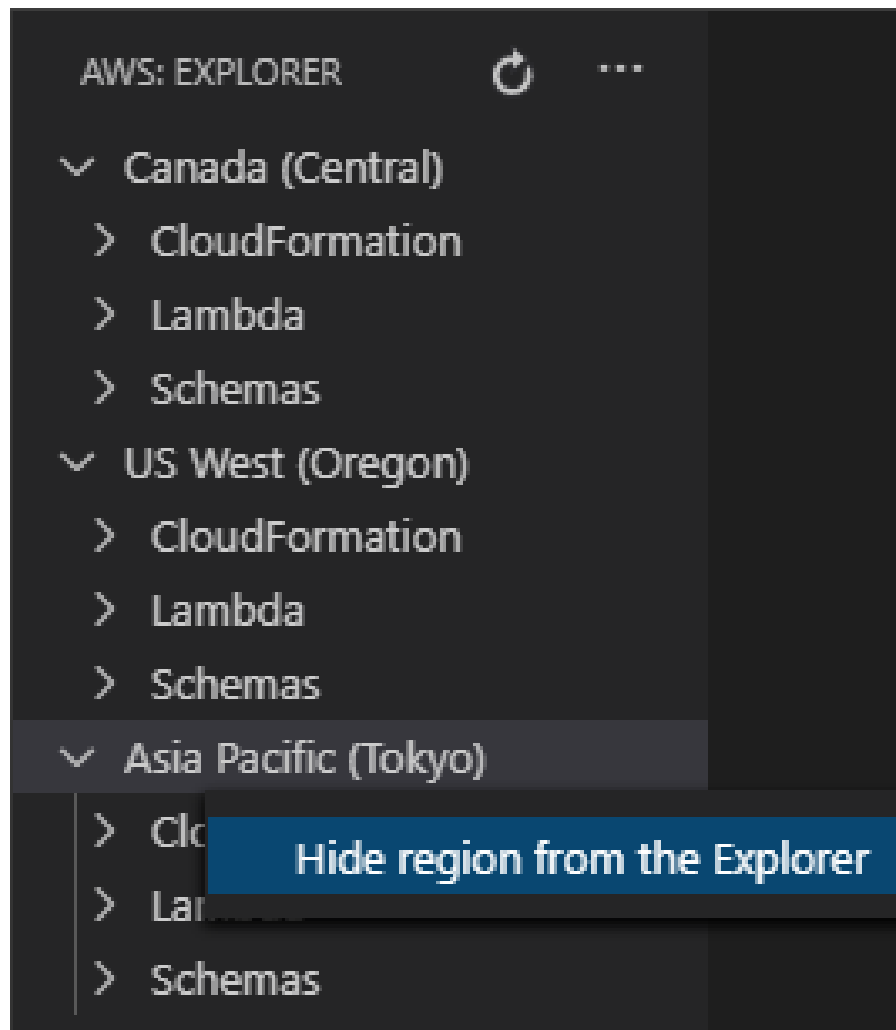
A lista contém apenas as regiões disponíveis para adicionar ao AWS Explorer. As regiões que você já adicionou não aparecem na lista.

4. Verifique se a região agora aparece no AWS Explorer.



Ocultar uma região do AWS Explorer

1. Escolha o ícone da AWS na Activity bar (Barra de atividades) para abrir o AWS Explorer.
2. Escolha uma das regiões na lista e abra seu menu de contexto.



3. Selecione Ocultar região do Explorer.

Configurar a cadeia de ferramentas

O AWS Toolkit for Visual Studio Code oferece suporte a várias linguagens que você pode usar para interagir com AWS. Este tópico descreve como configurar a cadeia de ferramentas para cada uma dessas linguagens.

Configurar uma cadeia de ferramentas para .NET Core

1. Verifique se você tem o AWS Toolkit for VS Code [instalado](#) (p. 3).
2. Instale a [extensão C#](#). Essa extensão permite ao VS Code ao depurar aplicativos .NET Core.
3. Abra um aplicativo AWS Serverless Application Model (AWS SAM) ou [crie um](#) (p. 64).
4. Abra a pasta que contém `template.yaml`.

Configurar uma cadeia de ferramentas para Node.js

Ao depurar uma função do Lambda TypeScript diretamente do código-fonte (a configuração do lançamento "target": "code"), o compilador TypeScript deve ser instalado globalmente. Para obter mais informações sobre como instalar isso, consulte [TypeScript](https://www.npmjs.com/) em <https://www.npmjs.com/>.

1. Verifique se você tem o AWSToolkit for VS Code [instalado](#) (p. 3).
2. Abra um aplicativo AWS SAM ou [crie um](#) (p. 64).
3. Abra a pasta que contém `template.yaml`.

Configurar uma cadeia de ferramentas para Python

1. Verifique se você tem o AWSToolkit for VS Code [instalado](#) (p. 3).
2. Instale a [extensão Python para Visual Studio Code](#). Essa extensão permite ao VS Code ao depurar aplicativos Python.
3. Abra um aplicativo AWS SAM ou [crie um](#) (p. 64).
4. Abra a pasta que contém `template.yaml`.
5. Abra um terminal na raiz do aplicativo 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 você tem o AWSToolkit for VS Code [instalado](#) (p. 3).
2. Instalar a [Extensão Java e Java 11](#). Esta extensão permite que o VS Code reconheça funções Java.
3. Instalar a [Extensão do depurador Java](#). Essa extensão permite que o VS Code depurar aplicativos Java.
4. Abra um aplicativo AWS SAM ou [crie um](#) (p. 64).
5. Abra a pasta que contém `template.yaml`.

Configurar uma cadeia de ferramentas para Go

1. Verifique se você tem o AWSToolkit for VS Code [instalado](#) (p. 3).
2. Ir 1.14 ou superior é necessário para depurar funções Go Lambda.
3. Instalar a [Extensão Go](#).

Note

A versão 0.25.0 ou superior é necessária para depurar tempos de execução do Go 1.15+.

4. Instale as ferramentas Go usando o [Paleta de comandos](#):
 - a. Na paleta de comando do, escolha `Go: Install/Update Tools`.

- b. No conjunto de caixas de seleção, selecione `liveops`.
5. Abra um aplicativo AWS SAM ou [crie um \(p. 64\)](#).
6. Abra a pasta que contém `template.yaml`.

Como usar a cadeia de ferramentas

Depois de configurar a cadeia de ferramentas, é possível usá-la para [executar ou depurar \(p. 66\)](#) o aplicativo AWS SAM.

Navegação no AWS Toolkit for Visual Studio Code

Este tópico descreve como navegar no AWS Toolkit for Visual Studio Code . Antes de ler este tópico, certifique-se de primeiro [instalar o toolkit \(p. 2\)](#).

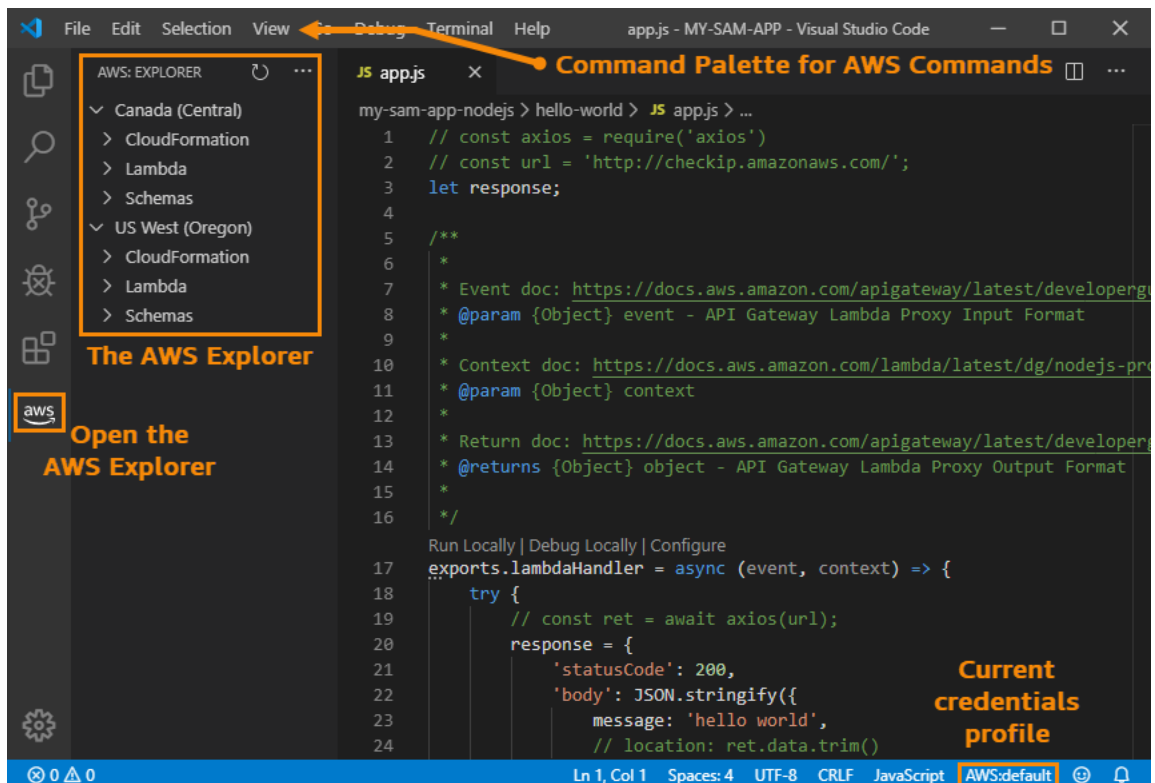
Para obter informações gerais sobre a interface de usuário (IU) do Visual Studio Code, consulte a seção [documentação da IU](#) Para o VS Code.

Componentes fundamentais da IU

A seguir estão os componentes básicos da IU do AWSToolkit for VS Code.

O AWS Explorer

- A imagem a seguir mostra os componentes básicos da IU do AWS Explorer. Consulte [Trabalhar com serviços da AWS no AWS Explorer \(p. 22\)](#) para obter mais detalhes sobre o que você pode fazer com eles.

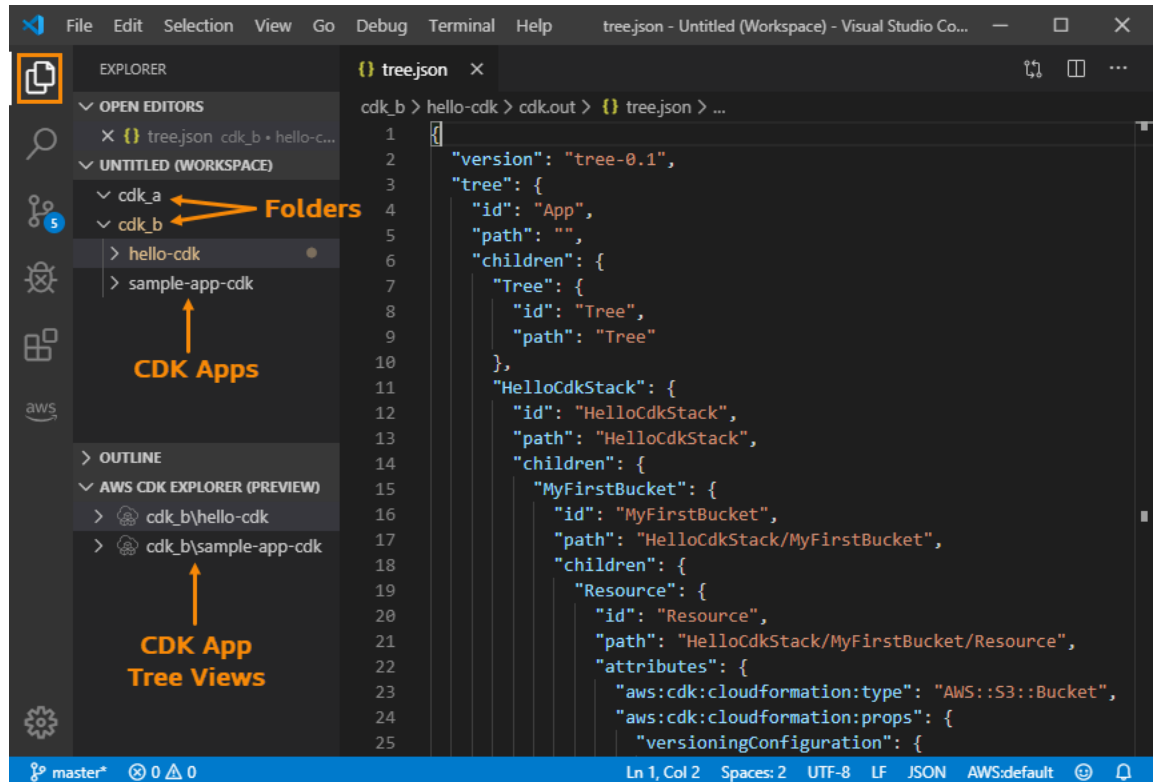


O AWS Explorer é mais formalmente conhecido como AWS: Barra lateral do Explorer.

O AWS CDK Explorer

- A imagem a seguir mostra os componentes básicos da IU do AWS CDK Explorer. Consulte [Trabalhar com o AWS CDK Explorer \(p. 60\)](#) para obter mais detalhes sobre o que você pode fazer com eles.

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



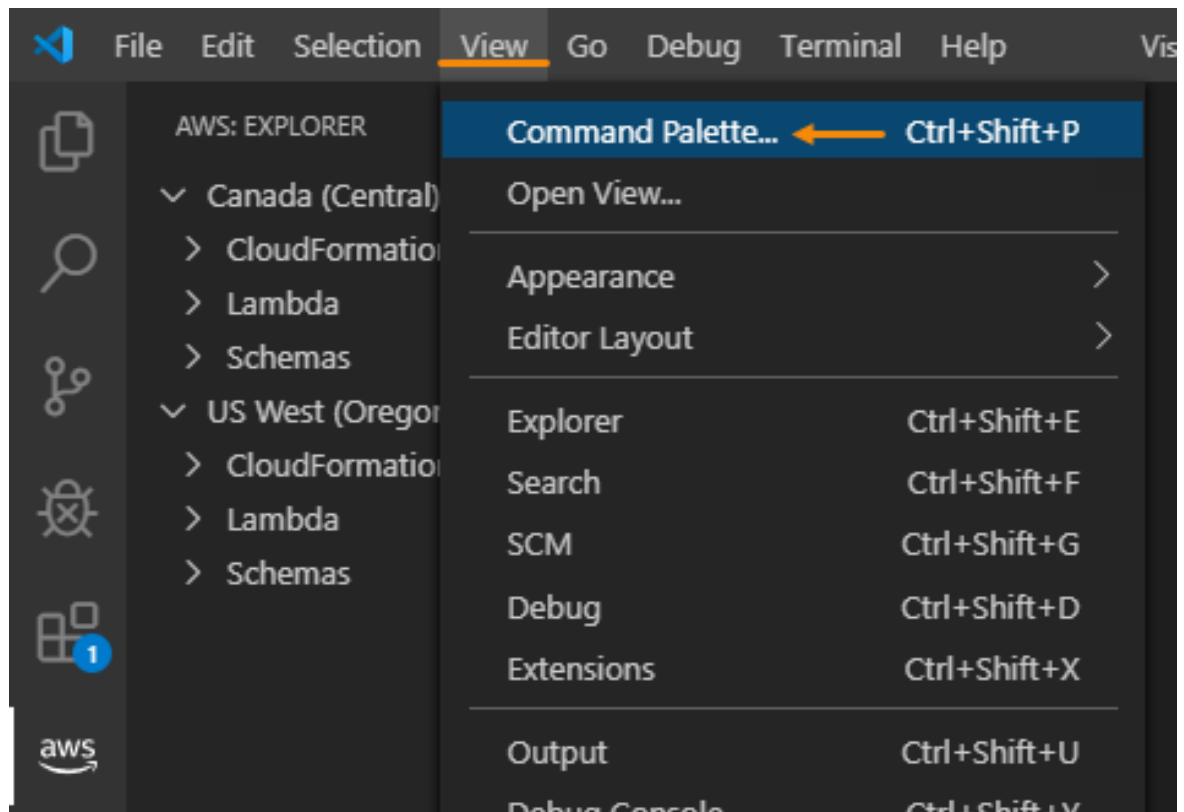
Locais de Comando

Você pode encontrar os comandos do Toolkit for VS Code em vários locais.

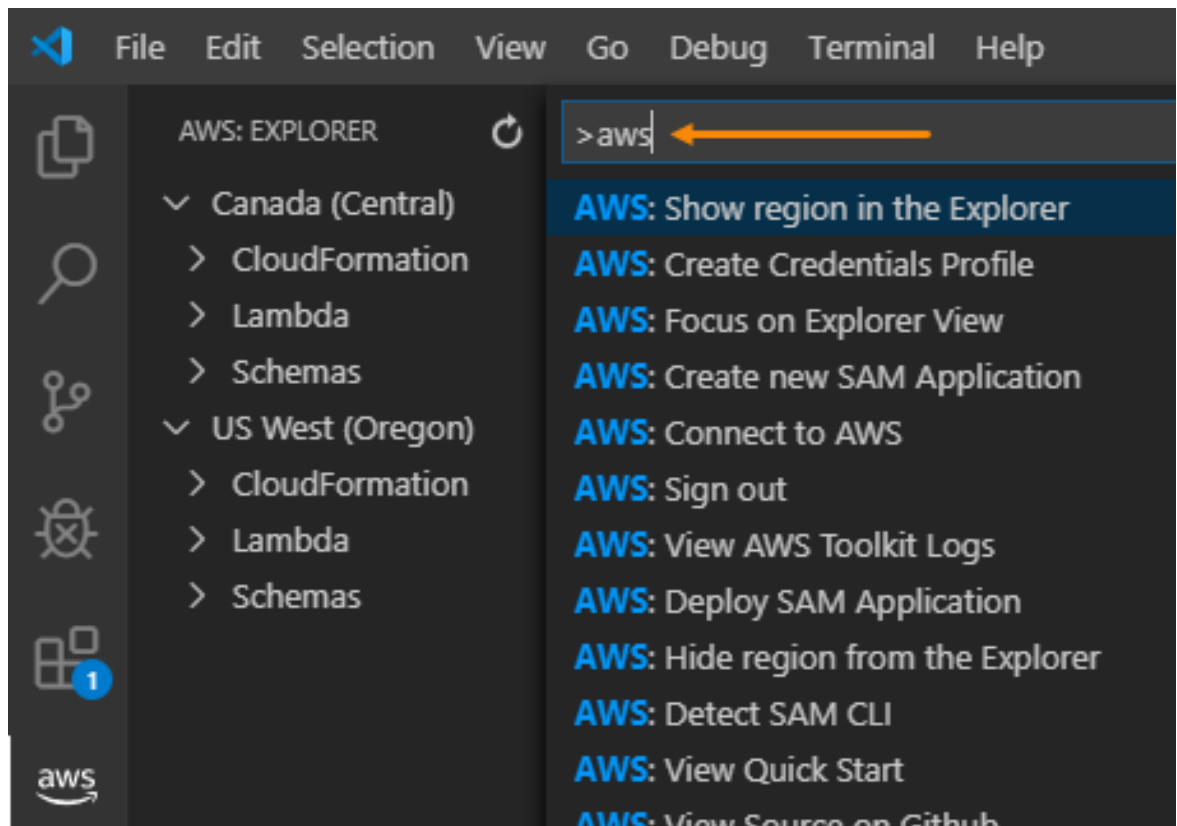
Paleta de comandos de código do Visual Studio

A Paleta de comandos de código do VS contém todos os comandos que você pode usar para trabalhar com o AWS Serviços da . Para visualizar esses comandos, abra a Paleta de comandos de código do VS no Exibir menu. Ou use os seguintes atalhos:

- Windows e Linux: pressione **Ctrl+Shift+P**.
- macOS: pressione **Shift+Command+P**.

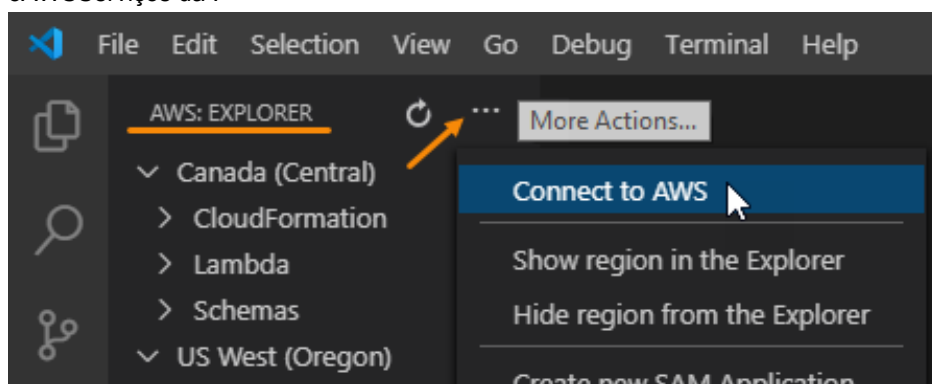


Em seguida, você pode filtrar por "AWS".



AWSExplorerMenu

O AWS Explorer tem um menu Mais Ações que contém os comandos mais comuns para trabalhar com os serviços da AWS.



Como trabalhar com os serviços da AWS

O AWS Toolkit for Visual Studio Code permite visualizar e trabalhar com vários serviços da AWS. Ele pressupõe que você já tenha [instalado e configurado](#) (p. 2) o Toolkit for VS Code em seu sistema.

Tópicos

- [Trabalhar com recursos experimentais](#) (p. 22)
- [Trabalhar com serviços da AWS no AWS Explorer](#) (p. 22)
- [Trabalho com o Amazon Elastic Container Service](#) (p. 59)
- [Trabalhar com o AWS CDK Explorer](#) (p. 60)
- [Trabalhar com aplicativos sem servidor](#) (p. 63)

Trabalhar com recursos experimentais

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

Warning

Como os recursos experimentais continuam sendo testados e atualizados, eles podem ter problemas de usabilidade. E as características experimentais podem ser removidas do AWS Toolkit for Visual Studio Code sem aviso prévio.

