



Guida per l'utente

AWS Toolkit per VS Code



AWS Toolkit per VS Code: Guida per l'utente

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

I marchi e l'immagine commerciale di Amazon non possono essere utilizzati in relazione a prodotti o servizi che non siano di Amazon, in una qualsiasi modalità che possa causare confusione tra i clienti o in una qualsiasi modalità che denigri o discrediti Amazon. Tutti gli altri marchi non di proprietà di Amazon sono di proprietà delle rispettive aziende, che possono o meno essere associate, collegate o sponsorizzate da Amazon.

Table of Contents

AWS Toolkit for Visual Studio Code	1
Che cos'è il AWS Toolkit for Visual Studio Code	1
Informazioni correlate	1
Sviluppatore Amazon Q e Amazon CodeWhisperer	2
Scarica il Toolkit	3
Scaricamento del Toolkit da VS Code Marketplace	3
Toolkit IDE aggiuntivi da AWS	3
Nozioni di base	4
Installazione del Toolkit for VS Code	4
Prerequisiti	4
Scaricamento e installazione di AWS Toolkit for Visual Studio Code	4
Prerequisiti opzionali	5
Connessione a AWS	6
Prerequisiti	6
Apertura del pannello di accesso	7
Connessione a AWS dal Toolkit	7
Autenticazione per Amazon CodeCatalyst	8
AWS Regioni che cambiano	9
Aggiungere una regione all' AWS Explorer	9
Nascondi una regione dall' AWS Explorer	10
Configurazione della toolchain	10
Configurare una toolchain per .NET Core	10
Configurazione della toolchain per Node.js	10
Configura una toolchain per Python	11
Configurare una toolchain per Java	11
Configura una toolchain per Go	12
Usare la tua catena di strumenti	12
Autenticazione e accesso	13
IAM Identity Center	13
IAMcredenziali	13
Creare un IAM utente	14
Creazione di un file di credenziali condiviso da AWS Toolkit for Visual Studio Code	15
Aggiungi profili di credenziali aggiuntivi	16
AWS ID del costruttore	17

Utilizzo di un processo di credenziali esterno	17
Lavorare con AWS	18
Caratteristiche sperimentali	19
AWS Explorer	19
Amazon CodeCatalyst	20
Che cos'è Amazon CodeCatalyst?	20
Iniziare con CodeCatalyst	21
Utilizzo delle risorse di CodeCatalyst	21
Lavorare con ambienti di sviluppo	25
Risoluzione dei problemi	28
Amazon API Gateway	29
AWS App Runner	29
Prerequisiti	30
Prezzi	33
Creazione di servizi App Runner	33
Gestione dei servizi App Runner	37
AWSCompositore di applicazioni	39
Lavorare con Application Composer AWS	40
AWS CDK	41
AWS CDK applicazioni	42
AWS CloudFormationStack	44
Eliminazione di unAWS CloudFormationpila	44
Crea un CloudFormation modello	45
AmazonCloudWatchRegistri	47
Visualizzazione dei gruppi di registro log e dei flussi di registro CloudWatch	47
Utilizzo di CloudWatch eventi di log	48
Ricerca nei gruppi di log	50
Amazon ECR	53
Prerequisiti	53
Utilizzo di Amazon ECR con Toolkit for VS Code	55
Amazon ECS	64
Utilizzo di IntelliSense per i file di definizione delle attività	64
Amazon ECS Exec	65
Amazon EventBridge	68
Lavorare con Amazon EventBridge Schemas	68
Access Analyzer di AWS IAM	70

Utilizzo di AWS IAM Access Analyzer	71
AWS IoT	75
Prerequisiti di AWS IoT	75
AWS IoT Oggetti	76
AWS IoT Certificati	77
Policy AWS IoT	80
Funzioni AWS Lambda	84
Interagire con le funzioni Lambda remote	84
Amazon Redshift	90
Lavorare con Amazon Redshift	90
Amazon S3	95
Utilizzo delle risorse S3	95
Lavorare con oggetti S3	97
Applicazione Serverless AWS	101
Nozioni di base	101
Esecuzione e debug delle funzioni Lambda escludendo le risorse del modello AWS SAM ...	109
Esecuzione e debug delle risorse locali di Amazon API Gateway	114
Opzioni di configurazione per il debug di applicazioni serverless	118
Risoluzione dei problemi	126
AWS Systems Manager	128
Presupposti e prerequisiti	129
Autorizzazioni IAM per i documenti di automazione di Systems Manager	129
Creazione di un nuovo documento di automazione di Systems Manager	130
Apertura di un documento di automazione di Systems Manager esistente	130
Modifica di un documento di automazione di Systems Manager	131
Creazione di un documento di automazione di Systems Manager	132
Eliminazione di un documento di automazione di Systems Manager	132
Esecuzione di un documento di automazione di Systems Manager	133
Risoluzione dei problemi	133
AWS Step Functions	134
AWS Step Functions e VS Code	134
Threat Composer	145
Lavorare con Threat Composer	145
Risorse	146
IAM autorizzazioni per l'accesso alle risorse	147
Aggiunta di risorse e interazione con le risorse esistenti	148

Creazione e modifica delle risorse	149
Sicurezza	152
Protezione dei dati	152
Cronologia dei documenti	154
.....	clx

AWS Toolkit for Visual Studio Code

Questa è la guida per l'utente del AWS Toolkit for VS Code. Se stai cercando AWS Toolkit for Visual Studio, consulta la [Guida per gli utenti per AWS Toolkit for Visual Studio](#).

Che cos'è il AWS Toolkit for Visual Studio Code

Toolkit for VS Code è un'estensione open source per l'editor di Visual Studio Code (VS Code). Questa estensione semplifica lo sviluppo, il debug locale e la distribuzione di applicazioni serverless che utilizzano Amazon Web Services (AWS).

Argomenti

- [Nozioni di base su AWS Toolkit for Visual Studio Code](#)
- [Utilizzo di AWS servizi e strumenti](#)

Informazioni correlate

Utilizza le risorse seguenti per accedere al codice sorgente per il kit di strumenti o visualizzare i problemi attualmente aperti.

- [Codice sorgente](#)
- [Utilità di monitoraggio dei problemi](#)

Per ulteriori informazioni sull'editor di codice di Visual Studio, visita <https://code.visualstudio.com/>.

Sviluppatore Amazon Q e Amazon CodeWhisperer

A partire dal 30 aprile 2024, Amazon CodeWhisperer fa ora parte di Amazon Q Developer, ciò include suggerimenti di codice in linea e scansioni di sicurezza di Amazon Q Developer. Scarica [l'IDE estensione Amazon Q Developer da VS Code Marketplace](#) per iniziare.

Per informazioni dettagliate sul servizio Amazon Q Developer, consulta la [Amazon Q Developer User Guide](#). Per informazioni dettagliate sui piani e sui prezzi di Amazon Q, consulta la guida ai [prezzi di Amazon Q](#).

Scaricamento del Toolkit for VS Code

Puoi scaricarlo, installarlo e configurarlo AWS Toolkit for Visual Studio Code tramite VS Code Marketplace nel tuo IDE. Per istruzioni dettagliate, consulta la sezione [Download e installazione](#) nell'argomento Guida introduttiva di questa Guida per l'utente.

Scaricamento del Toolkit da VS Code Marketplace

In alternativa, puoi scaricare i file di AWS Toolkit for Visual Studio Code installazione accedendo a [VS Code Marketplace](#) dal tuo browser web.

Toolkit IDE aggiuntivi da AWS

Oltre a AWS Toolkit for Visual Studio Code, offre AWS anche IDE Toolkit per JetBrains e Visual Studio.

AWS Toolkit for JetBrains collegamenti

- Segui questo link per [scaricarlo AWS Toolkit for JetBrains dal](#) JetBrains Marketplace.
- Per ulteriori informazioni AWS Toolkit for JetBrains, consulta la Guida per l'[AWS Toolkit for JetBrains](#) utente.

Collegamenti al Toolkit for Visual Studio

- Segui questo link per [scaricare il Toolkit for Visual Studio da Visual](#) Studio Marketplace.
- Per ulteriori informazioni sul Toolkit for Visual Studio, consulta [la Toolkit for](#) Visual Studio User Guide.

Nozioni di base su AWS Toolkit for Visual Studio Code

AWS Toolkit for Visual Studio Code rende disponibili AWS i tuoi servizi e le tue risorse direttamente dal tuo ambiente di sviluppo integrato (IDE) VS Code.

Per iniziare, i seguenti argomenti descrivono come configurare, installare e configurare AWS Toolkit for Visual Studio Code.

Argomenti

- [Installazione del AWS Toolkit for Visual Studio Code](#)
- [Connessione a AWS](#)
- [AWS Regioni che cambiano](#)
- [Configurazione della toolchain](#)

Installazione del AWS Toolkit for Visual Studio Code

Prerequisiti

Per iniziare a lavorare con AWS Toolkit for Visual Studio Code VS Code, devono essere soddisfatti i seguenti prerequisiti. Per ulteriori informazioni sull'accesso a tutti i AWS servizi e le risorse disponibili da AWS Toolkit for Visual Studio Code, consulta la [the section called “Prerequisiti opzionali”](#) sezione di questa guida.

- VS Code richiede un sistema operativo Windows, macOS o Linux.
- AWS Toolkit for Visual Studio Code Richiede di lavorare con la versione 1.73.0 di VS Code o una versione successiva.

Per ulteriori informazioni su VS Code o per scaricare l'ultima versione di VS Code, visitate il sito Web dedicato ai [download di VS Code](#).

Scaricamento e installazione di AWS Toolkit for Visual Studio Code

Puoi scaricarlo, installarlo e configurarlo AWS Toolkit for Visual Studio Code tramite VS Code Marketplace nel tuo IDE. In alternativa, puoi scaricare i file di AWS Toolkit for Visual Studio Code installazione accedendo a [VS Code Marketplace](#) dal tuo browser web.

Installazione di AWS Toolkit for Visual Studio Code da VS Code IDE Marketplace

1. Apri l' AWS Toolkit for Visual Studio Code estensione nel tuo VS Code IDE con il seguente link:
[Apri il VS Code Marketplace.](#)

Note

Se VS Code non è già in esecuzione sul computer, questa operazione potrebbe richiedere alcuni istanti durante il caricamento di VS Code.

2. Dall' AWS Toolkit for Visual Studio Code estensione in VS Code Marketplace, scegli Installa per iniziare il processo di installazione.
3. Quando richiesto, scegli di riavviare VS Code per completare il processo di installazione.

Prerequisiti opzionali

Prima di poter utilizzare determinate funzionalità di AWS Toolkit for Visual Studio Code, è necessario disporre di quanto segue:

- Account Amazon Web Services (AWS): non è necessario utilizzare un AWS account AWS Toolkit for Visual Studio Code, ma la funzionalità è notevolmente limitata senza di esso. Per ottenere un AWS account, vai alla [AWS home page](#). Scegli Crea un AWS account o Completa la registrazione (se hai già visitato il sito).
- Sviluppo del codice: pertinente SDK per il linguaggio che desideri utilizzare. Puoi scaricare dai seguenti link o utilizzare il tuo gestore di pacchetti preferito:
 - .NET SDK: <https://dotnet.microsoft.com/download>
 - Node.js SDK: <https://nodejs.org/en/download>
 - Python SDK: <https://www.python.org/downloads>
 - Giava SDK: <https://aws.amazon.com/corretto/>
 - Vai SDK: <https://golang.org/doc/install>
- AWS SAM CLI— Si tratta di uno AWS CLI strumento che consente di sviluppare, testare e analizzare localmente le applicazioni serverless. Questo non è necessario per l'installazione del toolkit. Tuttavia, ti consigliamo di installarlo (e Docker, descritto di seguito) perché è necessario per qualsiasi AWS Serverless Application Model (AWS SAM) funzionalità, ad esempio. [Creazione di una nuova applicazione serverless \(locale\)](#)

Per ulteriori informazioni, consulta [Installazione di AWS SAM CLI nella Guida per gli AWS Serverless Application Model sviluppatori](#).

- Docker: AWS SAM CLI richiede questa piattaforma contenitore software open source. Per ulteriori informazioni e istruzioni di download, consulta [Docker](#).
- Package Manager: un gestore di pacchetti che consente di scaricare e condividere il codice dell'applicazione.
 - .NET: [NuGet](#)
 - Node.js: [npm](#)
 - Python: [pip](#)
 - Java: [Gradle o Maven](#)

Connessione a AWS

La maggior parte delle risorse di Amazon Web Services (AWS) viene gestita tramite un AWS account. Non è necessario un AWS account per utilizzare AWS Toolkit for Visual Studio Code, tuttavia le funzioni del Toolkit sono limitate senza una connessione.

Se in precedenza hai impostato un AWS account e l'autenticazione tramite un altro AWS servizio (come il AWS Command Line Interface), rileva AWS Toolkit for Visual Studio Code automaticamente le tue credenziali.

Prerequisiti

Se sei nuovo utente AWS o non l'hai ancora creato, ci sono 3 passaggi principali per collegarlo AWS Toolkit for Visual Studio Code al tuo AWS account:

1. Registrazione di un AWS account: puoi creare un AWS account dal [portale di AWS registrazione](#). Per informazioni dettagliate sulla configurazione di un nuovo AWS account, consulta l'argomento [Panoramica](#) nella Guida per l'utente alla AWS configurazione.
2. Configurazione dell'autenticazione: Esistono 3 metodi principali per autenticarsi con l' AWS AWS Toolkit for Visual Studio Code account da. Per ulteriori informazioni su ciascuno di questi metodi, consulta l'argomento [Autenticazione e accesso](#) in questa Guida per l'utente.
3. Autenticazione AWS dal Toolkit: è possibile connettersi al AWS proprio account dal Toolkit completando le procedure descritte nelle seguenti sezioni di questa Guida per l'utente.

Apertura del pannello di accesso

Completate una delle seguenti procedure per aprire il pannello di accesso al AWS Toolkit.

Per aprire il pannello AWS Toolkit Sign In da Explorer: AWS

1. Da AWS Toolkit for Visual Studio Code, **EXPLORER** Espandi.
2. Espandi le altre azioni... menu selezionando il... icona.
3. Da Altre azioni... menu, scegli **Connect to per AWS** aprire il pannello di accesso a AWS Toolkit.

Per aprire il pannello AWS Toolkit Sign In utilizzando il pallet di comandi VS Code:

1. Aprire il pallet dei comandi premendo **Shift+Command+P** (Windows) **Ctrl+Shift+P**.
2. Entra **AWS: Add a New Connection** nel campo di ricerca.
3. Seleziona **AWS: Add a New Connection** per aprire il pannello di accesso al AWS Toolkit.

Connessione a AWS dal Toolkit

Effettua l'autenticazione e connettiti con SSO

Per autenticarsi e connettersi AWS utilizzando AWS IAM Identity Center, completare la procedura seguente.

Note

L'autenticazione con AWS Builder ID o IAM Identity Center avvia il portale di AWS autorizzazione nel browser Web predefinito. Ogni volta che le credenziali scadono, questo processo deve essere ripetuto per rinnovare la connessione tra l'account e il. AWS AWS Toolkit for Visual Studio Code

Autentica e connettiti con Identity Center AWS IAM

1. Dal pannello AWS Toolkit Sign In, scegli la scheda Workforce, quindi seleziona il pulsante Continua per procedere.
2. Dal pannello Accedi con IAM Identity Center, inserisci Start URL per la tua organizzazione. Questo ti URL viene fornito da un amministratore o dall'help desk della tua azienda.

3. Seleziona la tua AWS regione dal menu a discesa. Questa è la AWS regione che ospita la tua directory di identità.
4. Scegli il pulsante Continua e conferma che desideri aprire il sito Web di richiesta di AWS autorizzazione nel tuo browser Web predefinito.
5. Segui le istruzioni nel tuo browser web predefinito, riceverai una notifica quando il processo di autorizzazione è completo, puoi chiudere il browser e tornare a VS Code.

Autentica e connettiti con le credenziali IAM

Per autenticarsi e connettersi AWS utilizzando le IAM credenziali, completare la procedura seguente.

Autentica e connettiti con le credenziali IAM

1. Dal pannello AWS Toolkit Sign In, scegli IAMCredenziali, quindi seleziona il pulsante Continua per procedere.
2. Inserisci il **Profile Name** **Access Key**, e **Secret Key** del tuo AWS account nei campi forniti, quindi scegli il pulsante Continua per aggiungere il profilo al tuo file di configurazione e connettere il Toolkit al tuo account. AWS
3. Toolkit AWS Explorer si aggiorna per visualizzare i AWS servizi e le risorse dell'utente una volta completata l'autenticazione e stabilita una connessione.

Autenticazione per Amazon CodeCatalyst

Per iniziare a lavorare con CodeCatalyst il Toolkit, esegui l'autenticazione e connettiti con il tuo AWS Builder ID o le credenziali di IAM Identity Center.

Le seguenti procedure descrivono come autenticare e connettere il Toolkit al tuo account. AWS

Autentica e connettiti con un Builder ID AWS

1. Dal pannello AWS Toolkit Sign In, scegli la scheda Workforce, quindi seleziona il pulsante Continua per procedere.
2. Nella parte superiore del SSO pannello Accedi con, scegli il link Vai all'accesso.
3. Segui le istruzioni nel tuo browser web predefinito, ricevi una notifica quando il processo di autorizzazione è completo, puoi chiudere il browser e tornare a VS Code.

Autentica e connettiti con Identity Center IAM

1. Dal pannello AWS Toolkit Sign In, scegli la scheda Workforce, quindi seleziona il pulsante Continua per procedere.
2. Dal pannello Accedi con IAM Identity Center, inserisci Start URL per la tua organizzazione. Questo ti URL viene fornito da un amministratore o dall'help desk della tua azienda.
3. Seleziona la tua AWS regione dal menu a discesa. Questa è la AWS regione che ospita la tua directory di identità.
4. Scegli il pulsante Continua e conferma che desideri aprire il sito Web di richiesta di AWS autorizzazione nel tuo browser Web predefinito.
5. Segui le istruzioni nel tuo browser web predefinito, riceverai una notifica quando il processo di autorizzazione è completo, puoi chiudere il browser e tornare a VS Code.

AWS Regioni che cambiano

Una AWS regione specifica dove vengono gestite AWS le tue risorse. La tua AWS regione predefinita viene rilevata quando ti connetti al tuo AWS account da AWS Toolkit for Visual Studio Code, visualizzata automaticamente in AWS Explorer.

Le sezioni seguenti descrivono come aggiungere o nascondere una regione dall'AWS Explorer.

Aggiungere una regione all' AWS Explorer

Completate la seguente procedura per aggiungere una regione all' AWS Explorer.

1. Da VS Code, apri la palette dei comandi espandendo Visualizza nel menu principale e scegliendo Command Palette. Oppure utilizzare i seguenti tasti di scelta rapida:
 - Windows e Linux — Premi **Ctrl+Shift+P**.
 - macOS: premere. **Shift+Command+P**
2. Dalla palette dei comandi, cerca **AWS: Show or Hide Regions** e scegli AWS: Mostra o nascondi regioni per visualizzare un elenco di regioni disponibili.
3. Dall'elenco, seleziona le AWS regioni che desideri aggiungere all'AWS Explorer.
4. Scegli il pulsante OK per confermare le tue scelte e aggiornare AWS Explorer.

Nascondi una regione dall' AWS Explorer

Per nascondere una regione dalla visualizzazione AWS Explorer, complete la procedura seguente.

1. In AWS Explorer, individuate la AWS regione che desiderate nascondere.
2. Apri il menu contestuale per (fai clic con il pulsante destro del mouse) per la regione che desideri nascondere.
3. Scegli Mostra o nascondi regioni per aprire le opzioni AWS: Mostra o nascondi regioni in VS Code.
4. Deseleziona le regioni che desideri nascondere nella vista AWS Explorer.

Configurazione della toolchain

AWS Toolkit for Visual Studio CodeSupporta più lingue in tutti iAWS servizi. Nelle sezioni seguenti viene descritto come configurare la toolchain per diverse viene descritto come configurare la toolchain per diverse viene descritto come configurare la toolchain.

Configurare una toolchain per .NET Core

1. Assicurati di avere [installato](#) ilAWS Toolkit for VS Code.
2. Installa [l'estensione C#](#). Questa estensione consente a VS Code di eseguire il debug delle applicazioni .NET Core.
3. Apri un'applicazione AWS Serverless Application Model (AWS SAM) o [creane una](#).
4. Apri la cartella che contiene `template.yaml`.

Configurazione della toolchain per Node.js

1. Assicurati di avere [installato](#) ilAWS Toolkit for VS Code.
2. Apri un'applicazione AWS SAM oppure [creane una](#).
3. Apri la cartella che contiene `template.yaml`.

Note

Quando si esegue il debug di una funzione TypeScript Lambda direttamente dal codice sorgente (la configurazione di avvio è presente `"target": "code"`),

il TypeScript compilatore deve essere installato a livello globale o in quello del `projettpackage.json`.

Configura una toolchain per Python

1. Assicurati di avere [installato](#) ilAWS Toolkit for VS Code.
2. Installa [l'estensione Python per Visual Studio Code](#). Questa estensione consente a VS Code di eseguire il debug delle applicazioni Python.
3. Apri un'applicazione AWS SAM oppure [creane una](#).
4. Apri la cartella che contiene `template.yaml`.
5. Apri un terminale nella root dell'applicazione e configura `virtualenv` eseguendo `python -m venv ./venv`.

Note

Devi configurare `virtualenv` una sola volta per sistema.

6. Attiva `virtualenv` eseguendo una delle seguenti operazioni:
 - Bash shell: `./venv/Scripts/activate`
 - PowerShell: `./venv/Scripts/Activate.ps1`

Configurare una toolchain per Java

1. Assicurati di avere [installato](#) ilAWS Toolkit for VS Code.
2. Installa [l'estensione Java e Java 11](#). Questa estensione consente a VS Code di riconoscere le funzioni Java.
3. Installa [l'estensione del debugger Java](#). Questa estensione consente a VS Code di eseguire il debug delle applicazioni Java.
4. Apri un'applicazione AWS SAM oppure [creane una](#).
5. Apri la cartella che contiene `template.yaml`.

Configura una toolchain per Go

1. Assicurati di avere [installato](#) ilAWS Toolkit for VS Code.
2. Go 1.14 o versioni successive è necessario per il debug delle funzioni Go Lambda.
3. Installa l'[estensione Go](#).

Note

È necessaria la versione 0.25.0 o successiva per il debug dei runtime Go1.15+.

4. Installa gli strumenti Go usando la [palette dei comandi](#):
 - a. Dalla palette dei comandi, scegliGo: Install/Update Tools.
 - b. Dal set di caselle di controllo, selezionadlv egopls.
5. Apri un'applicazione AWS SAM oppure [creane una](#).
6. Apri la cartella che contiene `template.yaml`.

Usare la tua catena di strumenti

Una volta configurata la toolchain, puoi utilizzarla [per l'esecuzione o per effettuare il debug](#) dell'applicazione AWS SAM.

Autenticazione e accesso per il AWS Toolkit for Visual Studio Code

Non è necessario autenticarsi con AWS per iniziare a utilizzare. AWS Toolkit for Visual Studio Code Tuttavia, la maggior parte AWS delle risorse viene gestita tramite un AWS account. Per accedere a tutti i AWS Toolkit for Visual Studio Code servizi e le funzionalità, dovrai autenticarti con AWS IAM Identity Center AWS Builder ID o credenziali IAM.

I seguenti argomenti contengono dettagli aggiuntivi su ciascun tipo di credenziale.

Per informazioni dettagliate su come connettersi a AWS in AWS Toolkit for Visual Studio Code con le credenziali esistenti, consulta l' AWS argomento [Connessione a](#) in questa Guida per l'utente.

Argomenti

- [AWS IAM Identity Center](#)
- [AWS IAMcredenziali](#)
- [AWS Builder ID per sviluppatori](#)
- [Utilizzo di un processo di credenziali esterno](#)

AWS IAM Identity Center

AWS IAM Identity Center è la best practice consigliata per la gestione dell'autenticazione AWS dell'account.

Per istruzioni dettagliate su come configurare IAM Identity Center for Software Development Kits (SDK), consulta la sezione sull'[autenticazione di IAM Identity Center](#) della Guida di riferimento agli AWS SDK e agli strumenti.

Per i dettagli su come autenticare e connettere il AWS toolkit con le credenziali IAM Identity Center esistenti, consulta l' AWS argomento [Connect to](#) in questa guida per l'utente.

AWS IAMcredenziali

AWS IAMautenticazione delle credenziali con il tuo AWS account tramite chiavi di accesso memorizzate localmente.

Per informazioni dettagliate su come autenticare e connettere il AWS toolkit con le AWS IAM credenziali esistenti, consulta l' AWS argomento [Connect to di questa Guida per l'utente](#).

Le sezioni seguenti descrivono come configurare IAM le credenziali per l'autenticazione con l'account da. AWS AWS Toolkit for Visual Studio Code

Important

Prima di configurare IAM le credenziali per l'autenticazione con il tuo AWS account, tieni presente che:

- Se hai già impostato IAM le credenziali tramite un altro AWS servizio (come il AWS CLI), rileva AWS Toolkit for Visual Studio Code automaticamente tali credenziali e le rende disponibili in VS Code.
- AWS consiglia di utilizzare l'autenticazione IAM Identity Center. Per ulteriori informazioni sulle AWS IAM best practice, consulta la sezione Procedure consigliate [per la sicurezza nella IAM](#) sezione della AWS Identity and Access Management User Guide.
- Per evitare rischi per la sicurezza, non utilizzate IAM gli utenti per l'autenticazione quando sviluppate software appositamente creato o lavorate con dati reali. Utilizza invece la federazione con un provider di identità come [What is IAM Identity Center?](#) nella Guida AWS IAM Identity Center per l'utente.

Creare un IAM utente

Prima di poter configurare l'autenticazione con il AWS Toolkit for Visual Studio Code tuo AWS account, devi completare il Passaggio 1: Creare IAM l'utente e il Passaggio 2: Ottieni le chiavi di accesso nell'argomento [Autenticazione con credenziali a lungo termine](#) nella Guida di riferimento agli strumenti AWS SDKs e agli strumenti.

Note

Fase 3: L'aggiornamento del file delle credenziali condivise nella Tools Reference AWS SDKs Guide è facoltativo.

Se completi il passaggio 3, rileva AWS Toolkit for Visual Studio Code automaticamente le tue credenziali durante le [the section called "Creazione di un file di credenziali condiviso da AWS Toolkit for Visual Studio Code"](#) operazioni riportate di seguito.

Se non hai completato la Fase 3, ti AWS Toolkit for Visual Studio Code guiderà attraverso il processo di creazione di un `credentials` file file come descritto di seguito [the section called “Creazione di un file di credenziali condiviso da AWS Toolkit for Visual Studio Code”](#).

Creazione di un file di credenziali condiviso da AWS Toolkit for Visual Studio Code

Il file di configurazione condiviso e il file delle credenziali condivise memorizzano le informazioni sulla configurazione e sulle credenziali per i tuoi account. AWS Per ulteriori informazioni sulla configurazione e le credenziali condivise, consulta la sezione [Where are configuration settings stored?](#) della Guida per l'utente di AWS Command Line Interface .

Creazione di un file di credenziali condiviso tramite il AWS Toolkit for Visual Studio Code

1. Aprire il pallet dei comandi premendo **Shift+Command+P (Ctrl+Shift+P)** Windows).
2. Entra **AWS: Add a New Connection** nel campo di ricerca.
3. Seleziona **AWS: Add a New Connection** per aprire il pannello di accesso al AWS Toolkit.
4. Dal pannello AWS Toolkit Sign In, scegli IAMCredenziale, quindi seleziona il pulsante Continua per procedere.
5. Inserisci il **Profile Name** **Access Key**, e **Secret Key** del tuo AWS account nei campi forniti, quindi scegli il pulsante Continua per aggiungere il profilo al tuo file di configurazione e connettere il Toolkit al tuo account. AWS
6. Toolkit AWS Explorer si aggiorna per visualizzare i AWS servizi e le risorse dell'utente una volta completata l'autenticazione e stabilita una connessione.

Note

In questo esempio, supponiamo che `[Profile_Name]` contiene errori di sintassi e causa il fallimento dell'autenticazione.

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

Di seguito è riportato un esempio di messaggio di registro generato in risposta a un tentativo di autenticazione fallito.