Você pode habilitar recursos experimentais para específicos AWS serviços no AWS Toolkit Seção do do Configurações painel no VS Code IDE.

1. Como editar o AWS configurações no VS Code, escolha Arquivo, Preferences (Preferências), Configurações.
2. No Configurações painel, expanda Extensões e escolha AWS Toolkit.
3. Sob AWS: Experiments, marque as caixas de seleção para os recursos experimentais que você deseja acessar antes do lançamento. Se você quiser desativar um recurso experimental, desmarque a caixa de seleção relevante.

Trabalhar com serviços da AWS no AWS Explorer

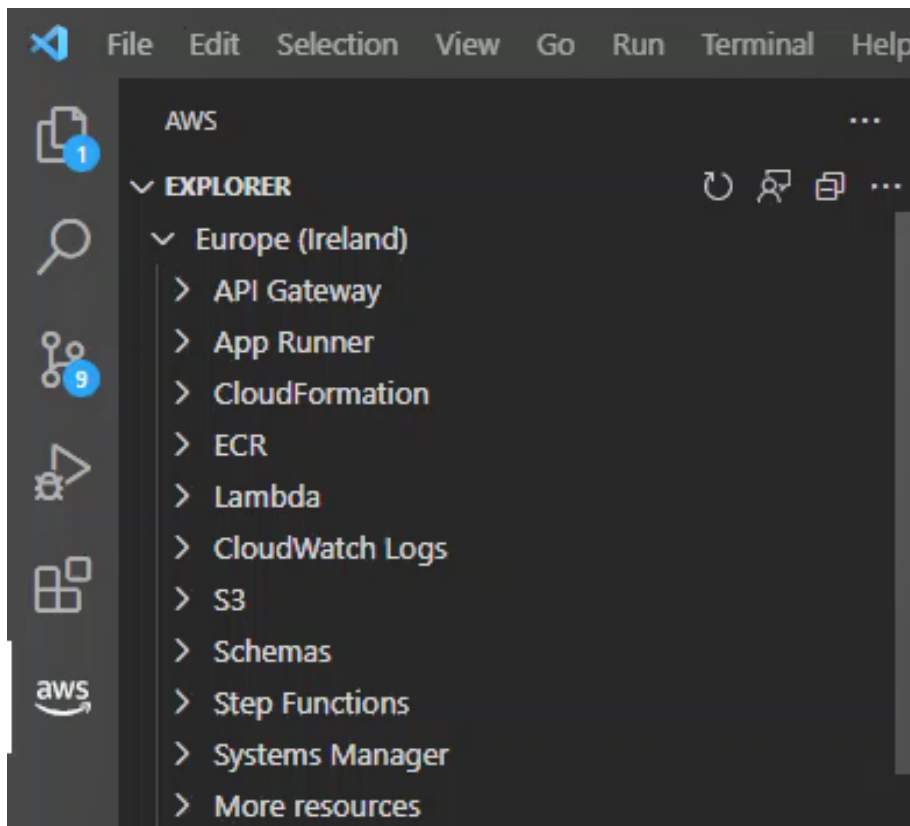
O AWS Explorer oferece uma visualização de alguns serviços da AWS com os quais pode trabalhar ao usar o AWS Toolkit for Visual Studio Code. Consulte uma visualização de alto nível do AWS Explorer no tópico de [navegação](#) (p. 18).

Esta seção fornece informações sobre como acessar e usar o AWS Explorer no VS Code. Pressupõem-se que você já tenha [instalado e configurado](#) (p. 2) o Toolkit for VS Code em seu sistema.

Alguns pontos importantes:

- Se o toolkit estiver instalado e configurado corretamente, você verá os itens no AWS Explorer. Para visualizar o AWS Explorer, escolha o AWS Ícone na Barra de atividades.

Por exemplo:



- Determinados recursos exigem certas permissões da AWS. Por exemplo, para ver o AWS Lambda Funções no AWS conta, as credenciais que você configurou em [Estabelecer credenciais \(p. 4\)](#) O deve incluir pelo menos permissões do Lambda somente 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 são imediatamente visíveis no AWS Explorer, você pode ir para [Mais recursos](#) e escolha entre centenas de recursos que podem ser adicionados à interface.

Por exemplo, é possível escolher `AWS::CodeArtifact::Repository` a partir da seleção de tipos de recursos disponíveis. Depois que esse tipo de recurso for adicionado ao [Mais recursos](#), você pode expandir a entrada para exibir uma lista de recursos que criam repositórios CodeArtifact diferentes com suas próprias propriedades e atributos. Além disso, você pode descrever as propriedades e atributos de recursos em modelos formatados em JSON, que podem ser salvos para criar novos recursos na AWS Cloud.

Tópicos

- [Trabalhar com o Amazon API Gateway \(p. 24\)](#)
- [Uso do AWS App Runner com a AWS Toolkit for Visual Studio Code \(p. 24\)](#)
- [Trabalho com AWS CloudFormation Pilhas do \(p. 30\)](#)
- [Trabalhando com o CloudWatch Logs usando o AWS Toolkit for Visual Studio Code \(p. 31\)](#)
- [Trabalhar com o Amazon EventBridge \(p. 33\)](#)
- [Trabalhar com funções do AWS Lambda \(p. 35\)](#)
- [Trabalhar com o Amazon S3 \(p. 41\)](#)
- [Trabalhar com documentos de Automação do Systems Manager \(p. 45\)](#)
- [Trabalho com AWS Step Functions \(p. 49\)](#)
- [Trabalhar com recursos do \(p. 56\)](#)

Trabalhar com o Amazon API Gateway

Você pode navegar e executar recursos remotos do API Gateway em seu conectadoAWSconta usando oAWS Toolkit for Visual Studio Code.

Note

Esse recurso não oferece suporte à depuração.

Para procurar e executar recursos remotos do API Gateway

1. NoAWSExplorer, escolhaAPI Gatewaypara expandir o menu. Os recursos do API Gateway remoto estão listados.
2. Encontre o recurso API Gateway que você deseja chamar, abra o menu de contexto (clique com o botão direito do mouse) e escolhaInvocar o emAWS.
3. No formulário de parâmetros, especifique os parâmetros de chamada.
4. Para executar o recurso API Gateway remoto, escolhaInvoke. Os resultados são detidos noSaída de código VSvisualizar.

Uso do AWS App Runner com a 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 para um aplicativo web escalável e seguro noAWSCloud. Ao usá-lo, você não precisa aprender novas tecnologias, decidir qual serviço de computação usar ou saber como provisionar e configurarAWSrecursos da AWS.

Você pode usarAWS App Runnerpara criar e gerenciar serviços com base em umimagem de origemoucódigo fonte. Se você usar uma imagem de origem, poderá escolher uma imagem de contêiner pública ou privada armazenada em um repositório de imagens. O App Runner oferece suporte aos seguintes fornecedores de repositório de imagens

- Amazon Elastic Container Registry (Amazon ECR): Armazena imagens privadas na sua conta da AWS.
- Amazon Elastic Container Registry Público (Amazon ECR Público): Armazena imagens legíveis publicamente.

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

Prerequisites

Para interagir com o App Runner usando o AWS Toolkit for Visual Studio Code O requer o seguinte:

- Uma conta da AWS
- Uma versão do AWS Toolkit for Visual Studio Code que apresentaAWS App Runner

Além desses requisitos principais, certifique-se de que todos os usuários relevantes do IAM tenham permissões para interagir com o serviço 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 repositório do GitHub. Você precisará dessas informações ao criar o serviço do App Runner.

Configurando permissões do IAM para o App Runner

A maneira mais fácil de conceder as permissões necessárias para o App Runner é anexar um existente AWS política gerenciada para o relevante AWS Identity and Access Management entidade (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 executem todas as ações do App Runner.
- `AWSAppRunnerReadOnlyAccess`: Permitir 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 de serviço, você deverá criar a seguinte função de acesso para o serviço 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 o VS Code Paleta de comandos do.

Note

OA `AWSServiceRoleForApprunner` função vinculada a serviços permite AWS App Runner Para concluir as seguintes tarefas:

- Faça push de logs para grupos de logs do Amazon CloudWatch 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 a função vinculada a 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 obter mais informações, consulte [Gerenciamento de identidade e acesso para o App Runner](#) no AWS App Runner Guia do desenvolvedor.

Obtendo fontes de serviço para o App Runner

Você pode usar AWS App Runner para implantar serviços a partir de uma imagem de origem 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 de um público ou privado AWS registro de imagens.

- 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 Galeria Pública do Amazon ECR em <https://gallery.ecr.aws/>.

Note

Você também pode obter o URI para um repositório privado do Amazon ECR diretamente de AWS Explorer No Toolkit for VS Code:

- AbertoAWSExplorere expanda oECRnó para exibir a lista de repositórios para issoAWSRegião :
- Clique com o botão direito em um repositórioURI do repositório dePara copiar o link para a área de transferência.

Você especifica o URI para o repositório de imagens ao configurar sua instância de serviço com o VS CodePaleta de comandos do

Para obter mais informações, consulte[Serviço App Runner baseado em uma imagem de origem](#)noAWS App RunnerGuia do desenvolvedor.

Source code

Para que seu código-fonte seja implantado em umAWS App Runner, esse código deve ser armazenado em um repositório Git mantido por um provedor de repositório suportado. O App Runner suporta um provedor de repositório de código fonte: [GitHub](#).

Para obter informações sobre a configuração de um repositório do GitHub, consulte o[Documentação de conceitos básicos](#)no GitHub

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

Important

Para criar conexões do GitHub, você deve usar o console do App Runner (<https://console.aws.amazon.com/apprunner>) para criar uma conexão que vincule o GitHub aoAWS. Você pode selecionar as conexões que estão disponíveis noConexões GitHubpágina ao configurar sua instância de serviço com o VS CodesPaleta de comandos do. Para obter mais informações, consulte[Gerenciar conexões App Runner](#)noAWS App RunnerGuia do desenvolvedor.

A instância de serviço App Runner fornece um tempo de execução gerenciado que permite que seu código seja compilado e executado.AWS App RunnerNo momento, o oferece suporte aos seguintes tempos de execução:

- Runtime gerenciado do Python
- Runtime de Node.js gerenciado

Como parte da configuração do serviço, você fornece informações sobre como o serviço App Runner constrói e inicia seu serviço. Você pode inserir essas informações usando oPaleta de comandos doou especifique um formato YAMLAquivo de configuração App Runner. Os valores neste arquivo instruem o App Runner a criar e iniciar seu serviço e fornecer contexto de tempo de execução. Isso inclui configurações de rede e variáveis de ambiente relevantes. O arquivo de configuração é denominadoapprunner.yaml. Ele é adicionado automaticamente ao diretório raiz do repositório do aplicativo.

Pricing

Você é cobrado pelos recursos de computação e memória que seu aplicativo usa. Além disso, se você automatizar suas implantações, também pagará uma taxa mensal definida para cada aplicativo que cobre todas as implantações automatizadas para esse mês. Se você optar por implantar a partir do código-fonte, você também paga uma taxa de compilação pelo tempo necessário para o App Runner para criar um contêiner a partir do código-fonte.

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

Tópicos

- [Criar serviços do App Runner \(p. 27\)](#)
- [Gerenciar serviços App Runner \(p. 29\)](#)

Criar serviços do App Runner

Você pode criar um serviço App Runner no Toolkit for VS Code usando o **AWSExplorer** e **VS Codes** Paleta de comandos do. Depois de optar por criar um serviço em um específico **AWS Região**, etapas numeradas fornecidas pelo **Paleta de comandos** do **Orienta** você durante o processo de configurar a instância de serviço na qual o aplicativo é executado.

Antes de criar um serviço de App Runner, verifique se você concluiu o **Pré-requisitos do (p. 24)**. Isso inclui fornecer as permissões relevantes do IAM e confirmar o repositório de origem específico que você deseja implantar.

Para criar um serviço App Runner

1. Aberto **AWSExplorer**, se ainda não estiver aberto.
2. Clique com botão direito **App Runner** e escolha **Criar serviço**.

O **Paleta de comandos** do **Exibe**.
3. para o **Selecione** um tipo de local de código-fonte, escolha **ECR** ou **Repositório**.

Se escolher **ECR**, você especifica uma imagem de contêiner em um repositório mantido pelo Amazon Elastic Container Registry. Se escolher **Repositório**, você especifica um repositório de código-fonte mantido por um provedor de repositório compatível. Atualmente, o App Runner suporta **GitHub** Como provedor de repositório de códigos-fonte.

Implantação do ECR

1. para o **Selecione** ou insira um repositório de imagens, escolha ou insira o URL do repositório de imagens mantido pelo registro privado do Amazon ECR ou pela Galeria Pública do Amazon ECR.

Note

Se você especificar um repositório da Galeria Pública do Amazon ECR, verifique se as implantações automáticas estão desativadas porque o App Runner não oferece suporte a implantações automáticas para uma imagem em um repositório do ECR Public.

Implantações automáticas são desativadas por padrão, e isso é indicado quando o ícone no **Paleta de comandos** do cabeçalho apresenta uma linha diagonal através dele. Se você optar por ativar implantações automáticas, uma mensagem informará que essa opção pode incorrer em custos adicionais.

2. Se o **Paleta de comandos** do **Relatórios de etapas** que **Nenhuma** etiqueta encontrada, você precisa 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, você precisa da função de acesso **ECR, App Runner accessRole**, o que permite ao App Runner acessar imagens do Amazon Elastic Container Registry (Amazon ECR) na sua conta. Escolha o ícone "+" no **Paleta de comandos** do cabeçalho para criar automaticamente essa função. (Uma função de acesso não será necessária se sua imagem estiver armazenada no Amazon ECR Public, onde as imagens estão disponíveis publicamente.)
4. para o **Port**, insira a porta IP usada pelo serviço (Porta 8000, por exemplo).
5. para o **Configurar variáveis de ambiente**, você pode especificar um arquivo que contenha variáveis de ambiente usadas para personalizar o comportamento na instância de serviço. Ou você pode ignorar esta etapa.
6. para o **Nome** e seu serviço, insira um nome exclusivo sem espaços e pressione **Digite**.

7. para oSelecionar configuração da instância, escolha uma combinação de unidades de CPU e memória em GB para sua instância de serviço.

Quando o serviço está sendo criado, o status muda deCriarparaRunning.

8. Depois que o serviço começar a ser executado, clique com o botão direito e escolhaCopiar URL do serviço.
9. Para acessar seu aplicativo implantado, cole o URL copiado na barra de endereços do navegador da Web.

Implantar a partir de um repositório remoto

1. para oSelecione uma conexão, escolha uma conexão que vincule o GitHub aoAWS. As conexões que estão disponíveis para seleção estão listadas noConexões GitHubpágina no console do App Runner.
2. para oSelecione um repositório remoto do GitHub, escolha ou insira um 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 oSelecione uma ramificação, escolha qual ramo Git do seu código-fonte você deseja implantar.
4. para oEscolha a fonte de configuração, especifique como você deseja definir sua configuração de tempo de execução.

Se escolherUsar arquivo de configuração, sua instância de serviço é definida por configurações definidas peloapprunner.yamlArquivo de configuração. Este arquivo está no diretório raiz do repositório do aplicativo.

Se escolherConfigure todas as configurações aqui, use oPaleta de comandosPara especificar o seguinte:

- Tempo de execução: SelecionePython 3ouNodejs 12.
 - Comando compilação: Digite o comando para criar seu aplicativo no ambiente de tempo de execução da instância de serviço.
 - Comando para iniciar: Insira o comando para iniciar seu aplicativo no ambiente de tempo de execução da instância de serviço.
5. para oPort, insira a porta IP usada pelo serviço (Porta8000, por exemplo).
 6. para oConfigurar variáveis de ambiente, você pode especificar um arquivo que contenha variáveis de ambiente usadas para personalizar o comportamento na instância de serviço. Ou você pode ignorar esta etapa.
 7. para oNomeie seu serviço, insira um nome exclusivo sem espaços e pressioneDigite.
 8. para oSelecionar configuração da instância, escolha uma combinação de unidades de CPU e memória em GB para sua instância de serviço.

Quando o serviço está sendo criado, o status muda deCriarparaRunning.

9. Depois que o serviço começar a ser executado, clique com o botão direito e escolhaCopiar URL do serviço.
10. Para acessar seu aplicativo implantado, cole o URL copiado na barra de endereços do navegador da Web.

Note

Se a tentativa de criar um serviço App Runner falhar, o serviço mostrará um status deFalha na criaçãoemAWS Explorer. Para obter dicas de solução de problemas, consulte[Quando a criação do serviço falha](#)noGuia do desenvolvedor App Runner.

Gerenciar serviços App Runner

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

- [Pausar e retomar serviços do App Runner \(p. 29\)](#)
- [Implantar serviços do App Runner \(p. 29\)](#)
- [Visualizando fluxos de registros para o App Runner \(p. 30\)](#)
- [Excluir serviços do App Runner \(p. 30\)](#)

Pausar e retomar serviços do App Runner

Se você precisar desativar o aplicativo da Web temporariamente e impedir que o código seja executado, você pode pausar oAWSServiço App Runner. O App Runner reduz a capacidade computacional do serviço para zero. Quando você estiver pronto para executar seu aplicativo novamente, retome o serviço App Runner. O App Runner provisiona nova capacidade computacional, implanta seu aplicativo nele e executa o aplicativo.

Important

Você é cobrado pelo App Runner somente quando ele estiver em execução. Portanto, você pode pausar e retomar seu aplicativo conforme necessário para gerenciar custos. Isso é particularmente útil em cenários de desenvolvimento e teste.

Para pausar o serviço App Runner

1. AbertoAWSExplorer, se ainda não estiver aberto.
2. AmpliarApp RunnerComo exibir a lista de serviços.
3. Clique com o botão direito no serviço e escolhaPause.
4. Na caixa de diálogo exibida, escolhaConfirmar.

Enquanto o serviço está pausando, o status do serviço muda deRunningparaPausinge depois paraPaused.

Para retomar seu serviço App Runner

1. AbertoAWSExplorer, se ainda não estiver aberto.
2. AmpliarApp RunnerComo exibir a lista de serviços.
3. Clique com o botão direito no serviço e escolhaRetomar.

Enquanto o serviço está sendo retomado, o status do serviço muda deRetomandoparaRunning.

Implantar serviços do App Runner

Se você escolher a opção de implantação manual para seu serviço, precisará iniciar explicitamente cada implantação em seu serviço.

1. AbertoAWSExplorer, se ainda não estiver aberto.
2. AmpliarApp RunnerComo exibir a lista de serviços.
3. Clique com o botão direito no serviço e escolhaIniciar a implantação.
4. Enquanto seu aplicativo está sendo implantado, o status do serviço muda deImplantar oparaRunning.
5. Para confirmar se o aplicativo foi implantado com sucesso, clique com o botão direito do mouse no mesmo serviço e escolhaCopiar URL do serviço.

6. Para acessar seu aplicativo da Web implantado, cole o URL copiado na barra de endereços do navegador da Web.

Visualizando fluxos de registros para o App Runner

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

1. AmpliarApp RunnerComo exibir a lista de instâncias de serviço.
2. Expanda uma instância de serviço específica para visualizar a lista de grupos de log. (Um grupo de logs é um grupo de fluxos de log que compartilham as mesmas configurações de retenção, monitoramento e controle de acesso.)
3. Clique com o botão direito em um grupo de registrosExibir fluxos de log.
4. DoPaleta de comandos do, escolha um fluxo de logs do grupo.

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

Excluir serviços do App Runner

Important

Se você excluir seu serviço App Runner, ele será removido permanentemente e seus dados armazenados serão excluídos. Se você precisar recriar o serviço, o App Runner precisará buscar sua fonte novamente e construí-la se for um repositório de código. Seu aplicativo da Web obtém um novo domínio do App Runner.

1. AbertoAWSExplorer, se ainda não estiver aberto.
2. AmpliarApp RunnerComo exibir a lista de serviços.
3. Clique com o botão direito no serviço e escolhaExcluir serviço.
4. NoPaleta de comandos do, insiradeletee pressioneDigitePara confirmar.

O serviço excluído exibe oExcluirstatus e, em seguida, o serviço desaparece da lista.

Trabalho comAWS CloudFormationPilhas do

O AWS Toolkit for Visual Studio Code oferece suporte para pilhas do [AWS CloudFormation](#). Usando o Toolkit for VS Code Code, você pode executar determinadas tarefas com oAWS CloudFormationpilhas do tipo, como excluí-las.

Tópicos

- [Exclusão de umaAWS CloudFormationPilha \(p. 30\)](#)

Exclusão de umaAWS CloudFormationPilha

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

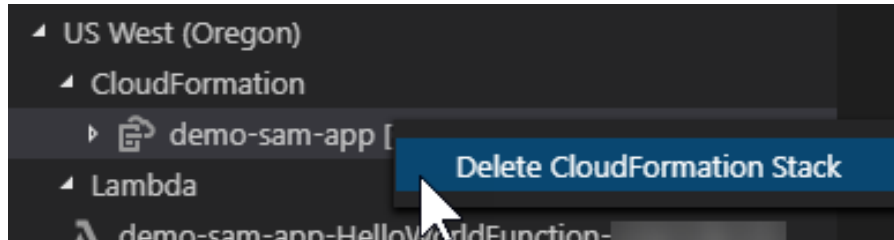
Prerequisites

- Verifique se o sistema atende aos pré-requisitos especificados em[Instalação do Toolkit for VS Code \(p. 2\)](#).
- Certifique-se de que as credenciais que você configurou em [Estabelecer credenciais \(p. 4\)](#) incluem acesso adequado de leitura/gravação ao serviço do AWS CloudFormation. Se noAWSExplorer,

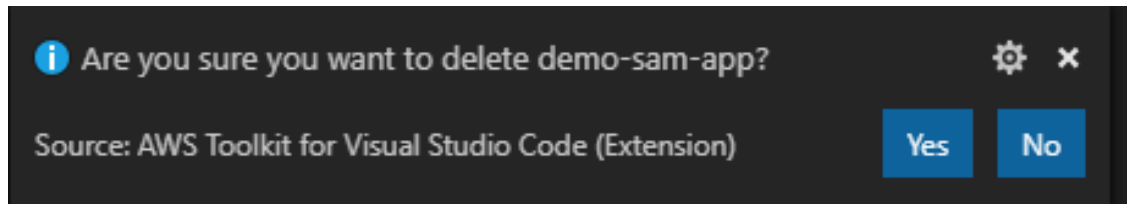
em, emCloudFormationVocê verá uma mensagem semelhante a “Erro ao carregar os recursos do CloudFormation”, verifique as permissões anexadas a essas credenciais. As alterações feitas nas permissões levarão alguns minutos para afetar oAWSExplorerno VS Code.

Delete 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.

Trabalhando com o CloudWatch Logs usando o AWS Toolkit for Visual Studio Code

O Amazon CloudWatch Logs permite centralizar os logs de todos os sistemas, aplicativos e AWS Serviços que você usa em um único serviço altamente escalável. Você pode facilmente visualizá-los, 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 Guia 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 AWS conta.

Tópicos

- [Exibindo grupos de log e fluxos de log do CloudWatch usando o AWS Toolkit for Visual Studio Code \(p. 31\)](#)
- [Trabalhando com eventos de log do CloudWatch em fluxos de log usando o AWS Toolkit for Visual Studio Code \(p. 32\)](#)

Exibindo grupos de log e fluxos de log 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 log 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 de logs e fluxos de log](#) no Guia do usuário do Amazon CloudWatch.

Tópicos

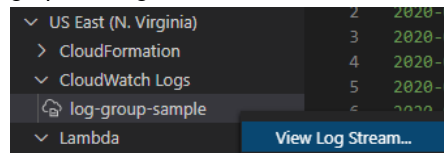
- [Visualizar grupos e fluxos de logs com o CloudWatch Logs](#) (p. 32)

Visualizar grupos e fluxos de logs com o CloudWatch Logs

1. No VS Code, escolha **Exibir, Explorador** **Abri** **AWSE** **Explorador**.
2. Clique no ícone da barra de ferramentas **CloudWatch Logs** Para expandir a lista de grupos de logs.

Os grupos de logs para a atual AWS região é exibida sob a guia **CloudWatch Logs**.

3. Para exibir os fluxos de log em um grupo de logs, clique com o botão direito do mouse no nome do grupo de registros e escolha **Exibir fluxo de logs**.



4. Na **Paleta de comandos**, selecione um fluxo de log do grupo a ser exibido.

Note

O **Paleta de comandos** exibe um carimbo de data/hora para o último evento em cada stream.

O **Log Stream (Fluxo de logs) Editor** (p. 32) inicia para exibir os eventos de log do fluxo.

Trabalhando com eventos de log do CloudWatch em fluxos de log usando o AWS Toolkit for Visual Studio Code

Depois de abrir o **Stream de log editor**, você pode acessar os eventos de log em cada fluxo. Eventos de log são registros de atividade registrada pelo aplicativo ou recurso que está sendo monitorado.

Tópicos

- [Exibindo e copiando informações de fluxo de log](#) (p. 32)
- [Salve o conteúdo do editor de fluxo de log em um arquivo local](#) (p. 33)

Exibindo e copiando informações de fluxo de log

Quando você abre um fluxo de log, o **Log Stream (Fluxo de logs) editor** exibe a sequência desse fluxo de eventos de log.

1. Para encontrar um fluxo de logs para visualizar, abra o **Log Stream (Fluxo de logs) Editor** (consulte [Exibindo grupos de logs e fluxos de log do CloudWatch](#) (p. 31)).

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

2. Você pode visualizar e copiar informações sobre os eventos do fluxo usando as seguintes opções:

- Exibir eventos por hora: Exibir os eventos de log mais recentes e antigos escolhendo Carregar eventos mais recentes ou Carregar eventos mais antigos.

Note

O Log Stream (Fluxo de logs) o editor carrega inicialmente um lote das mais recentes 10.000 linhas de eventos de log ou 1 MB de dados de log (o que for menor). Se escolher Carregar eventos mais recentes, o editor exibe eventos que foram registrados após o último lote ter sido carregado. Se escolher Carregar eventos mais antigos, o editor exibe um lote de eventos que ocorreram antes dos exibidos no momento.

- Copiar eventos de log: Selecione os eventos para copiar, clique com o botão direito do mouse e selecione Copiar no menu.
- Copie o nome do fluxo de log: Clique com o botão direito do mouse Log Stream (Fluxo de logs) editor e escolha Copiar nome do fluxo de logs.

Note

Você também pode usar a Paleta de comandos do Para executar o AWS: Copiar nome do fluxo de logs.

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

Você pode baixar o conteúdo do editor de fluxo de log do CloudWatch para um logArquivo na máquina local.

Note

Com essa opção, você salva para arquivar apenas os eventos de log exibidos no momento no editor de fluxo de log. Por exemplo, se o tamanho total de um fluxo de log for 5MB e apenas 2 MB forem carregados no editor, seu arquivo salvo também conterá apenas 2 MB de dados de log. Para exibir mais dados a serem salvos, escolha Carregar eventos mais recentes ou Carregar eventos mais antigos no editor.

1. Para encontrar um fluxo de logs para copiar, abra o Fluxos de log Editor de do (consulte [Exibindo grupos de logs e fluxos de log do CloudWatch \(p. 31\)](#)).
2. Selecione o Save (Salvar) ícone ao lado da guia exibindo o nome do fluxo de log.

Note

Você também pode usar a Paleta de comandos do Para executar o AWS: Salvar conteúdo do fluxo de log atual.

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

Trabalhar com o Amazon EventBridge

O AWS Toolkit for Visual Studio Code (VS Code) fornece suporte para [Amazon EventBridge](#). Usando o Toolkit for VS Code, você pode trabalhar com determinados aspectos do EventBridge, como esquemas.

Tópicos

- [Trabalhar com esquemas do Amazon EventBridge \(p. 34\)](#)

Trabalhar com esquemas do Amazon EventBridge

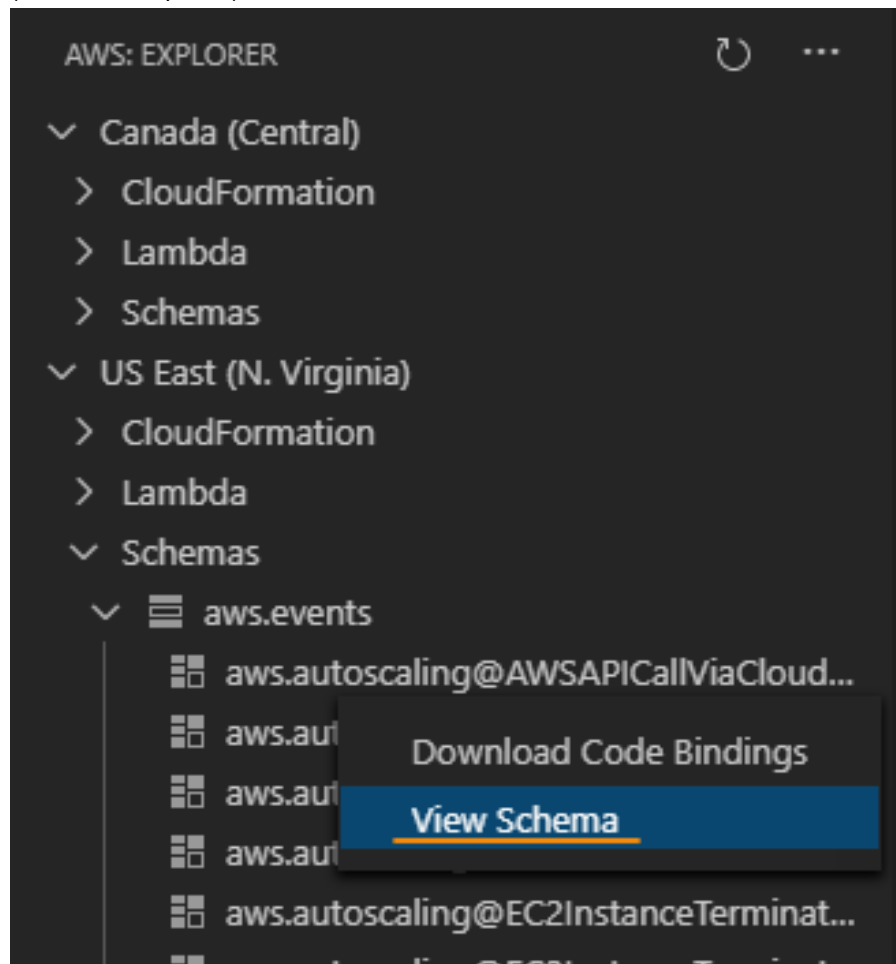
Você pode usar o AWS Toolkit for Visual Studio Code (VS Code) para realizar várias operações em [Esquemas do Amazon EventBridge](#).

Prerequisites

- Verifique se o sistema atende aos pré-requisitos especificados em [Instalação do Toolkit for VS Code](#) (p. 2).
- O esquema EventBridge com o qual deseja trabalhar deve estar disponível no AWS conta. Se não estiver, crie um ou carregue-o. Consulte [Amazon EventBridge Schemas](#) no [Guia do usuário do Amazon EventBridge](#).
- Abra a barra lateral do [AWS Explorer](#) (p. 18).

Visualizar um esquema disponível

1. No AWS Explorer, expanda Schemas (Esquemas).
2. Expanda o nome do registro que contém o esquema que deseja visualizar. Por exemplo, muitos dos esquemas que a AWS oferece estão no registro aws.events.
3. Para visualizar um esquema no editor, abra o menu de contexto do esquema e escolha View Schema (Visualizar esquema).

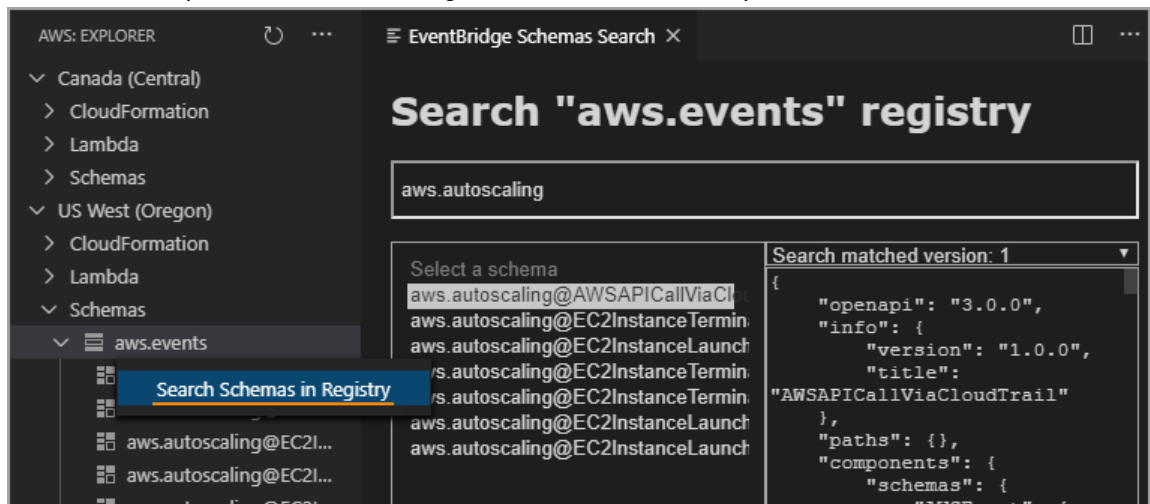


Localizar um esquema disponível

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). NoPesquisa de esquemas EventBridgeCaixa de diálogo, comece a digitar 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

Trabalhar com funções do AWS Lambda

O AWS Toolkit for Visual Studio Code oferece suporte para funções do [AWS Lambda](#). Usando o Toolkit for VS Code Code, é possível criar código para as funções do Lambda que fazem parte de [aplicativos sem servidor](#). Além disso, você pode invocar funções do Lambda localmente ou naAWS.

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

Tópicos

- [Interação com funções remotas do Lambda \(p. 36\)](#)

Interação com funções remotas do Lambda

Usando o Toolkit for VS Code, você pode interagir com [AWS Lambda](#) de várias maneiras, conforme descrito posteriormente neste tópico.

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

Note

Se você já criou funções do Lambda usando o AWS Management Console ou de alguma outra forma, você pode invocá-los do toolkit. Para criar uma nova função (usando o VS Code) que você pode implantar no AWS Lambda, é necessário primeiro [Criar um aplicativo sem servidor \(p. 64\)](#).

Tópicos

- [Prerequisites \(p. 36\)](#)
- [Invocar uma função do Lambda \(p. 36\)](#)
- [Excluir uma função do Lambda \(p. 38\)](#)
- [Importar uma função do Lambda \(p. 39\)](#)
- [Carregar uma função do Lambda \(p. 40\)](#)

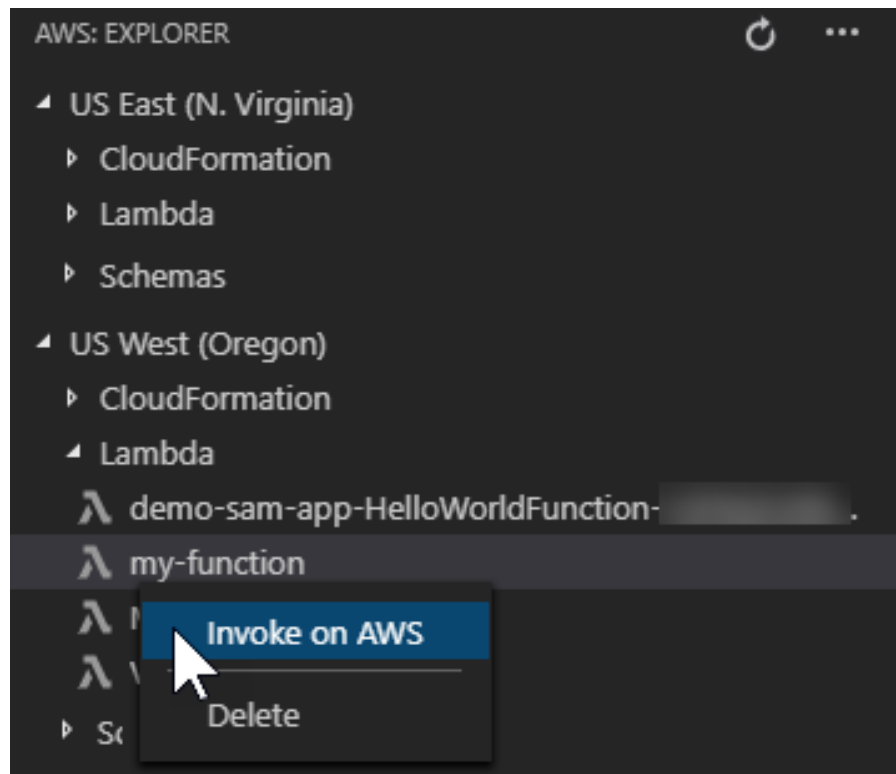
Prerequisites

- Verifique se o sistema atende aos pré-requisitos especificados em [Instalação do Toolkit for VS Code \(p. 2\)](#).
- Certifique-se de que as credenciais que você configurou em [Estabelecer credenciais \(p. 4\)](#) incluem acesso adequado de leitura/gravação ao serviço do AWS Lambda. Se no [AWS Explorer](#), em [Lambda](#) Você verá uma mensagem semelhante a “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.

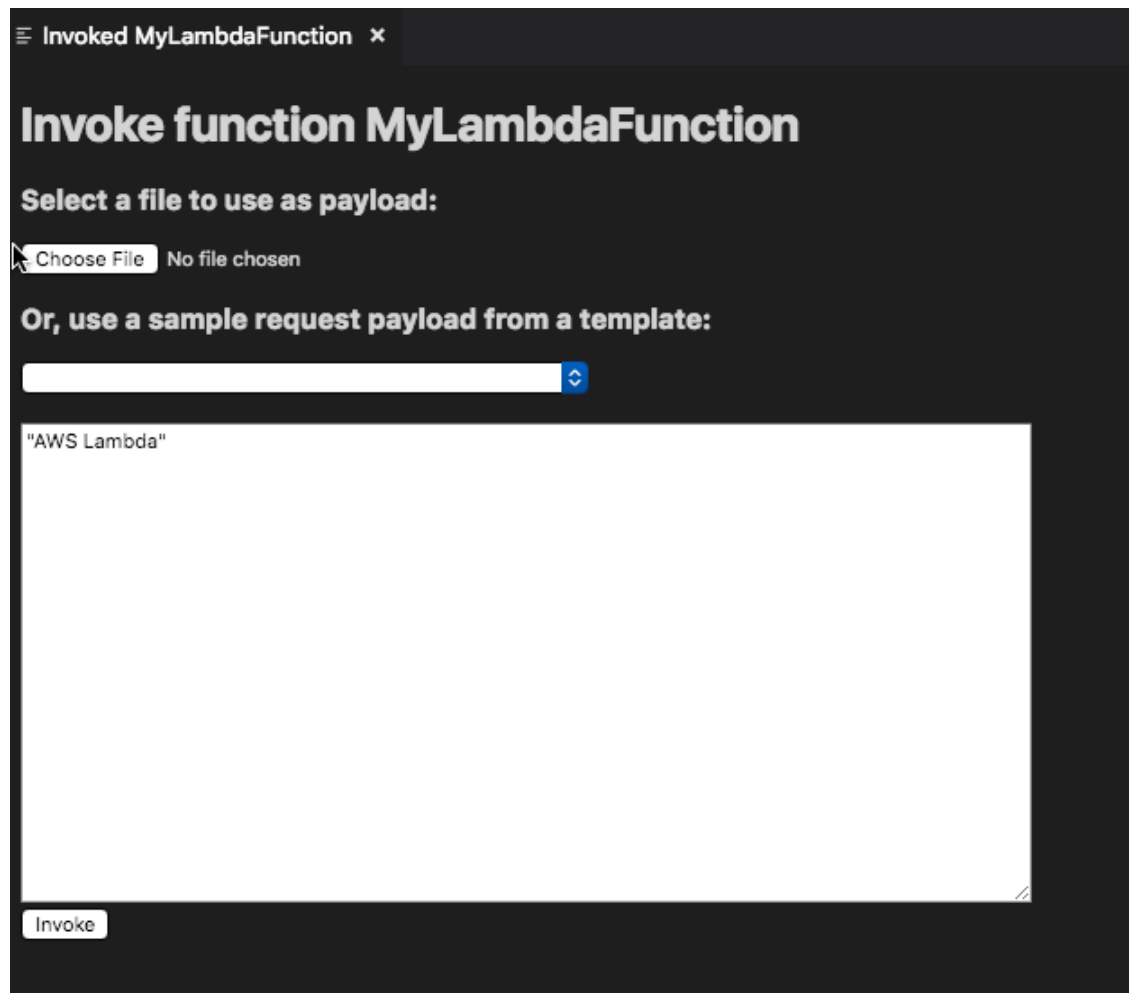
Invocar uma função do Lambda

Você pode invocar uma função do Lambda no [AWS No Toolkit for VS Code](#).

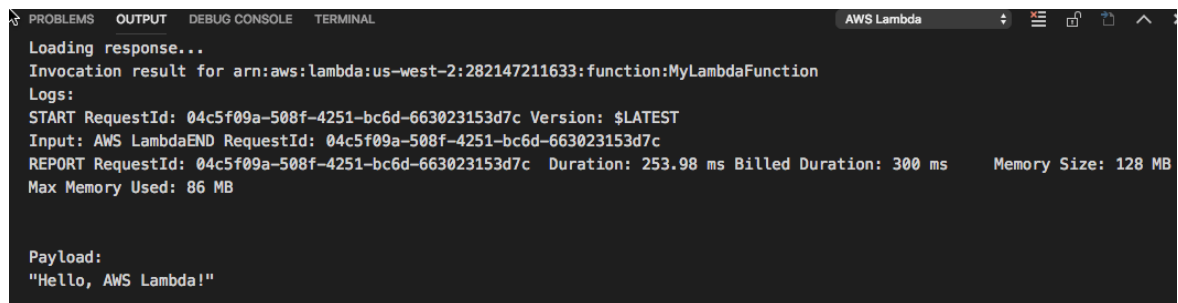
1. No [AWS Explorer](#), escolha o nome da função do Lambda que você deseja invocar e abra o menu de contexto.



2. Selecione Invocar o em AWS.
3. Na janela de invocação, insira a entrada de que sua função do Lambda precisa. 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 faria para qualquer outro projeto usando o VS Code.



Excluir uma função do Lambda

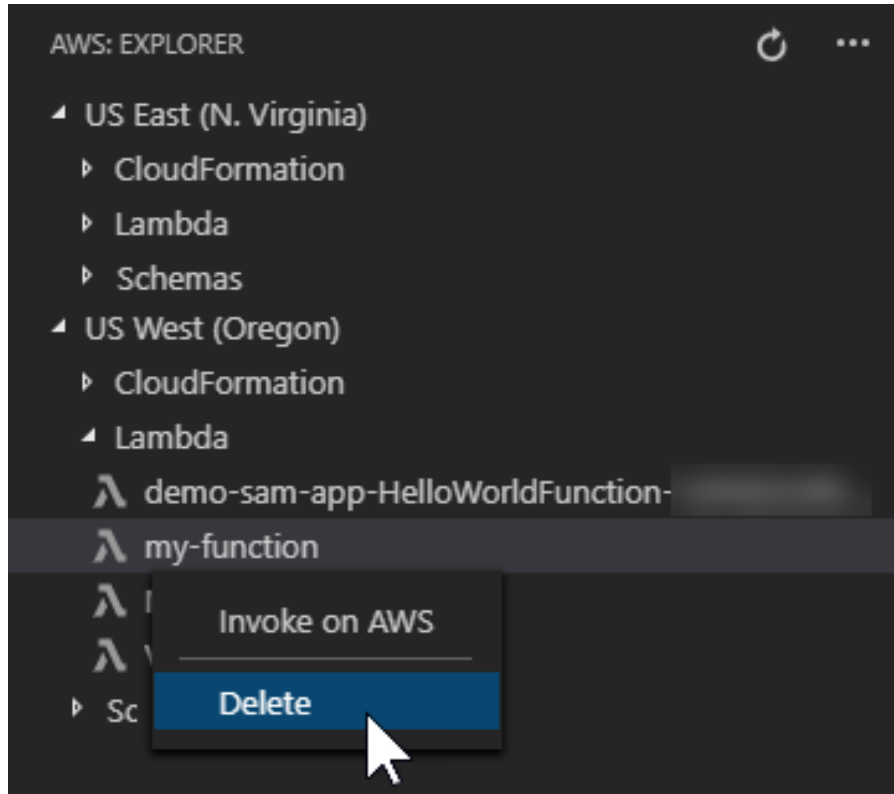
Você também pode excluir uma função do Lambda usando o mesmo menu de contexto.