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

Aggiungi profili di credenziali aggiuntivi

Puoi aggiungere più credenziali ai tuoi file di configurazione. Per farlo, apri la palette dei comandi e scegli AWS Toolkit Create Credentials Profile. Questo aprirà il file delle credenziali. In questa pagina, puoi aggiungere un nuovo profilo sotto il tuo primo profilo, come mostrato nell'esempio seguente:

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

AWS Builder ID per sviluppatori

Un AWS Builder ID è un AWS account aggiuntivo facoltativo o richiesto per determinati servizi. AWS Per informazioni dettagliate sul metodo di autenticazione AWS Builder ID, consulta l'argomento [Accedi con AWS Builder ID nella Guida per l'utente di AWS accesso](#).

Per informazioni dettagliate su come autenticare e connettere il AWS toolkit con il tuo AWS Builder ID esistente, consulta l' AWS argomento [Connect to](#) di questa guida per l'utente.

Utilizzo di un processo di credenziali esterno

Puoi configurarliAWS Toolkit for Visual Studio Code per i processi di credenziali che non sono direttamente supportati daAWS, modificando il tuo `shared config file`.

La modifica dei processi `shared config file` per le credenziali è la stessa sia per ilAWS Command Line Interface.AWS Toolkit for Visual Studio Code Per informazioni dettagliate su come configurare le credenziali esterne, vedere l'argomento [Sourcing delle credenziali con un processo esterno](#) nella Guida per l'AWS Command Line Interfaceutente.

Utilizzo di AWS servizi e strumenti

Ti AWS Toolkit for Visual Studio Code mette a disposizione AWS servizi, strumenti e risorse direttamente in VS Code. Di seguito è riportato un elenco di argomenti della guida che coprono ogni servizio Toolkit for VS Code e le relative funzionalità. Scegliete un servizio o uno strumento per ulteriori informazioni sulle sue funzioni, su come configurarlo e sull'utilizzo delle funzionalità di base.

Argomenti

- [Utilizzo delle caratteristiche sperimentali](#)
- [Utilizzo AWS dei servizi in AWS Explorer](#)
- [Amazon CodeCatalyst per VS Code](#)
- [Utilizzo di Amazon API Gateway](#)
- [Utilizzo AWS App Runner con AWS Toolkit for Visual Studio Code](#)
- [AWSCompositore di applicazioni](#)
- [AWS CDKper VS Code](#)
- [Utilizzo diAWS CloudFormationpile](#)
- [Lavorare conCloudWatchRegistra utilizzando ilAWS Toolkit for Visual Studio Code](#)
- [Utilizzo di Amazon Elastic Container Registry](#)
- [Amazon Elastic Container Service](#)
- [Utilizzo di schemi Amazon EventBridge](#)
- [Access Analyzer di AWS IAM](#)
- [Lavorare con AWS IoT in AWS Toolkit for Visual Studio Code](#)
- [Lavorare con le funzioni AWS Lambda](#)
- [Amazon Redshift nel Toolkit for VS Code](#)
- [Lavorare con Amazon S3](#)
- [Utilizzo di applicazioni serverless](#)
- [Utilizzo dei documenti di automazione di Systems Manager](#)
- [Lavorare con AWS Step Functions](#)
- [Utilizzo di Threat Composer](#)
- [Utilizzo delle risorse](#)

Utilizzo delle caratteristiche sperimentali

Le funzionalità sperimentali offrono l'accesso anticipato alle funzionalità del AWS Toolkit for Visual Studio Code prima che vengano pubblicati ufficialmente.

Warning

Poiché le funzionalità sperimentali continuano a essere testate e aggiornate, potrebbero avere problemi di usabilità. E le funzionalità sperimentali possono essere rimosse dal AWS Toolkit for Visual Studio Code senza preavviso.

È possibile abilitare funzionalità sperimentali per specifiche AWS servizi nel AWS Kit di strumenti Sezione dell'Impostazione nel tuo IDE del codice VS.

1. Per modificare AWS impostazioni in VS Code, scegli File, Preferenze, Impostazioni.
2. Nell'Impostazione riquadro, espandi Estensione e scegli AWS Kit di strumenti.
3. **UNDER AWS: Esperimenti**, seleziona le caselle di controllo per le funzionalità sperimentali a cui desideri accedere prima del rilascio. Se si desidera disattivare una funzione sperimentale, deselezionare la relativa casella di controllo.

Utilizzo AWS dei servizi in AWS Explorer

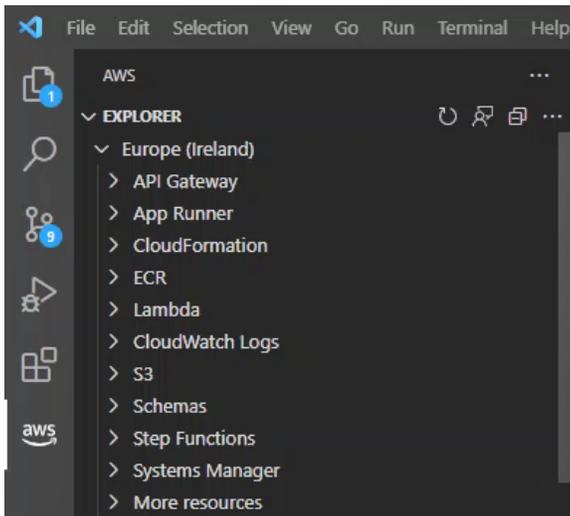
AWS Explorer offre una panoramica di alcuni dei AWS servizi con cui è possibile lavorare durante l'utilizzo di AWS Toolkit for Visual Studio Code.

Questa sezione fornisce informazioni su come accedere e utilizzare AWS Explorer in VS Code. Si presuppone che tu abbia già [installato e configurato](#) il Toolkit for VS Code sul tuo sistema.

Alcuni punti importanti:

- Se il toolkit è installato e configurato correttamente, dovresti vedere gli elementi in Explorer. AWS Per vedere AWS Explorer, scegli l'AWS icona nella barra delle attività.

Per esempio:



- Alcune funzionalità richiedono determinate AWS autorizzazioni. Ad esempio, per visualizzare AWS Lambda le funzioni del tuo AWS account, le credenziali configurate [Autenticazione e accesso](#) devono includere almeno le autorizzazioni Lambda di sola lettura. Consulta gli argomenti seguenti per ulteriori informazioni sulle autorizzazioni necessarie per ogni funzionalità.
- Se desideri interagire con AWS servizi che non sono immediatamente visibili in AWS Explorer, puoi andare su Altre risorse e scegliere tra centinaia di risorse da aggiungere all'interfaccia.

Ad esempio, puoi scegliere AWS Toolkit:CodeArtifact: :Repository dalla selezione dei tipi di risorse disponibili. Dopo aver aggiunto questo tipo di risorsa a Altre risorse, è possibile espandere la voce per visualizzare un elenco di risorse che creano CodeArtifact repository diversi con proprietà e attributi propri. Inoltre, puoi descrivere le proprietà e gli attributi delle risorse in modelli in formato JSON, che possono essere salvati per creare nuove risorse nel Cloud. AWS

Amazon CodeCatalyst per VS Code

Che cos'è Amazon CodeCatalyst?

Amazon CodeCatalyst è uno spazio di collaborazione basato sul cloud per i team di sviluppo software. Tramite AWS Toolkit for Visual Studio Code, puoi visualizzare e gestire le tue CodeCatalyst risorse direttamente da VS Code. Puoi anche lavorare direttamente nel cloud avviando CodeCatalyst Dev Environment. Per ulteriori informazioni sul CodeCatalyst servizio, consulta la [Amazon CodeCatalyst User Guide](#).

I seguenti argomenti descrivono come connettere VS Code e come utilizzarlo CodeCatalyst dal Toolkit for VS Code. CodeCatalyst

Argomenti

- [Guida introduttiva CodeCatalyst e al Toolkit for VS Code](#)
- [Lavorare con CodeCatalyst le risorse Amazon in VS Code](#)
- [Lavorare con il Toolkit in un ambiente di sviluppo](#)
- [Risoluzione dei problemi relativi ad Amazon CodeCatalyst e VS Code](#)

Guida introduttiva CodeCatalyst e al Toolkit for VS Code

Per iniziare a lavorare con CodeCatalyst VS Code, segui queste procedure.

Argomenti

- [Creazione di un CodeCatalyst account](#)
- [Connessione del Toolkit AWS con CodeCatalyst](#)

Creazione di un CodeCatalyst account

È necessario disporre di un AWS Builder ID o AWS IAM Identity Center delle credenziali attivi a cui connettersi CodeCatalyst dal Toolkit for VS Code. Per ulteriori informazioni su AWS Builder ID, IAM Identity Center e CodeCatalyst credenziali, consulta la CodeCatalyst sezione [Configurazione con della Guida per l'utente](#). CodeCatalyst

Connessione del Toolkit AWS con CodeCatalyst

Per connettere il AWS Toolkit al tuo CodeCatalyst account, consulta la CodeCatalyst sezione [Autenticazione per Amazon](#) nell' AWS argomento Connessione a di questa Guida per l'utente.

Lavorare con CodeCatalyst le risorse Amazon in VS Code

Le seguenti sezioni forniscono una panoramica delle funzionalità di gestione CodeCatalyst delle risorse di Amazon disponibili nel Toolkit for VS Code.

Per ulteriori informazioni sugli ambienti di sviluppo e su come accedervi da CodeCatalyst, consulta la sezione [Dev Environments](#) nella Amazon CodeCatalyst User Guide.

Le seguenti sezioni descrivono come creare, aprire e lavorare con Dev Environments da VS Code.

Argomenti

- [Clona un repository](#)

- [Apertura di un ambiente di sviluppo](#)
- [Creazione di un ambiente di CodeCatalyst sviluppo](#)
- [Creazione di un ambiente di sviluppo da un repository di terze parti](#)
- [CodeCatalyst comandi in VS Code](#)

Clona un repository

CodeCatalyst è un servizio basato sul cloud che richiede la connessione al cloud per lavorare sui progetti. CodeCatalyst Se preferisci lavorare su un progetto localmente, puoi clonare i tuoi CodeCatalyst repository sul tuo computer locale e sincronizzarlo con il tuo CodeCatalyst progetto online, la prossima volta che ti connetti al cloud.

Per clonare un repository dal tuo CodeCatalyst account a VS Code with the AWS Toolkit, completa i seguenti passaggi:

Note

Se stai clonando un repository da un servizio di terze parti, ti potrebbe essere richiesto di autenticarti con le credenziali di quel servizio.

Durante la clonazione del repository, VS Code mostra l'avanzamento nella finestra di stato del repository di clonazione. Dopo la clonazione del repository, la pagina Vuoi aprire il repository clonato? viene visualizzato un messaggio.

1. Dal Toolkit for VS Code, espandi DEVELOPERTOOLS\explorer.
2. Espandi CodeCatalyst, scegli Clone Repository.
3. Nella finestra di dialogo Seleziona un CodeCatalyst archivio, cerca il repository che desideri clonare, quindi selezionalo per aprire la finestra di dialogo Scegli una cartella da clonare.
4. Scegli Seleziona la posizione del repository per chiudere il prompt e iniziare a clonare il repository.
5. Dalla finestra di dialogo, scegliete una delle seguenti opzioni per completare il processo di clonazione:
 - Per aprire il repository nella finestra VS Code corrente, scegli Apri.
 - Per aprire il repository in una nuova finestra di VS Code, scegli Apri in una nuova finestra.
 - Per completare il processo di clonazione senza aprire il repository, chiudi la finestra di dialogo.

Apertura di un ambiente di sviluppo

Per aprire un ambiente di sviluppo esistente in VS Code, completa i seguenti passaggi.

Note

La selezione dell'ambiente di sviluppo avvia il processo di connessione a VS Code CodeCatalyst aprendo l'ambiente di sviluppo. Durante questo processo, VS Code mostra gli aggiornamenti sullo stato di avanzamento in una finestra di CodeCatalyst stato. La finestra di stato si aggiorna quando il processo è completo.

- Se l'ambiente di sviluppo non viene aperto, lo stato viene aggiornato con informazioni sul motivo per cui il processo non è riuscito e un collegamento per aprire i registri del processo.
- Se il processo ha esito positivo, il tuo Dev Environment si apre in una nuova finestra, da VS Code.

1. Dal Toolkit for VS Code, espandi DEVELOPERTOOLS\explorer.
2. Espandi CodeCatalyst scegli Open Dev Environment per aprire la finestra di dialogo Seleziona un ambiente di CodeCatalyst sviluppo in VS Code.
3. Dalla finestra di dialogo Seleziona un CodeCatalyst ambiente di sviluppo, scegli l'ambiente di sviluppo che desideri aprire.

Creazione di un ambiente di CodeCatalyst sviluppo

Per creare un nuovo ambiente di sviluppo, completa i seguenti passaggi:

Note

Quando crei un nuovo ambiente di sviluppo, osserva quanto segue:

- AWS consiglia di specificare un alias perché semplifica l'organizzazione e migliora le funzionalità di ricerca per gli ambienti di sviluppo.
- Dev Environments salva il tuo lavoro in modo persistente. Ciò significa che il tuo ambiente di sviluppo può essere interrotto senza perdere il tuo lavoro. L'interruzione dell'ambiente di sviluppo riduce i costi necessari per mantenere l'ambiente di sviluppo attivo e funzionante.
- L'archiviazione è l'unica impostazione che non può essere modificata dopo la creazione dell'ambiente di sviluppo.

- VS Code mostra lo stato di avanzamento della creazione dell'ambiente di sviluppo in una finestra di stato. Dopo aver creato l'ambiente di sviluppo, VS Code apre l'ambiente di sviluppo in una nuova finestra e il pulsante Ti fidi degli autori dei file in questa cartella? Viene visualizzato anche il prompt. Accetta i termini e le condizioni per continuare a lavorare nel tuo ambiente di sviluppo.

1. Dal Toolkit for VS Code, espandi DEVELOPERTOOLS\explorer.
2. CodeCatalystEspandi e scegli l'opzione Crea ambiente di sviluppo per aprire il menu Crea un ambiente di CodeCatalyst sviluppo in VS Code.
3. Dalla sezione Codice sorgente, scegli una delle seguenti opzioni:
 - Usa un CodeCatalyst repository esistente: crea un ambiente di sviluppo da un CodeCatalyst repository esistente. È necessario selezionare il CodeCatalyst progetto e il ramo.
 - Crea un ambiente di sviluppo vuoto: crea un ambiente di sviluppo vuoto.
4. (Facoltativo) Nella sezione Alias, inserite un nome alternativo per il vostro ambiente di sviluppo.
5. (Facoltativo) Dalla sezione Dev Environments Configuration, modifica le seguenti impostazioni per soddisfare le tue esigenze specifiche.
 - Elaborazione: scegli Modifica calcolo per modificare la quantità di potenza di elaborazione assegnata al tuo sistema. RAM
 - Timeout: scegli Edit Timeout per modificare la quantità di tempo di inattività del sistema consentito prima che l'ambiente di sviluppo venga interrotto.
 - Archiviazione: scegli Modifica dimensione di archiviazione per modificare la quantità di spazio di archiviazione assegnata al tuo sistema.
6. Scegli Create Dev Environment per creare il tuo nuovo ambiente di sviluppo cloud.

Creazione di un ambiente di sviluppo da un repository di terze parti

È possibile creare ambienti di sviluppo da un repository di terze parti connettendosi al repository come origine.

Il collegamento a un repository di terze parti come sorgente viene gestito a livello di progetto in. CodeCatalyst Per istruzioni e dettagli aggiuntivi su come connettere un repository di terze parti al tuo ambiente di sviluppo, consulta l'argomento [Collegamento di un repository di origine](#) nella Amazon CodeCatalyst User Guide.

CodeCatalyst comandi in VS Code

Esistono comandi VS Code aggiuntivi assegnati a funzionalità CodeCatalyst correlate che non vengono visualizzate direttamente nel AWS Toolkit.

Per visualizzare un elenco di comandi assegnati CodeCatalyst dalla palette dei comandi, completa i seguenti passaggi:

1. Dal Toolkit for VS Code, espandi DEVELOPERTOOLS\l'explorer.
2. Scegli Mostra CodeCatalyst comandi per aprire la palette dei comandi con una ricerca precompilata per CodeCatalyst
3. Seleziona un CodeCatalyst comando dall'elenco per attivarlo.

Lavorare con il Toolkit in un ambiente di sviluppo

Gli ambienti di sviluppo sono ambienti di elaborazione virtuali per Amazon CodeCatalyst. Le seguenti sezioni descrivono come creare, avviare e utilizzare ambienti di sviluppo utilizzando AWS Toolkit for Visual Studio Code.

Per informazioni dettagliate sugli ambienti di sviluppo, consulta l'argomento [Dev Environments](#) nella Amazon CodeCatalyst User Guide.

Configurazione dell'ambiente di sviluppo con devfiles

La `devfile` specifica è un formato a standard aperto YAML che può essere utilizzato per definire configurazioni per gli ambienti di sviluppo. Ogni ambiente di sviluppo ha un devfile. Se crei un ambiente di sviluppo senza un repository o da un repository che non contiene un devfile, viene applicato automaticamente un valore predefinito all'origine. I file Devfile possono essere aggiornati da o dal tuo. CodeCatalyst IDE I processi per aggiornare un devfile in un'istanza locale o remota di VS Code sono identici, ma se aggiorni un devfile localmente, devi inviare gli aggiornamenti al tuo repository di origine prima che gli aggiornamenti abbiano effetto.

Per informazioni dettagliate sulla configurazione degli ambienti di sviluppo con devfiles, consulta l'argomento [Configuring your Dev Environment](#) nella Amazon CodeCatalyst User Guide.

La procedura seguente descrive come modificare il tuo devfile da un'istanza remota del Toolkit mentre è in esecuzione in un ambiente di sviluppo.

Important

Se lo modifichi `Devfile` da VS Code, tieni presente quanto segue:

- La modifica del nome del devfile o del nome del componente devfile sostituisce il contenuto della directory principale. Tutti i contenuti precedenti vengono persi e sono irrecuperabili.
- Se create un ambiente di sviluppo senza un devfile nella cartella principale o un ambiente di sviluppo non associato a un repository di origine, al momento della creazione viene generato un devfile con le impostazioni di configurazione predefinite per il vostro ambiente di sviluppo.
- [Per istruzioni su come definire e configurare il file `Devfile`, consultate la documentazione relativa all'aggiunta dei comandi sul sito `devfile.io`.](#)

1. Dal Toolkit for VS Code, espandi DEVELOPERTOOLS l'explorer.
2. Espandi CodeCatalyst scegli Open Devfile per aprirlo `devfile.yaml` in una nuova finestra dell'editor, all'interno del tuo attuale ambiente di sviluppo.
3. Dall'editor VS Code, aggiorna il tuo devfile, quindi salva le modifiche.
4. La prossima volta che avvii il tuo ambiente di sviluppo, la configurazione viene aggiornata in modo che corrisponda alle specifiche definite nel tuo `Devfile`.

Autenticazione e connessione AWS dal tuo ambiente di sviluppo

Per accedere a tutte le AWS risorse dal tuo ambiente di sviluppo, devi autenticarti e connettere l'istanza remota del Toolkit al tuo account. AWS L'istanza remota del Toolkit si autentica automaticamente con le credenziali ereditate dall'istanza locale del Toolkit all'avvio del Dev Environment.

Le procedure per aggiornare le credenziali per un'istanza remota del Toolkit sono identiche all'esperienza di autenticazione nell'istanza locale del Toolkit. Per istruzioni dettagliate su come aggiornare le credenziali, autenticarsi e connettersi AWS dal Toolkit, consultate la AWS sezione [Connessione a](#) nell'argomento Guida introduttiva di questa Guida per l'utente.

Per ulteriori informazioni su ciascuno dei metodi di AWS autenticazione compatibili con il AWS Toolkit for Visual Studio Code, vedere l'argomento [Autenticazione e accesso](#) in questa Guida per l'utente.

Utilizzo del Toolkit for VS Code negli ambienti di sviluppo

Dopo aver aperto o creato un ambiente di sviluppo in VS Code, puoi lavorare dal Toolkit for VS Code, in modo simile a come puoi fare da un'istanza locale di VS Code. Gli ambienti di sviluppo che eseguono VS Code sono configurati per installare automaticamente il AWS Toolkit e connettersi con il tuo ID AWS Builder.

Interruzione di un ambiente di sviluppo

Per interrompere il tuo attuale ambiente di sviluppo:

1. Dal Toolkit for VS Code, espandi DEVELOPERTOOLS\l'explorer.
2. Espandi CodeCatalyste scegli Stop Dev Environment.
3. Quando richiesto da VS Code, conferma di voler interrompere il tuo Dev Environment.
4. Il tuo ambiente di sviluppo si è interrotto correttamente quando VS Code chiude la connessione remota e torna a un'istanza di sviluppo locale.

Apertura delle impostazioni dell'ambiente di sviluppo

Per aprire le impostazioni per il tuo ambiente di sviluppo corrente, completa i seguenti passaggi:

Note

Non è possibile modificare la quantità di spazio di archiviazione assegnato all'ambiente di sviluppo dopo averlo creato.

1. Dal Toolkit for VS Code, espandi DEVELOPERTOOLS\l'explorer.
2. Espandi CodeCatalyste scegli Apri impostazioni per aprire la vista Impostazioni dell'ambiente di sviluppo, per il tuo ambiente di sviluppo corrente.
3. Nella vista Dev Environment Settings (Impostazioni dell'ambiente di sviluppo), le seguenti sezioni contengono opzioni per il tuo ambiente di sviluppo:
 - Alias: visualizza e modifica l'alias assegnato al tuo ambiente di sviluppo.
 - Stato: visualizza lo stato attuale dell'ambiente di sviluppo, il progetto a cui è assegnato e interrompi il tuo ambiente.
 - Devfile: visualizza il nome e la posizione del tuo ambiente Devfile di sviluppo. Apri il tuo Devfile scegliendo il pulsante Apri nell'editor.

- **Compute Settings (Impostazioni di calcolo):** modifica le dimensioni e il valore predefinito per `Timeout Length (Durata timeout)` per il tuo ambiente di sviluppo.

Risoluzione dei problemi relativi ad Amazon CodeCatalyst e VS Code

I seguenti argomenti affrontano potenziali problemi tecnici quando si lavora con Amazon CodeCatalyst e VS Code.

Argomenti

- [Versione VS Code](#)
- [Autorizzazioni per Amazon CodeCatalyst](#)
- [Connessione a un ambiente di sviluppo dal Toolkit for VS Code](#)

Versione VS Code

La tua versione di VS Code dovrebbe configurare un gestore per `vscode://` URIs il tuo sistema. Senza questo gestore, non puoi accedere a tutte le CodeCatalyst funzionalità del AWS Toolkit. Ad esempio, si verifica un errore durante l'avvio di un ambiente di sviluppo da VS Code Insiders. Questo perché VS Code Insiders gestisce `vscode-insiders://` URIs e non gestisce `vscode://` URIs

Autorizzazioni per Amazon CodeCatalyst

Di seguito sono riportati i requisiti di autorizzazione dei file per lavorare con CodeCatalyst : AWS Toolkit for Visual Studio Code

- Imposta le tue autorizzazioni di accesso per il `~/.ssh/config` file su `read write`. Limita `write` le autorizzazioni per tutti gli altri utenti.
- Imposta le tue autorizzazioni di accesso solo per i `~/.ssh/id_rsa` file `~/.ssh/id_dsa` and `read readLimit write` e `execute` autorizzazioni per tutti gli altri utenti.
- Il `globals.context.globalStorageUri.fsPath` file deve trovarsi in una posizione scrivibile.

Connessione a un ambiente di sviluppo dal Toolkit for VS Code

Se ricevi il seguente errore quando tenti di connetterti a un ambiente di sviluppo da: AWS Toolkit for Visual Studio Code

```
~/.ssh/configLa tua aws-devenv- * sezione potrebbe non essere aggiornata.
```

- Scegli Open config. pulsante per aprire il `~/ .ssh/config` file in VS Code Editor.
- Dall'editor, seleziona ed elimina il contenuto della Host `aws-devenv-*` sezione.
- Salva le modifiche che hai apportato Host `aws-devenv-*` a `~/ .ssh/config`. Quindi, chiudi il file.
- Tentativo di connessione a un ambiente di sviluppo dal Toolkit for VS Code.

Utilizzo di Amazon API Gateway

È possibile sfogliare ed eseguire risorse API Gateway remote nella tua connessione AWS account usando il AWS Toolkit for Visual Studio Code.

Note

Questa caratteristica non supporta il debug.

Per sfogliare ed eseguire risorse remote API Gateway

1. Nella AWS Esploratore, scegli Gateway API per espandere il menu. Sono elencate le risorse remote API Gateway.
2. Individuare la risorsa API Gateway che si desidera richiamare, aprire il menu contestuale (pulsante destro del mouse) e scegliere Richiama su AWS.
3. Nel modulo parametri, specificare i parametri di richiamo.
4. Per eseguire la risorsa API Gateway remota, scegliere Richiamo. I risultati sono riportati nell'Uscita codice VS visualizzazione.

Utilizzo AWS App Runner con AWS Toolkit for Visual Studio Code

[AWS App Runner](#) offre un modo rapido, semplice ed economico per la distribuzione dal codice sorgente o da un'immagine del contenitore direttamente a un'applicazione web scalabile e sicura nel cloud. AWS Usandolo, non è necessario apprendere nuove tecnologie, decidere quale servizio di elaborazione utilizzare o sapere come fornire e configurare le risorse. AWS

Puoi utilizzarlo AWS App Runner per creare e gestire servizi basati su un'immagine sorgente o un codice sorgente. Se utilizzi un'immagine sorgente, puoi scegliere un'immagine di container pubblica o

privata archiviata in un repository di immagini. App Runner supporta i seguenti provider di repository di immagini:

- Amazon Elastic Container Registry (Amazon ECR): memorizza immagini private nel tuo account. AWS
- Amazon Elastic Container Registry Public (Amazon ECR Public): archivia le immagini leggibili pubblicamente.

Se scegli l'opzione del codice sorgente, puoi eseguire l'implementazione da un repository di codice sorgente gestito da un provider di repository supportato. Attualmente, App Runner [GitHub](#) funge da provider di repository di codice sorgente.

Prerequisiti

Per interagire con App Runner utilizzando il è AWS Toolkit for Visual Studio Code necessario quanto segue:

- Un account AWS
- Una versione di AWS Toolkit for Visual Studio Code queste funzionalità AWS App Runner

Oltre a questi requisiti fondamentali, assicurati che tutti gli utenti IAM pertinenti dispongano delle autorizzazioni per interagire con il servizio App Runner. Inoltre, è necessario ottenere informazioni specifiche sulla fonte del servizio, come l'URI dell'immagine del contenitore o la connessione al GitHub repository. Queste informazioni sono necessarie per creare il servizio App Runner.

Configurazione delle autorizzazioni IAM per App Runner

Il modo più semplice per concedere le autorizzazioni necessarie per App Runner consiste nell'allegare una policy AWS gestita esistente all'entità pertinente AWS Identity and Access Management (IAM), in particolare un utente o un gruppo. App Runner fornisce due policy gestite che è possibile allegare agli utenti IAM:

- `AWSAppRunnerFullAccess`: consente agli utenti di eseguire tutte le operazioni di App Runner.
- `AWSAppRunnerReadOnlyAccess`: consente agli utenti di elencare e visualizzare i dettagli relativi alle risorse di App Runner.

Inoltre, se scegli un repository privato da Amazon Elastic Container Registry (Amazon ECR) come origine del servizio, devi creare il seguente ruolo di accesso per il servizio App Runner:

- `AWSAppRunnerServicePolicyForECRAccess`: consente ad App Runner di accedere alle immagini Amazon Elastic Container Registry (Amazon ECR) nel tuo account.

È possibile creare questo ruolo automaticamente durante la configurazione dell'istanza del servizio con Command Palette di VS Code.

Note

Il ruolo `AWSServiceRoleForAppRunner` collegato al servizio consente di AWS App Runner completare le seguenti attività:

- Invia i log ai gruppi di log di Amazon CloudWatch Logs.
- Crea regole Amazon CloudWatch Events per iscriverti all'immagine push di Amazon Elastic Container Registry (Amazon ECR).