Warning

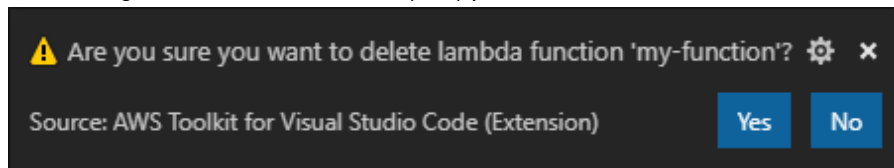
Não use este procedimento para excluir funções do Lambda associadas ao [AWS CloudFormation](#) (por exemplo, a função do Lambda que foi criada quando [Criar um aplicativo sem](#)

[servidor \(p. 64\)](#) anteriormente neste guia). Essas funções devem ser excluídas por meio da pilha do AWS CloudFormation.

1. NoAWSExplorer, 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 código de um function Lambda remoto para seu espaço de trabalho VS Code para edição e depuração.

Note

O kit de ferramentas só oferece suporte à importação de funções do Lambda usando tempos de execução de Node.js e Python suportados.

1. NoAWSExplorer, escolha o nome da função do Lambda que você deseja importar e abra o menu de contexto.

2. SelecioneImportar...
3. Escolha uma pasta para a qual importar o código do Lambda. Pastas fora do workspace atual serão adicionadas ao workspace atual.
4. Após o download, o Toolkit adiciona o código ao seu espaço de trabalho e abre o arquivo que contém o código do manipulador do Lambda. O kit de ferramentas também cria uma configuração de execução, 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 “Executando e depurando um aplicativo sem servidor a partir do modelo \(local\)” \(p. 66\)](#).

Carregar uma função do Lambda

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

Warning

O kit de ferramentas não consegue verificar se o código funciona. Certifique-se de que o código funcione antes de atualizar as funções do Lambda de produção.

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

Carregar um arquivo.zip pré-fabricado

- Selecione Arquivo Zip no menu Seleção rápida.
- Escolha um arquivo.zip do seu sistema de arquivos e confirme o upload com a caixa de diálogo modal. Isso carrega o arquivo.zip como está e atualiza imediatamente o Lambda após a implantação.

Faça upload de um diretório como está

- Selecione Diretório no menu Seleção rápida.
- Escolha um diretório no sistema de arquivos.
- Selecione No (Não) quando solicitado a construir o diretório, confirme o upload com a caixa de diálogo modal. Isso carrega o diretório como está e atualiza imediatamente o Lambda após a implantação.

Crie e carregue um diretório

Note

Isso requer o AWS SAM CLI.

- Selecione Diretório no menu Seleção rápida.
- Escolha um diretório do sistema de arquivos do.
- Selecione Yes (Sim) quando solicitado a construir o diretório, confirme o upload com a caixa de diálogo modal. Isso constrói o código no diretório usando o AWS SAM CLI `sam 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 você deseja alterar o manipulador vinculado à função do Lambda, você pode fazê-lo através do AWS Management Console ou o AWS CLI.

Trabalhar com o Amazon S3

Os tópicos a seguir descrevem como usar o AWS Toolkit for Visual Studio Code para trabalhar com [Amazon S3](#) baldes e objetos em uma AWS conta.

Tópicos

- [Trabalhar com buckets do Amazon S3](#) (p. 41)
- [Trabalhar com objetos do Amazon S3](#) (p. 43)

Trabalhar com buckets do Amazon S3

Cada objeto armazenado no Amazon S3 reside em um bucket. Você pode usar buckets para agrupar objetos relacionados da mesma forma como usa um diretório para agrupar arquivos em um sistema de arquivos.

Tópicos

- [Criar um bucket do Amazon S3](#) (p. 41)
- [Adicionar uma pasta a um bucket do Amazon S3](#) (p. 42)
- [Excluir um bucket do Amazon S3](#) (p. 42)
- [Configurar a exibição de itens do Amazon S3](#) (p. 42)

Criar um bucket do Amazon S3

1. No **AWS Explorer**, clique com o botão direito no ícone e, em seguida, escolha **Criar bucket**. Ou escolha o ícone **Criar bucket** em frente ao ícone.



2. No **Nome do bucket**, insira um nome válido para o bucket. Pressione **Enter** para confirmar.

O novo bucket é exibido sob a guia **S3**.

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 exclusivo globalmente. Se alguma outra conta já criou um bucket com o nome que você escolheu, será necessário usar outro nome.

Você pode verificar o resultado se você não puder criar um bucket. Por exemplo, um erro `BucketAlreadyExists` ocorre se você usar um nome de bucket já em uso. Para obter mais informações, consulte [Restrições e limitações do bucket](#) no Guia do usuário do Amazon Simple Storage Service.

Depois de criar um bucket, você poderá copiar seu nome e o nome de recurso da Amazon (ARN) para a área de transferência. Clique com o botão direito do mouse na entrada do bucket e selecione a opção relevante no menu.

Adicionar uma pasta a um bucket do Amazon S3

Você organiza o conteúdo de um bucket agrupando objetos em pastas. Você também pode criar pastas dentro de outras pastas.

1. No AWSExplorer, escolha o Amazon S3 Para exibir a lista de buckets.
2. Clique com o botão direito do mouse em um bucket ou pasta e escolha Criar pasta. Ou escolha o Criar pasta ícone do outro lado do nome do bucket ou pasta.
3. Digite um Nome da pasta e pressione Digite.

Excluir um bucket do Amazon S3

Quando você exclui um bucket, você também exclui as pastas e objetos que ele contém. Você deve confirmar que deseja fazer isso antes que a exclusão seja realizada.

Note

Para excluir apenas uma pasta, não o balde inteiro, use o AWS Management Console.

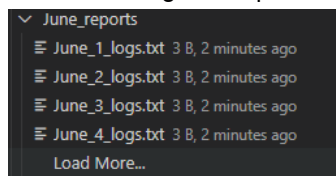
1. No AWSExplorer, escolha o S3 Para expandir a lista de buckets.
2. Clique com o botão direito do mouse no bucket a ser excluído e escolha Delete S3 Bucket (Excluir bucket do S3).
3. Insira o nome do bucket para confirmar a exclusão e pressione Digite.

Note

Se o bucket contiver objetos, o bucket será esvaziado antes de ser excluído. Isso pode levar algum tempo se for necessário excluir todas as versões de milhares de objetos. Depois da exclusão, uma notificação é exibida.

Configurar a exibição de itens do Amazon S3

Se você estiver trabalhando com um grande número de objetos ou pastas do Amazon S3, poderá especificar quantos são exibidos ao mesmo tempo. Quando o número máximo for exibido, você pode escolher Carregar mais para exibir o próximo lote.



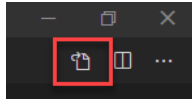
1. No VS Code Barra de atividades, escolha Extensões.
2. Para a AWSExtensão do toolkit, escolha o ícone de configurações e, a seguir, escolha Configurações da extensão.
3. No Configurações página, role até o AWS > S3: Máximo de itens por página Configuração do.
4. Altere o valor padrão para o número de itens do S3 que você deseja exibir antes de optar por carregar mais.

Note

O intervalo de valores aceitos está entre 3 e 1000. 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 só vez. (Por padrão, você pode criar até 100 buckets em cada AWS conta.)

5. Feche oConfiguraçõesPágina para confirmar suas alterações.

Você também pode atualizar as configurações em um arquivo formatado em JSON escolhendo oConfigurações abertas (JSON)ícone no canto superior direito doConfigurações.



Trabalhar com objetos do Amazon S3

Os objetos são as entidades fundamentais armazenadas no Amazon S3. Os objetos consistem em metadados e dados de objeto.

Tópicos

- [Fazer upload de um arquivo em um bucket do Amazon S3 \(p. 43\)](#)
- [Fazer download de um objeto do Amazon S3 \(p. 44\)](#)
- [Excluir um objeto do Amazon S3 \(p. 44\)](#)
- [Gerar um URL pré-assinado para um objeto do Amazon S3 \(p. 44\)](#)

Fazer upload de um arquivo em um bucket do Amazon S3

Você pode usar a interface do Toolkit ou oPaleta de comandos doPara fazer upload de um arquivo em um bucket.

Ambos os métodos permitem que você carregue um arquivo do ambiente de um usuário e o armazene como um objeto S3 naAWS Cloud. Você pode carregar um arquivo para um bucket ou para uma pasta que organiza o conteúdo desse bucket.

Faça upload de um arquivo em um bucket do S3 usando a interface

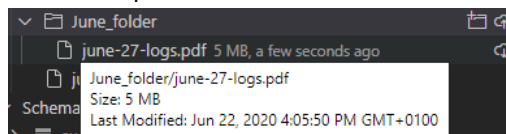
1. NoAWS Explorer, escolha oAmazon S3Para exibir a lista de buckets.
2. Clique com o botão direito do mouse em um bucket ou pasta nesse bucket e, a seguir, escolhaUpload de arquivo. Ou escolha oUpload de arquivoícone do outro lado do nome do bucket ou pasta.

Note

Se você clicar com o botão direito do mouse em um objeto S3, poderá escolherFazer upload para o principal. Isso permite que você adicione um arquivo à pasta ou bucket que contém o arquivo selecionado.

3. Usando o gerenciador de arquivos do sistema, selecione um arquivo e escolhaUpload de arquivo.

O arquivo selecionado é carregado como um objeto S3 para o bucket ou pasta. A entrada de cada objeto descreve o tamanho do objeto armazenado e há quanto tempo ele foi carregado. Você pode pausar sobre a listagem do objeto para exibir o caminho, o tamanho e a hora em que ele foi modificado pela última vez.



Faça upload do arquivo atual em um bucket do S3 usando oPaleta de comandos do

1. Para selecionar um arquivo para upload, escolha a guia do arquivo.

2. Pressione **Ctrl+Shift+P** para exibir a Paleta de comandos do.
3. Na Paleta de comandos do, comece a digitar a frase `upload file` para exibir o **AWS: Upload File** comando. Escolha o comando quando ele for exibido.
4. para a Etapa 1: Selecione um arquivo para fazer upload do, você pode escolher o arquivo atual ou procurar outro arquivo.
5. para a Etapa 2: Selecione um bucket do S3 para upload do, escolha um bucket na lista.

O arquivo selecionado é carregado como um objeto S3 para o bucket ou pasta. A entrada de cada objeto descreve o tamanho do objeto armazenado e há quanto tempo ele foi carregado. Você pode pausar sobre a listagem do objeto para exibir o caminho, o tamanho e a hora em que ele foi modificado pela última vez.

Fazer download de um objeto do Amazon S3

Você pode baixar objetos em um bucket do Amazon S3 a partir do AWS Nuvem para seu sistema.

1. No AWS Explorer, escolha o S3 Para exibir a lista de buckets.
2. Em um bucket ou em uma pasta em um bucket, clique com o botão direito do mouse em um objeto e, a seguir, escolha **Fazer download do como**. Ou escolha **Fazer download do ícone** do outro lado do nome do objeto.
3. Usando o gerenciador de arquivos do sistema, selecione uma pasta de destino, insira um nome de arquivo e escolha **Baixar**.

Depois que um arquivo é baixado, você pode abri-lo na interface IDE usando o **AWSToolkit logs no Resultado Guia**. Dependendo do sistema operacional, pressione **Cmd ou Ctrl** clique no link para a localização do arquivo.

Excluir um objeto do Amazon S3

Você pode excluir permanentemente um objeto se ele estiver em um bucket sem versionamento. Mas, para buckets habilitados para versão, 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 objetos](#) no Guia do usuário do Amazon Simple Storage Service.

1. No AWS Explorer, escolha o S3 Para exibir a lista de buckets.
2. Em um bucket ou pasta em um bucket, clique com o botão direito do mouse em um objeto e, a seguir, escolha **Excluir**.
3. Escolha **Delete** para confirmar.

Gerar um URL pré-assinado para um objeto do Amazon S3

Com URLs pré-assinadas, o proprietário de um objeto pode compartilhar objetos privados do Amazon S3 com outras pessoas ao conceder permissão de tempo limitado para download de objetos. Para obter mais informações, consulte [Compartilhar um objeto com uma pre-signed URL](#) no Guia do usuário do Amazon S3.

1. No AWS Explorer, escolha o S3 Para exibir a lista de buckets.
2. Em um bucket ou pasta em um bucket, clique com o botão direito do mouse em um objeto e, a seguir, escolha **Gerar URL pre-signed**.
3. Na Paleta de comandos do, insira o número de minutos em que o URL pode ser usado para acessar o objeto. Pressione **Digite** para confirmar.

O código **VSStatus Bar** (Barra de status) confirma que o URL pré-assinado para o objeto foi copiado para a área de transferência.

Trabalhar com documentos de Automação do Systems Manager

AWS Systems Manager oferece visibilidade e controle da sua infraestrutura no AWS. O Systems Manager oferece uma interface de usuário unificada para que seja possível visualizar dados operacionais de vários AWS serviços e automatize tarefas operacionais em todo o seu AWS recursos da AWS.

A Documento do Systems Manager. Define as ações que o Systems Manager realiza nas suas instâncias gerenciadas. Um documento de automação é um tipo de documento do Systems Manager usado para executar 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 com o AWS Toolkit for Visual Studio Code.

Tópicos

- [Pressupostos e pré-requisitos do \(p. 45\)](#)
- [Permissões do IAM para documentos do Systems Manager Automation \(p. 45\)](#)
- [Criar um novo documento de Automação do Systems Manager \(p. 46\)](#)
- [Abrir um documento de Automação do Systems Manager \(p. 46\)](#)
- [Editar um documento de Automação do Systems Manager \(p. 47\)](#)
- [Publicar um documento de Automação do Systems Manager \(p. 47\)](#)
- [Excluir um documento de Automação do Systems Manager \(p. 48\)](#)
- [Executar um documento de Automação do Systems Manager \(p. 48\)](#)
- [Solução de problemas de documentos do Systems Manager Automation no Toolkit for VS Code \(p. 48\)](#)

Pressupostos e pré-requisitos do

Antes de começar, certifique-se:

- Você instalou o Visual Studio Code e a versão mais recente do AWS Toolkit for Visual Studio Code. Para obter mais informações, consulte [Instalar o AWS Toolkit for Visual Studio Code \(p. 2\)](#).
- Você está familiarizado com o Systems Manager. Para obter mais informações, consulte o [Manual do usuário do AWS Systems Manager](#).
- Você está familiarizado com os casos de uso do Systems Manager Automation. Para obter mais informações, consulte [AWS Automação do Systems Manager](#) no AWS Systems Manager Guia do usuário do.

Permissões do IAM para documentos do Systems Manager Automation

No Toolkit for VS Code, você deve ter um perfil de credenciais que contenha o AWS Identity and Access Management Permissões (IAM) necessárias para criar, editar, publicar e excluir documentos do Systems Manager Automation. O documento de política a seguir define as permissões do IAM necessárias que podem ser usadas em uma política 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 [Criando políticas do IAM](#) no Manual do usuário do IAM. Para obter informações sobre como configurar o perfil de credenciais do, consulte [Configuração do AWS credenciais](#) (p. 5).

Criar um novo documento de Automação do Systems Manager

Você pode criar um novo documento de automação no JSON ou YAML. Usar o Visual Studio Code. Quando você cria um novo documento de automação, ele será apresentado em um arquivo sem título. Você pode nomear seu arquivo e salvá-lo no VS Code, no entanto, o nome do arquivo não está visível para AWS.

Como criar um novo documento de Automação

1. Abra o VS Code.
2. No **Exibir Menu**, selecione **Paleta de comandos** do **Para abrir a Paleta de comandos**.
3. Na **Paleta de comandos**, digite **AWS: Criar um novo documento do Systems Manager localmente**.
4. Escolha um dos modelos iniciais para um exemplo do Hello World.
5. Escolha **JSON** ou **YAML**.

Um novo documento de automação é criado.

Note

Seu novo documento de automação no VS Code não aparece automaticamente em AWS. Você deve publicá-lo para AWS antes que você possa executá-lo.

Abrir um documento de Automação do Systems Manager

É possível usar o **AWS Explorer** para encontrar documentos existentes do Systems Manager Automation. Quando você abre um documento de automação existente, ele aparece como um arquivo sem título no VS Code.

Para abrir o documento de automação

1. Abra o VS Code.
2. Na navegação à esquerda, escolha **AWS** para abrir o **AWS Explorador**.
3. No **AWS Explorador**, para **Systems Manager (Gerenciador de sistemas)**, escolha o ícone de download no documento que você deseja abrir e escolha a versão do documento. O arquivo será aberto no formato dessa versão. Caso contrário, escolha qualquer **Baixar como JSON** ou **Baixar como YAML**.

Note

Salvar localmente um documento de automação como um arquivo no VS Code não faz com que ele apareça em AWS. Ele precisa ser publicado para AWS antes de executar.

Editar um documento de Automação do Systems Manager

Se você possui algum documento de automação, eles aparecerão no **Propriedade de mim** categoria de documentos do Systems Manager na **AWS Explorer**. Você pode possuir documentos de automação que já existem no AWS ou você pode possuir documentos novos ou atualizados nos quais você publicou anteriormente no AWS do VS Code.

Quando você abre um documento de automação para edição no VS Code, você pode fazer mais com ele do que pode na AWS Management Console. Por exemplo:

- Há validação de esquema em ambos **JSON** e **YAML** formatos do.
- Existem trechos disponíveis no editor de documentos para você criar qualquer um dos tipos de etapas de automação.
- Há suporte de preenchimento automático em várias opções no **JSON** e **YAML**.

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**, navegue até o documento no qual você deseja definir a versão padrão, abra o menu de contexto (clique com o botão direito do mouse) do documento e selecione **Definir a versão padrão**.

Note

Se o documento escolhido tiver apenas uma versão do, você não poderá alterar o padrão.

Publicar um documento de Automação do Systems Manager

Depois de editar seu documento de automação no VS Code, você pode publicá-lo no AWS.

Para publicar seu documento de automação

1. Abra o documento de Automação que você deseja publicar usando o procedimento descrito em [Abrir um documento de Automação do Systems Manager \(p. 46\)](#).
2. Faça as alterações desejadas para serem publicadas. Para obter mais informações, consulte [Editar um documento de Automação do Systems Manager \(p. 47\)](#).
3. Na parte superior direita do arquivo aberto, escolha o ícone de upload.
4. Na caixa de diálogo de fluxo de trabalho de publicação, selecione a **AWS Região** na qual você deseja publicar o documento de Automação.
5. Se você estiver publicando um novo documento, escolha **Criação rápida**. Caso contrário, escolha **Atualização rápida** para atualizar um documento de automação existente em que **AWS Região** :
6. Insira o nome para este documento de automação.

Quando você publica uma atualização em um documento de automação existente para AWS, uma nova versão é adicionada ao documento.

Excluir um documento de Automação do Systems Manager

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

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

Para excluir o documento de automação

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

Executar um documento de Automação do Systems Manager

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

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

Note

A execução de um documento de automação pode criar novos recursos no AWS e pode incorrer em custos de faturamento. Recomendamos enfaticamente que você entenda o que seu documento de automação criará em sua conta antes de executá-lo.

Solução de problemas de documentos do Systems Manager Automation no Toolkit for VS Code

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

Salvar um documento de automação no VS Code não publica o documento Automation no AWS. Para obter mais informações sobre como publicar o documento de Automação, consulte [Publicar um documento de Automação do Systems Manager](#) (p. 47).

A publicação do meu documento Automation falhou com um erro de permissões.

Certifique-se de que o AWS perfil de credenciais tem as permissões necessárias para publicar documentos de automação. Para um exemplo de política de permissões, consulte [Permissões do IAM para documentos do Systems Manager Automation](#) (p. 45).

Publiquei meu documento de automação para o AWS, mas eu não vejo isso no AWS Management Console.

Certifique-se de que você publicou o documento para a mesma AWS Região em que você está navegando no AWS Management Console.

Excluí meu documento de automação, mas ainda estou sendo cobrado pelos recursos que ele criou.

A exclusão de um documento de automação não exclui os recursos que ele criou ou modificou. Identifique o AWSrecursos que você criou a partir do [AWSConsole de gerenciamento de cobrança](#), explore suas cobranças e escolha quais recursos excluir de lá.

Trabalho com AWS Step Functions

O AWS Toolkit for Visual Studio Code (VS Code) fornece suporte para [AWS Step Functions](#). Usando o Toolkit for VS Code, você pode criar, atualizar e executar máquinas de estado Step Functions.

Tópicos

- [Trabalho com AWS Step Functions \(p. 49\)](#)

Trabalho com AWS Step Functions

Você pode usar o AWS Toolkit for Visual Studio Code (VS Code) para realizar várias operações com [Máquinas de estado do](#).

Tópicos

- [Prerequisites \(p. 49\)](#)
- [Trabalhe com máquinas de estado no VS Code \(p. 49\)](#)
- [Modelos de máquina de estado \(p. 53\)](#)
- [Visualização de gráficos de máquina de estado \(p. 53\)](#)
- [Trechos de código \(p. 54\)](#)
- [Preenchimento e validação de código \(p. 55\)](#)

Prerequisites

- Verifique se o sistema atende aos pré-requisitos especificados em [Instalação do Toolkit for VS Code \(p. 2\)](#) e, em seguida, instale o kit de ferramentas.
- Verifique se você configurou suas credenciais antes de abrir o AWSExplorer.