Non devi creare manualmente il ruolo collegato al servizio. Quando crei un ruolo collegato AWS App Runner al servizio AWS Management Console o utilizzando operazioni API richiamate da AWS Toolkit for Visual Studio Code, AWS App Runner crea automaticamente questo ruolo collegato al servizio.

Per ulteriori informazioni, consulta [Identity and Access Management per App Runner](#) nella Guida per gli sviluppatori di AWS App Runner.

Recupero delle origini di servizio per App Runner

Puoi utilizzare AWS App Runner per distribuire servizi da un'immagine sorgente o da un codice sorgente.

Source image

Se esegui la distribuzione da un'immagine sorgente, puoi ottenere un collegamento al repository di quell'immagine da un registro di immagini privato o pubblico. AWS

- Registro privato Amazon ECR: puoi copiare l'URI di un repository privato che utilizza la console Amazon ECR all'indirizzo <https://console.aws.amazon.com/ecr/repositories>.

- Registro pubblico Amazon ECR: puoi copiare l'URI di un repository pubblico che utilizza la galleria pubblica di Amazon ECR all'indirizzo <https://gallery.ecr.aws/>.

Note

Puoi anche ottenere l'URI per un repository Amazon ECR privato direttamente da AWS Explorer in Toolkit for VS Code:

- Apri AWS Explorer ed espandi il nodo ECR per visualizzare l'elenco dei repository per quella regione. AWS
- Fai clic con il pulsante destro del mouse su un repository e seleziona Copy Repository URI (Copia URI del repository) per copiare il collegamento negli appunti.

Si specifica l'URI per l'archivio di immagini durante la configurazione dell'istanza del servizio con la palette di comandi di VS Code

Per ulteriori informazioni, consulta [Servizio App Runner basato su un'immagine sorgente](#) nella Guida per gli sviluppatori di AWS App Runner .

Source code

Affinché il codice sorgente venga distribuito su un AWS App Runner servizio, tale codice deve essere archiviato in un repository Git gestito da un provider di repository supportato. App Runner supporta un provider di repository di codice sorgente: [GitHub](#)

Per informazioni sulla configurazione di un GitHub repository, consulta la documentazione [introduttiva](#) su. GitHub

Per distribuire il codice sorgente a un servizio App Runner da un GitHub repository, App Runner stabilisce una connessione a. GitHub Se il tuo repository è privato (ovvero non è accessibile pubblicamente su GitHub), devi fornire ad App Runner i dettagli di connessione.

Important

Per creare GitHub connessioni, è necessario utilizzare la console App Runner (<https://console.aws.amazon.com/apprunner>) per creare una connessione a cui si collega. GitHub AWS È possibile selezionare le connessioni disponibili nella pagina delle GitHubconnessioni durante la configurazione dell'istanza del servizio con Command Palette di VS Code.

Per ulteriori informazioni, consulta [Gestione delle connessioni di App Runner](#) nella Guida per gli sviluppatori di AWS App Runner .

L'istanza del servizio App Runner fornisce un runtime gestito che consente la creazione e l'esecuzione del codice. AWS App Runner attualmente supporta i seguenti runtime:

- Runtime gestito da Python
- Runtime gestito da Node.js

Come parte della configurazione del servizio, fornisci informazioni su come il servizio App Runner crea e avvia il servizio. Puoi inserire queste informazioni utilizzando il pannello dei comandi o specificando un [file di configurazione di App Runner](#) formattato come YAML. I valori in questo file indicano ad App Runner come creare e avviare il servizio e forniscono il contesto di runtime. Sono incluse le impostazioni di rete e le variabili di ambiente pertinenti. Il file di configurazione è denominato `apprunner.yaml`. Viene aggiunto automaticamente alla directory principale del repository dell'applicazione.

Prezzi

Ti viene addebitato il costo delle risorse di calcolo e memoria utilizzate dall'applicazione. Inoltre, se automatizzi le implementazioni, paghi anche una tariffa mensile fissa per ogni applicazione che copre tutte le implementazioni automatiche per quel mese. Se scegli di eseguire l'implementazione dal codice sorgente, paghi anche una tariffa di creazione per il periodo di tempo impiegato da App Runner per creare un container dal codice sorgente.

Per ulteriori informazioni, consulta la sezione [Prezzi di AWS App Runner](#).

Argomenti

- [Creazione di servizi App Runner](#)
- [Gestione dei servizi App Runner](#)

Creazione di servizi App Runner

È possibile creare un servizio App Runner in Toolkit for VS Code utilizzando Explorer e AWS la Command Palette di VS Code. Dopo aver scelto di creare un servizio in una AWS regione

specifica, i passaggi numerati forniti da Command Palette guidano l'utente attraverso il processo di configurazione dell'istanza del servizio su cui viene eseguita l'applicazione.

Prima di creare un servizio App Runner, verifica di aver soddisfatto tutti i [prerequisiti](#). Ciò include specificare le autorizzazioni IAM pertinenti e confermare il repository sorgente specifico che desideri implementare.

Creazione di un servizio App Runner

1. Apri AWS Explorer, se non è già aperto.
2. Fai clic con il pulsante destro del mouse sul nodo App Runner e scegli Create Service (Crea servizio).

Viene visualizzata la palette dei comandi.

3. Per Select a source code location type (Seleziona un tipo di posizione del codice sorgente), scegli ECR o Repository.

Se scegli ECR, specifichi un'immagine di container in un repository gestito da Amazon Elastic Container Registry. Se scegli Repository, specifichi un repository di codice sorgente gestito da un provider di repository supportato. Attualmente, App Runner supporta [GitHub](#) come provider di repository di codice sorgente.

Implementazione da ECR

1. Per Select or enter an image repository (Seleziona o immetti un repository di immagini), scegli o inserisci l'URL del repository di immagini gestito dal tuo registro privato Amazon ECR o dalla galleria pubblica di Amazon ECR.

Note

Se specifichi un repository dalla galleria pubblica di Amazon ECR, assicurati che le implementazioni automatiche siano disattivate: App Runner non supporta questo tipo di implementazioni per immagini in un repository pubblico ECR.

Le distribuzioni automatiche sono disattivate per impostazione predefinita e ciò viene indicato quando l'icona nell'intestazione Command Palette presenta una linea diagonale che la attraversa. Se decidi di attivare le implementazioni automatiche, sarà visualizzato un messaggio per informarti che questa opzione può comportare costi aggiuntivi.

2. Se il passaggio Command Palette riporta che non sono stati trovati tag, è necessario tornare indietro di un passaggio per selezionare un repository che contenga un'immagine del contenitore con tag.
3. Se utilizzi un registro privato Amazon ECR, hai bisogno del ruolo di accesso ECR, AppRunnerECR AccessRole, che consente ad App Runner di accedere alle immagini di Amazon Elastic Container Registry (Amazon ECR) nel tuo account. Scegli l'icona «+» nell'intestazione Command Palette per creare automaticamente questo ruolo. Non è necessario un ruolo di accesso se l'immagine è archiviata nella galleria pubblica di Amazon ECR, dove le immagini sono pubblicamente disponibili.
4. Per Port (Porta), inserisci la porta IP utilizzata dal servizio (porta 8000, ad esempio).
5. Per Configure environment variables (Configura variabili d'ambiente), puoi specificare un file contenente variabili di ambiente utilizzate per personalizzare il comportamento nell'istanza del servizio. Oppure puoi ignorare questa fase.
6. Per Name your service (Assegna un nome al servizio), inserisci un nome univoco senza spazi e premi Invio.
7. Per Select instance configuration (Seleziona configurazione dell'istanza), scegli una combinazione di unità CPU e memoria in GB per la tua istanza di servizio.

Quando il servizio viene creato, il suo stato cambia da Creating (Creazione in corso) a Running (In esecuzione).

8. Appena il servizio è in esecuzione, fai clic con il pulsante destro del mouse e scegli Copy Service URL (Copia URL del servizio).
9. Per accedere all'applicazione implementata, incolla l'URL copiato nella barra degli indirizzi del browser Web.

Implementazione da un repository remoto

1. Per Seleziona una connessione, scegli una connessione a cui GitHub collegare AWS. Le connessioni disponibili per la selezione sono elencate nella pagina delle GitHub connessioni della console App Runner.
2. Per Seleziona un GitHub repository remoto, scegli o inserisci un URL per l'archivio remoto.

I repository remoti già configurati con la gestione del controllo del codice sorgente (SCM) di Visual Studio Code sono disponibili per la selezione. Potrai inoltre incollare un collegamento al repository, se non è presente nell'elenco.

3. Per **Select a branch** (Seleziona un ramo), scegli il ramo Git del codice sorgente che desideri implementare.
4. Per **Choose configuration source** (Scegli sorgente di configurazione), specifica come desideri definire la configurazione del runtime.

Se scegli **Use configuration file** (Utilizza file di configurazione), l'istanza del servizio sarà configurata dalle impostazioni definite dal file di configurazione di `apprunner.yaml`. Questo file si trova nella directory principale del repository dell'applicazione.

Se scegli **Configura tutte le impostazioni qui**, usa la palette **Comando** per specificare quanto segue:

- **Runtime**: scegli Python 3 o Nodejs 12.
 - **Build command** (Comando build): immetti il comando per creare l'applicazione nell'ambiente di runtime dell'istanza del servizio.
 - **Start command** (Comando avvio): immetti il comando per avviare l'applicazione nell'ambiente di runtime dell'istanza del servizio.
5. Per **Port** (Porta), inserisci la porta IP utilizzata dal servizio (porta `8000`, ad esempio).
 6. Per **Configure environment variables** (Configura variabili d'ambiente), puoi specificare un file contenente variabili di ambiente utilizzate per personalizzare il comportamento nell'istanza del servizio. Oppure puoi ignorare questa fase.
 7. Per **Name your service** (Assegna un nome al servizio), inserisci un nome univoco senza spazi e premi **Invio**.
 8. Per **Select instance configuration** (Seleziona configurazione dell'istanza), scegli una combinazione di unità CPU e memoria in GB per la tua istanza di servizio.

Quando il servizio viene creato, il suo stato cambia da **Creating** (Creazione in corso) a **Running** (In esecuzione).

9. Appena il servizio è in esecuzione, fai clic con il pulsante destro del mouse e scegli **Copy Service URL** (Copia URL del servizio).
10. Per accedere all'applicazione implementata, incolla l'URL copiato nella barra degli indirizzi del browser Web.

Note

Se il tentativo di creare un servizio App Runner non riesce, il servizio mostra lo stato Create failed (Creazione non riuscita) in AWS Explorer. Per suggerimenti sulla risoluzione dei problemi, consulta [Quando la creazione del servizio non riesce](#) nella Guida per gli sviluppatori di App Runner.

Gestione dei servizi App Runner

Dopo aver creato un servizio App Runner, puoi gestirlo utilizzando il riquadro AWS Explorer per eseguire le seguenti attività:

- [Sospensione e ripresa dei servizi App Runner](#)
- [Implementazione dei servizi App Runner](#)
- [Visualizzazione dei flussi di log per App Runner](#)
- [Eliminazione di servizi App Runner](#)

Sospensione e ripresa dei servizi App Runner

Se è necessario disattivare temporaneamente l'applicazione Web e interrompere l'esecuzione del codice, è possibile sospendere il servizio AWS App Runner. App Runner riduce a zero la capacità di calcolo del servizio. Quando vuoi eseguire nuovamente l'applicazione, puoi riprendere il servizio App Runner. App Runner fornisce una nuova capacità di calcolo, implementa l'applicazione su di essa ed esegue l'applicazione.

Important

App Runner ti verrà fatturato solo se è in esecuzione. Pertanto, potrai sospendere e riprendere l'applicazione secondo necessità per gestire i costi. Ciò è particolarmente utile negli scenari di sviluppo e test.

Sospensione del servizio App Runner

1. Apri AWS Explorer, se non è già aperto.
2. Espandi App Runner per visualizzare l'elenco dei servizi.

3. Fai clic con il pulsante destro del mouse sul servizio e scegli Pause (Metti in pausa).
4. Nella finestra di dialogo visualizzata, scegli Confirm (Conferma).

Mentre il servizio è in pausa, lo stato del servizio cambia da Running (In esecuzione) a Pausing (In pausa) e poi a Paused (Sospeso).

Come riprendere il servizio App Runner

1. Apri AWS Explorer, se non è già aperto.
2. Espandi App Runner per visualizzare l'elenco dei servizi.
3. Fai clic con il pulsante destro del mouse sul servizio e scegli Resume (Riprendi).

Mentre il servizio si sta riavviando, lo stato del servizio cambia da Resuming (In riavvio) a Running (In esecuzione).

Implementazione dei servizi App Runner

Se scegli l'opzione di implementazione manuale per il servizio, è necessario avviare esplicitamente ogni implementazione sul servizio.

1. Apri AWS Explorer, se non è già aperto.
2. Espandi App Runner per visualizzare l'elenco dei servizi.
3. Fai clic con il pulsante destro del mouse sul servizio e scegli Start Deployment (Avvia implementazione).
4. Mentre l'applicazione viene implementata, lo stato del servizio cambia da Deploying (In implementazione) a Running (In esecuzione).
5. Per verificare che l'applicazione sia stata implementata correttamente, fai clic con il pulsante destro del mouse sullo stesso servizio e scegli Copy Service URL (Copia URL del servizio).
6. Per accedere all'applicazione Web implementata, incolla l'URL copiato nella barra degli indirizzi del browser Web.

Visualizzazione dei flussi di log per App Runner

Usa CloudWatch Logs per monitorare, archiviare e accedere ai tuoi flussi di log per servizi come App Runner. Un flusso di log è una sequenza di log eventi che condividono la stessa origine.

1. Espandi App Runner per visualizzare l'elenco delle istanze dei servizi.
2. Espandi un'istanza di servizio specifica per visualizzare l'elenco dei gruppi di log. Un gruppo di log è un gruppo di flussi di log che condividono le stesse impostazioni di conservazione, monitoraggio e controllo degli accessi.
3. Fai clic con il pulsante destro del mouse su un gruppo di log e scegli View Log Streams (Visualizza flussi di log).
4. Dalla palette dei comandi, scegli un flusso di log dal gruppo.

L'editor VS Code mostra l'elenco degli eventi di registro che compongono lo stream. Puoi decidere di caricare nell'editor eventi passati o più recenti.

Eliminazione di servizi App Runner

Important

Se elimini il servizio App Runner, questo viene rimosso in modo permanente e i dati archiviati vengono eliminati. Se devi ricreare il servizio, App Runner deve recuperare nuovamente il tuo sorgente e crearlo daccapo se si tratta di un repository di codice. La tua applicazione Web ottiene un nuovo dominio App Runner.

1. Apri AWS Explorer, se non è già aperto.
2. Espandi App Runner per visualizzare l'elenco dei servizi.
3. Fai clic con il pulsante destro del mouse su un servizio e scegli Delete Service (Elimina servizio).
4. Nella palette dei comandi, inserisci delete, quindi premi Invio per confermare.

Il servizio eliminato si troverà nello stato Deleting (Eliminazione in corso) e poi scomparirà dall'elenco.

AWSCompositore di applicazioni

È possibile utilizzare il AWS Toolkit for Visual Studio Code per utilizzare il servizio AWS Application Composer. AWS Application Composer è un generatore visivo per AWS applicazioni che aiuta a progettare l'architettura delle applicazioni e a visualizzare l'infrastruttura. AWS CloudFormation

[Per informazioni dettagliate sul servizio Application Composer, consulta la Guida per l'utente di AWS Application Composer. AWS](#)

I seguenti argomenti descrivono come utilizzare AWS Application Composer da. AWS Toolkit for Visual Studio Code

Argomenti

- [Utilizzo di AWS Application Composer dal Toolkit](#)

Utilizzo di AWS Application Composer dal Toolkit

AWS Application Composer for the AWS Toolkit for Visual Studio Code consente di progettare visivamente le applicazioni tramite una tela interattiva. È inoltre possibile utilizzare Application Composer per visualizzare e modificare AWS CloudFormation e AWS Serverless Application Model (AWS SAM) modelli. Mentre lavori con Application Composer, le modifiche vengono archiviate in modo persistente, consentendoti di passare senza interruzioni dalla modifica dei file direttamente nell'editor VS Code all'utilizzo della tela interattiva.

[Per informazioni dettagliate sul servizio AWS Application Composer, informazioni introduttive e tutorial, consulta la Guida per l'utente del AWS servizio Application Composer.](#)

Le sezioni seguenti descrivono come accedere al servizio AWS Application Composer da. AWS Toolkit for Visual Studio Code

Accesso a AWS Application Composer dal Toolkit

Esistono 3 modi principali per accedere ad AWS Application Composer dal Toolkit.

Accesso a AWS Application Composer da un modello esistente

1. Da VS Code, apri un file modello esistente nell'editor VS Code.
2. Dalla finestra dell'editor, fai clic sul pulsante AWS Application Composer situato nell'angolo in alto a destra della finestra dell'editor.
3. AWS Application Composer apre e visualizza il file modello nella finestra dell'editor VS Code.

Accesso a AWS Application Composer dal menu contestuale (tasto destro del mouse)

1. Da VS Code, fai clic con il pulsante destro del mouse sul file modello che desideri aprire con AWS Application Composer.

2. Nel menu contestuale, scegli l'opzione Apri con App Composer.
3. AWS Application Composer apre e visualizza il file modello in una nuova finestra dell'editor VS Code.

Accesso a AWS Application Composer dalla Command Palette

1. Da VS Code apri la palette dei comandi premendo **Cmd + Shift + P** o **Ctrl + Shift + P** (Windows)
2. Nel campo di ricerca, inserisci **AWS Application Composer** e scegli AWS Application Composer quando viene inserito nei risultati.
3. Scegli il file modello che desideri aprire, AWS Application Composer apre e visualizza il file modello in una nuova finestra dell'editor VS Code.

AWS CDK per VS Code

Si tratta di una documentazione di pre-rilascio di una caratteristica nella versione di anteprima ed è soggetta a modifiche.

La AWS CDK servizi ti permette di lavorare con [AWS Cloud Development Kit \(AWS CDK\)](#) applicazioni, oppure applicazioni. Informazioni dettagliate su AWS CDK nella [AWS Cloud Development Kit \(AWS CDK\) Guida per gli sviluppatori](#).

Le app AWS CDK sono composte da blocchi predefiniti noti come [costrutti](#), che includono al loro interno definizioni per gli stack AWS CloudFormation e le risorse AWS. Utilizzando AWS CDK Explorer, puoi visualizzare gli [stack](#) e le [risorse](#) che sono definite nei costrutti AWS CDK. Questa visualizzazione è fornita in un' struttura di riquadro Strumenti di sviluppo all'interno dell'editor di codice di Visual Studio (VS Code).

Questa sezione fornisce informazioni su come accedere e utilizzare AWS CDK nell'editor di codice VS. Si presuppone che tu abbia già [installato e configurato](#) il Toolkit for VS Code per il tuo IDE locale.

Argomenti

- [Lavorare con AWS CDK le applicazioni](#)

Lavorare con AWS CDK le applicazioni

Questa è la documentazione preliminare per una funzionalità della versione di anteprima. ed è soggetta a modifiche.

Usa l'AWS CDK Explorer nel AWS Toolkit for VS Code per visualizzare e lavorare AWS CDK con le applicazioni.

Prerequisiti

- Assicurati che il tuo sistema soddisfi i prerequisiti specificati in [Installazione del Toolkit for VS Code](#).
- Installa l'interfaccia a riga di AWS CDK comando, come descritto nelle prime sezioni di [Guida introduttiva](#) o AWS CDK nella Guida per gli AWS Cloud Development Kit (AWS CDK) sviluppatori.

Important

La AWS CDK versione deve essere 1.17.0 o successiva. Utilizza **cdk --version** sulla riga di comando per vedere quale versione è in esecuzione.

Visualizza un'applicazione AWS CDK

Utilizzando AWS Toolkit for VS AWS CDK Code Explorer, puoi gestire [gli](#) [stack](#) e le risorse archiviati nei costrutti CDK delle tue app. AWS CDK Explorer visualizza le risorse in una visualizzazione ad albero utilizzando le informazioni definite nel `tree.json` file, che viene creato quando si esegue il comando. **cdk synth** Per impostazione predefinita, il `tree.json` file si trova nella `cdk.out` directory di un'app.

Per iniziare a utilizzare Toolkit AWS CDK Explorer, devi creare un'CDK applicazione.

1. Completa i primi passaggi del [tutorial Hello World disponibile](#) nella [Guida per gli AWS CDK sviluppatori](#).

Important

Quando arrivi alla fase del tutorial Deploying the Stack, fermati e torna a questa guida.

Note

È possibile eseguire i comandi forniti nel tutorial, ad esempio, **mkdir cdk init**, sulla riga di comando di un sistema operativo o in una finestra del terminale all'interno dell'editor VS Code.

2. Dopo aver completato i passaggi richiesti del CDK tutorial, apri il CDK contenuto che hai creato nell'editor VS Code.
3. Dal riquadro AWS di navigazione, espandi l'intestazione CDK(Anteprima). CDKLe applicazioni e le risorse associate vengono ora visualizzate nella visualizzazione ad albero di CDK Explorer.

Note importanti

- Quando carichi CDK app nell'editor VS Code, puoi caricare più cartelle contemporaneamente. Ogni cartella può contenere più CDK app, come mostrato nell'immagine precedente. AWS CDK Explorer trova le app nella directory principale del progetto e nelle relative sottodirectory dirette.
- Quando esegui le prime fasi del tutorial, potresti notare che l'ultimo comando eseguito è **cdk synth**, che genera il file `tree.json`. Se modifichi alcuni aspetti di un'CDKapp, ad esempio aggiungi altre risorse, devi eseguire nuovamente il comando per vedere le modifiche riflesse nella visualizzazione ad albero.

Esegui altre operazioni su un' AWS CDK app

Puoi utilizzare l'editor VS Code per eseguire altre operazioni su un'CDKapp, proprio come faresti utilizzando la riga di comando del tuo sistema operativo o altri strumenti. Ad esempio, è possibile aggiornare i file di codice nell'editor e distribuire l'app utilizzando una finestra di VS Code Terminal.

Per provare questo tipo di azioni, usa l'editor VS Code per continuare il [tutorial di Hello World](#) nella Guida per gli AWS CDK sviluppatori. Assicurati di eseguire l'ultimo passaggio, Distruggere le risorse dell'app, in modo da non incorrere in costi imprevisti per il tuo account. AWS

Utilizzo di AWS CloudFormation

AWS Toolkit for Visual Studio Code fornisce il supporto per gli stack [AWS CloudFormation](#).

Utilizzando il Toolkit for VS Code, puoi eseguire determinate attività con AWS CloudFormation stack, ad esempio eliminarli.

Argomenti

- [Eliminazione di un AWS CloudFormation stack](#)
- [Crea un AWS CloudFormation modello utilizzando il AWS Toolkit for Visual Studio Code](#)

Eliminazione di un AWS CloudFormation stack

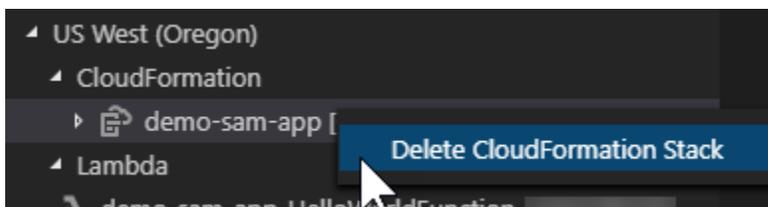
È possibile utilizzare il AWS Toolkit for Visual Studio Code per eliminare gli stack AWS CloudFormation.

Prerequisiti

- Assicurarsi che il sistema soddisfi i prerequisiti specificati nella sezione [Installazione di Toolkit for VS Code](#).
- Verificare che le credenziali configurate in [Autenticazione e accesso](#) includano l'accesso in lettura/scrittura appropriato al servizio AWS CloudFormation. Se in AWS Explorer, in CloudFormation, viene visualizzato un messaggio simile a "Errore durante il caricamento delle risorse CloudFormation", controllare le autorizzazioni allegate a tali credenziali. Le modifiche apportate alle autorizzazioni richiedono alcuni minuti per essere applicate ad AWS Explorer nel codice VS.

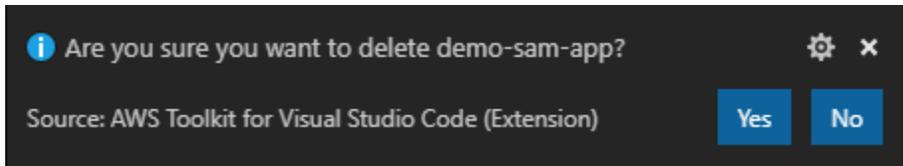
Eliminare uno stack CloudFormation

1. In AWS Explorer, aprire il menu contestuale dello stack AWS CloudFormation da eliminare.



2. Scegliere Elimina CloudFormation Stack.

3. Nel messaggio visualizzato, scegliere Yes (Sì) per confermare l'eliminazione.



Dopo averlo eliminato, lo stack non è più elencato in AWS Explorer.

Crea un AWS CloudFormation modello utilizzando il AWS Toolkit for Visual Studio Code

AWS Toolkit for Visual Studio Code Possono aiutarti nella scrittura di modelli AWS CloudFormation e SAM.

Prerequisiti

Toolkit for VS Code e prerequisiti per le credenziali

- Prima di poter accedere al CloudFormation servizio dal Toolkit for VS Code, devi soddisfare i requisiti indicati nella [guida per l'utente Installing the Toolkit for VS Code](#).
- Le credenziali create [Autenticazione e accesso](#) devono includere l'accesso appropriato in lettura/scrittura al servizio. AWS CloudFormation

Note

Se il CloudFormationservizio visualizza un messaggio di Errore durante il caricamento CloudFormation delle risorse, controlla le autorizzazioni che hai allegato a tali credenziali. Tieni inoltre presente che l'aggiornamento delle modifiche apportate alle autorizzazioni in Explorer potrebbe richiedere alcuni minuti.AWS

CloudFormation prerequisiti del modello

- Installa e abilita l'estensione [Redhat Developer YAML](#) VS Code.
- È necessario essere connessi a Internet quando si utilizza l'estensione Redhat Developer YAML VS Code perché viene utilizzata per scaricare e incassare schemi JSON sul computer.

Scrivere un CloudFormation modello con YAML Schema Support

Il toolkit utilizza il supporto del linguaggio YAML e gli schemi JSON per semplificare il processo di scrittura e i modelli SAM. CloudFormation Funzionalità come la convalida della sintassi e il completamento automatico non solo velocizzano il processo, ma aiutano anche a migliorare la qualità del modello. Quando si seleziona uno schema per il modello, si consiglia di seguire le procedure consigliate.

CloudFormation modello

- Il file ha un'estensione `.yaml` o `.yml`.
- Il file ha un nodo Risorse di primo livello. **AWSTemplateFormatVersion**

Modello SAM

- Tutti i criteri già descritti per CloudFormation
- Il file ha un nodo Transform di primo livello, contenente un valore che inizia con `AWS::Serverless`.

Lo schema verrà applicato al momento della modifica del file. Ad esempio, uno schema di modello SAM verrà applicato dopo aver aggiunto una trasformazione serverless a un CloudFormation modello e aver salvato il file.

Convalida della sintassi

L'estensione YAML applicherà automaticamente la convalida del tipo al modello. Ciò evidenzia le voci con tipi non validi per una determinata proprietà. Se passi il mouse su una voce evidenziata, le estensioni mostrano le azioni correttive.

Completamento automatico

Quando aggiungi nuovi campi, valori enumerati o altri [tipi di risorse](#), puoi avviare la funzionalità di completamento automatico dell'estensione YAML digitando `Ctrl + space`.

Lavorare con CloudWatch Registra utilizzando il AWS Toolkit for Visual Studio Code

Amazon CloudWatch Logs ti consente di centralizzare i log di tutti i tuoi sistemi, applicazioni e AWS servizi che utilizzi, in un unico servizio altamente scalabile. È quindi possibile visualizzarli facilmente, cercarli per codici di errore o modelli specifici, filtrarli in base a campi specifici o archivarli in modo sicuro per analisi future. Per ulteriori informazioni, vedere [Che cos'è Amazon CloudWatch Registra?](#) nell'Amazon CloudWatch Guida per l'utente.

I seguenti argomenti descrivono come utilizzare AWS Toolkit for Visual Studio Code con cui lavorare CloudWatch. Effettua il login con AWS con il tuo account.

Argomenti

- [Visualizzazione dei gruppi di registro e dei flussi di registro CloudWatch utilizzando AWS Toolkit for Visual Studio Code](#)
- [Utilizzo di CloudWatch registro degli eventi nei flussi di log tramite l'AWS Toolkit for Visual Studio Code](#)
- [Ricerca nei gruppi di CloudWatch log](#)

Visualizzazione dei gruppi di registro e dei flussi di registro CloudWatch utilizzando AWS Toolkit for Visual Studio Code

Un flusso di log è una sequenza di eventi di log che condividono la stessa origine. Ciascuna origine di registri separata in CloudWatch Logs costituisce un flusso di log distinto.

Un gruppo di log è un gruppo di flussi di log che condividono le stesse impostazioni di conservazione, monitoraggio e controllo degli accessi. Puoi definire i gruppi di log e specificare quali flussi inserire in ciascun gruppo. Non vi è alcun limite al numero di flussi di log che possono appartenere a un gruppo di log.

Per ulteriori informazioni, consulta [Utilizzo di gruppi di registro e flussi di registro](#) nella Guida per l'utente di Amazon CloudWatch.

Argomenti

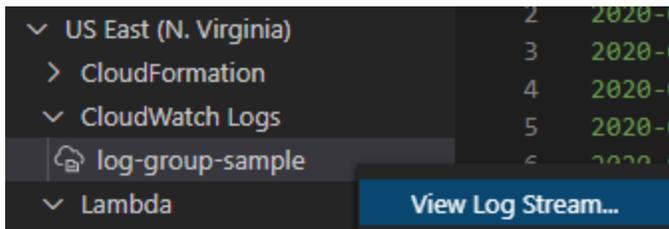
- [Visualizzazione dei gruppi di registro log e dei flussi di registro con il nodo CloudWatch Logs](#)

Visualizzazione dei gruppi di registro log e dei flussi di registro con il nodo CloudWatch Logs

1. In VS Code, scegli **Visualizzazione**, **Esploratore** Per aprire **AWS Explorer**.
2. Fai clic sul nodo **CloudWatch Logs** per espandere l'elenco dei gruppi di registro.

I gruppi di registro per la regione AWS corrente vengono visualizzati sotto il nodo **CloudWatch Logs**.

3. Per visualizzare i flussi di registro in un gruppo di registro, fai clic con il pulsante destro del mouse sul nome del gruppo di registro e scegli **Visualizzazione dei flussi di registro**.



4. Da **Tavolozza dei comandi**, seleziona un flusso di registro dal gruppo da visualizzare.

Note

La **Tavolozza dei comandi** visualizza un timestamp per l'ultimo evento in ogni flusso.

La [Log Stream \(Flusso di log\) editore](#) viene avviato per visualizzare gli eventi di log dello stream.

Utilizzo di CloudWatch registro degli eventi nei flussi di log tramite l'AWS Toolkit for Visual Studio Code

Dopo aver aperto il **Log di Steam Editor**, puoi accedere al registro eventi in ogni flusso. Un registro eventi è una serie di registri di attività registrate dall'applicazione o dalla risorsa monitorata.

Argomenti

- [Visualizzazione e copia delle informazioni sul flusso di registro](#)
- [Salvataggio del contenuto dell'editor del flusso di registro in un file locale](#)

Visualizzazione e copia delle informazioni sul flusso di registro

Quando si apre un flusso di log, Log Stream (Flusso di log) Editor visualizza la sequenza del registro eventi.

1. Per trovare un flusso di log da visualizzare, apri l'Log Stream (Flusso di log) Editor (consulta [Visualizzazione dei gruppi di registro log e dei flussi di registro CloudWatch](#)).

Ogni riga in cui è elencato un evento mostra la data e l'ora di registrazione.

2. Puoi visualizzare e copiare informazioni sugli eventi del flusso utilizzando le seguenti opzioni:
 - Visualizza gli eventi per ora: Visualizza gli eventi di registro più recenti e meno recenti scegliendo Carica degli eventi più recenti o Carica eventi precedenti.

Note

L'editor Log Stream (Flusso di registro) carica inizialmente un batch delle 10.000 righe del registro eventi più recenti o 1 MB di dati di registro (a seconda di quale sia più piccolo). Se scegli Load newer events (Carica eventi più recenti), l'editor visualizza gli eventi registrati dopo il caricamento dell'ultimo batch. Se scegli Load older events (Carica eventi meno recenti), l'editor visualizza un batch di eventi che si sono verificati prima di quelli attualmente visualizzati.

- Copia degli eventi di log: Selezionare gli eventi da copiare, quindi fare clic con il pulsante destro del mouse Copy (Copia) dal menu.
- Copia il nome del log stream: Fare clic con il pulsante destro Log Stream (Flusso di log) editor e scegli Nome flusso di log.

Note

È possibile utilizzare anche l'Tavolozza dei comandi eseguireAWS Nome flusso di log di Toolkit Copy.

Salvataggio del contenuto dell'editor del flusso di registro in un file locale

Puoi scaricare il contenuto del CloudWatch editor di log stream su un file logfile sul computer locale.

Note

Con questa opzione, puoi salvare nel file solo il registro eventi attualmente visualizzato nell'editor del flusso di registro. Ad esempio, se le dimensioni totali di un flusso di registro sono di 5 MB e solo 2 MB sono caricati nell'editor, anche il file salvato conterrà solo 2 MB di dati di registro. Per visualizzare altri dati da salvare, scegli Load newer events (Carica eventi più recenti) o Load older events (Carica eventi meno recenti) nell'editor.

1. Per trovare un flusso di log da copiare, apri l'Editor di flussi di log (consulta [Visualizzazione dei gruppi di registro log e dei flussi di registro CloudWatch](#)).
2. Seleziona Save (Salva) accanto alla scheda che mostra il nome del flusso di log.

Note

È possibile utilizzare anche l'editor di comandi `executeAWSToolkit Save` il contenuto corrente del flusso di log.

3. Utilizza la finestra di dialogo per selezionare o creare una cartella di download per il file di log e fai clic su Save (Salva).

Ricerca nei gruppi di CloudWatch log

È possibile utilizzare Search Log Group per cercare tutti i flussi di log in un gruppo di log.

Per informazioni dettagliate sul servizio Amazon CloudWatch Logs, consulta l'argomento [Working with Log Groups and Log Streams](#) nella Amazon CloudWatch User Guide.

Ricerca di gruppi di log dalla palette di comandi VS Code

Per cercare gruppi di log dalla palette di comandi VS Code, completa i seguenti passaggi.

Per informazioni dettagliate sui filtri e i pattern di Amazon CloudWatch Logs, consulta la sezione sulla [sintassi dei filtri e dei pattern](#) nella Amazon CloudWatch User Guide.

1. Da VS Code, apri la palette dei comandi premendo **cmd+shift+p** (windows:**ctrl+shift+p**).

2. Dalla palette dei comandi, inserisci il comando **AWS: Search Log Group**, quindi selezionalo per aprire la finestra di dialogo del gruppo di log di ricerca in VS Code e segui le istruzioni per continuare.

Note

Dal primo prompt, hai la possibilità di cambiare AWS regione prima di procedere ai passaggi successivi.

3. Dal prompt Select Log Group (1/3), scegli il gruppo di log in cui desideri cercare.
4. Dal prompt Select Time Filter (2/3), scegli il filtro temporale da applicare alla ricerca.
5. Dal prompt Search Log Group... (3/3), inserisci il tuo modello di ricerca nell'apposito campo, quindi premi il **Enter** tasto per continuare o il tasto per annullare la **ESC** ricerca.
6. I risultati della ricerca si aprono nell'editor VS Code, quando la ricerca è completa.

Ricerca di gruppi di log da AWS Explorer

Per cercare gruppi di log da AWS Toolkit for Visual Studio Code Explorer, completare la procedura seguente.

1. Da AWS Toolkit for Visual Studio Code Explorer espandi CloudWatch.
2. Apri il menu contestuale per (fai clic con il pulsante destro del mouse) per il gruppo di log di ricerca in cui desideri cercare, quindi scegli Search Log Group per aprire il prompt di ricerca.
3. Segui le istruzioni selezionando un intervallo di tempo per continuare.
4. Quando richiesto, inserisci il modello di ricerca nell'apposito campo, quindi premi il **Enter** tasto per continuare o il **ESC** tasto per annullare la ricerca.
5. I risultati della ricerca si aprono nell'editor VS Code quando la ricerca è completa.

Lavorare con i risultati del registro di ricerca

Dopo aver completato con successo una ricerca nei gruppi di CloudWatch log, i risultati della ricerca vengono aperti nell'editor VS Code. Le seguenti procedure descrivono come lavorare con i risultati dei log di ricerca.

 Note

Quando si visualizza un singolo flusso di log, le seguenti funzionalità sono limitate ai risultati del flusso di log attualmente attivo.

Salvataggio dei risultati del gruppo di log di ricerca

Per salvare localmente i risultati del gruppo del registro di ricerca, completa i seguenti passaggi.

1. Dai risultati del gruppo del registro di ricerca, scegli il pulsante con l'icona Salva registro su file, situato nell'angolo in alto a destra dell'editor VS Code.
2. Dal prompt Salva con nome, specifica il nome e la posizione in cui desideri salvare il file.
3. Scegliendo OK, il file viene salvato sul computer locale.

Modifica dell'intervallo di tempo (l'intervallo di tempo)

Per modificare l'intervallo di tempo attivo nei risultati del gruppo del registro di ricerca, completa i passaggi seguenti.

1. Dai risultati del gruppo del registro di ricerca, scegli Cerca per data... pulsante icona, situato nell'angolo in alto a destra dell'editor VS Code.
2. Dal prompt Select Time Filter, scegli un nuovo intervallo di tempo per i risultati del registro di ricerca.
3. I risultati vengono aggiornati quando viene chiuso il prompt Select Time Filter.

Modifica del modello di ricerca

Per modificare il modello di ricerca attivo nei risultati del gruppo del registro di ricerca, completa i passaggi seguenti.

1. Dai risultati del gruppo del registro di ricerca, scegli Cerca per modello... pulsante icona, situato nell'angolo in alto a destra dell'editor VS Code.
2. Dal prompt Search Log Group, inserisci il nuovo modello di ricerca nel campo fornito.
3. Premi il **Enter** tasto per chiudere il prompt e aggiornare i risultati con il nuovo modello di ricerca.

Utilizzo di Amazon Elastic Container Registry

Amazon Elastic Container Registry (Amazon ECR) è un servizio di registro di container AWS gestito, sicuro e scalabilità. Diverse funzioni del servizio Amazon ECR sono accessibili dal Toolkit for VS Code Explorer.

- Creazione di un repository.
- Creazione di un servizio AWS App Runner per il repository o l'immagine taggata.
- Accesso ai tag di immagine e agli URI o ARN del repository.
- Eliminazione di tag e repository di immagini.

Puoi anche accedere all'intera gamma di funzioni Amazon ECR tramite la console VS Code integrando l'AWSCLI e altre piattaforme, con VS Code.

Per ulteriori informazioni su Amazon ECR, consulta la pagina [Che cos'è Amazon ECR?](#) nella Guida per l'utente di Amazon Elastic Container Registry.

Prerequisiti

Devi completare questi passaggi per accedere al servizio Amazon ECR da VS Code Explorer.

Creazione di un utente IAM

Prima di poter accedere a un AWS servizio, come Amazon ECR, devi fornire le credenziali. In questo modo il servizio può stabilire se l'utente dispone delle autorizzazioni di accesso alle risorse presenti nel servizio. È sconsigliabile accedere AWS direttamente tramite le credenziali dell'AWS account root. Utilizza invece AWS Identity and Access Management (IAM) per creare un utente IAM e quindi aggiungilo a un gruppo IAM con autorizzazioni amministrative. Puoi quindi accedere ad AWS usando un URL speciale e le credenziali dell'utente IAM.

Se hai effettuato la AWS registrazione a senza creare un utente IAM per te stesso, puoi crearne uno utilizzando la console IAM.

Per creare un utente amministratore, scegli una delle seguenti opzioni.

Scelta di un modo per gestire il tuo amministratore	Per	By	Puoi anche
In IAM Identity Center (Consigliato)	Usa credenziali a breve termine per accedere a AWS. Ciò è in linea con le best practice per la sicurezza. Per informazioni sulle best practice, consulta Best practice per la sicurezza in IAM nella Guida per l'utente di IAM.	Segui le istruzioni riportate in Nozioni di base nella Guida per l'utente di AWS IAM Identity Center.	Configura l'accesso programmatico seguendo quanto riportato in Configurazione della AWS CLI per utilizzare AWS IAM Identity Center nella Guida per l'utente di AWS Command Line Interface.
In IAM (Non consigliato)	Usa credenziali a lungo termine per accedere a AWS.	Segui le istruzioni in Creazione del primo utente e gruppo di utenti IAM di amministrazione nella Guida per l'utente IAM.	Configura l'accesso programmatico seguendo quanto riportato in Gestione delle chiavi di accesso per gli utenti IAM nella Guida per l'utente IAM.

Per accedere come nuovo utente IAM, disconnettiti dallaAWS console, quindi utilizza l'URL seguente. Nel seguente URL, dove `your_aws_account_id` è il tuo numero diAWS account senza i trattini (ad esempio se il tuo numero diAWS account è1234-5678-9012, il tuo IDAWS account è123456789012):

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

Immettere il nome utente e la password di IAM appena creati. Una volta effettuato l'accesso, la barra di navigazione visualizza "your_user_name @ your_aws_account_id".

Se non desideri che l'URL per la tua pagina di accesso contenga il tuo ID account di AWS, è possibile creare un alias dell'account. Nel pannello di controllo IAM, seleziona Personura e inserisci un alias dell'account. Questo può essere il nome della tua azienda. Per maggiori informazioni, consulta [l'IDAWS account e il suo alias](#) nella Guida per l'utente di IAM.

Per effettuare l'accesso dopo aver creato un alias dell'account, utilizzare il seguente URL:

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

Per verificare il link di accesso degli utenti IAM al tuo account, apri la console IAM e controlla in IAM users sign-in link (Link di accesso utenti IAM) nel pannello di controllo.

Per ulteriori informazioni su IAM, consulta la [Guida per l'utente AWS Identity and Access Management](#).

Inste configura Docker

È possibile installare e configurare Docker selezionando il sistema operativo preferito dalla guida per l'utente di [Install Docker Engine](#) e seguendo le istruzioni.

Installazione e configurazione dellaAWS CLI versione 2

Installa e configura laAWS CLI versione 2 selezionando il tuo sistema operativo preferito dalla guida per l'utente di [installazione, aggiornamento e disinstallazione dellaAWS CLI versione 2](#).

Argomenti

- [Lavorare con il servizio Amazon Elastic Container Registry in VS Code](#)

Lavorare con il servizio Amazon Elastic Container Registry in VS Code

È possibile accedere al servizio Amazon Elastic Container Registry (Amazon ECR) direttamente dalAWSEsplora in VS Code e usarlo per inviare un'immagine del programma a un repository Amazon ECR. Per iniziare, è necessario eseguire le seguenti operazioni:

1. Crea un Dockerfile che contenga le informazioni necessarie per creare un'immagine.
2. Crea un'immagine da quel Dockerfile e tagga l'immagine per l'elaborazione.

3. Crea un repository all'interno della tua istanza Amazon ECR.
4. Invia l'immagine con tag al tuo repository.

Sezioni

- [Prerequisiti](#)
- [1. Creazione di un Dockerfile](#)
- [2. Crea la tua immagine dal tuo Dockerfile](#)
- [3. Creazione di un nuovo repository](#)
- [4. Invia, tira ed elimina immagini](#)

Prerequisiti

Prima di poter utilizzare la funzione di servizio Amazon ECR del Toolkit for VS Code, devi soddisfare questi requisiti [prerequisiti](#).

1. Creazione di un Dockerfile

Docker utilizza un file chiamato Dockerfile per definire un'immagine che può essere inviata e archiviata in un repository remoto. Prima di poter caricare un'immagine in un repository ECR, è necessario creare un Dockerfile e quindi creare un'immagine da quel Dockerfile.

Creazione di un Dockerfile

1. Usa il Toolkit for VS Code explorer per accedere alla directory in cui desideri archiviare il tuo Dockerfile.
2. Creare un nuovo file denominato Dockerfile.

Note

VS Code potrebbe richiedere di selezionare un tipo di file o un'estensione di file. In questo caso, seleziona `testo normale`. VS Code ha un'estensione «dockerfile». Tuttavia, non è consigliato l'uso. Questo perché l'estensione potrebbe causare conflitti con alcune versioni di Docker o altre applicazioni associate.

Modifica il tuo Dockerfile usando VS Code

Se il Dockerfile ha un'estensione file, aprire il menu contestuale per il file (clic con il pulsante destro del mouse) e rimuovere l'estensione.

Dopo aver rimosso l'estensione del file dal Dockerfile:

1. Apri il Dockerfile vuoto direttamente in VS Code.
2. Copia il contenuto del seguente esempio nel tuo Dockerfile:

Example modello immagine di Dockerfile

```
FROM ubuntu:18.04

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

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

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

EXPOSE 80

CMD /root/run_apache.sh
```

Questo è un Dockerfile che utilizza un'immagine Ubuntu 18.04. Le istruzioni aggiornano le cache dei pacchetti. Installa i pacchetti software per il server Web e infine scrivi il contenuto «Hello World!» nella root del documento del server Web. L'istruzione espone la porta 80 sul container e il comando avvia il server Web.

3. Salva il tuo Dockerfile.

⚠ Important

Assicurati che il tuo Dockerfile non abbia un'estensione allegata al nome. Un Dockerfile con estensioni potrebbe causare conflitti con alcune versioni di Docker o altre applicazioni associate.

2. Crea la tua immagine dal tuo Dockerfile

Il Dockerfile creato contiene le informazioni necessarie per creare un'immagine per un programma. Prima di poter eseguire il push di quell'immagine nella tua istanza Amazon ECR, devi prima creare l'immagine.

Crea un'immagine dal tuo Dockerfile

1. Usa l'interfaccia della riga di comando Docker o una CLI integrata con la tua istanza di Docker per navigare nella directory che contiene il tuo Dockerfile.
2. Eseguirebuild Dockercomando per creare l'immagine definita nel Dockerfile.

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

3. EseguireImmagini Dockercomando per verificare che la creazione dell'immagine sia andata a buon fine.

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

Example Output di esempio:

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

4.

Note

Questo passaggio non è necessario per creare o spingere la tua immagine, ma puoi vedere come funziona l'immagine del programma quando viene eseguita.

Per eseguire l'immagine appena creata, utilizzare il comando Docker.

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

La pl'opzione specificata nell'esempio precedente mappa l'oggetto esposto porta 80 sul contenitore per porta 80 del sistema host. Se stai eseguendo Docker localmente, vai a <http://localhost:80> utilizzando il tuo browser Web. Se il programma è stato eseguito correttamente, un «Hello World!» viene visualizzata l'istruzione.

Per ulteriori informazioni su Esegui il Docker comando, vedere [riferimento di riferimento di riferimento di](#) sul sito Web Docker.

3. Creazione di un nuovo repository

Per caricare la tua immagine nella tua istanza Amazon ECR, crea un nuovo repository in cui possa essere archiviata.

Creazione di un nuovo repository Amazon ECR

1. Dal codice VS Bar delle attività, scegli il AWS Icona del Toolkit.
2. Espandere il AWS Esploratore menù.
3. Individua il valore predefinito AWS Regione associata al tuo AWS conto. Quindi, selezionalo per visualizzare un elenco dei servizi disponibili tramite il Toolkit for VS Code.
4. Seleziona ECR + opzione per iniziare il Creazione di un nuovo repository processo.
5. Segui le istruzioni per completare il processo.
6. Una volta completata l'operazione, è possibile accedere al nuovo repository dal contenuto ECR Sezione della AWS Menu Explorer.

4. Invia, tira ed elimina immagini

Dopo aver creato un'immagine dal Dockerfile e creato un repository, puoi inserire la tua immagine nel tuo repository Amazon ECR. Inoltre, utilizzando il **AWSExplorer** con Docker e **AWSCLI**, è possibile effettuare le seguenti operazioni:

- Estrae un'immagine dal tuo repository.
- Elimina un'immagine archiviata nel tuo repository.
- Elimina il tuo repository.

Autentica Docker con il tuo registro predefinito

L'autenticazione è necessaria per lo scambio di dati tra istanze Amazon ECR e Docker. Per autenticare Docker con il registro:

1. Apri un sistema operativo da riga di comando connesso alla tua istanza di **AWSCLI**.
2. Utilizzo dell'`get-login-password` metodo per autenticarsi nel registro ECR privato.

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

Important

Nel comando precedente, è necessario aggiornare sia il contenuto **regione** la **AWS_account_id** dalle informazioni specifiche per il tuo **AWS** conto.

Aggiungere un tag all'immagine e inviarla al tuo repository

Dopo aver autenticato Docker con l'istanza di **AWS**, invia un'immagine al tuo repository.

1. Utilizzo dell'Immagini Docker comando per visualizzare le immagini memorizzate localmente e identificare quella che si desidera etichettare.

```
docker images
```

Example Output di esempio:

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

2. Contrassegna la tua immagine con il tag Docker comando.

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

3. Invia l'immagine con tag al tuo repository con il contenuto tag Docker comando.

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

Example Output di esempio:

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

Dopo che l'immagine con tag è stata caricata correttamente nel tuo repository, è visibile nella AWS Menu Explorer.

Estrae un'immagine da Amazon ECR

- Puoi estrarre un'immagine nella tua istanza locale di tag Docker comando.

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

Example Output di esempio:

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

Eliminazione di un'immagine dal repository Amazon ECR

Sono disponibili due metodi per eliminare un'immagine da VS Code. Il primo metodo consiste nell'utilizzare ilAWSExplorer.

1. DaAWSExplorer, espandi laECRmenù
2. Espandere il repository da cui si desidera eliminare un'immagine
3. Scegli il tag immagine associato all'immagine che desideri eliminare, aprendo il menu contestuale (tasto destro del mouse)
4. SelezionaElimina il tag...opzione per eliminare tutte le immagini memorizzate associate a quel tag

Eliminazione di un'immagine utilizzando ilAWSCLI

- È anche possibile eliminare un'immagine dal repository con il contenutoAWSecr batch-delete-imagecomando.

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

Example Output di esempio:

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

Elimina un repository dalla tua istanza Amazon ECR

Esistono due metodi per eliminare un repository da VS Code. Il primo metodo consiste nell'utilizzare ilAWSExplorer.

1. DaAWSExplorer, espandi laECRmenù
2. Scegli il repository che desideri eliminare aprendo il menu contestuale (clic con il pulsante destro del mouse)
3. SelezionaElimina repository...opzione al repository scelto

Eliminazione di un repository Amazon ECR daAWSCLI

- È possibile eliminare un repository con ilAWSecr delete-repositorycomando.

Note

Per impostazione predefinita, non è possibile eliminare un repository che contiene immagini. Tuttavia, il--forceflag lo consente.

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

Example Output di esempio:

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

Amazon Elastic Container Service

LaAWS Toolkit for Visual Studio Codefornisce un certo supporto per[Amazon Elastic Container Service \(Amazon ECS\)](#). Toolkit for VS Code ti assiste in determinate operazioni correlate a Amazon ECS, ad esempio nella creazione di definizioni di attività.

Argomenti

- [Utilizzo di file di definizione delle attività di IntelliSense per Amazon ECS](#)
- [Dirigente di Amazon Elastic Container Service in AWS Toolkit for Visual Studio Code](#)

Utilizzo di file di definizione delle attività di IntelliSense per Amazon ECS

Una delle cose che è possibile fare quando si utilizza Amazon Elastic Container Service (Amazon ECS) è creare definizioni di attività, come descritto nella sezione[Creazione di una definizione di attività](#)dagliGuida per sviluppatori Amazon Elastic Container Service. Quando si installa ilAWS Toolkit for Visual Studio Code, l'installazione include la funzionalità IntelliSense per i file di definizione delle attività di Amazon ECS.

Prerequisiti

- Assicurarsi che il sistema soddisfi i prerequisiti specificati nella[Installazione di Toolkit for VS Code](#).

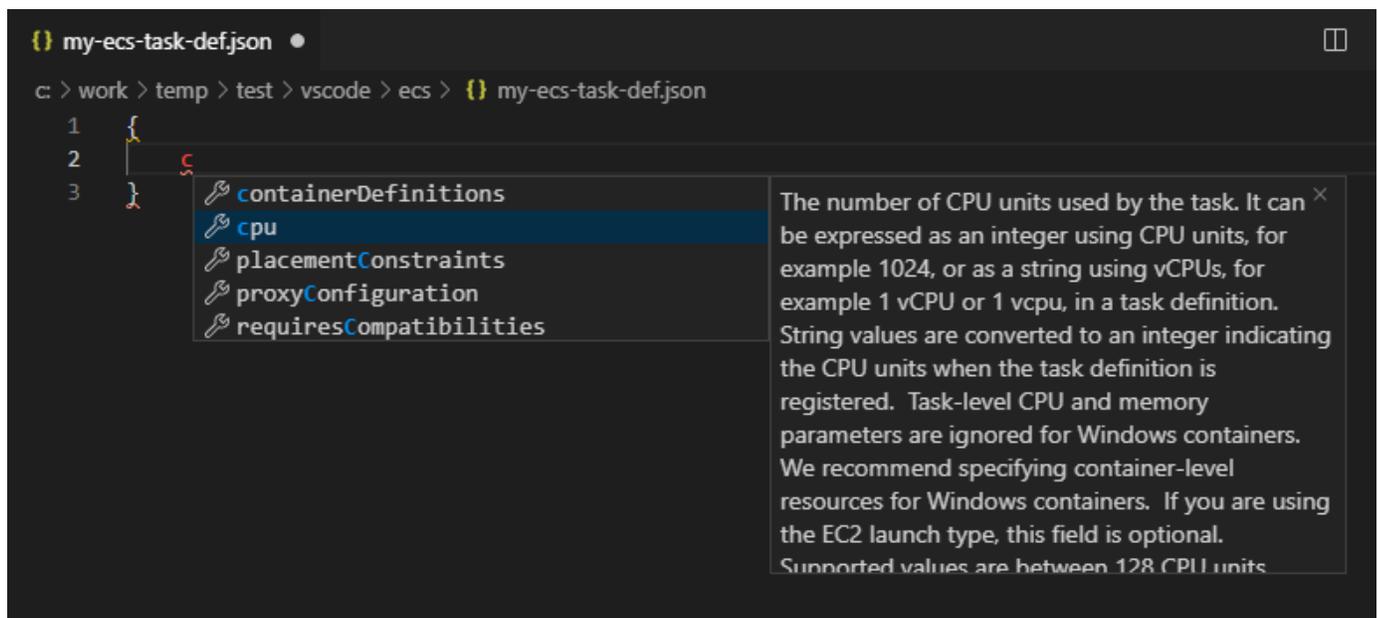
Utilizzare IntelliSense nei file di definizione delle attività di Amazon ECS

Nell'esempio seguente viene illustrato come sfruttare IntelliSense nei file di definizione delle attività di Amazon ECS.

1. Creare un file JSON per la definizione delle attività Amazon ECS. Il nome del file deve terminare con `ecs-task-def.json`, ma può essere preceduto da caratteri aggiuntivi.

Per questo esempio, creare un file denominato `my-ecs-task-def.json`

2. Aprire il file in un editor VS Code e immettere le parentesi graffe iniziali.
3. Immettere la lettera "c" come se si volesse aggiungere `cpu` alla definizione. Osservare la finestra di dialogo IntelliSense che si apre, che è simile alla seguente.



Dirigente di Amazon Elastic Container Service in AWS Toolkit for Visual Studio Code

Puoi emettere singoli comandi in un contenitore Amazon Elastic Container Service (Amazon ECS) con, utilizzando AWS Toolkit for Visual Studio Code la funzionalità Amazon ECS Exec.

⚠ Important

L'attivazione e la disabilitazione di Amazon ECS Exec modifica lo stato delle risorse nel tuo account. AWS Sono inclusi l'arresto e il riavvio del servizio. Alterare lo stato delle risorse mentre Amazon ECS Exec è abilitato può portare a risultati imprevedibili. Per ulteriori

informazioni su Amazon ECS, consulta la guida per sviluppatori [Using Amazon ECS Exec for Debugging](#).