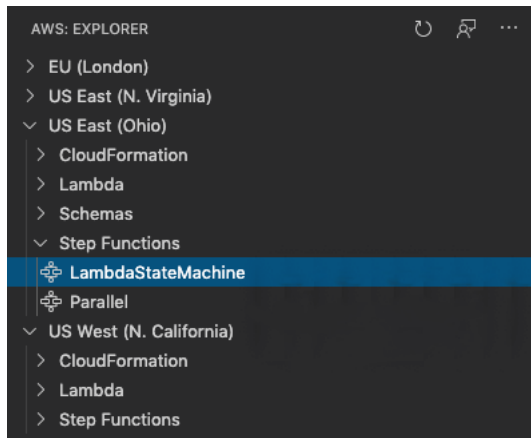
Trabalhe com máquinas de estado no VS Code

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, listar máquinas de estado existentes, executá-las e fazer download delas. 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:

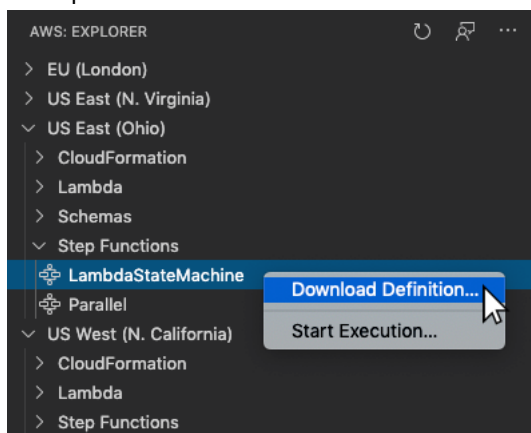
1. Abrir o AWSExplorer.
2. Selecione Step Functions
3. Verifique se ele lista todas as máquinas de estado em sua conta.



Baixar uma máquina de estado

Para baixar uma máquina de estado:

1. NoAWSExplorerClique 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.



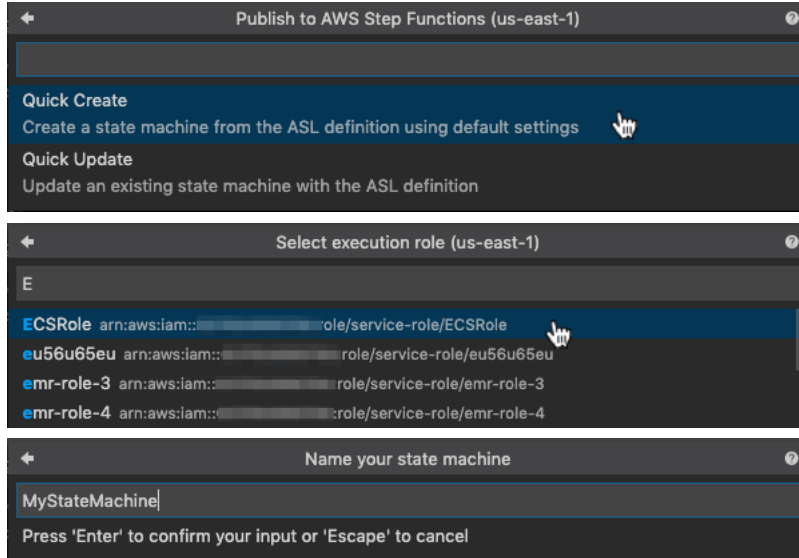
Criar uma máquina de estado

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).

```
1  Publish to Step Functions | Render graph
2  {
3    "StartAt": "FirstState",
4    "States": {
5      "FirstState": {
6        "Type": "Task",
7        "Resource": "arn:aws:lambda:us-east-1:123456789012:function:Function",
8        "Next": "ChoiceState"
9      },
10     "ChoiceState": {
11       "Type": "Choice",
```

3. Selecione Quick Create (Criação rápida), escolha uma função e nomeie sua máquina de estado.



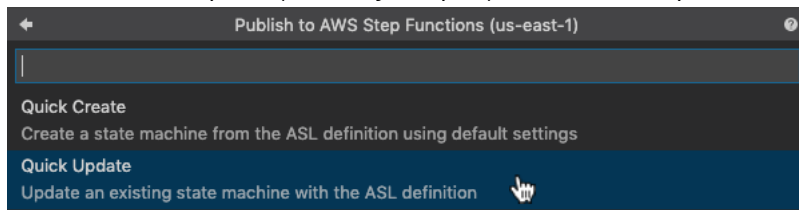
Atualizar uma máquina de estado

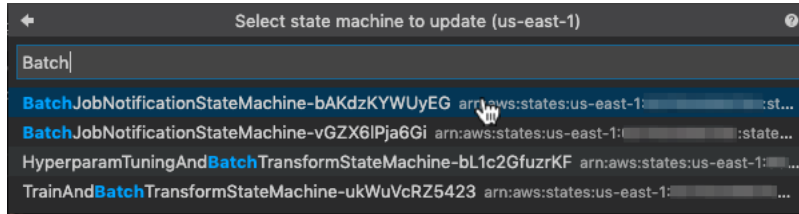
Para atualizar uma máquina de estado:

1. Edite o arquivo ASL com sua definição de máquina de estado.
2. Selecione Publish to Step Functions (Publicar no Step Functions).

```
1  Publish to Step Functions | Render graph
2  {
3    "StartAt": "FirstState",
4    "States": {
5      "FirstState": {
6        "Type": "Task",
7        "Resource": "arn:aws:lambda:us-east-1:123456789012:function:Function",
8        "Next": "ChoiceState"
9      },
10     "ChoiceState": {
11       "Type": "Choice",
```

3. Selecione Quick Update (Atualização rápida) e escolha a máquina de estado que deseja atualizar.

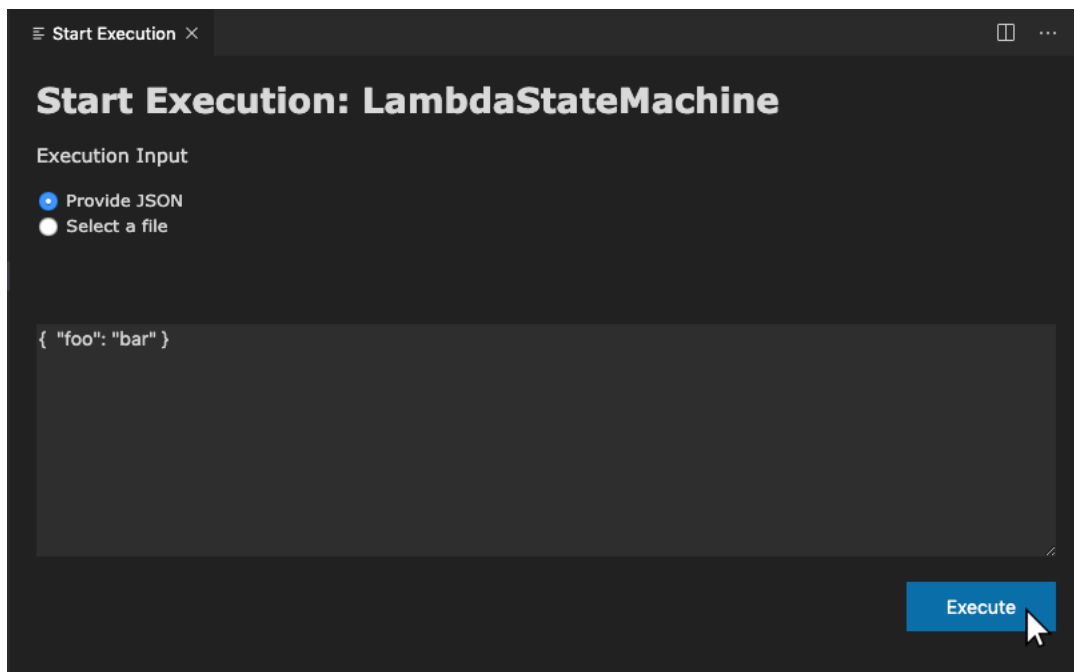
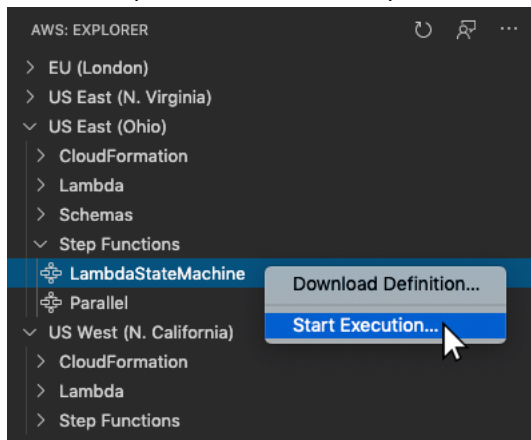




Executar uma máquina de estado

Para executar uma máquina de estado:

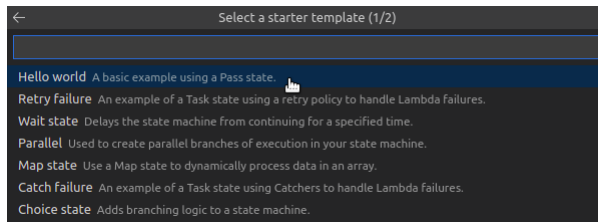
1. NoAWSExplorerClique com o botão direito do mouse na máquina de estado que deseja executar.
2. Forneça entrada para a máquina de estado. Você pode tentar a entrada de um arquivo e a entrada em uma caixa de texto.
3. Inicie a máquina de estado e verifique se ela é executada com êxito.



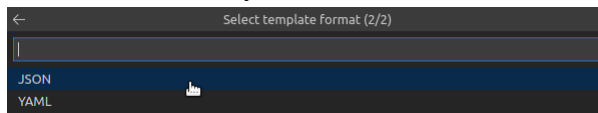
Modelos de máquina de estado

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. Abrir oPaleta de comandos dono VS Code.
2. SelectAWS: Criar uma nova máquina de estado Step Functions.
3. Escolha o modelo que deseja usar.



4. Escolha se você deseja usar o formato de modelo JSON ou YAML.



Visualização de gráficos de máquina de estado

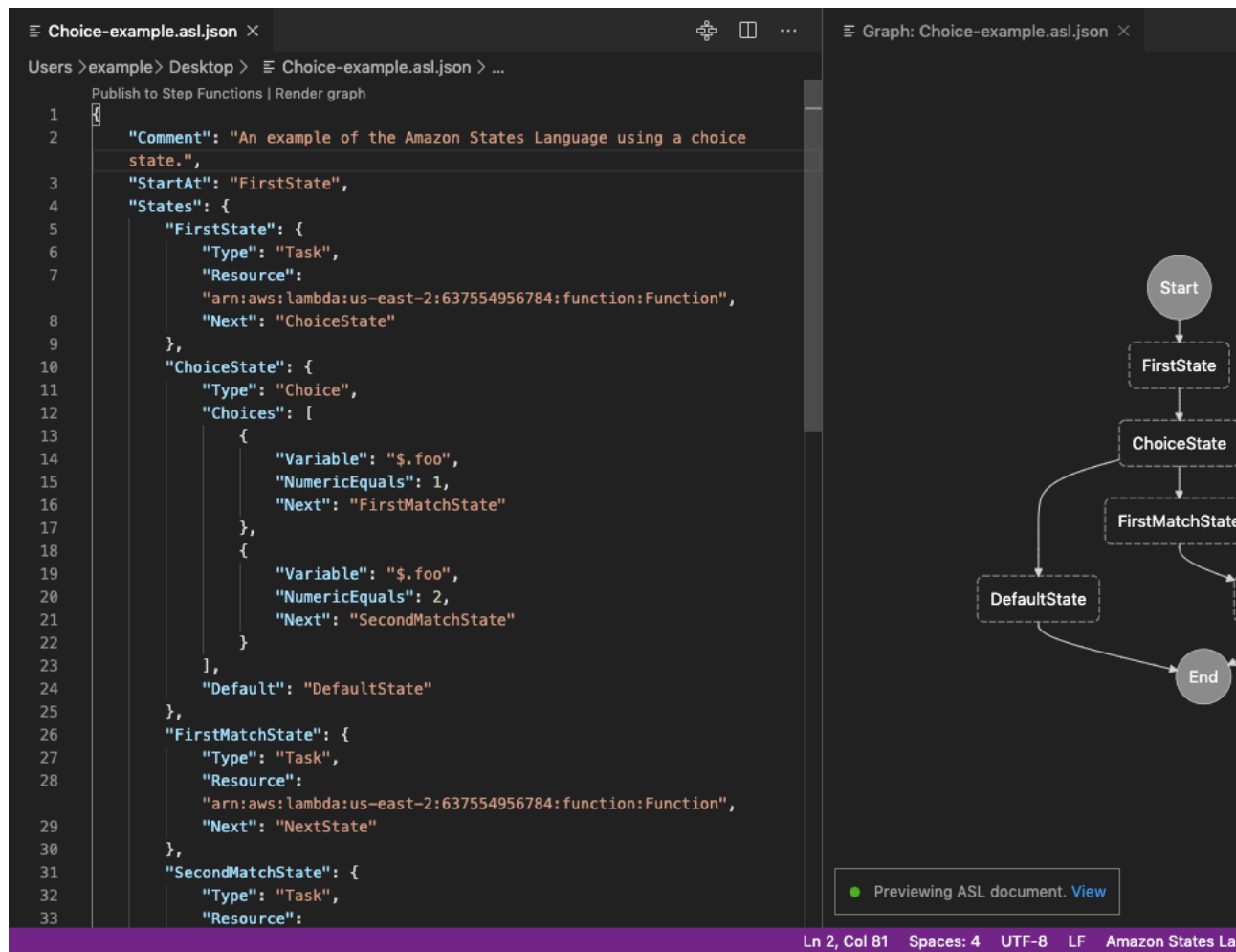
As visualizações de gráfico permitem que você veja a aparência da sua máquina de estado no formato gráfico. Quando você criar uma visualização de gráfico, outra guia será aberta e exibirá uma visualização da máquina de estado JSON ou YAML. Depois você poderá comparar a definição de máquina de estado 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:

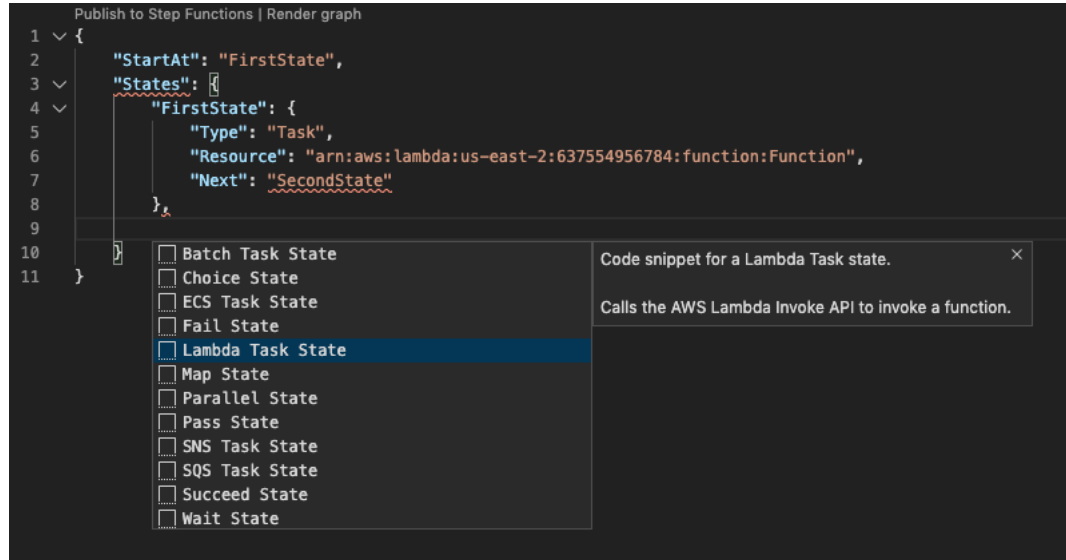
1. Defina sua máquina de estado.
2. Abrir oPaleta de comandos dono VS Code.
3. Para criar uma visualização, use o botão de visualização no canto superior direito ou escolhaAWSGráfico de máquina de estado de visualização.



Trechos de código

Os trechos de código permitem inserir seções curtas do código. Para usar trechos de código:

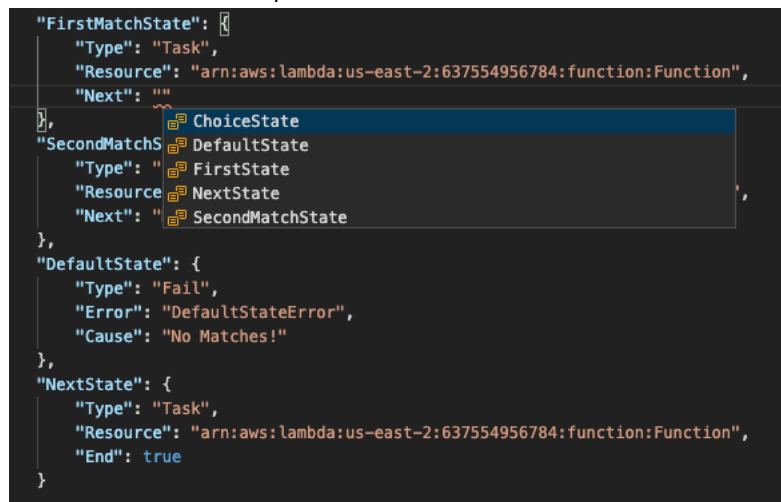
1. Abra um arquivo e salve-o com a extensão `.asl.json` para o formato JSON ou `.asl.yaml` para o formato YAML.
2. Crie uma nova máquina de estado com a propriedade `States` (Estados).
3. Coloque o cursor dentro de `States` (Estados).
4. Use a combinação de teclas `Control + Space` e selecione seu trecho 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.

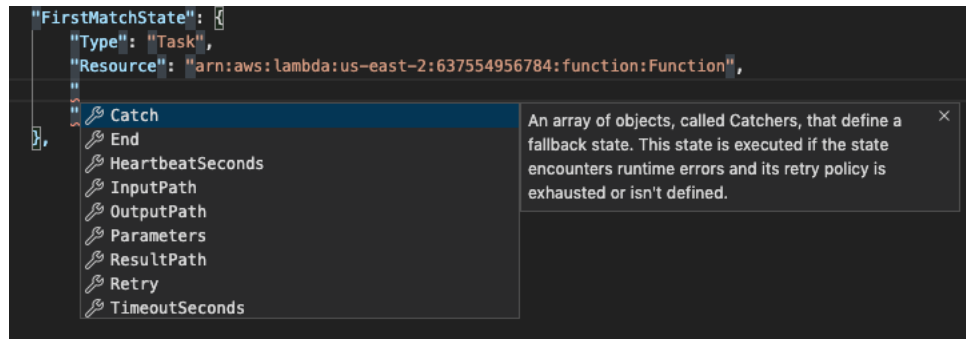


Preenchimento e validação de código

Para ver como funciona o preenchimento de código:

1. Crie vários estados.
2. Coloque o cursor após uma propriedade Next (Próximo), StartAt (Iniciar às) ou Default (Padrão).
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





Trabalhar com recursos do

Além de acessar AWS serviços listados por padrão no AWS Explorer, você também pode ir para Recursos e escolher entre centenas de recursos para adicionar à interface. Dentro do 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 Streams, AWS Funções do IAM, distribuições Amazon VPC e Amazon CloudFront.

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 você selecionar o `AWS::Lambda::Function` tipo de recurso, você pode acessar os recursos que definem funções diferentes, suas propriedades e seus atributos.

Depois de adicionar um tipo de recurso aos Recursos, você pode interagir com ele e com seus recursos das seguintes maneiras:

- Exibir uma lista de recursos existentes que estão disponíveis no atual AWS Região para esse tipo de recurso.
- Exibir uma versão somente leitura do arquivo JSON que descreve um recurso.
- Copiar o identificador do recurso para o recurso.
- Visualizar a AWS documentação que explica a finalidade do tipo de recurso e do esquema (nos formatos JSON e YAML) para modelar um recurso.
- Criar um novo recurso editando e salvando um modelo formatado em JSON que esteja em conformidade com um esquema.*
- Atualizar ou excluir 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 recursos experimentais continuam sendo testados e atualizados, eles podem ter problemas de usabilidade. E as características experimentais podem ser removidas do AWS Toolkit for Visual Studio Code sem aviso prévio.

Para permitir o uso de recursos experimentais para recursos, abra o Configurações painel no VS Code IDE e expanda Extensões e escolha o AWS Toolkit.

Sob AWS: Experiments, selecione Modificação JSON Resource para permitir criar, atualizar e excluir recursos do.

Para obter mais informações, consulte [Trabalhar com recursos experimentais \(p. 22\)](#).

Permissões do IAM para acessar recursos

Você precisa de específicas AWS Identity and Access Management permissões para acessar os recursos associados aos AWS Serviços da . Por exemplo, uma entidade do IAM, como um usuário ou uma função, requer permissões do Lambda para acessar `AWS::Lambda::Function` recursos da AWS.

Além das permissões para recursos de serviço, uma entidade do IAM requer permissões para permitir que o Toolkit for VS Code ligue AWS Operações da Cloud Control API em seu nome. As operações da Cloud Control API permitem que o usuário ou a função do IAM acesse e atualize os recursos remotos.

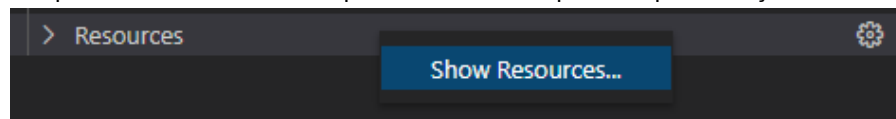
A maneira mais fácil de conceder permissões é anexar o AWS política gerenciada, `PowerUserAccess`, para a entidade do IAM que está chamando essas operações de API usando a interface do Toolkit. Essa política gerenciada concede uma variedade de permissões para executar tarefas de desenvolvimento de aplicativos, incluindo chamadas de operações de API.

Para obter permissões específicas que definem operações de API permitidas em recursos remotos, consulte o [AWS Guia do usuário da Cloud Control API](#).

Adicionando e interagindo com recursos existentes

1. No AWS Explorer, clique com o mouse e escolha **Mostrar recursos**.

Um painel exibe uma lista de tipos de recursos disponíveis para seleção.