Prerequisiti di Amazon ECS Exec

Prima di poter utilizzare la funzionalità Amazon ECS Exec, è necessario soddisfare alcune condizioni preliminari.

Requisiti di Amazon ECS

A seconda che le attività siano ospitate su Amazon EC2 o AWS Fargate (Fargate) Amazon ECS Exec ha requisiti di versione diversi.

- Se utilizzi Amazon EC2, devi utilizzare un'AMI ottimizzata per Amazon ECS rilasciata dopo il 20 gennaio 2021, con una versione dell'agente 1.50.2 o successiva. Ulteriori informazioni sono disponibili nella guida per gli sviluppatori sulle [AMI ottimizzate per Amazon ECS](#).
- Se utilizzi AWS Fargate, devi utilizzare la versione 1.4.0 o successiva della piattaforma. Ulteriori informazioni sui requisiti di Fargate sono disponibili nella guida per gli sviluppatori sulle [versioni della piattaforma AWS Fargate](#).

AWS configurazione dell'account e autorizzazioni IAM

Per utilizzare la funzionalità Amazon ECS Exec, devi avere un cluster Amazon ECS esistente associato al tuo account. AWS Amazon ECS Exec utilizza Systems Manager per stabilire una connessione con i container del cluster e richiede autorizzazioni dei ruoli IAM specifiche per le attività per comunicare con il servizio SSM.

Puoi trovare informazioni sui ruoli e sulle policy IAM specifiche per Amazon ECS Exec nella sezione [IAM permissions required for ECS Exec](#) della Guida per gli sviluppatori.

Utilizzo di Amazon ECS Exec

Puoi abilitare o disabilitare Amazon ECS Exec direttamente da AWS Explorer nel Toolkit for VS Code. Dopo aver abilitato Amazon ECS Exec, puoi scegliere i contenitori dal menu Amazon ECS e quindi eseguire comandi su di essi.

Abilitazione di Amazon ECS Exec

1. Da AWS Explorer, individua ed espandi il menu Amazon ECS.

2. Espandere il cluster con il servizio che si desidera modificare.
3. Aprire il menu contestuale (clic con il pulsante destro del mouse) del servizio e selezionare **Enable Command Execution** (Abilita esecuzione del comando).

 **Important**

Questa operazione avvia una nuova implementazione del servizio e potrebbe richiedere alcuni minuti. Per ulteriori informazioni, consultare la nota all'inizio di questa sezione.

Disabilitazione di Amazon ECS Exec

1. Da AWS Explorer, individua ed espandi il menu Amazon ECS.
2. Espandi il cluster che ospita il servizio che desideri.
3. Aprire il menu contestuale (clic con il pulsante destro del mouse) del servizio e selezionare **Disable Command Execution** (Disabilita esecuzione del comando).

 **Important**

Questa operazione avvia una nuova implementazione del servizio e potrebbe richiedere alcuni minuti. Per ulteriori informazioni, consulta la nota all'inizio di questa sezione.

Esecuzione di comandi su un container

Per eseguire comandi su un contenitore utilizzando AWS Explorer, è necessario abilitare Amazon ECS Exec. Se non è abilitato, consulta la procedura [Enabling ECS Exec](#) in questa sezione.

1. Da AWS Explorer, individua ed espandi il menu Amazon ECS.
2. Espandi il cluster che ospita il servizio che desideri.
3. Espandere il servizio per elencare i container associati.
4. Aprire il menu contestuale (clic con il pulsante destro del mouse) sul container e scegliere **Run Command in Container** (Esegui comando nel container).
5. Si aprirà un prompt con un elenco di attività in esecuzione, scegli il Task ARN che desideri.

Note

Se per quel servizio è in esecuzione solo un'attività, questa verrà selezionata automaticamente e questo passaggio verrà ignorato.

- Quando richiesto, digita il comando che desideri eseguire e premi Invio per elaborarlo.

Utilizzo di schemi Amazon EventBridge

LaAWS Toolkit for Visual Studio Code(VS Code) fornisce supporto per[Amazon EventBridge](#). Utilizzando Toolkit for VS Code, puoi utilizzare alcuni aspetti di EventBridge, come gli schemi.

Argomenti

- [Lavorare con Amazon EventBridge Schemas](#)

Lavorare con Amazon EventBridge Schemas

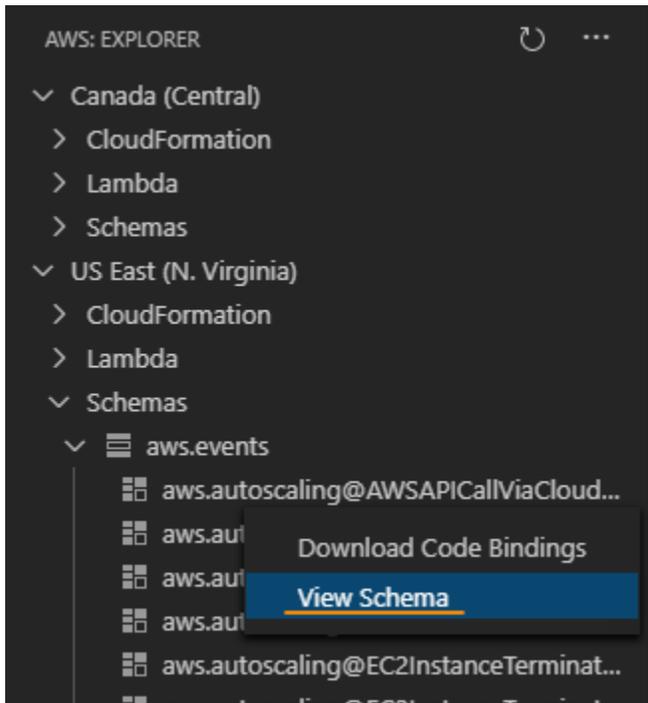
Puoi utilizzare AWS Toolkit for Visual Studio Code (VS Code) per eseguire varie operazioni sugli [EventBridge schemi Amazon](#).

Prerequisiti

- Assicurati che il tuo sistema soddisfi i prerequisiti specificati in [Installazione del Toolkit for VS Code](#).
- Lo EventBridge schema con cui vuoi lavorare deve essere disponibile nel tuo AWS account. In caso contrario, crearlo o caricarlo. Consulta [Amazon EventBridge Schemas](#) nella [Amazon EventBridge User Guide](#).

Visualizzazione di uno schema disponibile

- In AWS Explorer, espandere Schemas (Schemi).
- Espandere il nome del registro che contiene lo schema da visualizzare. Ad esempio, molti degli schemi AWS forniti si trovano nel registro aws.events.
- Per visualizzare uno schema nell'editor, aprire il menu contestuale dello schema, quindi scegliere View Schema (Visualizza schema).

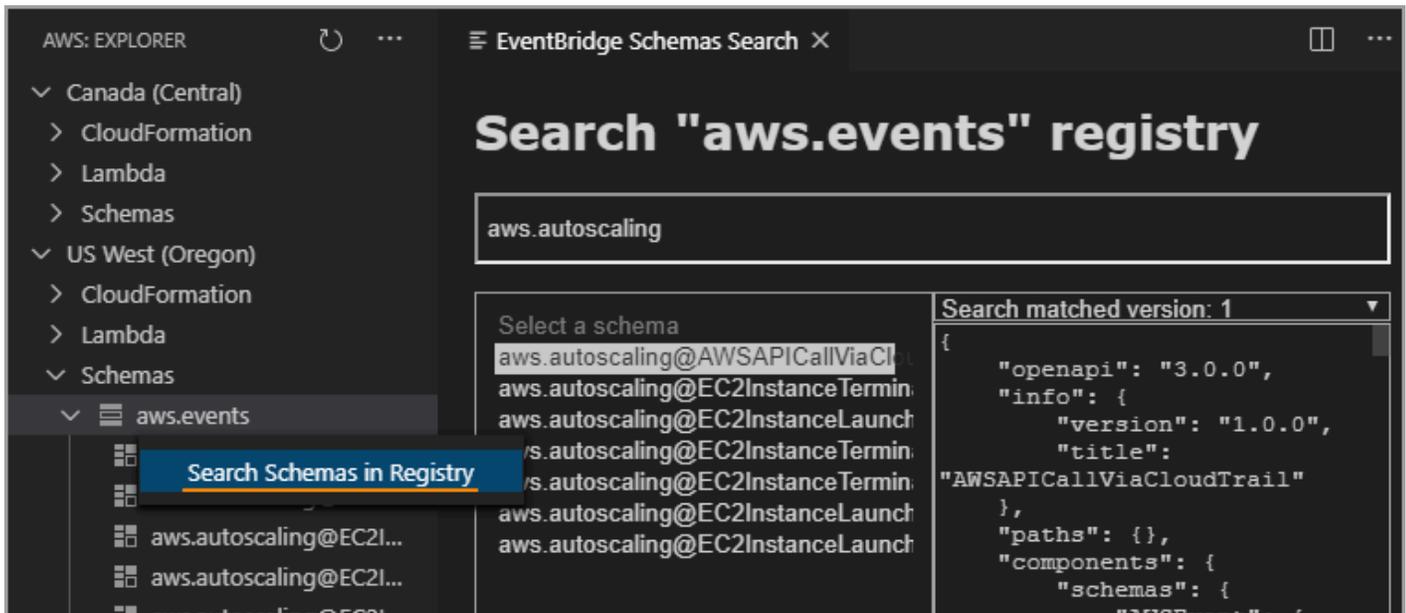


Individuazione di uno schema disponibile

In AWS Explorer, eseguire una o più delle operazioni seguenti:

- Iniziare a digitare il titolo dello schema che si desidera trovare. In AWS Explorer vengono evidenziati i titoli degli schemi che contengono una corrispondenza. (È necessario espandere un registro per visualizzare i titoli evidenziati.)
- Aprire il menu contestuale per Schemas (Schemi), quindi scegliere Search Schemas (Cerca schemi). In alternativa, espandere Schemas (Schemi), aprire il menu contestuale per il registro che contiene lo schema che si desidera trovare, quindi scegliere Search Schemas in Registry (Cerca schemi nel registro). Nella finestra di dialogo EventBridge Schemas Search, iniziate a digitare il titolo dello schema che desiderate trovare. Nella finestra di dialogo vengono visualizzati i titoli degli schemi che contengono una corrispondenza.

Per visualizzare lo schema nella finestra di dialogo, selezionare il titolo dello schema.



Generazione del codice per uno schema disponibile

1. In AWS Explorer, espandere Schemas (Schemi).
2. Espandere il nome del registro che contiene lo schema per il quale si desidera generare codice.
3. Fare clic con il pulsante destro del mouse sul titolo dello schema, quindi scegliere Download code bindings (Scarica associazioni codice).
4. Nelle pagine della procedura guidata risultanti, scegliere quanto segue:
 - La Version (Versione) dello schema.
 - Il linguaggio di associazione codice
 - La cartella dell'area di lavoro in cui si desidera archiviare il codice generato sul computer di sviluppo locale

Access Analyzer di AWS IAM

È possibile eseguire controlli delle politiche di [AWS Identity and Access Management \(IAM\) Access Analyzer](#) sulle IAM politiche create in AWS CloudFormation modelli, piani Terraform e documenti relativi alle JSON policy, utilizzando IAM Access Analyzer in. AWS Toolkit for Visual Studio Code

IAMI controlli delle policy di Access Analyzer includono la convalida delle policy e i controlli personalizzati delle policy. La convalida delle politiche consente di convalidare le IAM politiche

in base agli standard descritti nella sezione [Grammatica del linguaggio delle IAM JSON policy](#) e nelle [migliori pratiche di AWS sicurezza negli IAM argomenti, disponibili nella Guida](#) per l'utente. AWS Identity and Access Management I risultati della convalida delle politiche includono avvisi di sicurezza, errori, avvertenze generali e suggerimenti sulle politiche.

Puoi anche eseguire controlli personalizzati delle policy per i nuovi accessi, in base ai tuoi standard di sicurezza. A ogni controllo delle policy personalizzate per i nuovi accessi è associato un addebito. Per informazioni dettagliate sui prezzi, consulta il sito dei [prezzi di AWS IAM Access Analyzer](#). Per informazioni dettagliate sui controlli delle policy di IAM Access Analyzer, consulta l'argomento [Controlli per la convalida delle policy](#) nella Guida per l'AWS Identity and Access Management utente.

I seguenti argomenti descrivono come utilizzare i controlli delle policy di IAM Access Analyzer in. AWS Toolkit for Visual Studio Code

Argomenti

- [Utilizzo di AWS IAM Access Analyzer](#)

Utilizzo di AWS IAM Access Analyzer

Le sezioni seguenti descrivono come eseguire la convalida delle IAM policy e i controlli personalizzati delle policy in. AWS Toolkit for Visual Studio Code Per ulteriori dettagli, vedere i seguenti argomenti nella Guida per l' AWS Identity and Access Management utente: [Convalida delle policy di IAM Access Analyzer e controlli delle policy](#) personalizzati di [IAMAccess Analyzer](#).

Prerequisiti

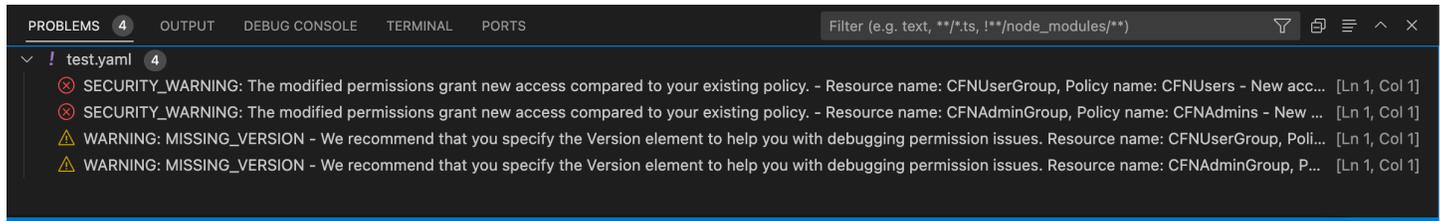
I seguenti prerequisiti devono essere soddisfatti prima di poter utilizzare i controlli delle policy di IAM Access Analyzer dal Toolkit.

- Installa Python versione 3.6 o successiva.
- Installa il [IAMPolicy Validator per AWS CloudFormation](#) o il [IAMPolicy Validator per Terraform](#) richiesto dagli CLI strumenti Python e specificato nella finestra Policy Checks. IAM
- Configura le credenziali del tuo ruolo. AWS

IAMAccess Analyzer: controlli delle policy

È possibile eseguire controlli delle politiche per i AWS CloudFormation modelli, i piani Terraform e i documenti relativi alle JSON politiche, utilizzando il. AWS Toolkit for Visual Studio Code I risultati del

controllo sono visualizzabili nel pannello VS Code Problems. L'immagine seguente mostra il pannello VS Code Problems.



IAMAccess Analyzer offre 4 tipi di controlli:

- Convalida della politica
- CheckAccessNotGranted
- CheckNoNewAccess
- CheckNoPublicAccess

Le seguenti sezioni descrivono come eseguire ogni tipo di controllo.

Note

Configura le credenziali del tuo AWS ruolo prima di eseguire qualsiasi tipo di controllo. I file supportati includono i seguenti tipi di documenti: AWS CloudFormation modelli, piani Terraform e JSON documenti Policy

I riferimenti ai percorsi dei file vengono in genere forniti dall'amministratore o dal team di sicurezza e possono essere un percorso di file di sistema o un bucket Amazon S3. URI

Per utilizzare un bucket Amazon S3URI, il tuo ruolo attuale deve avere accesso al bucket Amazon S3.

Viene addebitato un costo per ogni controllo della policy personalizzato. Per informazioni dettagliate sulla politica personalizzata, consulta la guida ai prezzi di [AWS IAMAccess Analyzer](#).

Esecuzione della politica di convalida

Il controllo Validate Policy, noto anche come convalida delle politiche, convalida la policy rispetto alla grammatica e alle best practice IAM delle policy. AWS Per ulteriori informazioni, consulta la sezione [Grammatica del linguaggio delle IAM JSON politiche e le migliori pratiche di AWS sicurezza negli IAM argomenti, disponibili nella Guida](#) per l'utente. AWS Identity and Access Management

1. Da VS Code, apri un file supportato che contiene AWS IAM le politiche, nell'editor VS Code.
2. Per aprire i controlli dei criteri di IAM Access Analyzer, apri il VS Code Command Palette premendo **CRTL+Shift+P**, cerca **IAM Policy Checks**, quindi fai clic per aprire il riquadro IAM Policy Checks nell'editor VS Code.
3. Dal riquadro IAM Policy Checks, seleziona il tipo di documento dal menu a discesa.
4. Nella sezione Convalida criteri, scegli il pulsante Esegui convalida dei criteri per eseguire il controllo Convalida dei criteri.
5. Dal pannello Problemi di VS Code, esamina i risultati del controllo delle politiche.
6. Aggiorna la tua politica e ripeti questa procedura, eseguendo nuovamente il controllo di convalida della politica fino a quando i risultati del controllo della politica non mostreranno più avvisi o errori di sicurezza.

In esecuzione CheckAccessNotGranted

CheckAccessNotGranted è un controllo personalizzato delle politiche per verificare che IAM azioni specifiche non siano consentite dalla politica.

Note

I riferimenti ai percorsi dei file vengono in genere forniti dall'amministratore o dal team di sicurezza e possono essere un percorso di file di sistema o un bucket Amazon S3. URI
Per utilizzare un bucket Amazon S3URI, il tuo ruolo attuale deve avere accesso al bucket Amazon S3. È necessario specificare almeno un'azione o una risorsa e il file deve essere strutturato secondo il seguente esempio:

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

1. Da VS Code, apri un file supportato che contiene AWS IAM Policies, nell'editor VS Code.
2. Per aprire i controlli dei criteri di IAM Access Analyzer, apri il VS Code Command Palette premendo **CRTL+Shift+P**, cerca **IAM Policy Checks**, quindi fai clic per aprire il riquadro IAM Policy Checks nell'editor VS Code.
3. Dal riquadro IAM Policy Checks, seleziona il tipo di documento dal menu a discesa.

4. Dalla sezione Custom Policy Checks, seleziona. CheckAccessNotGranted
5. Nel campo di immissione di testo, puoi inserire un elenco separato da virgole che contiene azioni e risorse. ARNs È necessario fornire almeno un'azione o una risorsa.
6. Scegli il pulsante Run Custom Policy Check.
7. Dal pannello Problemi di VS Code, esamina i risultati del controllo delle politiche. I controlli delle politiche personalizzate restituiscono un PASS FAIL risultato.
8. Aggiorna la polizza e ripeti questa procedura, eseguendo nuovamente il CheckAccessNotGranted controllo fino alla sua restituzionePASS.

In esecuzione CheckNoNewAccess

CheckNoNewAccess è un controllo personalizzato delle politiche per verificare se la politica consente un nuovo accesso rispetto a una politica di riferimento.

1. Da VS Code, apri un file supportato che contiene AWS IAM Policies, nell'editor VS Code.
2. Per aprire i controlli dei criteri di IAM Access Analyzer, apri il VS Code Command Palette premendo **CRTL+Shift+P**, cerca **IAM Policy Checks**, quindi fai clic per aprire il riquadro IAM Policy Checks nell'editor VS Code.
3. Dal riquadro IAM Policy Checks, seleziona il tipo di documento dal menu a discesa.
4. Dalla sezione Custom Policy Checks, seleziona. CheckNoNewAccess
5. Inserisci un documento relativo alla JSON politica di riferimento. In alternativa, è possibile fornire un percorso di file che faccia riferimento a un documento di JSON policy.
6. Seleziona il tipo di politica di riferimento che corrisponde al tipo del documento di riferimento.
7. Scegli il pulsante Run Custom Policy Check.
8. Dal pannello Problemi di VS Code, esamina i risultati del controllo delle politiche. I controlli delle politiche personalizzate restituiscono un PASS FAIL risultato.
9. Aggiorna la polizza e ripeti questa procedura, eseguendo nuovamente il CheckNoNewAccess controllo fino alla sua restituzionePASS.

In esecuzione CheckNoPublicAccess

CheckNoPublicAccess è un controllo personalizzato delle politiche per verificare se la politica garantisce l'accesso pubblico ai tipi di risorse supportati all'interno del modello.

Per informazioni specifiche sui tipi di risorse supportati, consulta gli [terraform-iam-policy-validator](#) GitHub archivi [cloudformation-iam-policy-validator](#)and.

1. Da VS Code, apri un file supportato che contiene AWS IAM Policies, nell'editor VS Code.
2. Per aprire i controlli dei criteri di IAM Access Analyzer, apri il VS Code Command Palette premendo **CRTL+Shift+P**, cerca **IAM Policy Checks**, quindi fai clic per aprire il riquadro IAM Policy Checks nell'editor VS Code.
3. Dal riquadro IAM Policy Checks, seleziona il tipo di documento dal menu a discesa.
4. Dalla sezione Custom Policy Checks, seleziona. CheckNoPublicAccess
5. Scegli il pulsante Esegui il controllo delle politiche personalizzate.
6. Dal pannello Problemi di VS Code, esamina i risultati del controllo delle politiche. I controlli delle politiche personalizzate restituiscono un PASS FAIL risultato.
7. Aggiorna la polizza e ripeti questa procedura, eseguendo nuovamente il CheckNoNewAccess controllo fino alla sua restituzionePASS.

Lavorare con AWS IoT in AWS Toolkit for Visual Studio Code

AWS IoT nell'AWS Toolkit for Visual Studio Code consente di interagire con AWS IoT servizio, riducendo al minimo le interruzioni del flusso di lavoro in VS Code. Questa guida per l'utente è pensata per essere utili per iniziare a utilizzare la AWS IoT caratteristiche del servizio che sono disponibili in AWS Toolkit for Visual Studio Code. Ulteriori informazioni su AWS IoT service, consulta la guida per gli sviluppatori [Che cos'è AWS IoT?](#)

Prerequisiti di AWS IoT

Per iniziare a usare AWS IoT da Toolkit for VS Code, assicurati che AWS account e VS Code soddisfano i requisiti di queste guide:

- Per AWS requisiti dell'account e AWS autorizzazioni utente specifiche per AWS IoT service, vedere [il Nozioni di base su AWS IoT Core Guida per sviluppatori](#).
- Per i requisiti specifici di Toolkit for VS Code, consultare [il Configurazione del Toolkit for VS Code Guida per l'utente](#).

AWS IoT Oggetti

AWS IoT collegamento di dispositivi AWS servizi e risorse cloud. Puoi connettere i tuoi dispositivi a AWS IoT usando oggetti chiamati cose. Un oggetto è una rappresentazione di un'entità logica o un dispositivo specifico. Può trattarsi di un dispositivo fisico o un sensore, ad esempio una lampadina o un interruttore su un muro. Ulteriori informazioni su AWS IoT cose, consulta la guida per gli sviluppatori [Gestione di dispositivi con AWS IoT](#).

Gestione AWS IoT cose

Il Toolkit for VS Code ha diverse funzionalità che rendono il tuo AWS IoT gestione delle cose più efficiente. Questi sono i modi in cui puoi utilizzare il toolkit VS Code per gestire il tuo AWS IoT things:

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

Per creare un oggetto

1. Da a AWSExplorer, espandilo Tintestazione del servizio e contestuale (pulsante destro del mouse) Oggetti.
2. Scegliere Creazione di oggetti dal menu contestuale per aprire una finestra di dialogo.
3. Segui la richiesta inserendo un nome per il tuo oggetto IoT nella Nome oggetto.
4. Una volta completata questa operazione, un seguito dal nome specificato sarà visibile nella Oggettosezione.

Collegamento di un certificato a un oggetto

1. Da a AWSExplorer, espandilo Tsezione di servizio.
2. In Oggettisottosezione, individuare il cosadove si allega il certificato.
3. contestuale (pulsante destro del mouse) cosae scegli Collegamento di un certificatodal menu contestuale, per aprire un selettore di input con un elenco dei tuoi certificati.
4. Dall'elenco, scegliere la ID del certificato che corrisponde al certificato che vuoi allegare alla tua cosa.

5. Al termine, il certificato è accessibile nella AWS Explorer, come elemento della cosa a cui l'hai attaccata.

Scollegamento di un certificato da un oggetto

1. Da AWS Explorer, espandilo sezione servizi
2. Nella Oggetti sottosezione, individuare il cosa da cui si desidera distaccare un certificato.
3. contestuale (pulsante destro del mouse) cosa e scegli Scollegare il certificato dal menu contestuale.
4. Una volta completata questa operazione, il certificato distaccato non verrà più visualizzato sotto quella cosa nella AWS Explorer, ma sarà comunque accessibile dal Certificati sottosezione.

Per eliminare un oggetto

1. Da AWS Explorer, espandilo sezione di servizio.
2. Nella Oggetti sottosezione, individuare il cosa che desideri eliminare.
3. contestuale: seleziona (pulsante destro del mouse) e scegli Eliminazione di un oggetto dal menu contestuale per eliminarlo.
4. Al termine, l'elemento eliminato non sarà più disponibile dal Oggetti sottosezione.

Note

Nota: È possibile eliminare solo un elemento a cui non è allegato un certificato.

AWS IoT Certificati

I certificati sono un modo comune per creare una connessione sicura tra AWS IoT servizi e dispositivi. I certificati X.509 sono certificati digitali che usano lo standard di infrastruttura a chiave pubblica X.509 per associare una chiave pubblica a un'identità contenuta in un certificato. Ulteriori informazioni su AWS IoT certificati, consulta la guida per gli sviluppatori [Autenticazione \(IoT\)](#).

Gestione dei certificati

Il toolkit VS Code offre una varietà di modi per gestire i tuoi AWS IoT certificati, direttamente dall'AWS Cost.

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

Per creare un AWS IoT certificato

È possibile utilizzare un certificato X.509 per connettersi con l'istanza di AWS IoT.

1. Da AWS Explorer, espandi la sezione servizio e contestuale (pulsante destro del mouse) Certificati.
2. Scegliere Creazione di un certificato dal menu contestuale per aprire una finestra di dialogo.
3. Selezionare una directory nel file system locale in cui salvare la key pair RSA e il certificato X.509.

Note

- I nomi di file predefiniti contengono l'ID del certificato come prefisso.
- Solo il certificato X.509 viene archiviato con il tuo account AWS, tramite il servizio AWS IoT.
- La key pair RSA può essere emessa una sola volta, salvandola in una posizione sicura nel file system quando richiesto.
- Se il certificato o la key pair non possono essere salvati nel tuo file system in questo momento, allora AWS Toolkit elimina il certificato dal tuo account AWS.

Per modificare lo stato di un certificato

Lo stato di un certificato individuale viene visualizzato accanto al suo ID in AWS Explorer e può essere impostato su: attivo, inattivo o revocato.

 Note

- Il certificato richiede un'attivazione prima di poterlo utilizzare per collegare il dispositivo a AWS IoT Service.
- Un record inattivo il certificato può essere attivato, indipendentemente dal fatto che sia stato disattivato in precedenza o sia inattivo per impostazione predefinita.
- Un certificato che è stato revocato non può essere riattivato.

1. Da AWS Explorer, espandilo sezione di servizio.
2. Nella Certificati sottosezione, individuare il certificato da modificare.
3. Contesto: selezionare (fare clic con il pulsante destro del mouse) sul certificato per aprire un menu di scelta rapida che visualizza le opzioni di modifica dello stato disponibili per quel certificato
 - Se un certificato ha lo stato inattivo, scegliere Attiva per modificare lo stato di su attivo.
 - Se un certificato ha lo stato attivo, scegliere Disattiva per modificare lo stato di su inattivo.
 - Se un certificato ha uno dei due stati attivo o inattivo, scegliere Revoca per modificare lo stato di su revocata.

 Note

Ognuna di queste azioni di modifica dello stato è disponibile anche se si seleziona un certificato allegato a un oggetto mentre è visualizzato nella Oggetti sottosezione.

Per allegare una policy IoT a un certificato

1. Da AWS Explorer, espandilo sezione di servizio.
2. Nella Certificati sottosezione, individuare il certificato da modificare.
3. contestuale: selezionare (pulsante destro del mouse) e scegliere Collegamento di policy dal menu contestuale, per aprire un selettore di input con un elenco delle politiche disponibili.
4. Scegli la policy da collegare al certificato.

5. Al termine, il criterio selezionato verrà aggiunto al certificato come voce di sottomenu.

Per scollegare una policy IoT da un certificato

1. Da aAWSExplorer, espandiloTsezione di servizio.
2. NellaCertificatisottosezione, individuare il certificato da modificare.
3. Espandi il certificato e individua la policy da scollegare.
4. contestuale: selezionare (pulsante destro del mouse) e scegliereDistaccadal menu contestuale.
5. Al termine, la politica non sarà più un elemento accessibile dal tuo certificato, ma sarà disponibile dalPolicysottosezione.

Eliminazione di un certificato

1. Da aAWSExplorer, espandiloTSMS?.
2. NellaCertificatisottosezione, individuare il certificato da eliminare.
3. contestuale: selezionare (pulsante destro del mouse) e scegliereEliminazione di un certificatodal menu contestuale.

Note

Non puoi eliminare un certificato se è collegato a un oggetto o ha uno stato attivo. È possibile eliminare un certificato a cui sono allegati criteri.

Policy AWS IoT

AWS IoTLe policy principali sono definite tramite documenti JSON, ognuno dei quali contiene una o più dichiarazioni di policy. Le politiche definiscono comeAWS IoT,AWSe il dispositivo può interagire tra loro. Per ulteriori informazioni su come creare un documento policy, consulta la guida per gli sviluppatori[Policy IoT](#).

Note

Le policy denominate hanno più versioni in modo da poterle ripristinare. NelAWSExplorer, le tue politiche IoT sono elencate nellaPolicysottosezione, nel servizio IoT. È possibile

visualizzare le versioni dei criteri espandendo un criterio. La versione predefinita è indicata da un asterisco.

Gestione di policy

Toolkit for VS Code offre vari modi per gestire i tuoi AWS IoT policy di servizio. Questi sono modi in cui è possibile gestire o modificare le politiche direttamente da AWS Explorer nel codice VS:

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

Per creare una policy AWS IoT

Note

Puoi creare una nuova policy dalla AWS Explorer, ma il documento JSON che definisce la politica deve già esistere nel tuo file system.

1. Da AWS Explorer, espandi la sezione di servizio.
2. contestuale (pulsante destro del mouse) Policy sottosezione e scegli Crea criterio da un documento, per aprire Nome policy campo di input.
3. Inserisci un nome e segui le istruzioni per aprire una finestra di dialogo che ti chiede di selezionare un documento JSON dal tuo file system.
4. Scegliere il file JSON che contiene le definizioni dei criteri, la politica sarà disponibile nella AWS Explorer quando questo è completo.

Per caricare un nuovo AWS IoT versione della policy

È possibile creare una nuova versione di una policy caricando un documento JSON nella policy.

 Note

Il nuovo documento JSON deve essere presente nel file system per creare una nuova versione utilizzando il `AWSCost`.

1. Da `AWSExplorer`, espandilo `T` sezione di servizio.
2. Espandere la `Policy` sottosezione per visualizzare i tuoi `AWS IoT` norme
3. contestuale: selezionare (pulsante destro del mouse) la policy da aggiornare e scegliere `Crea nuova versione da Document`.
4. Quando si apre la finestra di dialogo, scegliere il file JSON che contiene gli aggiornamenti alle definizioni dei criteri.
5. La nuova versione sarà accessibile dalla tua politica nella `AWSCost`.

Per modificare un `AWS IoT` versione della policy

Un documento politico può essere aperto e modificato utilizzando `VS Code`. Al termine della modifica del documento, puoi salvarlo nel file system. Quindi puoi caricarlo sul `AWS IoT` servizio dal `AWSCost`.

1. Da `AWSExplorer`, espandilo `T` sezione di servizio.
2. Espandere la `Policy` sottosezione e individuare la policy da aggiornare. `Crea criterio da un documento` per aprire `Nome policy` campo di input.
3. Espandere il criterio che si desidera aggiornare e quindi selezionare contestualmente (fare clic con il pulsante destro del mouse) la versione del criterio che si desidera modificare.
4. Scegliere `Visualizzazione` dal menu di scelta rapida per aprire la versione della politica in `VS Code`
5. All'apertura del documento di policy, apportare e salvare le modifiche desiderate.

 Note

A questo punto, le modifiche apportate alla politica vengono salvate solo nel file system locale. Per aggiornare la versione e seguirla con `AWSExplorer`, ripeti i passaggi descritti nella [Upload a new policy version](#) procedura.

Per selezionare una nuova versione del criterio (default)

1. Da aAWSExplorer, espandiloTsezione di servizio.
2. Espandere laPolicysottosezione e individuare la policy da aggiornare.
3. Espandere il criterio che si desidera aggiornare e quindi selezionare contestualmente (fare clic con il pulsante destro del mouse) la versione del criterio che si desidera impostare e scegliereImposta come impostazione predefinita.
4. Al termine, la nuova versione predefinita selezionata avrà una stella accanto.

Per eliminare policy

Note

Prima di poter eliminare una policy o una versione di policy, ci sono delle condizioni che devono essere soddisfatte.

- Non puoi eliminare una policy se è allegata a un certificato.
- Non è possibile eliminare un criterio se dispone di versioni non predefinite.
- Non è possibile eliminare la versione predefinita di un criterio a meno che non sia selezionata una nuova versione predefinita o non venga eliminata l'intera policy.
- Prima di poter eliminare un intero criterio, è necessario eliminare prima tutte le versioni non predefinite di tale criterio.

1. Da aAWSExplorer, espandiloTsezione di servizio.
2. Espandere laPolicysottosezione e individuare la policy da aggiornare.
3. Espandere il criterio che si desidera aggiornare e quindi selezionare contestualmente (fare clic con il pulsante destro del mouse) la versione del criterio che si desidera eliminare e scegliereElimina.
4. Quando una versione viene eliminata, non sarà più visibile da Explorer.
5. Quando l'unica versione rimasta per un criterio è quella predefinita, è possibile selezionare contestualmente (fare clic con il pulsante destro del mouse) sulla politica principale e scegliereEliminaper eliminarlo.

Lavorare con le funzioni AWS Lambda

Il AWS Toolkit for Visual Studio Code fornisce il supporto per le funzioni [AWS Lambda](#). Con Toolkit for VS Code, puoi creare codice per le funzioni Lambda che fanno parte di [applicazioni serverless](#). Inoltre, puoi richiamare funzioni Lambda localmente o su AWS.

Lambda è un servizio di calcolo completamente gestito che esegue il codice in risposta a eventi generati dal codice personalizzato o da vari servizi AWS come Amazon Simple Storage Service (Amazon S3), Amazon DynamoDB, Amazon Kinesis, Amazon Simple Notification Service (Amazon SNS) e Amazon Cognito.

Argomenti

- [Interagire con le funzioni Lambda remote](#)

Interagire con le funzioni Lambda remote

Utilizzando il Toolkit for VS Code, è possibile interagire con [AWS Lambda](#) funziona in vari modi, come descritto più avanti in questo argomento.

Per ulteriori informazioni su Lambda, consulta la [Guida per gli sviluppatori di AWS Lambda](#).

Note

Se sono già state create le funzioni Lambda utilizzando la AWS Management Console o in un altro modo, è possibile richiamarle dal kit di strumenti. Per creare una nuova funzione (utilizzando VS Code) su cui è possibile distribuire AWS Lambda, devi prima [creazione di un'applicazione serverless](#).

Argomenti

- [Prerequisiti](#)
- [Richiamare una funzione Lambda](#)
- [Eliminare una funzione Lambda](#)
- [Importare una funzione Lambda](#)
- [Caricare una funzione Lambda](#)

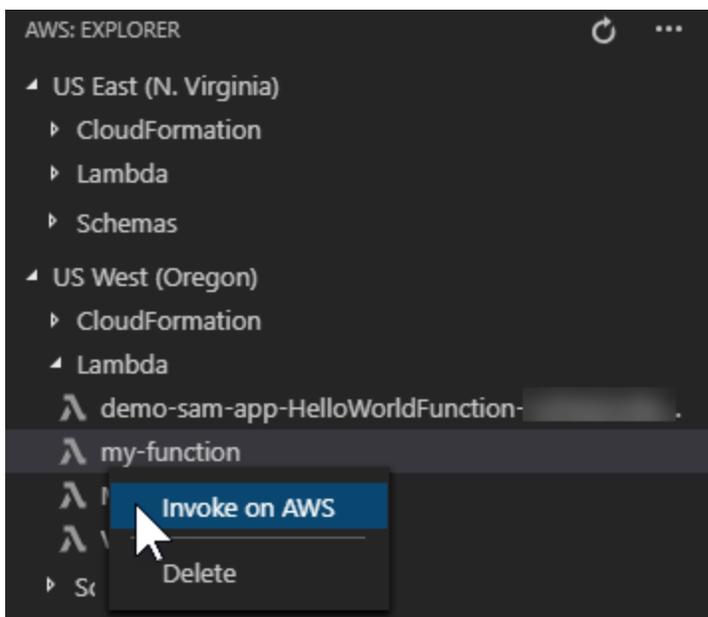
Prerequisiti

- Assicurarsi che il sistema soddisfi i prerequisiti specificati nella sezione [Installazione di Toolkit for VS Code](#).
- Verificare che le credenziali configurate in [Autenticazione e accesso](#) includano l'accesso in lettura/scrittura appropriato al servizio AWS Lambda. Se in AWS Explorer, in Lambda, viene visualizzato un messaggio simile a "Errore durante il caricamento delle risorse Lambda", controlla le autorizzazioni associate a tali credenziali. Le modifiche apportate alle autorizzazioni richiedono alcuni minuti per essere applicate ad AWS Explorer nel codice VS.

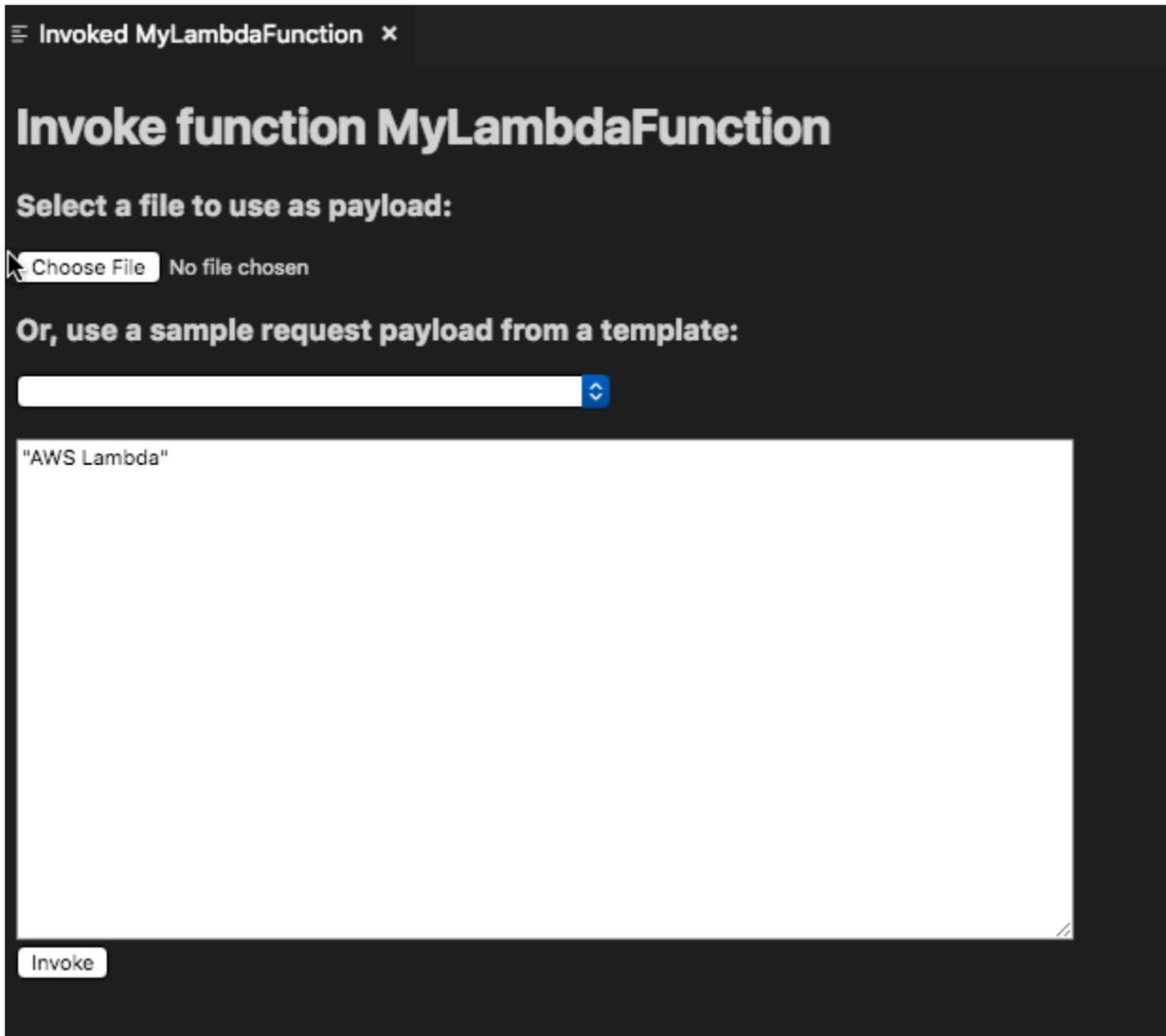
Richiamare una funzione Lambda

È possibile richiamare una funzione Lambda su AWS dal Toolkit for VS Code.

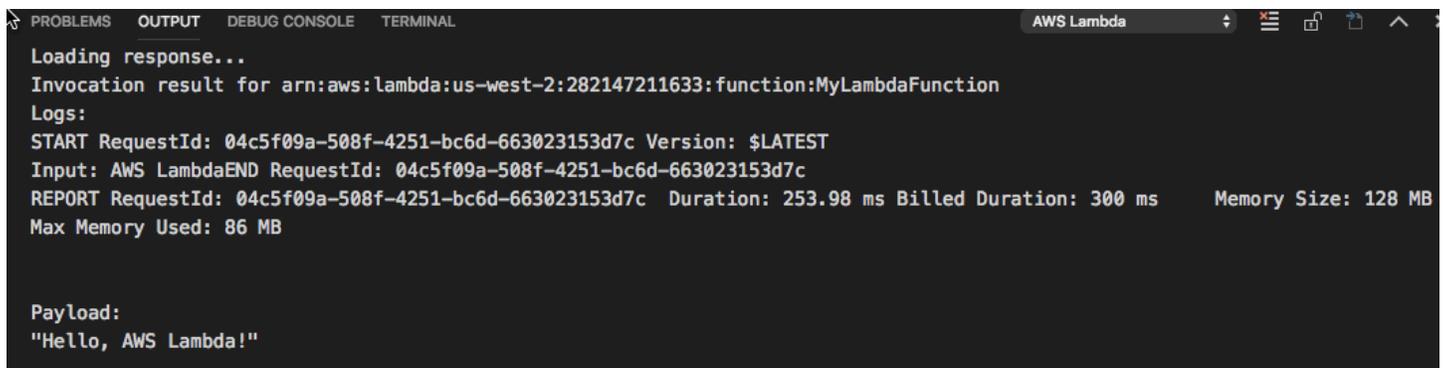
1. In AWS Explorer, scegli il nome della funzione Lambda da richiamare, quindi apri il relativo menu contestuale.



2. Scegli **Invoke on AWS** (Richiama su AWS).
3. Nella finestra di richiamata che si apre, inserire l'input necessario alla funzione Lambda. La funzione Lambda potrebbe, ad esempio, richiedere una stringa come input, come mostrato nella casella di testo.



L'output dell'esecuzione della funzione Lambda verrà visualizzato esattamente come per qualsiasi altro progetto che utilizza VS Code.



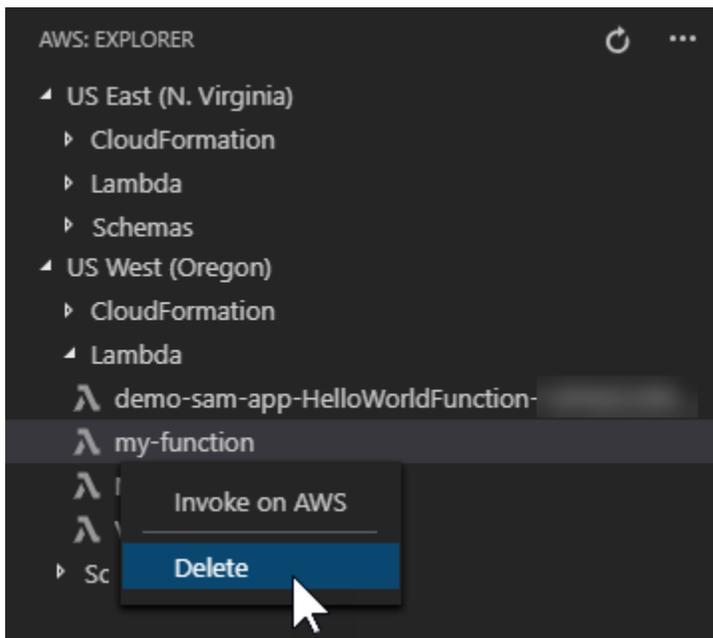
Eliminare una funzione Lambda

È inoltre possibile eliminare una funzione Lambda utilizzando lo stesso menu contestuale.

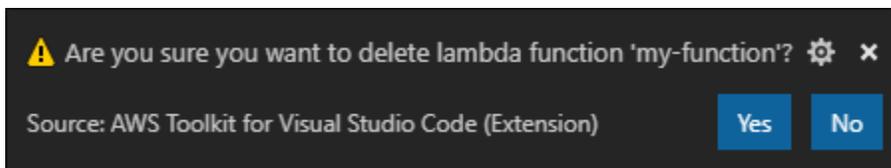
Warning

Non utilizzare questa procedura per eliminare le funzioni Lambda associate a [AWS CloudFormation](#) (ad esempio, la funzione Lambda creata precedentemente durante la [creazione di un'applicazione serverless](#) in questa guida). Queste funzioni devono essere eliminate attraverso la pila AWS CloudFormation.

1. In AWS Explorer, scegli il nome della funzione Lambda che si desidera eliminare e apri il relativo menu contestuale.



2. Scegli Delete (Elimina).
3. Nel messaggio visualizzato, scegliere Yes (Sì) per confermare l'eliminazione.



Dopo che la funzione è stata eliminata, non è più elencata in AWS Explorer.

Importare una funzione Lambda

È possibile importare codice da una funzione Lambda remota nell'area di lavoro VS Code per la modifica e il debug.

Note

Il toolkit supporta solo l'importazione delle funzioni Lambda utilizzando i runtime Node.js e Python supportati.

1. Nella **AWS Esploratore**, scegli il nome della funzione Lambda da importare, quindi apri il relativo menu contestuale.
2. Scegliere **Importa...**
3. Scegli una cartella in cui importare il codice Lambda. Le cartelle esterne all'area di lavoro corrente verranno aggiunte all'area di lavoro corrente.
4. Dopo il download, il Toolkit aggiunge il codice al workspace e apre il file contenente il codice del gestore Lambda. Il Toolkit crea anche un'configurazione di avvio, che viene visualizzato nel pannello di esecuzione VS Code in modo da poter eseguire e eseguire il debug localmente della funzione Lambda utilizzando **AWS Serverless Application Model**. Per ulteriori informazioni sull'utilizzo di **AWS SAM**, consultare [the section called “Esecuzione e debug di un'applicazione serverless dal modello \(locale\)”](#).

Caricare una funzione Lambda

Puoi aggiornare le funzioni Lambda esistenti con il codice locale. L'aggiornamento del codice in questo modo non utilizza la CLI **AWS SAM** per l'implementazione e non crea una pila **AWS CloudFormation**. Questa funzionalità può caricare una funzione Lambda con qualsiasi tempo di esecuzione supportato da Lambda.

Warning

Il toolkit non è in grado di verificare se il codice funziona. Assicurati che il codice funzioni prima di aggiornare le funzioni Lambda di produzione.

1. NellaAWS Esploratore, scegli il nome della funzione Lambda da importare, quindi apri il relativo menu contestuale.
2. Scegli Upload Lambda... (Carica Lambda...)
3. Scegli una delle tre opzioni per caricare la funzione Lambda. Le opzioni includono:

Caricamento di un archivio in formato zip predefinito

- ScegliereArchivio Zipdal menu di scelta rapida.
- Scegli un file in formato zip dal file system e conferma il caricamento con la finestra di dialogo modale. In tal modo il file in formato zip così com'è viene caricato e Lambda viene aggiornato subito dopo l'implementazione.

Caricamento di una directory così com'è

- ScegliereElencodal menu di scelta rapida.
- Scegli una directory dal file system.
- ScegliereNoquando viene richiesto di costruire la directory, quindi confermare il caricamento con la finestra di dialogo modale. In tal modo la directory così com'è viene caricata e Lambda viene aggiornato subito dopo l'implementazione.

Costruzione e caricamento di una directory

Note

Ciò richiede il fileAWS SAMCLI.

- ScegliereElencodal menu di scelta rapida.
- Scegli una directory dal file system.
- ScegliereSìquando viene richiesto di costruire la directory, quindi confermare il caricamento con la finestra di dialogo modale. In questo modo il codice viene costruito nella directory usando il comando `sam build` della CLI AWS SAM e Lambda viene aggiornato subito dopo l'implementazione.

Note

Il toolkit ti avviserà se non è in grado di rilevare un gestore corrispondente prima del caricamento. Se si desidera cambiare il gestore legato alla funzione Lambda, è possibile farlo tramite la funzione LambdaAWS Management Console o ilAWS CLI.

Amazon Redshift nel Toolkit for VS Code

Amazon Redshift è un servizio di data warehouse nel cloud in scala petabyte interamente gestito. Per informazioni dettagliate sul servizio Amazon Redshift, consulta il sommario delle guide per gli utenti di [Amazon Redshift](#).

I seguenti argomenti descrivono come lavorare con Amazon Redshift da AWS Toolkit for Visual Studio Code

Argomenti

- [Utilizzo di Amazon Redshift dal Toolkit for VS Code](#)

Utilizzo di Amazon Redshift dal Toolkit for VS Code

Le seguenti sezioni descrivono come iniziare a lavorare con Amazon Redshift da AWS Toolkit for Visual Studio Code

Per informazioni dettagliate sul servizio Amazon Redshift, consulta gli argomenti della [Amazon Redshift User Guide](#).

Nozioni di base

Prima di iniziare a lavorare con Amazon Redshift daAWS Toolkit for Visual Studio Code, è necessario soddisfare i seguenti requisiti.

1. Sei connesso ai tuoi AWS account dal Toolkit. Per ulteriori informazioni sulla connessione all'AWSaccount dal Toolkit, consulta l'AWSargomento [Connessione a](#) in questa guida per l'utente.
2. Hai creato un data warehouse fornito o senza server.

Se non hai ancora creato un cluster con provisioning Amazon Redshift Serverless o Amazon Redshift, le seguenti procedure descrivono come creare un data warehouse con un set di dati di esempio dalla console. AWS

Creazione di un data warehouse fornito

Per ulteriori dettagli sulla creazione di un data warehouse di cluster con provisioning di Amazon Redshift, consulta l'argomento [Creare un cluster Amazon Redshift](#) di esempio nella Guida introduttiva per l'utente di Amazon Redshift.

1. [Dal tuo browser Internet preferito, accedi alla Console di AWS gestione e apri la console Amazon Redshift all'indirizzo https://console.aws.amazon.com/redshift/.](https://console.aws.amazon.com/redshift/)
2. Dalla console Amazon Redshift, scegli la dashboard Provisioned Clusters.
3. Dalla dashboard Provisioned Clusters, scegli il pulsante Crea cluster per aprire il riquadro Crea cluster.
4. Completa i campi obbligatori nella sezione Configurazione del cluster.
5. Nella sezione Dati di esempio, seleziona la casella Carica dati di esempio per caricare il set di dati di esempio **Ticket** nel database predefinito **Dev** con lo **public** schema.
6. Nella sezione Configurazioni del database, inserisci i valori per i campi nome utente amministratore e password utente amministratore.
7. Scegli Crea cluster per creare il tuo data warehouse assegnato.

Creazione di un data warehouse senza server

Per ulteriori dettagli sulla creazione di un data warehouse di Amazon Redshift Serverless, consulta la sezione [Creazione di un data warehouse with Amazon Redshift Serverless nella](#) Guida introduttiva per l'utente di Amazon Redshift.

1. [Dal tuo browser Internet preferito, accedi alla Console di AWS gestione e apri la console Amazon Redshift all'indirizzo https://console.aws.amazon.com/redshift/.](https://console.aws.amazon.com/redshift/)
2. Dalla console Amazon Redshift, scegli il pulsante Prova Amazon Redshift Serverless per aprire il riquadro Inizia a usare Amazon Redshift Serverless.
3. Nella sezione Configurazioni, scegli l'opzione Use default settings radial.
4. Nella parte inferiore del riquadro Inizia a usare Amazon Redshift Serverless, scegli Salva configurazione per creare un data warehouse serverless con impostazioni predefinite per gruppo di lavoro, namespace, credenziali e crittografia.

Connessione a un data warehouse dal Toolkit

Esistono 3 metodi per connettersi a un database dal Toolkit:

- Nome utente e password del database
- AWS Secrets Manager
- Credenziali temporanee

Per connetterti a un database situato su un cluster o un data warehouse serverless esistente con provisioning tramite Toolkit, completa i passaggi seguenti.

Important

Se hai completato i passaggi indicati nella sezione Prerequisiti di questo argomento della Guida per l'utente e il tuo data warehouse non è visibile nel Toolkit explorer, assicurati di lavorare dall'area corretta AWS dell'explorer.

Connessione al data warehouse con il metodo del nome utente e della password del database

1. Da Toolkit Explorer, espandi il Regione AWS luogo in cui si trova il tuo data warehouse.
2. Espandi Redshift e scegli il tuo data warehouse per aprire la finestra di dialogo Seleziona un tipo di connessione in VS Code.
3. Dalla finestra di dialogo Seleziona un tipo di connessione, scegli il nome utente e la password del database e fornisci le informazioni richieste da ciascuna delle istruzioni.
4. I database, le tabelle e gli schemi disponibili sono visibili nel Toolkit Explorer quando Toolkit si connette al data warehouse e la procedura è completa.

Connessione al data warehouse con AWS Secrets Manager

Note

Questa procedura richiede un segreto del database di AWS Secrets Manager per essere completata. Per istruzioni su come impostare un segreto del database, consulta la sezione [Create an AWS Secrets Manager database secret](#) nella Guida per l'utente di AWSSecrets Manager.

1. Da Toolkit Explorer, espandi il Regione AWS luogo in cui si trova il tuo data warehouse.
2. Espandi Redshift e scegli il tuo data warehouse per aprire la finestra di dialogo Seleziona un tipo di connessione in VS Code.
3. Nella finestra di dialogo Seleziona un tipo di connessione, scegli Secrets Manager e fornisci le informazioni richieste da ciascuna delle istruzioni.
4. I database, le tabelle e gli schemi disponibili sono visibili in Toolkit Explorer quando Toolkit si connette al data warehouse e la procedura è completa.

Connessione al data warehouse con credenziali temporanee

1. Da Toolkit Explorer, espandi la AWS regione in cui esiste il tuo data warehouse.
2. Espandi Redshift e scegli il tuo data warehouse per aprire la finestra di dialogo Seleziona un tipo di connessione in VS Code.
3. Dalla finestra di dialogo Seleziona un tipo di connessione, scegli Credenziali temporanee e fornisci le informazioni richieste da ciascuna delle istruzioni.
4. I database, le tabelle e gli schemi disponibili sono visibili in Toolkit Explorer quando Toolkit si connette al data warehouse e la procedura è completa.

Modifica della connessione al data warehouse

È possibile modificare la connessione al data warehouse per cambiare il database a cui connettersi.

1. Da Toolkit Explorer, espandi il Regione AWS luogo in cui si trova il tuo data warehouse.
2. Espandi Redshift, fai clic con il pulsante destro del mouse sul data warehouse a cui sei connesso, scegli Modifica connessione e fornisci il nome del database a cui desideri connetterti.
3. I database, le tabelle e gli schemi disponibili sono visibili in Toolkit Explorer quando Toolkit si connette al data warehouse e la procedura è completa.