2. No painel de seleção, selecione os tipos de recurso para adicionar à AWS Explorer. Pressione **Retornar** ou escolha **OK** para confirmar.

Os tipos de recurso selecionados estão listados em **Recursos**.

Note

Se você já adicionou um tipo de recurso ao AWS Explorer, em seguida, desmarque a caixa de seleção desse tipo, ela não está mais listada em **Recursos** depois de escolher **OK**. Somente os tipos de recursos selecionados no momento estão visíveis no AWS Explorer.

3. Para exibir os recursos que já existem para um tipo de recurso, expanda a entrada desse tipo.

Uma lista de recursos disponíveis é exibida sob o tipo de recurso.

4. Para interagir com um recurso específico, clique com o botão direito do mouse em seu nome e selecione uma destas opções:

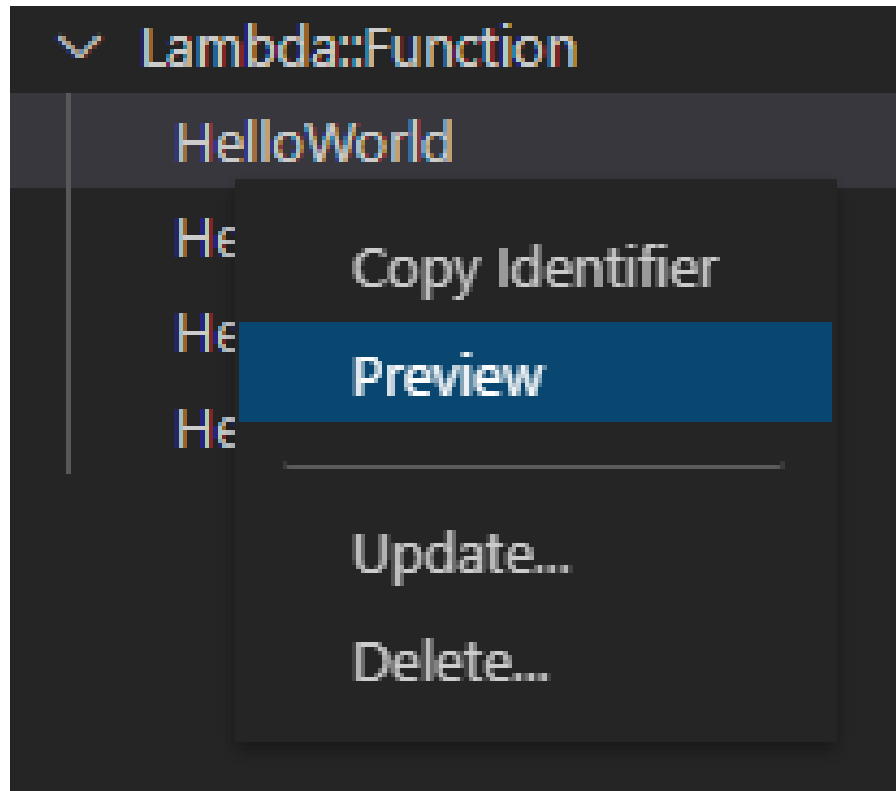
- Identificador do recurso de cópia: Copie o identificador do recurso específico para a área de transferência. (Por exemplo, `aws::dynamodb::table` recurso pode ser identificado usando `tableName` propriedade.)
- Demonstração do: Exiba uma versão somente leitura do modelo formatado em JSON que descreve o recurso.

Depois que o modelo de recurso for exibido, você poderá modificá-lo escolhendo o **Atualizar** ícone à direita da guia do editor. Para atualizar um recurso, você deve ter o **necessário** (p. 56) habilitado.

- Atualização: Edite o modelo formatado em JSON para o recurso em um editor VS Code. Para obter mais informações, consulte [Criar e editar recursos](#) (p. 58).
- Excluir: Exclua o recurso confirmando a exclusão em uma caixa de diálogo exibida. (A exclusão de recursos é atualmente um **bug** (p. 56) nesta versão do AWS Toolkit for Visual Studio Code.)

Warning

Se você excluir um recurso, qualquer AWS CloudFormation pilha que usa esse recurso falhará ao atualizar. Para corrigir essa falha de atualização, você precisa recriar o recurso ou remover a referência a ele na pilha AWS CloudFormation Modelo. Para obter mais informações, consulte esta [Artigo da Central de Conhecimento](#).



Criar e editar recursos

Important

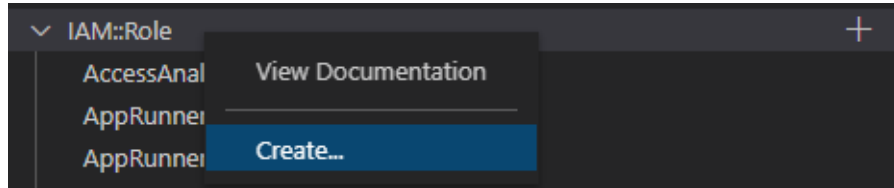
A criação e atualização de recursos é atualmente uma [limitação](#) (p. 56) nesta versão do AWS Toolkit for Visual Studio Code.

Criar um novo recurso envolve adicionar um tipo de recurso ao `Recursos` listar e, em seguida, editar um modelo formatado em JSON que define o recurso, suas propriedades e seus atributos.

Por exemplo, um recurso que pertence ao `AWS::SageMaker::UserProfile` tipo de recurso é definido com um modelo que cria um perfil de usuário para o Amazon SageMaker Studio. O modelo que define esse recurso de perfil de usuário deve estar em conformidade com o esquema de tipo de recurso para `AWS::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.

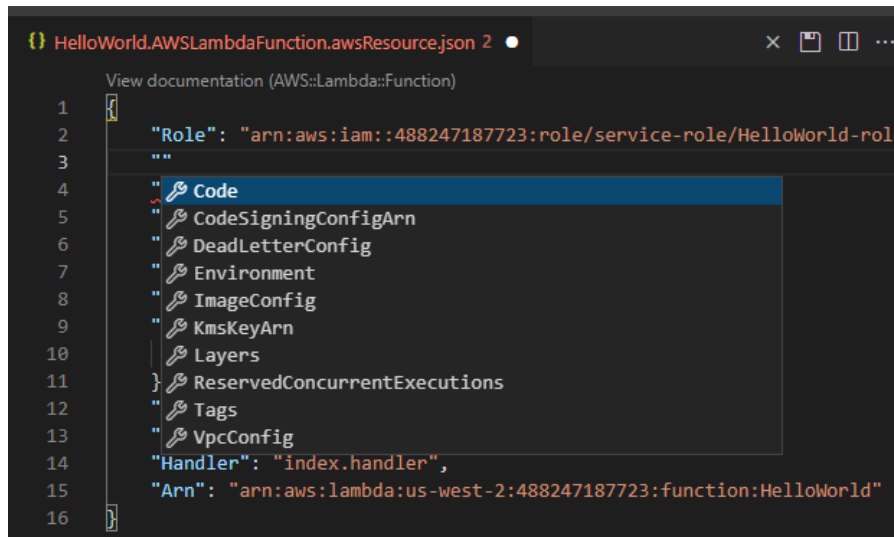
1. Adicione o tipo de recurso para o recurso que você deseja criar clicando com o botão direito do mouse no `Recursos` e escolhendo `Mostrar recursos`.
2. Depois que o tipo de recurso é adicionado em `Recursos`, escolha o ícone de mais (“+”) para abrir o arquivo de modelo em um novo editor.

Você também pode clicar com o botão direito do mouse no nome do recurso e escolher `Criar`. Você também pode acessar informações sobre como modelar o recurso escolhendo `Visualizar documentação`.



3. No editor, comece a definir propriedades que compõem o modelo de recurso. O recurso de preenchimento automático sugere nomes de propriedades que estão em conformidade com o esquema do modelo. Quando você passa o mouse sobre um tipo de propriedade, um painel exibe uma descrição do que ele é usado. Para obter informações detalhadas sobre o esquema, selecione **Visualizar documentação**.

Qualquer texto que não esteja em conformidade com o esquema de recurso é indicado por um sublinhado vermelho ondulado.



4. Depois de terminar de declarar seu recurso, escolha o **Save (Salvar)** ícone para validar seu modelo e salvar o recurso no controle remoto **AWSCloud**.

Se o modelo definir o recurso de acordo com o esquema, será exibida uma mensagem para confirmar se o recurso foi criado. (Se o recurso já existir, a mensagem confirma 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, uma mensagem será exibida para explicar que o recurso não pôde ser criado ou atualizado. Selecione **Visualizar logs** para identificar os elementos do modelo que você precisa corrigir.

Trabalho com o Amazon Elastic Container Service

O AWS Toolkit for Visual Studio Code O oferece algum suporte para [Amazon Elastic Container Service \(Amazon ECS\)](#). O Toolkit for VS Code ajuda você em determinados trabalhos relacionados ao Amazon ECS, como a criação de definições de tarefas.

Tópicos

- [Uso de arquivos de definição de tarefa do IntelliSense para Amazon ECS \(p. 60\)](#)

Uso de 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 tarefas, conforme descrito em [Como criar uma definição de tarefa](#) do Amazon Elastic Container Service. Quando você instala o AWS Toolkit for Visual Studio Code, a instalação inclui a funcionalidade IntelliSense para arquivos de definição de tarefas do Amazon ECS.

Prerequisites

- Verifique se o sistema atende aos pré-requisitos especificados em [Instalação do Toolkit for VS Code](#) (p. 2).

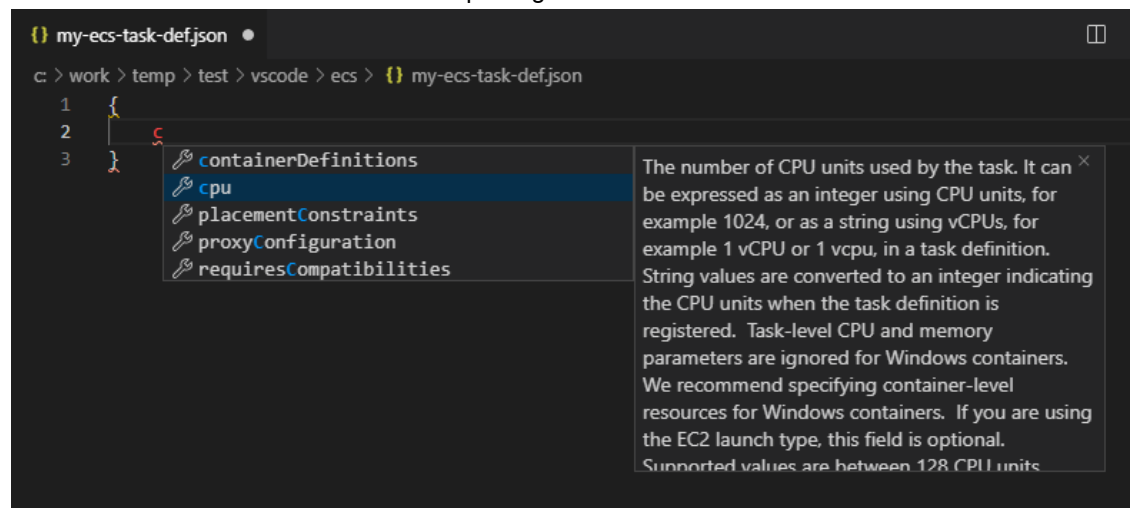
Usar o IntelliSense em arquivos de definição de tarefas do Amazon ECS

O exemplo a seguir mostra como usar o 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 de código do VS 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.



Trabalhar com o AWS CDK Explorer

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

O AWS CDK Explorer permite que você trabalhe com [AWS Cloud Development Kit \(CDK\)](#) aplicativos, ou aplicativos do. Encontre informações detalhadas sobre o AWS CDK no [AWS Cloud Development Kit \(CDK\) Guia do desenvolvedor](#).

Os aplicativos do AWS CDK são compostos por blocos de criação conhecidos como [constructos](#), que incluem definições para as suas 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 de árvore em um painel do Explorer dentro do editor do Visual Studio Code (VS Code). Consulte uma visualização de alto nível do AWS CDK Explorer no tópico de [navegação](#) (p. 19).

Esta seção fornece informações sobre como acessar e usar o AWS CDK Explorer no editor de código do VS. Ele pressupõe que você já [instalados e configurados](#) (p. 2) o Toolkit for VS Code de em seu sistema.

Note

Você pode desabilitar o AWS CDK Explorer para não exibi-lo no editor VS Code. No menu File (Arquivo), escolha Preferences (Preferências), Settings (Configurações). Em seguida, digite “cdk” na caixa Pesquisar e desmarque a caixa de seleção Habilitar o AWS Explorer CDK (Criar snapshot final?).

Tópicos

- [Trabalho com AWS CDK Applications](#) (p. 61)

Trabalho com AWS CDK Applications

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

Usar o AWS CDK Explorer no AWSToolkit for VS Code de para visualizar e trabalhar com AWS CDK aplicações.

Prerequisites

- Verifique se o sistema atende aos pré-requisitos especificados no [Instalação do Toolkit for VS Code](#) (p. 2).
- Instalar o AWS CDK, conforme descrito nas primeiras seções do [Conceitos básicos do AWS CDK](#) no [AWS Cloud Development Kit \(CDK\) Guia do desenvolvedor](#).

Important

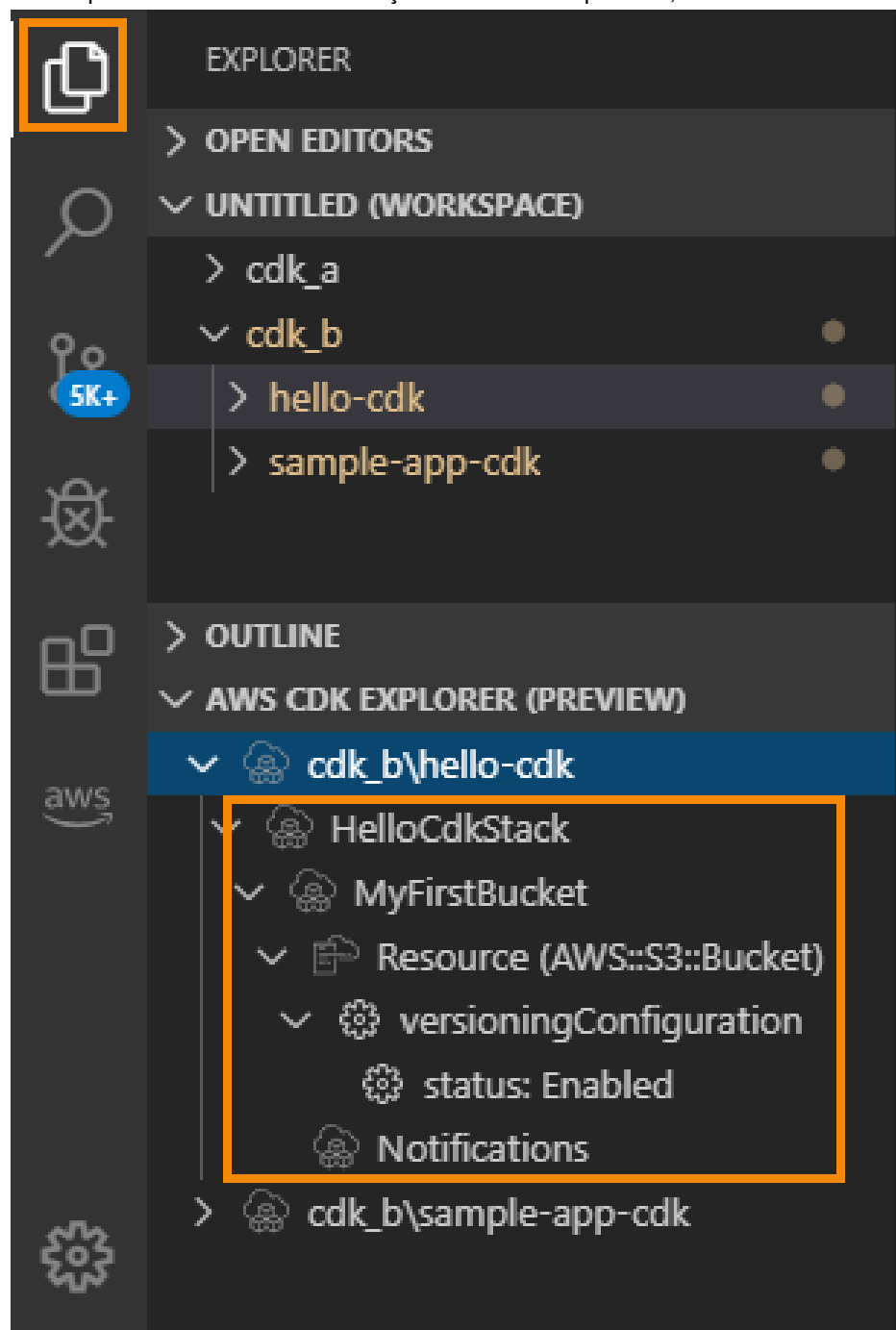
A versão do AWS CDK deve ser 1.17.0 ou posterior. Use **cdk --version** na linha de comando para ver qual versão está sendo executada.

Visualizar um AWS CDK Aplicativo

Usar o AWS CDK Explorer, você pode criar um aplicativo CDK ou carregar um existente. Em seguida, você pode visualizar o [Pilhas do recursos](#) que são definidos nos constructos CDK desse aplicativo.

Execute as primeiras etapas do [Tutorial Hello World](#) no Guia do desenvolvedor do AWS CDK. Pare quando chegar na etapa chamada Implantar a pilha. Você pode executar os comandos fornecidos no tutorial, por exemplo, **mkdir** **cdk** **init**, em uma linha de comando do sistema operacional ou em um Terminal dentro do editor de código VS.

Depois de concluir essa primeira parte do tutorial CDK, carregue a pasta resultante no editor de código do VS, se ainda não o tiver feito. Na parte inferior do Explorador de Código barra lateral encontrar AWS CDK Explorer abra o modo de exibição de árvore do aplicativo, conforme exibido.



O modo de exibição de árvore oferece uma representação visual das informações no `tree.json` do aplicativo CDK, que é criado ao executar o `cdk synth` comando. O arquivo está localizado no diretório `cdk.out` do aplicativo.

Observações importantes

- Ao carregar aplicativos CDK no editor de código do VS, você pode carregar várias pastas de uma só vez. Cada pasta pode conter vários aplicativos CDK, conforme mostrado na imagem anterior. O AWS CDK Explorer localiza aplicativos no diretório raiz do projeto e 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 os aspectos de um aplicativo CDK, por exemplo, adicionar mais recursos, será necessário executar esse comando novamente para ver as alterações refletidas no modo de exibição de árvore.

Executar outras operações em um AWS CDK App

Você pode usar o editor de código do VS para realizar outras operações em um aplicativo CDK, como faria ao usar a linha de comando do sistema operacional ou outras ferramentas. Por exemplo, você pode atualizar os arquivos de código no editor e implantar o aplicativo usando um VS Code Terminal Janela.

Para experimentar esses tipos de ações, use o editor de VS Code para continuar o [Tutorial Hello World](#) no AWS CDK Guia do desenvolvedor. Certifique-se de executar o último passo, **Destruindo os recursos do aplicativo**, para que você não incorra em custos inesperados para seu AWS conta.

Trabalhar com aplicativos sem servidor

O AWS Toolkit for Visual Studio Code oferece suporte para [aplicativos sem servidor](#). Usando o Toolkit for VS Code, você pode criar aplicativos sem servidor que contenham [AWS Lambda](#), depois, implantar os aplicativos em um [AWS CloudFormation](#) pilha.

Tópicos

- [Pressupostos e pré-requisitos](#) (p. 63)
- [Permissões do IAM para aplicativos sem servidor](#) (p. 64)
- [Criando um novo aplicativo sem servidor \(local\)](#) (p. 64)
- [Abrindo um aplicativo sem servidor \(local\)](#) (p. 65)
- [Executando e depurando um aplicativo sem servidor a partir do modelo \(local\)](#) (p. 66)
- [Implantar um aplicativo sem servidor no AWS Nuvem](#) (p. 67)
- [Excluir um aplicativo sem servidor do AWS Nuvem](#) (p. 68)
- [Executando e depurando funções do Lambda diretamente do código](#) (p. 68)
- [Executando e depurando recursos locais do Amazon API Gateway](#) (p. 71)
- [Opções de configuração para depurar aplicativos sem servidor](#) (p. 73)
- [Solução de problemas de aplicativos sem servidor](#) (p. 78)

Pressupostos e pré-requisitos

- Certifique-se de que o sistema atende aos pré-requisitos necessários especificados em [Instalar o AWS Toolkit for Visual Studio Code](#) (p. 2).
- Instalar a [AWS Serverless Application Model \(AWS SAM\)](#) Interface de linha de comando (CLI) e seus pré-requisitos. Consulte [AWS SAM CLI nos pré-requisitos de configuração](#) (p. 2). Se o Visual Studio Code estiver aberto quando você executar essas instalações, talvez seja necessário fechar e reabrir o editor.

- Identifique o padrãoAWSRegião em seuAWSArquivo de configuração. Para obter mais informações, consulte[Configurações de arquivos de configuração e credenciais](#)noAWS Command Line InterfaceGuia do usuário do. Se o Visual Studio Code estiver aberto quando você atualizar o arquivo de configuração, talvez seja necessário fechar e reabrir o editor.
- Depois de instalar o SDK do seu idioma, certifique-se de [configurar sua cadeia de ferramentas](#) (p. 15).
- Para garantir que você possa acessar oCodeLensrecurso noAWS SAMarquivos de modelo, instale o[Suporte a linguagem YAMLE](#)Extensão VS Code.

Permissões do IAM para aplicativos sem servidor

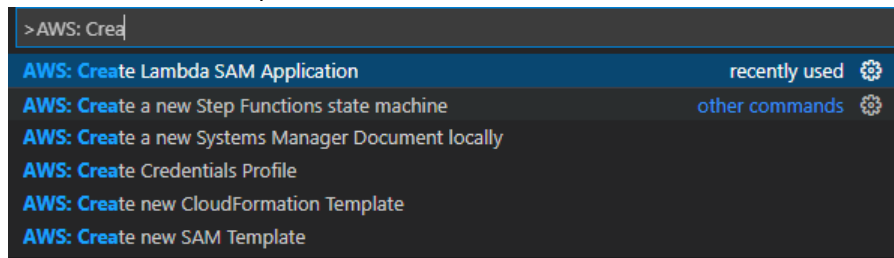
No Toolkit for VS Code, você deve ter um perfil de credenciais que contenha oAWS Identity and Access ManagementPermissões (IAM) necessárias para implantar e executar aplicativos sem servidor. Você deve ter acesso de leitura/gravação adequado aos seguintes serviços: AWS CloudFormation, IAM, Lambda, Amazon API Gateway, Amazon Simple Storage Service (Amazon S3) e Amazon Elastic Container Registry (Amazon ECR).

Para obter informações sobre como garantir que você tem as permissões necessárias para implantar e executar aplicativos sem servidor, consulte[Permissões](#)noAWS Serverless Application ModelGuia do desenvolvedor. Para obter informações sobre como definir seu perfil de credenciais, consulte[Configuração doAWSScredenciais](#) (p. 5).

Criando um novo aplicativo sem servidor (local)

Este procedimento mostra como criar um aplicativo sem servidor com o Toolkit for VS Code Code usando oAWS SAM. A saída desse procedimento é um diretório local no host de desenvolvimento que contém um aplicativo sem servidor de amostra, que você pode criar, testar localmente, modificar e implantar naAWSNuvem.

1. Para abrir oPaleta de comandos do, escolhaExibir,Paleta de comandos do, e, depois, insiraAWS.
2. SelecioneAWS: Criar aplicativo Lambda SAM.



Note

Se oAWS SAM CLI não está instalada, você recebe um erro no canto inferior direito do editor de VS Code. Se isso acontecer, verifique se você conheceu todos os[pressupostos e pré-requisitos](#) (p. 63).

3. Escolha o tempo de execução doAWS SAMaplicativo.

Note

Se você selecionar um dos tempos de execução com "(Imagem)", seu aplicativo será do tipo de pacoteImage. Se você selecionar um dos tempos de execução sem "(Imagem)", seu aplicativo será o tipozip. Para obter mais informações sobre a diferença entreImageezip tipos de pacotes, consulte[Pacotes de implantação do Lambda](#)noAWS LambdaGuia do desenvolvedor.

4. Dependendo do tempo de execução selecionado, você pode ser solicitado a selecionar um gerenciador de dependências e uma arquitetura de tempo de execução para seu aplicativo SAM.

Dependency Manager

Escolha entre Gradle ou Maven.

Note

Essa opção de ferramentas de automação de compilação está disponível apenas para tempos de execução Java.

Architecture

Escolha entre x86_64 ou arm64.

A opção de executar seu aplicativo sem servidor em um ambiente emulado baseado em ARM64, em vez do ambiente padrão baseado em x86_64, está disponível para os seguintes tempos de execução:

- nodejs12.x (ZIP e imagem)
- nodejs14.x (ZIP e imagem)
- python3.8 (ZIP e imagem)
- python3.9 (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 AWS CLI versão 1.33.0 ou posterior para permitir que aplicativos sejam executados em ambientes baseados em ARM64. Para obter mais informações, consulte [Prerequisites \(p. 2\)](#).

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 Digite, o Toolkit for VS Code leva alguns instantes para criar o projeto.

Quando o projeto é criado, seu aplicativo é adicionado ao workspace atual. Você o verá listado na janela do Explorer.

Abrindo um aplicativo sem servidor (local)

Para abrir um aplicativo sem servidor no host de desenvolvimento local, abra a pasta que contém o arquivo de modelo do aplicativo.

1. From the Arquivo, escolha Abrir pasta....
2. No Abrir pasta Caixa de diálogo, navegue até a pasta do aplicativo sem servidor que você deseja abrir.
3. Selecione o Selecionar pasta.

Quando você abre a pasta de um aplicativo, ela é adicionada à Explorer janela.

Executando e depurando um aplicativo sem servidor a partir do modelo (local)

Você pode usar o Toolkit for VS Code para configurar como depurar aplicativos sem servidor e executá-los localmente em seu ambiente de desenvolvimento.

Você começa a configurar o comportamento de depuração usando o VS Code [CodeLens](#) recurso para identificar uma função do Lambda elegível. O CodeLens permite interações com reconhecimento de conteúdo com seu código-fonte. Para obter informações sobre como garantir que você possa acessar o recurso CodeLens, consulte o [Pressupostos e pré-requisitos \(p. 63\)](#) Seção do anterior neste tópico.

Note

Neste exemplo, você depura um aplicativo que usa JavaScript. No entanto, você pode usar os recursos de depuração do Toolkit for VS Code com os seguintes idiomas e tempos de execução:

- 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
- Java — 8, 8.al2, 11
- Go — 1.x

Sua escolha de idioma também afeta a forma como o CodeLens detecta manipuladores Lambda qualificados. Para obter mais informações, consulte [Executando e depurando funções do Lambda diretamente do código \(p. 68\)](#).

Neste procedimento, você usará o aplicativo de exemplo criado na [Criando um novo aplicativo sem servidor \(local\) \(p. 64\)](#) No início deste tópico.

1. Para visualizar os arquivos do aplicativo no File Explorer do VS Code, escolha **Exibir, Explorer**.
2. Na pasta do aplicativo (por exemplo, `my-samp-app`), abra o `template.yaml` file.

Note

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

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

No indicador CodeLens para este recurso, escolha **Adicionar configuração de depuração**.

4. No **Paleta de comandos** do, selecione o tempo de execução em que seu AWS SAMo aplicativo será executado.
5. No editor para o `launch.json` arquivo, edite ou confirme valores para as seguintes propriedades de configuração:
 - `"name"` — Insira um nome amigável para o leitor para aparecer na **Configuração** campo suspenso no **Execução** do **Visualizar**.
 - `"target"` — Verifique se o valor é `"template"` Para que o AWS SAM Modelo é o ponto de entrada da sessão de depuração.
 - `"templatePath"` — Insira um caminho relativo ou absoluto para o `template.yaml` file.
 - `"logicalId"` — Certifique-se de que o nome corresponda ao especificado na `Resources` Seção do do AWS SAM Modelo. Neste caso, é o `HelloWorldFunction` do tipo `AWS::Serverless::Function`.

Para obter mais informações sobre essas e outras entradas no `launch.json` arquivo, consulte [Opções de configuração para depurar aplicativos sem servidor \(p. 73\)](#).

6. Se você estiver satisfeito com sua configuração de depuração, salve `launch.json`. Em seguida, para iniciar a depuração, escolha o botão verde “play” no `EXECUTAR` Visualizar.

Quando as sessões de depuração são iniciadas, o `CONSOLE DE DEPURAÇÃO` mostra a saída de depuração e exibe todos os valores retornados pela função do Lambda. (Ao depuração `AWS SAM` aplicativos, o `AWSToolkit` é selecionado como o `ResultadoChannel` no `ResultadoPainel`.)

Implantar um aplicativo sem servidor noAWSNuvem

Este exemplo mostra como implantar o aplicativo sem servidor criado na seção anterior ([Criando um novo aplicativo sem servidor \(local\) \(p. 64\)](#)) para o `AWSNuvem` usando o Toolkit for VS Code.

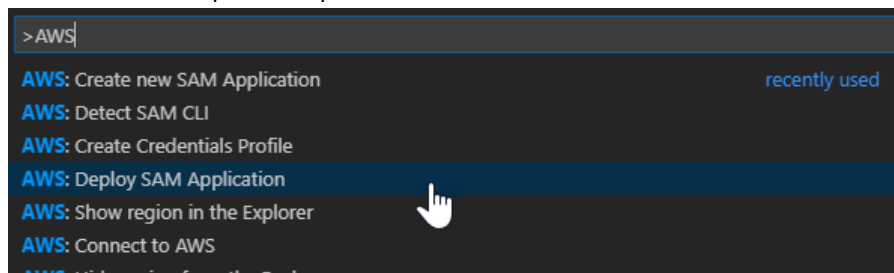
Antes de executar uma implantação, verifique se os seguintes itens estão cobertos:

- Para aplicativos com tipo de implantação `zip`, certifique-se de ter um nome de bucket do Amazon S3 exclusivo globalmente para usar para a implantação.
- Para aplicativos com tipo de implantação `Image`, certifique-se de ter um nome de bucket do Amazon S3 exclusivo globalmente e um URI de repositório do Amazon ECR para usar na implantação.

Para obter mais informações sobre os tipos de pacotes do Lambda, consulte [Pacotes de implantação do Lambda](#) no `AWS Lambda` Guia do desenvolvedor.

Para implantar um aplicativo sem servidor, siga estas etapas:

1. Para abrir o `Paleta de comandos` do, escolha `Exibir, Paleta de comandos` do, e, depois, insira `AWS`.
2. Selecione `AWS: Implane o aplicativo SAM`.



3. Escolha o arquivo `template.yaml` a ser usado para a implantação.
4. Selecione o `AWS` Região na qual será implantado.
5. Insira o nome de um bucket do Amazon S3 que essa implantação pode usar. O bucket deve estar na região em que você está implantando.

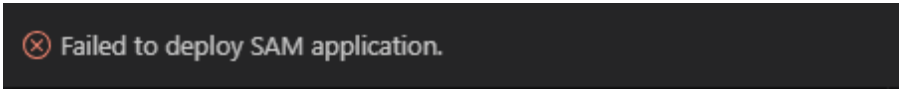
Warning

O nome do bucket do Amazon S3 deve ser globalmente exclusivo em todos os nomes de bucket existentes no Amazon S3. Portanto, você deve adicionar um identificador exclusivo ao nome fornecido no exemplo a seguir ou escolher um nome diferente.

6. Se o aplicativo 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 em que você está implantando.
7. Insira um nome, novo ou existente, para a pilha implantada.

8. Verifique o êxito da implantação noOUTPUTguia do VS Code.

Se ocorrer um erro, uma mensagem semelhante à seguinte será exibida na parte inferior direita:



Se isso acontecer, verifique o texto na guia OUTPUT para obter detalhes. Veja a seguir um exemplo de detalhes de erro:

```
Error with child process: Unable to upload artifact HelloWorldFunction referenced by
CodeUri parameter of HelloWorldFunction resource.
S3 Bucket does not exist. Execute the command to create a new bucket
aws s3 mb s3://pbart-my-sam-app-bucket
An error occurred while deploying a SAM Application. Check the logs for more
information by running the "View AWS Toolkit Logs" command from the Command Palette.
```

Neste exemplo, o erro ocorreu porque o bucket do Amazon S3 não existia.

Se ocorrer um erro, também é possível visualizar o AWS Toolkit Registros usando a Paleta de comandos. Para fazer isso, vá para oExibirMenu, selecionePaleta de comandos do, insiraAWS, e escolhaAWS: Exibir AWS Toolkit Logs.

Quando a implantação estiver concluída, você verá o aplicativo listado naAWSExplorer. Para saber como invocar a função do Lambda criada como parte do aplicativo, consulte[Interação com funções remotas do Lambda \(p. 36\)](#).

Excluir um aplicativo sem servidor doAWSNuvem

A exclusão de um aplicativo sem servidor envolve a exclusão doAWS CloudFormationpilha que você implantou anteriormente noAWSNuvem. Observe que esse procedimento não exclui o diretório do aplicativo do host local.

1. Abra a [AWS Explorer \(p. 22\)](#).
2. NoAWS: Explorer, expanda a Região que contém o aplicativo implantado que você deseja excluir e, em seguida, expandaAWS CloudFormation.
3. Abra o menu de contexto (clique com o botão direito do mouse) para o nome doAWS CloudFormationPilha que corresponde ao aplicativo sem servidor que você deseja excluir e, em seguida, escolhaExcluirAWS CloudFormationStack.
4. Para confirmar que você deseja excluir a pilha selecionada, escolhaYes (Sim).

Se a exclusão de pilha for bem-sucedida, o Toolkit for VS Code Code removerá o nome da pilha doAWS CloudFormationLista noAWSExplorer.

Executando e depurando funções do Lambda diretamente do código

Ao testar oAWS SAMaplicativo, você pode optar por executar e depurar apenas a função do Lambda e excluir outros recursos que oAWS SAMDefine o modelo. Essa abordagem envolve o uso doCodeLenspara identificar manipuladores de função Lambda no código-fonte que você pode invocar diretamente.

Os manipuladores do Lambda detectados pelo CodeLens dependem do idioma e do tempo de execução que você está usando para seu aplicativo.

Idioma/tempo de execução	CrITÉrios para funções do Lambda a serem identificadas pelos indicadores CodeLens
C# (dotnetcore2.1, 3.1; .NET 5.0)	<p>A função tem os seguintes recursos:</p> <ul style="list-style-type: none"> • É uma função pública de uma classe pública. • Ele tem um ou dois parâmetros. Com dois parâmetros, o segundo parâmetro deve implementar o <code>ILambdaContextInterface</code>. • Tem um <code>*.csproj</code> arquivo em sua pasta pai dentro da pasta do espaço de trabalho VS Code. <p>O extensão ms-dotnettools.csharp (ou qualquer extensão que forneça símbolos de idioma para C#) está instalada e ativada.</p>
JavaScript/TypeScript (Node.js 12.x, 14.x)	<p>A função tem os seguintes recursos:</p> <ul style="list-style-type: none"> • É uma função exportada com até três parâmetros. • Tem um <code>package.json</code> arquivo em sua pasta pai dentro da pasta do espaço de trabalho VS Code.
Python (3.6, 3.7, 3.8, 3.9)	<p>A função tem os seguintes recursos:</p> <ul style="list-style-type: none"> • É uma função de nível superior. • Tem um <code>requirements.txt</code> arquivo em sua pasta pai dentro da pasta do espaço de trabalho VS Code. <p>O extensão ms-python.python (ou qualquer extensão que forneça símbolos de idioma para Python) está instalada e ativada.</p>
Java (8, 8.al2, 11)	<p>A função tem os seguintes recursos:</p> <ul style="list-style-type: none"> • É uma função pública de uma classe pública, não abstrata. • Ele 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 um <code>java.io.InputStream</code> e um <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 um <code>java.io.InputStream</code>, um <code>java.io.OutputStream</code> e o último parâmetro deve ser <code>com.amazonaws.services.lambda.runtime.Context</code>.

Idioma/tempo de execução	Critérios para funções do Lambda a serem identificadas pelos indicadores CodeLens
	<ul style="list-style-type: none"> • Tem <code>umbuild.gradle</code>(Gradle) ou <code>pom.xml</code>(Maven) em sua pasta pai dentro da pasta do espaço de trabalho VS Code. <p><code>Oredhat.java</code>(ou qualquer extensão que forneça símbolos de linguagem para Java) está instalada e ativada. Essa extensão requer o Java 11, independentemente do tempo de execução do Java que você esteja usando.</p> <p>O <code>vscjava.vscode-java-debug</code> extensão (ou qualquer extensão que forneça um depurador Java) está instalada e ativada.</p>
lr (1.x)	<p>A função tem os seguintes recursos:</p> <ul style="list-style-type: none"> • É uma função de nível superior. • É preciso de 0 a 2 argumentos. Se houver dois argumentos, o primeiro argumento deverá implementar <code>context.Context</code>. • Ele retorna de 0 a 2 argumentos. Se houver mais de 0 argumentos, o último argumento deve implementar <code>error</code>. • Tem um <code>go.mod</code> arquivo dentro da pasta do espaço de trabalho VS Code. <p>O <code>extensão golang.go</code> está instalado, configurado e ativado.</p>

Para executar e depurar um aplicativo sem servidor diretamente do código do aplicativo

1. Para visualizar os arquivos do aplicativo no VS Code File Explorer, escolha **Exibir, Explorer**.
2. Na pasta do aplicativo (por exemplo, `my-samp-app`), expanda a pasta de funções (neste caso, `hello-world`) e abra `oapp.jsfile`.
3. No indicador CodeLens que identifica um manipulador de função Lambda elegível, escolha **Add Debug Configuration**.
4. Na Paleta de comandos do, selecione o tempo de execução em que seu AWS SAMo aplicativo será executado.
5. No editor para o `launch.json` arquivo, edite ou confirme valores para as seguintes propriedades de configuração:
 - `"name"`— Insira um nome amigável para o leitor para aparecer na Configuração campo suspenso no Execução do Visualizar.
 - `"target"`— Verifique se o valor é `"code"` para que um manipulador de funções do Lambda seja chamado diretamente.
 - `"lambdaHandler"`— Insira o nome do método em seu código que o Lambda chama para invocar sua função. Por exemplo, para aplicativos em JavaScript, o padrão é `app.lambdaHandler`.
 - `"projectRoot"`— Insira o caminho para o arquivo do aplicativo 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 do evento que você deseja fornecer à sua função do Lambda como entrada:
 - "json" Pares de chave/valor formatados em JSON que definem a carga do evento.
 - "path": Um caminho para o arquivo usado como carga útil do evento.

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

Para obter mais informações sobre essas e outras entradas no `launch.json` arquivo, consulte [Opções de configuração para depurar aplicativos sem servidor \(p. 73\)](#).

6.

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

Quando as sessões de depuração são iniciadas, o CONSOLE DE DEPURAÇÃO mostra a saída de depuração e exibe todos os valores que a função do Lambda retorna. (Ao depuração AWS SAM aplicativos, AWSToolkit é selecionado como o resultado do channel no Resultado Painel.)

Executando e depurando recursos locais do Amazon API Gateway

Você pode executar ou depurar AWS SAM Recursos locais do API Gateway, especificados em `template.yaml`, executando uma configuração de lançamento do VS Code de `type=aws-sam` com `invokeTarget.target=api`.

Note

O API Gateway oferece suporte a dois tipos de APIs, REST e HTTP. No entanto, o recurso API Gateway com o AWS Toolkit for Visual Studio Code só é compatível com APIs REST. Às vezes, as APIs HTTP são chamadas de "APIs do API Gateway V2".

Para executar e depurar recursos locais do API Gateway

1. Escolha uma das seguintes abordagens para criar uma configuração de execução para um AWS SAM Recurso API Gateway:
 - Opção 1: Visite o código-fonte do manipulador (arquivo.js, .cs ou .py) em seu AWS SAM projeto, passe o mouse sobre o manipulador Lambda e escolha o Adicionar configuração de depuração CodeLens. Em seguida, no menu, escolha o item marcado Evento da API.
 - Opção 2: Edite `launch.json` e crie uma nova configuração de execução usando a seguinte sintaxe.

```
{
  "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 VS Code, escolha a configuração de inicialização (nomeado `myConfig` no exemplo acima).
3. (Opcional) Adicione pontos de interrupção ao código do projeto do Lambda.
4. Clique no botão `Play` ou escolha `Execute` no painel.
5. No painel de saída, visualize os resultados.

Configuration

Quando você usa `invokeTarget.target` da propriedade `api`, o Toolkit altera a validação e o comportamento da configuração de inicialização para suportar um `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 execução solicita, por exemplo, `"path": "/hello"`.

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

httpMethod

Um dos seguintes verbos: "delete", "get", "head", "options", "patch", "post", "put".

payload

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

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

`payload.json` especifica uma carga útil JSON em linha.

headers

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

```
"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 o `queryString` da solicitação, por exemplo, "queryString": "abc=def&ghi=jkl".

aws

Como as informações de conexão são fornecidas. Para obter mais informações, consulte o `.AWS` propriedades de conexão ("aws") Tabela no [Opções de configuração para depurar aplicativos sem servidor](#) (p. 73) seção.

sam

Como o AWS SAM CLI constrói o aplicativo. Para obter mais informações, consulte o `.AWS` propriedades do SAM CLI ("sam") Tabela no [Opções de configuração para depurar aplicativos sem servidor](#) (p. 73) seção.

Opções de configuração para depurar aplicativos sem servidor

Quando você abre `olaunch.json` para editar configurações de depuração, você pode usar o VS Code [IntelliSense](#) recurso para visualizar e concluir automaticamente propriedades válidas. Para acionar o IntelliSense no editor, pressione `Ctrl+Barra de espaço`.

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

O IntelliSense permite localizar e definir propriedades para invocar funções do Lambda diretamente ou com o AWS SAM Modelo. Você também pode definir propriedades para "lambda" (como a função é executada), "sam" (Como o AWS SAM CLI constrói o aplicativo) e "aws" (Como as informações de conexão são fornecidas).

AWS SAM: Handler direto do Lambda invocação/Chamada 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 o AWS SAM CLI para construir e depurar localmente.
name	Especifica um nome amigável para o leitor a ser exibido na Depuração de configuração de execução Lista.
request	Especifica o tipo de configuração a ser executado pela extensão designada (<code>aws-sam</code>). Sempre definido como <code>direct-invoke</code> para iniciar a função do Lambda.
invokeTarget	<p>Especifica o ponto de entrada para chamar o recurso.</p> <p>Para invocar a função do Lambda diretamente, defina valores para o seguinte <code>invokeTarget</code> Campos de :</p> <ul style="list-style-type: none">• <code>target</code>— Definido como <code>code</code>.• <code>lambdaHandler</code>— O nome do manipulador da função do Lambda a ser invocado.• <code>projectRoot</code>— O caminho para o arquivo do aplicativo que contém o manipulador de funções do Lambda.• <code>architecture</code>— Arquitetura do processador do ambiente emulado no qual o aplicativo SAM Lambda local é executado. Para determinados tempos de execução, você pode escolher <code>arm64</code> em vez do padrão <code>x86_64</code> arquitetura. Para obter mais informações, consulte Criando um novo aplicativo sem servidor (local) (p. 64). <p>Por invocar os recursos do Lambda com o AWS SAM Modelo, defina valores para o seguinte <code>invokeTarget</code> Campos de :</p> <ul style="list-style-type: none">• <code>target</code>— Definido como <code>template</code>.• <code>templatePath</code>— O caminho para o AWS SAM arquivo de modelo do.