Eliminazione della connessione al data warehouse

1. Da Toolkit Explorer, espandi il Regione AWS luogo in cui si trova il tuo data warehouse.
2. Espandi Redshift, fai clic con il pulsante destro del mouse sul data warehouse con la connessione che desideri eliminare e scegli Elimina connessione. In questo modo vengono rimossi i database, le tabelle e gli schemi disponibili dal Toolkit Explorer.

3. Per riconnetterti al tuo data warehouse, scegli Fai clic per connetterti e fornisci le informazioni richieste da ciascuna delle istruzioni. Per impostazione predefinita, la riconnessione utilizza il metodo di autenticazione precedente per connettersi al data warehouse. Per utilizzare un metodo diverso, scegli la freccia indietro nella finestra di dialogo fino a raggiungere la richiesta di autenticazione.

Esecuzione di istruzioni SQL

Le procedure seguenti descrivono come creare ed eseguire istruzioni SQL nel database da AWS Toolkit for Visual Studio Code.

Note

Per completare i passaggi di ciascuna delle seguenti procedure, è necessario innanzitutto completare la sezione Connessione a un data warehouse dal Toolkit, disponibile in questo argomento della Guida per l'utente.

1. Da Toolkit explorer, espandi Redshift, quindi espandi il data warehouse che contiene il database su cui desideri eseguire la query.
2. Scegli Create-Notebook per specificare un nome di file e una posizione in cui archiviare il notebook localmente, quindi scegli OK per aprire il notebook nell'editor VS Code.
3. Dall'editor VS Code, inserisci le istruzioni SQL che desideri archiviare in questo notebook.
4. Scegli il pulsante Esegui tutto per eseguire le istruzioni SQL che hai inserito.
5. L'output delle istruzioni SQL viene visualizzato sotto le istruzioni inserite.

Aggiungere Markdown a un taccuino

1. Dal tuo taccuino nell'editor VS Code, scegli il pulsante Markdown per aggiungere una cella Markdown al tuo taccuino.
2. Inserisci il tuo Markdown nella cella fornita.
3. La cella Markdown può essere modificata utilizzando gli strumenti dell'editor situati nell'angolo in alto a destra della cella Markdown.

Aggiungere codice a un taccuino

1. Dal tuo taccuino nell'editor VS Code, scegli il pulsante Codice per aggiungere una cella Code al tuo taccuino.
2. Inserisci il codice nella cella fornita.
3. Puoi scegliere di eseguire il codice sopra o sotto la cella Codice selezionando il pulsante appropriato dagli strumenti dell'editor di celle, situato nell'angolo in alto a destra della cella Codice.

Lavorare con Amazon S3

Amazon Simple Storage Service (Amazon S3) Simple Storage Service (Amazon S3) è un servizio di archiviazione dati scalabile. Ti AWS Toolkit for Visual Studio Code consente di gestire gli oggetti e le risorse Amazon S3 direttamente da VS Code.

Per informazioni dettagliate sul servizio Amazon S3, consulta la Amazon [S3 User Guide](#).

I seguenti argomenti descrivono come utilizzare oggetti e risorse Amazon S3 da AWS Toolkit for Visual Studio Code

Argomenti

- [Utilizzo delle risorse Amazon S3](#)
- [Utilizzo degli oggetti Amazon S3](#)

Utilizzo delle risorse Amazon S3

Puoi usare Amazon S3 dal AWS Toolkit for Visual Studio Code per visualizzare, gestire e modificare i tuoi bucket Amazon S3 e altre risorse.

Le seguenti sezioni descrivono come utilizzare le risorse Amazon S3 di AWS Toolkit for Visual Studio Code Per informazioni sull'utilizzo di oggetti Amazon S3, come cartelle e file, da AWS Toolkit for Visual Studio Code, consulta l'argomento [Utilizzo degli oggetti S3](#) in questa Guida per l'utente.

Creazione di un bucket Amazon S3

1. Da Toolkit Explorer, apri il menu contestuale (fai clic con il pulsante destro del mouse) per il servizio S3 e scegli Create Bucket... . In alternativa, scegliete l'icona Create Bucket per aprire la finestra di dialogo Create Bucket.

2. Nel campo Bucket Name (Nome bucket), inserisci un nome valido per il bucket.

Premete Invio per creare il bucket e chiudere la finestra di dialogo. Il nuovo bucket viene quindi visualizzato nel servizio S3 nel toolkit.

Note

Poiché Amazon S3 consente di utilizzare il bucket come un URL oggetto accessibile pubblicamente, il nome del bucket scelto deve essere univoco a livello globale. Se un altro account ha già creato un bucket con il nome che desideri utilizzare, devi usare un nome diverso.

Se non riesci a creare un nuovo bucket, controlla i log del AWS Toolkit nella scheda Output. Se tenti di utilizzare un nome di bucket non valido, si verifica un errore.

`BucketAlreadyExists`

Per ulteriori informazioni, consulta [Restrizioni e limitazioni dei bucket](#) nella Guida per l'utente di Amazon Simple Storage Service.

Aggiunta di una cartella a un bucket Amazon S3

Puoi organizzare il contenuto di un bucket S3 raggruppando gli oggetti in cartelle. Puoi anche creare cartelle all'interno di cartelle.

1. Da Toolkit explorer, espandi il servizio S3 per visualizzare un elenco delle tue risorse S3.
2. Scegli l'icona Crea cartella per aprire la finestra di dialogo Crea cartella. In alternativa, apri il menu contestuale (fate clic con il pulsante destro del mouse) per un bucket o una cartella, quindi scegliete Crea cartella.
3. Immettete un valore nel campo Nome cartella e premete Invio per creare la cartella e chiudere la finestra di dialogo. La nuova cartella viene visualizzata nella risorsa S3 corrispondente nel menu del toolkit.

Eliminazione di un bucket Amazon S3

Quando elimini un bucket S3, elimini anche le cartelle e gli oggetti in esso contenuti. Quindi, quando tenti di eliminare un bucket, ti viene chiesto di confermare che desideri eliminarlo.

1. Dal menu principale del toolkit, espandi il servizio Amazon S3 per visualizzare un elenco delle tue risorse S3.

2. Apri il menu contestuale (fai clic con il pulsante destro del mouse) per un bucket o una cartella, quindi scegli Elimina bucket S3.
3. Quando richiesto, inserisci il nome del bucket nel campo di testo, quindi premi Invio per eliminare il bucket e chiudere la richiesta di conferma.

Note

Se il bucket contiene oggetti, viene svuotato prima di essere eliminato. Se si tenta di eliminare un numero elevato di risorse o oggetti contemporaneamente, l'eliminazione può richiedere del tempo. Una volta eliminati, ricevi una notifica che indica che sono stati eliminati con successo.

Utilizzo degli oggetti Amazon S3

I file, le cartelle e tutti gli altri dati archiviati in un bucket di risorse S3 sono noti come oggetti S3.

Le seguenti sezioni descrivono come utilizzare oggetti Amazon S3 da AWS Toolkit for Visual Studio Code. Per informazioni dettagliate sull'utilizzo delle risorse Amazon S3, come i bucket S3, da AWS Toolkit for Visual Studio Code, consulta l'argomento [Lavorare con le risorse S3](#) in questa guida per l'utente.

Impaginazione degli oggetti

Se lavori con un gran numero di oggetti e cartelle Amazon S3, l'impaginazione ti consente di specificare il numero di elementi che desideri visualizzare su una pagina.

1. Vai alla barra delle attività di VS Code e scegli Estensioni.
2. Dall'estensione AWS Toolkit, scegli l'icona delle impostazioni, quindi scegli Impostazioni estensione.
3. Nella pagina Impostazioni, scorri verso il basso fino all'impostazione AWS > S3: numero massimo di elementi per pagina.
4. Cambia il valore predefinito impostando il numero di elementi S3 che desideri vengano visualizzati prima che venga visualizzato «carica altro».

Note

I valori validi includono qualsiasi numero compreso tra 3 e 1000. Questa impostazione si applica solo al numero di oggetti o cartelle visualizzati contemporaneamente. Tutti i bucket che hai creato vengono visualizzati contemporaneamente. Per impostazione predefinita, è possibile creare fino a 100 bucket in ciascun account AWS.

5. Chiudi la pagina Impostazioni per confermare le modifiche.

Puoi anche aggiornare le impostazioni in un file in formato JSON scegliendo l'icona Apri impostazioni (JSON) nella parte superiore destra della pagina Impostazioni.

Caricamento e download di oggetti Amazon S3

Puoi caricare file archiviati localmente nei tuoi bucket Amazon S3 o scaricare oggetti Amazon S3 remoti sul tuo sistema locale, da AWS Toolkit for Visual Studio Code

Carica un file utilizzando il Toolkit

1. Da Toolkit explorer, espandi il servizio Amazon S3 per visualizzare un elenco delle tue risorse S3.
2. Scegli l'icona Carica file che si trova accanto a un bucket o a una cartella per aprire la finestra di dialogo Carica file. Oppure puoi aprire il menu contestuale (fai clic con il pulsante destro del mouse) e scegliere Carica file.

Note

Per caricare un file nella cartella o nella risorsa principale dell'oggetto, apri il menu contestuale (fai clic con il pulsante destro del mouse) per qualsiasi oggetto S3 e scegli Carica su principale.

3. Usa il file manager del tuo sistema per selezionare un file, quindi scegli Carica file per chiudere la finestra di dialogo e caricare il file.

Carica un file utilizzando la palette dei comandi

È possibile utilizzare l'interfaccia Toolkit o la Command Palette per caricare un file in un bucket.

1. Per selezionare un file da caricare, scegli la scheda del file in VS Code.
2. Premi Ctrl+Shift+P per visualizzare la palette dei comandi.
3. Nella palette dei comandi, inserisci la frase `upload file` per visualizzare un elenco di comandi consigliati.
4. Scegliete il comando AWS: Carica file per aprire la finestra di dialogo AWS: Carica file.
5. Quando richiesto, scegli il file che desideri caricare, quindi scegli il bucket in cui vuoi caricare il file.
6. Conferma il caricamento per chiudere la finestra di dialogo e iniziare il processo di caricamento. Una volta completato il caricamento, l'oggetto viene visualizzato nel menu del toolkit con metadati che includono la dimensione dell'oggetto, la data dell'ultima modifica e il percorso.

Download di un oggetto Amazon S3

1. Dal Toolkit explorer, espandi il servizio S3.
2. Da un bucket o da una cartella, apri il menu contestuale (fai clic con il pulsante destro del mouse) per un oggetto che desideri scaricare. Quindi, scegliete Scarica come per aprire la finestra di dialogo Scarica come. Oppure, in alternativa, scegliete l'icona Scarica come accanto all'oggetto.
3. Utilizzando il gestore di file del sistema, scegliete una cartella di destinazione, inserite un nome di file, quindi scegliete Scarica per chiudere la finestra di dialogo e avviare il download.

Modifica di oggetti remoti

Puoi utilizzarlo AWS Toolkit for Visual Studio Code per modificare gli oggetti Amazon S3 archiviati nelle tue risorse Amazon S3 remote.

1. Dal Toolkit explorer, espandi il servizio S3.
2. Espandi la risorsa S3 che contiene il file che desideri modificare.
3. Per modificare il file, scegli l'icona a forma di matita (Modifica file).
4. Per modificare un file aperto in modalità di sola lettura, visualizza il file nell'editor VS Code, quindi scegli l'icona a forma di matita situata nell'angolo in alto a destra dell'interfaccia utente.

Note

- Se riavvii o esci da VS Code, l'IDE si disconnette dalle risorse S3. Se alcuni file S3 remoti vengono modificati quando ti disconnetti, la modifica si interrompe. È necessario riavviare VS Code e riaprire la scheda di modifica per riprendere la modifica.
- Il pulsante Modifica file si trova nell'angolo in alto a destra dell'interfaccia utente. È visibile solo quando stai visualizzando attivamente un file di sola lettura nell'editor VS Code.
- I file non di testo non possono essere aperti in modalità di sola lettura. Si aprono sempre in modalità di modifica.
- Non puoi tornare alla modalità di sola lettura dalla modalità di sola modifica, solo il contrario.

Copiare il percorso di un oggetto Amazon S3

La procedura seguente descrive come copiare il percorso di un oggetto Amazon S3 da AWS Toolkit for Visual Studio Code

1. Dal Toolkit explorer, espandi il servizio S3.
2. Espandi il bucket di risorse che contiene l'oggetto di cui desideri copiare il percorso.
3. Apri il menu contestuale (fai clic con il pulsante destro del mouse) per l'oggetto di cui desideri copiare il percorso, quindi scegli Copia percorso per copiare il percorso dell'oggetto negli appunti locali.

Generazione di un URL prefirmato per un oggetto Amazon S3

Puoi condividere oggetti Amazon S3 privati con altri concedendo autorizzazioni limitate nel tempo per i download tramite la funzionalità URL predefinita. Per ulteriori informazioni, consulta [Condivisione di un oggetto con un URL predefinito](#).

1. Dal Toolkit explorer, espandi il servizio S3.
2. Da un bucket o da una cartella, apri il menu contestuale (fai clic con il pulsante destro del mouse) per un oggetto che desideri condividere. Quindi, scegliete Genera URL predefinito per aprire la palette Comandi.
3. Dalla palette dei comandi, inserite il numero di minuti in cui l'URL può essere utilizzato per accedere all'oggetto. Quindi, scegliete Invio per confermare e chiudere la finestra di dialogo.

4. Dopo la generazione dell'URL predefinito, la barra di stato del codice VS mostra l'URL predefinito per l'oggetto che è stato copiato negli appunti locali.

Eliminazione di un oggetto Amazon S3

Se un oggetto si trova in un bucket senza versioni, puoi eliminarlo definitivamente. Per i bucket con il controllo delle versioni abilitato, una richiesta di eliminazione non elimina definitivamente quell'oggetto. Anziché rimuovere l'oggetto, Amazon S3 inserisce un contrassegno di eliminazione nel bucket. Per ulteriori informazioni, vedere [Eliminazione](#) delle versioni degli oggetti.

1. Da Toolkit explorer, espandi il servizio S3 per visualizzare un elenco delle tue risorse S3.
2. Apri il menu contestuale (fai clic con il pulsante destro del mouse) per un oggetto che desideri eliminare, quindi scegli Elimina per aprire la finestra di dialogo di conferma.
3. Scegliete Elimina. per confermare che desideri eliminare l'oggetto S3. Quindi, chiudi la finestra di dialogo.

Utilizzo di applicazioni serverless

AWS Toolkit for Visual Studio Code Fornisce supporto per [Applicazione Serverless AWS](#). Gli argomenti seguenti descrivono come iniziare a creare e lavorare con applicazioni AWS Serverless Application Model (AWS SAM), da AWS Toolkit for Visual Studio Code.

Argomenti

- [Guida introduttiva alle applicazioni serverless](#)
- [Esecuzione e debug delle funzioni Lambda escludendo le risorse del modello AWS SAM](#)
- [Esecuzione e debug delle risorse locali di Amazon API Gateway](#)
- [Opzioni di configurazione per il debug di applicazioni serverless](#)
- [Risoluzione dei problemi delle applicazioni serverless](#)

Guida introduttiva alle applicazioni serverless

Le sezioni seguenti descrivono come iniziare a creare un file Applicazione Serverless AWS da AWS Toolkit for Visual Studio Code, using AWS Serverless Application Model (AWS SAM) e AWS CloudFormation stack.

Prerequisiti

Prima di poter creare o lavorare con un'Applicazione Serverless AWS, è necessario completare i seguenti prerequisiti.

Note

Le seguenti operazioni potrebbero richiedere l'uscita o il riavvio di VS Code prima del completamento delle modifiche.

- Installa l'interfaccia AWS SAM a riga di comando (CLI). Per ulteriori informazioni e istruzioni su come installare la AWS SAM CLI, consultate l'argomento [Installazione della AWS SAM CLI](#) in questa Guida per l'utente. AWS Serverless Application Model
- Dal tuo file di AWS configurazione, identifica la tua regione predefinita. AWS Per ulteriori informazioni sul file di configurazione, consulta l'argomento [Configurazione e impostazioni dei file di credenziali](#) nella Guida per l'AWS Command Line Interface utente.
- Installa l'SDK della tua lingua e configura la tua toolchain. Per ulteriori informazioni su come configurare la toolchain, AWS Toolkit for Visual Studio Code consulta l'argomento [Configurazione della toolchain in questa Guida per l'utente](#).
- Installa l'[estensione per il supporto del linguaggio YAML](#) dal marketplace VS Code. Ciò è necessario per la CodeLens funzionalità di accessibilità dei file AWS SAM modello. Per ulteriori informazioni in merito CodeLens, consulta la [CodeLens](#) sezione della documentazione di VS Code

Autorizzazioni IAM per applicazioni serverless

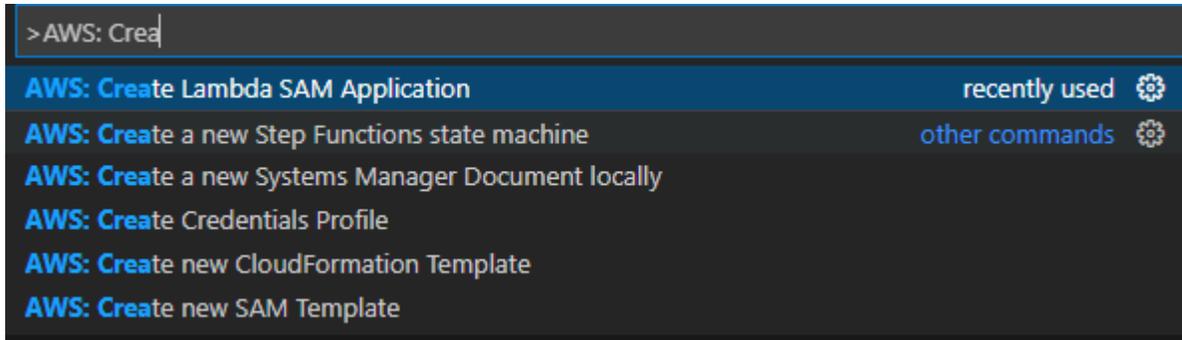
Nel Toolkit for VS Code è necessario disporre di un profilo di credenziali che contenga AWS Identity and Access Management le autorizzazioni (IAM) necessarie per distribuire ed eseguire applicazioni serverless. È necessario disporre di un accesso adeguato in lettura/scrittura ai seguenti servizi: AWS CloudFormation, IAM, Lambda, Amazon API Gateway, Amazon Simple Storage Service (Amazon S3) e Amazon Elastic Container Registry (Amazon ECR) Elastic Container Registry (Amazon ECR).

Per ulteriori informazioni sulla configurazione dell'autenticazione necessaria per distribuire ed eseguire applicazioni serverless, consulta la sezione [Gestione dell'accesso e delle autorizzazioni alle risorse](#) nella Guida per gli sviluppatori. AWS Serverless Application Model Per informazioni su come configurare le credenziali, consulta la sezione contenuta [AWS IAM credenziali](#) in questa Guida per l'utente.

Creazione di una nuova applicazione serverless (locale)

Questa procedura mostra come creare un'applicazione serverless con Toolkit for VS Code utilizzando AWS SAM. L'output di questa procedura è una directory locale sull'host di sviluppo contenente un'applicazione serverless di esempio, che è possibile creare, testare localmente, modificare e distribuire sul cloud. AWS

1. Per aprire la palette dei comandi, scegliete Visualizza, Command Palette, quindi immettete. AWS
2. Scegli AWSToolkit Create Lambda SAM Application.



Note

Se la AWS SAM CLI non è installata, viene visualizzato un errore nell'angolo in basso a destra dell'editor VS Code. In tal caso, verifica di aver soddisfatto tutti i [presupposti](#) e i prerequisiti.

3. Scegliete il runtime per la vostra AWS SAM applicazione.

Note

Se selezioni uno dei tempi di esecuzione con "(Image)", l'applicazione è di tipo pacchetto Image. Se selezioni uno dei tempi di esecuzione senza "(Image)", l'applicazione è di tipo Zip. Per ulteriori informazioni sulla differenza tra tipi di pacchetto Image e Zip, consulta [Pacchetti di implementazione Lambda](#) nella Guida per gli sviluppatori di AWS Lambda.

4. A seconda del runtime selezionato, è possibile che venga richiesto di selezionare un gestore delle dipendenze e un'architettura di runtime per l'applicazione SAM.

Dependency Manager

Scegli tra Gradle o Maven.

 Note

Questa scelta di strumenti di automazione delle build è disponibile solo per i runtime Java.

Architecture

Scegli tra x86_64 o arm64.

L'opzione per eseguire l'applicazione serverless in un ambiente emulato basato su ARM64 anziché nell'ambiente predefinito basato su x86_64 è disponibile per i seguenti runtime:

- nodejs12.x (ZIP e immagine)
- nodejs14.x (ZIP e immagine)
- python3.8 (ZIP e immagine)
- python3.9 (ZIP e immagine)
- python3.10 (ZIP e immagine)
- python3.11 (ZIP e immagine)
- python3.12 (ZIP e immagine)
- java8.al2 con Gradle (ZIP e immagine)
- java8.al2 con Maven (solo ZIP)
- java11 con Gradle (ZIP e immagine)
- java11 con Maven (solo ZIP)

 Important

È necessario installare la AWS CLI versione 1.33.0 o successiva per consentire l'esecuzione delle applicazioni in ambienti basati su ARM64. Per ulteriori informazioni, consulta [Prerequisiti](#).

5. Scegliere una località per il nuovo progetto. È possibile utilizzare una cartella dell'area di lavoro esistente se è aperta, selezionare una cartella diversa già esistente o creare una nuova cartella e selezionarla. Per questo esempio, scegliere There are no workspace folders open (Non sono aperte cartelle dell'area di lavoro) per creare una cartella denominata MY-SAM-APP.

6. Inserire un nome per il nuovo progetto. Per questo esempio, utilizzare `my-sam-app-nodejs`. Dopo aver premuto Invio, il Toolkit for VS Code Code impiega alcuni minuti per creare il progetto.

Quando il progetto viene creato, l'applicazione viene aggiunta all'area di lavoro corrente. Dovrebbe essere visualizzabile nella finestra Explorer.

Apertura di un'applicazione serverless (locale)

Per aprire un'applicazione serverless sull'host di sviluppo locale, aprite la cartella che contiene il file modello dell'applicazione.

1. Dal File, scegli Apri cartella... .
2. Nella finestra di dialogo Apri cartella, accedete alla cartella dell'applicazione serverless che desiderate aprire.
3. Scegliete il pulsante Seleziona cartella.

Quando si apre la cartella di un'applicazione, questa viene aggiunta alla finestra Explorer.

Esecuzione e debug di un'applicazione serverless dal modello (locale)

Puoi usare Toolkit for VS Code per configurare come eseguire il debug di applicazioni serverless ed eseguirle localmente nel tuo ambiente di sviluppo.

Si inizia a configurare il comportamento di debug utilizzando la funzionalità VS Code per identificare una [CodeLens](#) funzione Lambda idonea. CodeLens abilita interazioni basate sul contenuto con il codice sorgente. Per informazioni su come garantire l'accesso alla CodeLens funzionalità, consulta la [Prerequisiti](#) sezione precedente di questo argomento.

Note

In questo esempio, si esegue il debug di un'applicazione che utilizza JavaScript. Tuttavia, puoi utilizzare le funzionalità di debug di Toolkit for VS Code con i seguenti linguaggi e runtime:

- C# — .NET Core 2.1, 3.1; .NET 5.0
- JavaScript/TypeScript — Node.js 12. x, 14. x
- Python — 3.6, 3.7, 3.8, 3.9, 3.10, 3.11, 3.12
- Giava — 8, 8.al2, 11

- Vai — 1.x

La scelta della lingua influisce anche sul modo in cui CodeLens vengono rilevati i gestori Lambda idonei. Per ulteriori informazioni, consulta [Esecuzione e debug delle funzioni Lambda escludendo le risorse del modello AWS SAM](#).

In questa procedura, si utilizza l'applicazione di esempio creata nella [Creazione di una nuova applicazione serverless \(locale\)](#) sezione precedente di questo argomento.

1. Per visualizzare i file dell'applicazione in File Explorer di VS Code, scegliete Visualizza, Esplora.
2. Dalla cartella dell'applicazione (ad esempio, my-sample-app), apri il `template.yaml` file.

Note

Se utilizzate un modello con un nome diverso da `template.yaml`, l'CodeLens indicatore non è automaticamente disponibile nel file YAML. Ciò significa che è necessario aggiungere manualmente una configurazione di debug.

3. Nell'editor di `template.yaml`, vai alla `Resources` sezione del modello che definisce le risorse serverless. In questo caso, si tratta del tipo `AWS::Serverless::Function` di `HelloWorldFunction` risorsa.

Nell'CodeLens indicatore di questa risorsa, scegli Aggiungi configurazione di debug.

4. Nella palette dei comandi, selezionate il runtime in cui verrà eseguita l'AWS SAM applicazione.
5. Nell'editor del file `launch.json`, modifica o conferma i valori per le seguenti proprietà di configurazione:

- `"name"`: inserisci un nome facilmente leggibile da visualizzare nel campo a discesa Configuration (Configurazione) nella vista Run (Esegui).
- `"target"`— Assicuratevi che il valore sia `"template"` tale che il AWS SAM modello sia il punto di ingresso per la sessione di debug.
- `"templatePath"`: inserisci un percorso relativo o assoluto per il file `template.yaml`.
- `"logicalId"`— Assicuratevi che il nome corrisponda a quello specificato nella sezione Risorse del AWS SAM modello. In questo caso, è la `HelloWorldFunction` di tipo `AWS::Serverless::Function`.

Per ulteriori informazioni su questa e altre voci nel file `launch.json`, consulta [Opzioni di configurazione per il debug di applicazioni serverless](#).

6. Se la configurazione di debug ti soddisfa, salva `launch.json`. Quindi, per avviare il debug, scegli il pulsante verde «play» nella vista RUN.

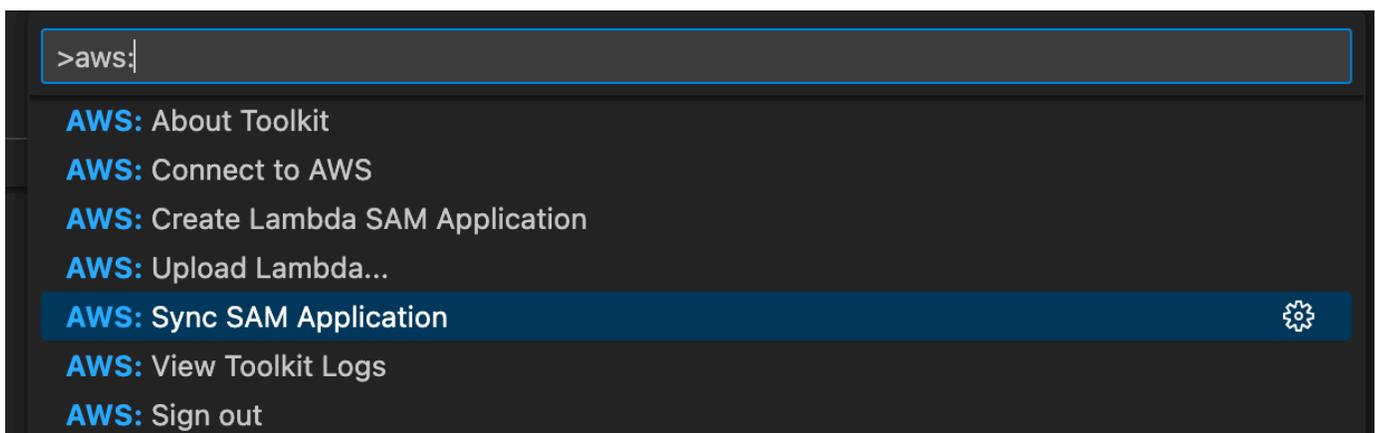
Quando vengono avviate le sessioni di debug, il pannello DEBUG CONSOLE (CONSOLE DI DEBUG) mostra l'output di debug e visualizza tutti i valori restituiti dalla funzione Lambda. (Durante il debug AWS SAM delle applicazioni, il AWSToolkit viene selezionato come canale di output nel pannello Output).

Sincronizzazione delle applicazioni AWS SAM

AWS Toolkit for Visual Studio Code esegue il comando AWS SAM CLI `aws sam sync` per distribuire le applicazioni serverless su Cloud AWS. Per ulteriori informazioni sulla AWS SAM sincronizzazione, consulta l'argomento di [riferimento sui comandi AWS SAM CLI](#) nella Guida per gli sviluppatori AWS Serverless Application Model.