Propriedade	Descrição
	<ul style="list-style-type: none"> <code>logicalId</code>— O nome do recurso <code>doAWS::Lambda::Function</code> ou <code>doAWS::Serverless::Function</code> para invocar. Você pode encontrar o nome do recurso no formato <code>YAML</code> <code>AWS SAM</code> Modelo. Observe que o AWS Toolkit reconhece implicitamente funções definidas com <code>PackageType: Image</code> no <code>AWS SAM</code> Template como <code>Baseado na imagem</code> Funções do Lambda. Para obter mais informações, consulte Pacotes de implantação do Lambda no <code>AWS Lambda</code> Guia do desenvolvedor.

Lambda ("**lambda**") propriedades

Propriedade	Descrição
<code>environmentVariables</code>	<p>Passa parâmetros operacionais para sua função do Lambda. Por exemplo, se estiver gravando em um bucket do Amazon S3, em vez de fixar no código o nome do bucket em que você está gravando, configure o nome do bucket como uma variável de ambiente.</p> <p>Note</p> <p>Ao especificar variáveis de ambiente para um aplicativo sem servidor, você deve adicionar configurações a ambos os <code>AWS SAM</code> Template <code>Template.yaml</code> e <code>launch.json</code> file.</p> <p>Exemplo de formatação para uma variável de ambiente no <code>AWS SAM</code> Modelo:</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 <code>launch.json</code> file:</p> <pre>"environmentVariables": { "SAMPLE1": "My sample 1 value" }</pre>
<code>payload</code>	<p>Fornece duas opções para a carga de eventos que você fornece à sua função do Lambda como entrada.</p> <ul style="list-style-type: none"> <code>"json"</code> Pares de chave/valor formatados em JSON que definem a carga do evento. <code>"path"</code>: Um caminho para o arquivo usado como carga útil do evento.
<code>memoryMB</code>	<p>Especifica megabytes (MB) de memória fornecida para executar uma função Lambda chamada.</p>

Propriedade	Descrição
<code>runtime</code>	Especifica o tempo de execução que a função Lambda usa. Para obter mais informações, consulte AWS Lambda Tempos de execução do .
<code>timeoutSec</code>	Define o tempo permitido, em segundos, antes da sessão de depuração expirar.
<code>pathMappings</code>	<p>Especifica onde o código local está em relação ao local em que ele é executado no contêiner.</p> <p>Por padrão, o Toolkit for VS Code define <code>localRoot</code> para a raiz de código da função Lambda no espaço de trabalho local e <code>remoteRoot</code> para <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 <code>workingDirectory</code> parâmetro no AWS CloudFormation Arquivo de modelo, pelo menos um <code>pathMapping</code> deve 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 <code>launch.json</code> file:</p> <pre>"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, o <code>remoteRoot</code> entrada deve ser o diretório de compilação. Para funções Lambda baseadas em Node.JS, você pode especificar apenas uma única entrada de mapeamento de caminho.

O Toolkit for VS Code usa o AWS SAM CLI para criar e depurar aplicativos sem servidor localmente. Você pode configurar o comportamento do AWS SAM Comandos da CLI usando propriedades do "sam" configuração no `launch.json` file.

AWS SAM CLI ("sam") propriedades

Propriedade	Descrição	Valor padrão
<code>buildArguments</code>	Configura como o <code>sam build</code> comando constrói seu código-fonte do Lambda. Para visualizar as opções de compilação, consulte sam build no AWS Serverless Application Model Guia do desenvolvedor.	String vazia

Propriedade	Descrição	Valor padrão
containerBuild	Indica se deseja construir sua função dentro de um contêiner Docker semelhante ao Lambda.	false
dockerNetwork	Especifica o nome ou ID de uma rede de Docker existente que o LambdaOs contêineres do Docker do devem se conectar com a rede de ponte padrão. Se não for especificado, os contêineres do Lambda do se conectam somente à rede de Docker de ponte padrão.	String vazia
localArguments	Especifica argumentos de chamada local adicionais.	String vazia
skipNewImageCheck	Especifica se o comando deve ignorar a imagem mais recente do Docker para o tempo de execução do Lambda.	false
template	Personalizar oAWS SAMmodelo usando parâmetros para inserir valores do cliente. Para obter mais informações, consulte Parâmetros no Guia do usuário do AWS CloudFormation .	"parameters":{}

AWSconexão ("aws") propriedades

Propriedade	Descrição	Valor padrão
credentials	Seleciona um perfil específico (por exemplo,profile:default) do seu arquivo de credencial para obterAWSCredenciais da .	OAWSCredenciais que suas existentes CompartilhadoAWSarquivo de configuração compartilhadoAWSarquivo de credenciais (p. 5) Forneça ao Toolkit for VS Code.
region	Define oAWSRegião do serviço (por exemplo, us-east-1).	O valorAWSRegião associada ao perfil de credenciais ativas.

Exemplo: Configuração de execução do modelo

Veja um arquivo de configuração de execução de umAWS SAMDestino do modelo:

```
{
  "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 execução de código

Aqui está 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": {}  
      }  
    }  
  ]  
}
```

Solução de problemas de aplicativos sem servidor

Este tópico detalha os erros comuns que você pode encontrar ao criar aplicativos sem servidor com o Toolkit for VS Code e como resolvê-los.

Tópicos

- [Como posso usar um samconfig.toml com uma configuração de inicialização do SAM?](#) (p. 78)
- [ERROR: "RuntimeError: O contêiner não existe"](#) (p. 79)
- [Erro: "Docker.errors.apiError: 500 Server Error... Você atingiu seu limite de taxa de pull."](#) (p. 79)
- [ERROR: "Erro do servidor 500: Montagem C:\Users\..."](#) (p. 79)
- [Usando WSL, visualizações da Web \(por exemplo, o "Chamar emAWS"form\) estão quebrados](#) (p. 79)
- [Depurando um aplicativo TypeScript, mas os pontos de interrupção não estão funcionando](#) (p. 79)

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

Especifique a localização da sua CLI SAM [samconfig.toml](#) configurando o `--config-file` argumento nos `sam.localArgumentsPropriedade` da sua configuração de execução. Por exemplo, se o arquivo `samconfig.toml` estiver localizado no nível superior do espaço de trabalho:

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

ERROR: “RuntimeError: O contêiner não existe”

Osam buildpode mostrar esse erro se o sistema não tiver espaço em disco suficiente para o contêiner do Docker. Se o armazenamento do sistema tiver apenas 1-2 GB de espaço disponível,sam buildpode 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... Você atingiu seu limite de taxa de pull.”

O Docker Hub limita as solicitações que usuários anônimos podem fazer. Se o sistema atingir o limite, o Docker falhará e esse erro aparecerá na visualização OUTPUT 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")
```

Certifique-se de que seuSistema DockerFoi autenticado com suas credenciais do Docker Hub.

ERROR: “Erro do servidor 500: Montagem C:\Users\...”

Os usuários do Windows podem ver esse erro de montagem do Docker ao depurarAWS SAMaplicações:

```
Fetching lambci/lambda:nodejs10.x Docker container image.....  
2019-07-12 13:36:58 Mounting C:\Users\<username>\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 ...
```

Tente atualizar as credenciais para seus drives compartilhados (nas configurações do Docker).

Usando WSL, visualizações da Web (por exemplo, o “Chamar emAWS“form) estão quebrados

Este 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 em[este problema de rastreamento WSL](#).

Depurando um aplicativo TypeScript, mas os pontos de interrupção não estão funcionando

Isso acontecerá se não houver um mapa de origem para vincular o arquivo JavaScript compilado ao arquivo TypeScript de origem. Para corrigir isso, abra suatsconfig.js e certifique-se de que a seguinte opção e valor estejam definidos:"inlineSourceMap": true.

Segurança do AWS Toolkit for VS Code

A segurança da nuvem na Amazon Web Services (AWS) é a nossa maior prioridade. Como cliente da AWS, você se contará com um datacenter e uma arquitetura de rede criados para atender aos requisitos das organizações com as maiores exigências de segurança. A segurança é uma responsabilidade compartilhada entre a AWS e você. O [modelo de responsabilidade compartilhada](#) descreve isso como a Segurança da nuvem e a Segurança na nuvem.

Segurança da nuvem - A AWS é responsável pela proteção da infraestrutura que executa todos os serviços oferecidos na Nuvem AWS e por fornecer serviços que você pode usar com segurança. Nossa responsabilidade de segurança é a maior prioridade na AWS, e a eficácia da nossa segurança é regularmente testada e verificada por auditores terceirizados como parte dos [Programas de Compatibilidade da AWS](#).

Segurança na nuvem: sua responsabilidade é determinada pelo produto da AWS que você está usando e por outros fatores, incluindo a confidencialidade dos dados, os requisitos da organização e as leis e regulamentos aplicáveis.

Esse AWS produto ou serviço segue o [Modelo de responsabilidade compartilhada](#). Por meio da Amazon Web Services específica (AWS) serviços que ele suporta. Para o AWS informações de segurança do serviço, consulte o [AWS Página da documentação de segurança do serviço](#) e [AWS serviços que estão no escopo de AWS esforços de conformidade por programa de conformidade](#).

Tópicos

- [Proteção de dados no AWS Toolkit for VS Code](#) (p. 80)
- [Identity and Access Management](#) para isso AWS Produto ou serviço (p. 81)
- [Registro em log e monitoramento no AWS Toolkit for VS Code](#) (p. 81)
- [Validação de conformidade para isso AWS Produto ou serviço](#) (p. 82)
- [Resiliência para isso AWS Produto ou serviço](#) (p. 82)
- [Segurança de infraestrutura para isso AWS Produto ou serviço](#) (p. 83)
- [Análise de configuração e vulnerabilidade no AWS Toolkit for VS Code](#) (p. 83)

Proteção de dados no AWS Toolkit for VS Code

O AWS [Modelo de responsabilidade compartilhada](#) aplica-se à proteção de dados no AWS Toolkit for VS Code. Conforme descrito nesse modelo, a AWS é responsável por proteger a infraestrutura global que executa toda a Nuvem AWS. Você é responsável por manter o controle sobre seu conteúdo hospedado nessa infraestrutura. Esse conteúdo inclui as tarefas de configuração e gerenciamento de segurança dos serviços da AWS que você usa. Para obter mais informações sobre a privacidade de dados, consulte as [Perguntas frequentes sobre privacidade de dados](#). Para obter mais informações sobre a proteção de dados na Europa, consulte a postagem do blog [AWS Shared Responsibility Model and GDPR](#) no Blog de segurança da AWS.

Para fins de proteção de dados, recomendamos que você proteja Conta da AWS credenciais e configurar contas de usuário individuais com 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 uma autenticação multifator (MFA) com cada conta.
- Use SSL/TLS para se comunicar com os recursos da AWS. Recomendamos TLS 1.2 ou posterior.
- Configure o registro em log das atividades da API e do usuário com o AWS CloudTrail.
- Use as soluções de criptografia da AWS com todos os controles de segurança padrão nos produtos da AWS.
- Use serviços gerenciados de segurança avançada, como o Amazon Macie, que ajuda a localizar e proteger dados pessoais armazenados no Amazon S3.
- Se forem necessários módulos criptográficos validados pelo FIPS 140-2 ao acessar a AWS por meio de uma interface de linhas de comando ou uma API, use um endpoint do FIPS. Para obter mais informações sobre endpoints do FIPS, consulte [Federal Information Processing Standard \(FIPS\) 140-2](#).

É altamente recomendável nunca colocar informações de identificação confidenciais, como endereços de e-mail dos seus clientes, em marcações ou campos de formato livre, como um campo Name (Nome). Isso inclui quando você trabalha com Toolkit for VS Code ou outros AWS serviços usando o console, a API, AWS CLI, ou AWS SDKs. Quaisquer dados inseridos em marcações ou campos de formato livre usados para nomes podem ser usados para logs de cobrança ou diagnóstico. Se você fornecer um URL para um servidor externo, recomendamos fortemente que não sejam incluídas informações de credenciais no URL para validar a solicitação a esse servidor.

Identity and Access Management para isso AWS Produto ou serviço

O AWS Identity and Access Management (IAM) é um serviço da Amazon Web Services (AWS) que ajuda a controlar o acesso aos recursos da AWS de forma segura. Os administradores do IAM controlam quem pode ser autenticado (assinado) e autorizado (tem permissões) para usar recursos em AWS Serviços da . O IAM é um serviço da AWS que pode ser usado sem custo adicional.

Para usar isso AWS produto ou serviço para acessar AWS, você precisa de um AWS Conta da e AWS Credenciais da . Para aumentar a segurança da conta da AWS, recomendamos usar um usuário do IAM para fornecer credenciais de acesso, em vez de usar as credenciais de sua conta da AWS.

Para obter detalhes sobre como trabalhar com o IAM, consulte [AWS Identity and Access Management](#).

Para obter uma visão geral dos usuários do IAM e por que eles são importantes para a segurança da sua conta, consulte [AWS Credenciais de segurança](#) e [Referência geral da Amazon Web Services](#).

Esse AWS produto ou serviço segue o [Modelo de responsabilidade compartilhada](#) Por meio da Amazon Web Services específica (AWS) serviços que ele suporta. para o AWS Informações de segurança do serviço, consulte o [AWS Página da documentação de segurança do serviço](#) e [AWS serviços que estão no escopo de AWS Esforços de conformidade por programa de conformidade](#).

Registro em log e monitoramento no AWS Toolkit for VS Code

Esse AWS O produto ou serviço fornece status e resultados no OUTPUT guia e o CONSOLE DE DEPURACÃO Guia. Você também pode exibir logs dessa atividade.

Para exibir os logs do Toolkit for VS Code

1. Abra View (Exibir), Command Palette (Paleta de comandos).

2. Tipo "AWS" na caixa de busca e escolha AWS: Exibir AWSLogs do Toolkit.

Validação de conformidade para isso AWS Produto ou serviço

Esse AWS produto ou serviço segue o [Modelo de responsabilidade compartilhada](#) Por meio da Amazon Web Services específica (AWS) serviços que ele suporta. para o AWS Informações de segurança do serviço, consulte o AWS [Página da documentação de segurança do serviço](#) e [AWS serviços que estão no escopo de AWS esforços de conformidade por programa de conformidade](#).

A segurança e a conformidade dos serviços da AWS são avaliadas por auditores externos como parte de vários programas de conformidade da AWS. Esses programas incluem o SOC, o PCI, o FedRAMP, o HIPAA e outros. A AWS fornece uma lista atualizada dos produtos da AWS no escopo de programas de compatibilidade específicos em [Produtos da AWS no escopo pelo programa de compatibilidade](#).

Os relatórios de auditoria de terceiros estão disponíveis para download por meio do AWS Artifact. Para obter mais informações, consulte [Fazer download dos relatórios no AWS Artifact](#).

Para obter mais informações sobre programas de conformidade da AWS, consulte [Programas de conformidade da AWS](#).

Sua responsabilidade de conformidade ao usar isso AWS produto ou serviço para acessar um AWS O serviço é determinado pela confidencialidade de seus dados, pelas metas de conformidade da sua empresa e pelas regulamentações e leis aplicáveis. Se o seu uso da AWS estiver sujeito à conformidade com padrões, como HIPAA, PCI ou FedRAMP, a AWS fornecerá recursos para ajudar:

- [Guias de início rápido de segurança e compatibilidade](#): guias de implantação que discutem as considerações de arquitetura e fornecem etapas para implantação de ambientes de linha de base focados em conformidade e segurança na AWS.
- [Whitepaper Architecting for HIPAA Security and Compliance](#)— Um whitepaper que descreve como as empresas podem usar AWS Para criar aplicativos compatíveis com HIPAA.
- [AWS Recursos de conformidade](#)— uma coleção de manuais e guias que pode ser aplicada ao seu setor e local.
- [AWS Config](#): um serviço que avalia até que ponto suas configurações de recursos estão compatíveis com práticas internas, diretrizes do setor e regulamentações.
- [AWS Security Hub](#): uma visão abrangente do estado da segurança na AWS que ajuda você a conferir a compatibilidade com os padrões e as práticas recomendadas do setor de segurança.

Resiliência para isso AWS Produto ou serviço

A infraestrutura global da Amazon Web Services (AWS) é baseada em regiões e zonas de disponibilidade da AWS.

As regiões da AWS fornecem várias zonas de disponibilidade separadas e isoladas fisicamente, que são conectadas com baixa latência, altas taxas de transferência e redes altamente redundantes.

Com as zonas de disponibilidade, você pode projetar e operar aplicações e bancos de dados que executam o failover automaticamente entre as zonas de disponibilidade, sem interrupção. As zonas de disponibilidade são mais altamente disponíveis, tolerantes a falhas e escaláveis que uma ou várias infraestruturas de data center tradicionais.

Para obter mais informações sobre regiões e zonas de disponibilidade da AWS, consulte [Infraestrutura global da AWS](#).

Esse AWS produto ou serviço segue o [Modelo de responsabilidade compartilhada](#) Por meio da Amazon Web Services específica (AWS) serviços que ele suporta. para o AWS Informações de segurança do serviço, consulte o [AWS Página da documentação de segurança do serviço](#) e [AWS serviços que estão no escopo de AWS esforços de conformidade por programa de conformidade](#).

Segurança de infraestrutura para isso AWS Produto ou serviço

Esse AWS produto ou serviço segue o [Modelo de responsabilidade compartilhada](#) Por meio da Amazon Web Services específica (AWS) serviços que ele suporta. para o AWS Informações de segurança do serviço, consulte o [AWS Página da documentação de segurança do serviço](#) e [AWS serviços que estão no escopo de AWS esforços de conformidade por programa de conformidade](#).

Análise de configuração e vulnerabilidade no AWS Toolkit for VS Code

Esse AWS produto ou serviço é liberado para o [Visual Studio Marketplace](#) À medida que novos recursos ou correções forem desenvolvidos. Essas atualizações às vezes incluem atualizações de segurança, portanto, é importante manter o Toolkit for VS Code atualizado.

Para verificar se as atualizações automáticas para extensões estão habilitadas

1. Abra a janela Settings (Configurações) escolhendo File (Arquivo), Preferences (Preferências), Settings (Configurações).
2. Expanda Features (Recursos) e escolha Extensions (Extensões).
3. Ajuste as configurações do seu ambiente.

Se você optar por desabilitar as atualizações automáticas para extensões, verifique se há atualizações do Toolkit for VS Code em intervalos apropriados para o seu ambiente.

Histórico do documento para o guia do usuário do AWS Toolkit for Visual Studio Code

Última grande atualização da documentação: 14 de outubro de 2021

A tabela a seguir descreve as alterações importantes em cada versão do AWS Toolkit for Visual Studio Code . Para receber notificações sobre atualizações dessa documentação, inscreva-se em um [feed RSS](#).

update-history-change	atualização da descrição do histórico	atualização da data do histórico
Support para recursos experimentais (p. 22)	Adicionado suporte para ativar recursos experimentais para AWS Serviços da .	14 de outubro de 2021
Suporte para AWS recursos (p. 56)	Adicionado suporte para acessar tipos de recursos junto com opções de interface para criar, editar e excluir recursos.	14 de outubro de 2021
Support para ambientes ARM64 (p. 63)	Agora você pode executar aplicativos sem servidor em ambientes emulados baseados em ARM64, bem como em ambientes baseados em x86_64.	1º de outubro de 2021
Aplicação sem servidor da AWS (p. 63)	Adicionado o suporte para execução de AWS SAM aplicativos na plataforma ARM64	30 de setembro de 2021
Suporte a App Runner (p. 24)	Adicionado o suporte para o AWS App Runner ao AWS Toolkit for Visual Studio Code .	11 de agosto de 2021
Funções do Go do (p. 66)	Adicionado suporte para depuração de funções Go locais.	10 de maio de 2021
Depuração de funções Java do (p. 66)	Adicionado suporte para depuração de funções Java locais.	22 de abril de 2021
Suporte a YAML para AWS Step Functions (p. 49)	Adicionado o suporte a YAML para AWS Step Functions.	4 de março de 2021
Depuração de recursos do Amazon API Gateway (p. 63)	Adicionado suporte para depuração de recursos locais do Amazon API Gateway.	1º de dezembro de 2020
Amazon API Gateway (p. 24)	Adicionado o suporte para o Amazon API Gateway.	1º de dezembro de 2020

Aplicação sem servidor da AWS (p. 63)	Adicionado o suporte para imagens de contêiner do Lambda com aplicativos sem servidor.	1º de dezembro de 2020
Suporte do AWS Systems Manager (p. 45)	Adicionado o suporte para documentos do Systems Manager Automation.	30 de setembro de 2020
CloudWatch Logs (p. 31)	Adicionado o suporte ao para o CloudWatch Logs	24 de agosto de 2020
Amazon S3 (p. 41)	Adicionado o suporte para o Amazon S3.	30 de julho de 2020
Suporte do AWS Step Functions (p. 49)	O suporte para AWS Step Functions foi adicionado.	31 de março de 2020
Conteúdo de segurança (p. 80)	Conteúdo de segurança adicionado.	6 de fevereiro de 2020
Trabalhar com esquemas do Amazon EventBridge (p. 34)	Suporte adicionado para esquemas do Amazon EventBridge	1º de dezembro de 2019
AWS CDK Explorer (p. 60)	Versão prévia do recurso do AWS CDK Explorer.	25 de novembro de 2019
Usando um processo de credencial externo (p. 10)	Foram adicionadas informações sobre o uso de um processo credencial externo para obterAWSCredenciais da .	25 de setembro de 2019
Usando o IntelliSense para arquivos de definição de tarefas (p. 60)	O suporte do IntelliSense foi adicionado para ser usado com os arquivos de definição de tarefas do Amazon ECS.	24 de setembro de 2019
Guia do usuário para o AWS Toolkit for Visual Studio Code (p. 1)	Versão de disponibilidade geral.	11 de julho de 2019
Guia do usuário para o AWS Toolkit for Visual Studio Code (p. 1)	Atualizamos a estrutura do documento por motivo de clareza e facilidade de uso.	3 de julho de 2019
Instalar oAWSToolkit for VS Code Code (p. 2)	Adicionamos informações sobre a instalação de SDKs de idioma para oferecer suporte a várias cadeias de ferramentas.	12 de junho de 2019
Configurar a cadeia de ferramentas (p. 15)	Foram adicionadas informações sobre como configurar várias cadeias de ferramentas.	12 de junho de 2019
Versão inicial (p. 84)	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.