La procedura seguente descrive come distribuire le applicazioni serverless Cloud AWS con Toolkit for VS `aws sam sync` Code.

1. Dal menu principale di VS Code, apri la palette dei comandi espandendo Visualizza e scegliendo Command Palette.
2. Dalla Command Palette cerca AWS e scegli Sincronizza l'applicazione SAM per iniziare a configurare la sincronizzazione.



3. Scegli la AWS regione con cui sincronizzare l'applicazione serverless.

- Scegliere il file `template.yaml` da utilizzare per la distribuzione.
- Seleziona un bucket Amazon S3 esistente o inserisci un nuovo nome di bucket Amazon S3 in cui distribuire l'applicazione.

Important

Il bucket Amazon S3 deve soddisfare i seguenti requisiti:

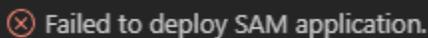
- Il bucket deve trovarsi nella regione con cui ti stai sincronizzando.
- Il nome di un bucket Amazon S3 deve essere univoco tra tutti i nomi di bucket esistenti in Amazon S3.

- Se l'applicazione serverless include una funzione con tipo di pacchetto Image, inserisci il nome di un repository Amazon ECR che questa implementazione può utilizzare. Il repository deve trovarsi nella regione a cui esegui l'implementazione.
- Seleziona uno stack di distribuzione dall'elenco delle distribuzioni precedenti o crea un nuovo stack di distribuzione inserendo un nuovo nome dello stack. Quindi, procedi con l'avvio del processo di sincronizzazione.

Note

Gli stack utilizzati nelle distribuzioni precedenti vengono richiamati per area di lavoro e regione.

- Durante il processo di sincronizzazione, lo stato della distribuzione viene registrato nella scheda Terminale di VS Code. Verifica che la sincronizzazione sia avvenuta correttamente dalla scheda Terminale, se si verifica un errore riceverai una notifica.



Note

Per ulteriori dettagli sulla sincronizzazione, i AWS Toolkit for Visual Studio Code log sono accessibili dalla palette dei comandi.

Per accedere ai AWS Toolkit for Visual Studio Code log dalla palette dei comandi, espandi Visualizza, scegli Command Palette, quindi **AWS: View AWS Toolkits Logs** cercala e selezionala quando compare nell'elenco.

Una volta completata la distribuzione, l'applicazione viene visualizzata nell'Explorer. AWS Per ulteriori informazioni su come richiamare la funzione Lambda creata come parte dell'applicazione, consulta l'argomento di questa Guida per [Interagire con le funzioni Lambda remote](#) l'utente.

Eliminazione di un'applicazione serverless dal Cloud AWS

L'eliminazione di un'applicazione serverless comporta l'eliminazione della pila AWS CloudFormation implementata precedentemente in AWS Cloud. Nota che questa procedura non elimina la directory dell'applicazione dall'host locale.

1. Apri la [AWS Explorer](#).
2. Nella finestra AWSToolkit Explorer, espandi l'area contenente l'applicazione distribuita che desideri eliminare, quindi espandi. AWS CloudFormation
3. Apri il menu contestuale (fai clic con il pulsante destro del mouse) per il nome AWS CloudFormation dello stack che corrisponde all'applicazione serverless che desideri eliminare, quindi scegli Elimina stack. AWS CloudFormation
4. Per confermare che desiderate eliminare lo stack selezionato, scegliete Sì.

Se l'eliminazione dello stack ha esito positivo, Toolkit for VS Code rimuove il nome dello stack dall'AWS CloudFormationelenco in Explorer. AWS

Esecuzione e debug delle funzioni Lambda escludendo le risorse del modello AWS SAM

Durante il test AWS SAM dell'applicazione, puoi scegliere di eseguire ed eseguire il debug solo della funzione Lambda ed escludere altre risorse definite AWS SAM dal modello. Questo approccio prevede l'utilizzo della [CodeLens](#) funzionalità per identificare i gestori di funzioni Lambda nel codice sorgente che è possibile richiamare direttamente.

I gestori Lambda rilevati da CodeLens dipendono dal linguaggio e dal runtime utilizzati per l'applicazione.

Linguaggio/tempo di esecuzione	Criteri per le funzioni Lambda da identificare mediante indicatori CodeLens
C# (dotnetcore2.1, 3.1;. NET5.0)	<p>La funzione presenta inoltre le seguenti caratteristiche:</p> <ul style="list-style-type: none">• È una funzione pubblica di una classe pubblica.• Ha uno o due parametri. Con due parametri , il secondo parametro deve implementare l'<code>ILambdaContext</code> interfaccia.• Ha un <code>*.csproj</code> file nella cartella principale all'interno della cartella Workspace VS Code. <p>L'estensione ms-dotnettools.csharp (o qualsiasi estensione) che fornisca simboli linguistici per C# è installata e abilitata.</p>
JavaScriptTypeScript /(Node.js 12.x, 14.x)	<p>La funzione presenta inoltre le seguenti caratteristiche:</p> <ul style="list-style-type: none">• È una funzione esportata con un massimo di tre parametri.• Ha un <code>package.json</code> file nella cartella principale all'interno della cartella dell'area di lavoro VS Code.
Python (3.7, 3.8, 3.9, 3.10, 3.11, 3.12)	<p>La funzione presenta inoltre le seguenti caratteristiche:</p> <ul style="list-style-type: none">• È una funzione di primo livello.• Ha un <code>requirements.txt</code> file nella cartella principale all'interno della cartella Workspace VS Code.

Linguaggio/tempo di esecuzione

Criteria per le funzioni Lambda da identificare mediante indicatori CodeLens

L'estensione [ms-python.python \(o qualsiasi estensione](#) che fornisca simboli linguistici per Python) è installata e abilitata.

Linguaggio/tempo di esecuzione	Criteri per le funzioni Lambda da identificare mediante indicatori CodeLens
Java (8, 8.al2, 11)	<p>La funzione presenta inoltre le seguenti caratteristiche:</p> <ul style="list-style-type: none">• È una funzione pubblica di una classe pubblica non astratta.• Ha uno, due o tre parametri:<ul style="list-style-type: none">• Un parametro: il parametro può essere qualsiasi cosa.• Due parametri: i parametri devono essere a <code>java.io.InputStream</code> e un <code>java.io.OutputStream</code> OR l'ultimo parametro deve essere <code>com.amazonaws.services.lambda.runtime.Context</code>.• Tre parametri: i parametri devono essere a <code>java.io.InputStream</code> e <code>java.io.OutputStream</code> AND l'ultimo parametro deve essere <code>com.amazonaws.services.lambda.runtime.Context</code>.• Ha un file <code>build.gradle</code> (Gradle) o <code>pom.xml</code> (Maven) nella cartella principale all'interno della cartella dell'area di lavoro VS Code. <p>L'estensione redhat.java (o qualsiasi estensione e che fornisca simboli linguistici per Java) è installata e abilitata. Questa estensione richiede Java 11, indipendentemente dal runtime Java in uso.</p>

Linguaggio/tempo di esecuzione	Criteri per le funzioni Lambda da identificare mediante indicatori CodeLens
	Il vscjava.vscode-java-debug l'estensione (o qualsiasi estensione che fornisce un debugger Java) è installata e abilitata.
Vai (1.x)	<p>La funzione presenta inoltre le seguenti caratteristiche:</p> <ul style="list-style-type: none"> • È una funzione di primo livello. • Richiede tra 0 e 2 argomenti. Nel caso di due argomenti, il primo argomento deve implementare <code>context.Context</code>. • Restituisce tra 0 e 2 argomenti. Se ci sono più di 0 argomenti, deve essere implementato l'ultimo argomento <code>error</code>. • Ha un <code>go.mod</code> file all'interno della cartella dell'area di lavoro VS Code. <p>L'estensione golang.go è installata, configurata e abilitata.</p>

Per l'esecuzione e il debug di un'applicazione serverless direttamente dal codice dell'applicazione

1. Per visualizzare i file dell'applicazione in VS Code File Explorer, scegli Visualizza, Esplora.
2. Dalla cartella dell'applicazione (ad esempio, my-sample-app), espandi la cartella delle funzioni (in questo caso, hello-world) e apri il `app.js` file.
3. Nell' CodeLens indicatore che identifica un gestore di funzioni Lambda idoneo, scegli. Add Debug Configuration
4. Nella palette dei comandi, seleziona il runtime in cui verrà eseguita l' AWS SAM applicazione.
5. Nell'editor del file `launch.json`, modifica o conferma i valori per le seguenti proprietà di configurazione:

- "name": inserisci un nome facilmente leggibile da visualizzare nel campo a tendina Configuration (Configurazione) nella vista Run (Esegui).
- "target": assicurarsi che il valore sia "code" in modo che venga richiamato direttamente un gestore della funzione Lambda.
- "lambdaHandler": inserisci il nome del metodo all'interno del codice sorgente chiamato da Lambda per eseguire la funzione. Ad esempio, per le applicazioni in JavaScript, l'impostazione predefinita è `app.lambdaHandler`.
- "projectRoot": inserisci il percorso del file dell'applicazione che contiene la funzione Lambda.
- "runtime": inserisci o conferma un tempo di esecuzione valido per l'ambiente di esecuzione Lambda, ad esempio "nodejs.12x".
- "payload": scegli una delle seguenti opzioni per definire come input il payload dell'evento da fornire alla funzione Lambda:
 - "json": coppie JSON chiave-valore formattate che definiscono il payload dell'evento.
 - "path": un percorso del file utilizzato come payload dell'evento.

Nell'esempio seguente, l'opzione definisce il "json" payload.

Per ulteriori informazioni su questa e altre voci nel file `launch.json`, consulta [Opzioni di configurazione per il debug di applicazioni serverless](#).

6.

Se sei soddisfatto della configurazione di debug, per iniziare il debug, scegli la freccia verde di riproduzione accanto a. RUN

All'avvio delle sessioni di debug, il DEBUGCONSOLE pannello mostra l'output di debug e visualizza tutti i valori restituiti dalla funzione Lambda. (Durante il debug delle AWS SAM applicazioni, AWS Toolkit viene selezionato come canale di output nel pannello Output.)

Esecuzione e debug delle risorse locali di Amazon API Gateway

Puoi eseguire o eseguire il debug AWS SAM Risorse locali di API Gateway, specificate in `template.yaml`, eseguendo una configurazione di lancio di VS Code di `type=aws-sam` con `invokeTarget.target=api`.

Note

API Gateway supporta due tipi di API, REST e HTTP. Tuttavia, la funzionalità API Gateway con AWS Toolkit for Visual Studio Code supporta solo API REST. A volte le API HTTP vengono chiamate "API V2 di API Gateway"

Per l'esecuzione e il debug delle risorse locali di Amazon API Gateway

1. Scegli uno dei seguenti approcci per creare una configurazione di avvio per una risorsa API Gateway AWS SAM:
 - Opzione 1: visita il codice sorgente del gestore (file .js, .cs o .py) nel AWS SAM, passa il mouse sopra il gestore Lambda e scegli il Aggiunta configurazione di debug CodeLens. Quindi, nel menu, scegli l'elemento contrassegnato come evento API.
 - Opzione 2: Modificare `launch.json` e crea una nuova configurazione di avvio utilizzando la sintassi seguente.

```
{
  "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. Nel codice VSEseguipannello, scegli la configurazione di lancio (denominatamyConfignell'esempio precedente).
3. (Facoltativo) Aggiungi punti di interruzione al codice del progetto Lambda.
4. Type (Tipo)F5oppure scegliPlay (Riproduci)nellaEseguiPannello.
5. Nel riquadro di output, visualizza i risultati.

Configurazione

Quando utilizzi l'api del valore proprietà `invokeTarget.target`, il kit di strumenti modifica la convalida e il comportamento della configurazione di avvio per supportare un campo dell'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": {}
}
```

Sostituisci i valori nell'esempio come segue:

invokeTarget.logicalId

Una risorsa API

path

Il percorso dell'API richiesto dalla configurazione di avvio, ad esempio "path": "/hello".

Deve essere un percorso API valido risolto da `template.yaml` specificato da `invokeTarget.templatePath`.

httpMethod

Può essere uno dei seguenti verbi: "delete", "get", "head", "options", "patch", "post", "put".

payload

Il payload JSON (corpo HTTP) da inviare nella richiesta, con la stessa struttura e le stesse regole del campo [lambda.payload](#).

`payload.path` punta a un file contenente il payload JSON.

`payload.json` specifica un payload JSON in linea.

headers

Mappa facoltativa delle coppie nome-valore, che consente di specificare intestazioni HTTP da includere nella richiesta, come mostrato nell'esempio seguente.

```
"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

Stringa facoltativa che imposta la `queryString` della richiesta, ad esempio "queryString": "abc=def&ghi=jkl".

AWS

Come vengono fornite le informazioni sulla connessione AWS. Per ulteriori informazioni, consulta la `.AWS` proprietà `connection` («aws») tabella nella tabella [Opzioni di configurazione per il debug di applicazioni serverless](#) Sezione.

sam

Come la CLI AWS SAM costruisce l'applicazione. Per ulteriori informazioni, consulta la `.AWS` proprietà dell'interfaccia della riga di comando SAM («sam») tabella nella tabella [Opzioni di configurazione per il debug di applicazioni serverless](#) Sezione.

Opzioni di configurazione per il debug di applicazioni serverless

Quando si apre il `launch.json` file per modificare le configurazioni di debug, è possibile utilizzare il codice VS [IntelliSense](#) funzionalità per visualizzare e completare automaticamente le proprietà valide. Per attivare IntelliSense nell'editor, premi `Ctrl+Barra spaziatrice`.

```

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

```

IntelliSense consente di trovare e definire le proprietà per richiamare le funzioni Lambda direttamente o con `AWS SAM Template`. È inoltre possibile definire proprietà per `"lambda"` (come viene eseguita la funzione), `"sam"` (come la CLI AWS SAM costruisce l'applicazione) e `"aws"` (come vengono fornite le informazioni sulla connessione AWS).

AWS SAM: richiamo diretto del gestore Lambda diretto/richiamo di Lambda basato sul modello

Proprietà	Descrizione
<code>type</code>	Specifica quale estensione gestisce la configurazione di avvio. Sempre impostato su <code>aws-sam</code> per utilizzare la CLI AWS SAM costruire ed eseguire il debug localmente.

Proprietà	Descrizione
<code>name</code>	Specifica un nome facilmente leggibile da visualizzare nell'elenco Debug launch configuration (Configurazione di avvio del debug).
<code>request</code>	Specifica il tipo di configurazione da eseguire dall'estensione designata (<code>aws-sam</code>). Sempre impostato su <code>direct-invoke</code> per avviare la funzione Lambda.

Proprietà	Descrizione
<p><code>invokeTarget</code></p>	<p>Specifica il punto di ingresso per richiamare la risorsa.</p> <p>Per richiamare direttamente la funzione Lambda, imposta i valori per i seguenti campi <code>invokeTarget</code> :</p> <ul style="list-style-type: none"> • <code>target</code>: impostato su <code>code</code>. • <code>lambdaHandler</code> : il nome del gestore della funzione Lambda da richiamare. • <code>projectRoot</code> : il percorso del file dell'applicazione contenente il gestore della funzione Lambda. • <code>architecture</code> — Architettura del processore dell'ambiente emulato in cui viene eseguita l'applicazione SAM Lambda locale. Per determinati runtime, è possibile scegliere <code>arm64</code> anziché l'impostazione di default <code>x86_64</code> architettura. Per ulteriori informazioni, consulta la pagina Creazione di una nuova applicazione serverless (locale) . <p>Per aver richiamato le risorse Lambda con <code>AWS SAMTemplate</code>, impostare i valori per quanto segue <code>invokeTarget</code> campi:</p> <ul style="list-style-type: none"> • <code>target</code>: impostato su <code>template</code>. • <code>templatePath</code> — Il percorso a <code>AWS SAMFile</code> di modello. • <code>logicalId</code> : il nome della risorsa della <code>AWS::Lambda::Function</code> o <code>AWS::Serverless::Function</code> da richiamare. È possibile trovare il nome della risorsa in formato <code>YAML AWS SAMTemplate Template</code>. Nota, <code>Kit di strumenti AWS</code> riconosce implicitamente le funzioni definite con <code>PackageType: Image</code> nella <code>AWS SAMTemplate</code> come Immagini di Funzioni Lambda. Per ulteriori informazioni, consulta Pacchetti di implementazione Lambda nella <code>AWS Lambda Guida</code> per gli sviluppatori.

Proprietà di Lambda ("**lambda**")

Proprietà	Descrizione
<code>environmentVariables</code>	<p>Passa i parametri operativi alla funzione Lambda. Se ad esempio si scrive in un bucket Amazon S3 anziché impostare come hard-coded il nome del bucket in cui si esegue la scrittura, configurare tale nome come una variabile di ambiente.</p> <div data-bbox="620 556 742 594"><p> Note</p></div> <p>Quando si specificano variabili di ambiente per un'applicazione serverless, è necessario aggiungere configurazioni a entrambi AWS SAMTemplate (<code>template.yaml</code>) e il <code>launch.json</code> file.</p> <p>Esempio di formattazione per una variabile di ambiente nella AWS SAMTemplate:</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>Esempio di formattazione per una variabile di ambiente nella <code>launch.json</code> file:</p> <pre>"environmentVariables": { "SAMPLE1": "My sample 1 value" }</pre>

Proprietà	Descrizione
	<ul style="list-style-type: none">"json": coppie chiave-valore in formato JSON che definiscono il payload dell'evento."path": un percorso del file utilizzato come payload dell'evento.
memoryMB	Specifica megabyte (MB) di memoria fornita per l'esecuzione di una funzione Lambda richiamata.
runtime	Specifica il tempo di esecuzione utilizzato dalla funzione Lambda. Per ulteriori informazioni, consulta Tempi di esecuzione AWS Lambda .
timeoutSec	Imposta il tempo permesso, in secondi, prima del timeout della sessione di debug.

Proprietà	Descrizione
pathMappings	<p>Specifica dove si trova il codice locale in relazione a dove viene eseguito nel contenitore.</p> <p>Per impostazione predefinita, il Toolkit per i set di codici VSlocalRoot alla radice del codice della funzione Lambda nell'area di lavoro locale e remoteRoot a /var/task , che è la directory di lavoro predefinita per il codice in esecuzione in Lambda. Se la directory di lavoro viene modificata nel Dockerfile o con il WorkingDirectory parametro nel campo AWS CloudFormationFile di modello, almeno un pathMapping deve essere specificata in modo che il debugger possa mappare correttamente i punti di interruzione impostati localmente al codice in esecuzione nel contenitore Lambda.</p> <p>Esempio di formattazione pathMappings nella launch.js on file:</p> <pre data-bbox="597 982 1507 1304"> "pathMappings": [{ "localRoot": " \${workspaceFolder}/sam-app/HelloWorldFunction ", "remoteRoot": " /var/task " }] </pre> <p>Avvertenze:</p> <ul data-bbox="597 1417 1507 1612" style="list-style-type: none"> • Per le funzioni Lambda basate su immagini .NET, remoteRoot entry deve essere la directory di compilazione. • Per le funzioni Lambda basate su Node.js, è possibile specificare solo una singola voce di mappatura del percorso.

Il Toolkit for VS Code utilizza il AWS SAM CLI per costruire ed eseguire il debug di applicazioni serverless a livello locale. È possibile configurare il comportamento dei comandi della CLI AWS SAM che utilizzano le proprietà della configurazione "sam" nel file launch.json.

Proprietà della CLI AWS SAM ("**sam**")

Proprietà	Descrizione	Valore predefinito
<code>buildArguments</code>	Configura il modo in cui il comando <code>sam build</code> costruisce il codice sorgente Lambda. Per visualizzare le opzioni di costruzione, consulta sam build nella Guida per gli sviluppatori di AWS Serverless Application Model.	Stringa vuota
<code>containerBuild</code>	Indica se costruire la funzione all'interno di un container Docker di tipo Lambda.	<code>false</code>
<code>dockerNetwork</code>	Specifica il nome o l'ID di una rete Docker esistente a cui devono connettersi i container Docker Lambda con la rete bridge di default. Se non specificato, i container Lambda si connettono solo alla rete Docker bridge di default.	Stringa vuota
<code>localArguments</code>	Specifica ulteriori argomenti di richiamo locale.	Stringa vuota
<code>skipNewImageCheck</code>	Specifica se il comando deve saltare il pull dell'ultima immagine Docker per il tempo di esecuzione Lambda.	<code>false</code>
<code>template</code>	Personalizzazione delleAWS SAMTemplate utilizzando i parametri per inserire i valori dei clienti. Per ulteriori	<code>"parameters": {}</code>

Proprietà	Descrizione	Valore predefinito
	informazioni, consultare Parametri nella Guida per l'utente AWS CloudFormation.	

Proprietà della connessione AWS ("**aws**")

Proprietà	Descrizione	Valore predefinito
<code>credentials</code>	Seleziona un profilo specifico (ad esempio, <code>profile:default</code>) dal file di credenziali per ottenere le credenziali AWS.	LaAWScredenziali che il tuo esistente condivisoAWSconfig file o condivisoAWSfile delle credenziali fornire al Toolkit for VS Code.
<code>region</code>	Imposta la regione AWS del servizio (ad esempio, <code>us-east-1</code>).	La regione AWS di default associata al profilo delle credenziali attive.

Esempio: Configurazione del lancio del modello

Ecco un file di configurazione di avvio di unAWS SAMtarget del modello:

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

```
    }  
  ]  
}
```

Esempio: Configurazione di avvio del codice

Ecco un esempio di file di configurazione di lancio per un target di funzione 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": {}  
      }  
    }  
  ]  
}
```

Risoluzione dei problemi delle applicazioni serverless

In questo argomento sono illustrati i comuni errori che possono verificarsi durante la creazione di applicazioni serverless con il Toolkit for VS Code e le relative modalità di risoluzione.

Argomenti

- [Come posso usare samconfig.toml con una configurazione di avvio SAM?](#)
- [Errore: «Errore di runtime: Il contenitore non esiste»](#)
- [Errore: «Docker.errors.apiError: 500 - Errore server in corso... Hai raggiunto il limite di pull rate.»](#)
- [Errore: «Errore del server 500: Montare C:\Users\...»](#)
- [Usando WSL, webviews \(ad esempio, «Invoke on»AWS«form\) sono rotti](#)
- [Debug di un'applicazione TypeScript, ma i punti di interruzione non funzionano](#)

Come posso usare samconfig.toml con una configurazione di avvio SAM?

Per specificare il percorso di interfaccia a riga di comando SAM [samconfig.toml](#) configurando il `--config-file` argomento nella `localArguments` proprietà della configurazione di avvio. Ad esempio, se il file `samconfig.toml` si trova al livello superiore dell'area di lavoro:

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

Errore: «Errore di runtime: Il contenitore non esiste»

La `sam build` il comando può mostrare questo errore se il sistema non dispone di spazio su disco sufficiente per il contenitore Docker. Se lo storage di sistema ha solo 1-2 GB di spazio disponibile, `sam build` potrebbe non riuscire durante l'elaborazione, anche se lo storage del sistema non è completamente pieno prima dell'avvio della compilazione. Per ulteriori informazioni, consulta [GitHub](#).

Errore: «Docker.errors.apiError: 500 - Errore server in corso... Hai raggiunto il limite di pull rate.»

Docker Hub limita le richieste che gli utenti anonimi possono effettuare. Se il sistema raggiunge il limite, Docker non riesce e questo errore viene visualizzato nella vista OUTPUT di 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")
```

Assicurati che il tuo Sistema Docker è autenticato con le credenziali Docker Hub.

Errore: «Errore del server 500: Montare C:\Users\...»

Gli utenti Windows potrebbero vedere questo errore di montaggio di Docker durante il debug AWS SAM applicazioni:

```
Fetching lambci/lambda:nodejs10.x Docker container image.....  
2019-07-12 13:36:58 Mounting C:\Users\\AppData\Local\Temp\ ... as /var/  
task:ro,delegated inside runtime container
```

```
Traceback (most recent call last):  
...  
requests.exceptions.HTTPError: 500 Server Error: Internal Server Error ...
```

Prova ad aggiornare le credenziali per le unità condivise (nelle impostazioni Docker).

Usando WSL, webviews (ad esempio, «Invoke on»AWS«form) sono rotti

Questo è un problema noto per gli utenti di Cisco VPN. Per ulteriori informazioni, consulta [GitHub](#).

Una soluzione alternativa è suggerita in [questo problema di tracciamento WSL](#).

Debug di un'applicazione TypeScript, ma i punti di interruzione non funzionano

Ciò accadrà se non esiste una mappa sorgente per collegare il file JavaScript compilato al file TypeScript sorgente. Per risolvere il problema, apri `iltsconfig.json` e assicurati che siano impostati i seguenti valori e opzioni: `"inlineSourceMap": true`.

Utilizzo dei documenti di automazione di Systems Manager

AWS Systems Manager offre visibilità e controllo dell'infrastruttura su AWS. Systems Manager offre un'interfaccia utente unificata attraverso la quale consultare dati operativi da diversi servizi AWS e automatizzare i processi operativi sulle risorse AWS.

Un [documento di Systems Manager](#) definisce le operazioni eseguite da Systems Manager sulle istanze gestite. Un documento di automazione è un tipo di documento di Systems Manager che si utilizza per l'esecuzione di attività comuni di manutenzione e implementazione, ad esempio la creazione o l'aggiornamento di una Amazon Machine Image (AMI). In questo argomento viene illustrato come creare, modificare ed eliminare documenti di automazione con AWS Toolkit for Visual Studio Code.

Argomenti

- [Presupposti e prerequisiti](#)
- [Autorizzazioni IAM per i documenti di automazione di Systems Manager](#)
- [Creazione di un nuovo documento di automazione di Systems Manager](#)
- [Apertura di un documento di automazione di Systems Manager esistente](#)
- [Modifica di un documento di automazione di Systems Manager](#)
- [Creazione di un documento di automazione di Systems Manager](#)
- [Eliminazione di un documento di automazione di Systems Manager](#)

- [Esecuzione di un documento di automazione di Systems Manager](#)
- [Risoluzione dei problemi dei documenti di Systems Manager Automation in Toolkit for VS Code](#)

Presupposti e prerequisiti

Prima di iniziare, assicurati di:

- Hai installato Visual Studio Code e l'ultima versione di AWS Toolkit for Visual Studio Code. Per ulteriori informazioni, consulta la pagina [Installazione del AWS Toolkit for Visual Studio Code](#).
- Sapere come funziona Systems Manager. Per ulteriori informazioni, consulta la [Guida per l'utente di AWS Systems Manager](#).
- Conoscere i casi d'uso di automazione di Systems Manager. Per ulteriori informazioni, consulta la sezione [Automazione di AWS Systems Manager](#) nella Guida per l'utente di AWS Systems Manager.

Autorizzazioni IAM per i documenti di automazione di Systems Manager

Nel Toolkit for VS Code è necessario disporre di un profilo di credenziali che contenga il AWS Identity and Access Management (IAM) le autorizzazioni necessarie per creare, modificare, pubblicare ed eliminare i documenti di automazione di Systems Manager. Il seguente documento di policy definisce le autorizzazioni IAM necessarie che possono essere utilizzate in una policy principale:

```
{
  "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": "*"
    }
  ]
}
```

```
]
}
```

Per informazioni sull'aggiornamento di una policy IAM, consulta la sezione [Creazione di policy IAM](#) nella Guida per l'utente di IAM. Per informazioni su come configurare il profilo di credenziali, consulta [AWS IAM credenziali](#).

Creazione di un nuovo documento di automazione di Systems Manager

È possibile creare un nuovo documento di automazione in JSON o YAML utilizzando Visual Studio Code. Quando si crea un nuovo documento di automazione, viene presentato come file senza titolo. È possibile assegnare un nome al file e salvarlo in VS Code, tuttavia il nome del file non è visibile a AWS.

Per creare un nuovo documento di automazione

1. Apri VS Code.
2. Sul menu di visualizzazione, scegli Tavolozza dei comandi per aprire la tavolozza dei comandi.
3. Nella tavolozza dei comandi, inserisci **AWSToolkit Creazione locale di un nuovo documento di Systems Manager**.
4. Scegli uno dei modelli di avvio per un esempio di Hello World.
5. Scegliere JSON o YAML.

Viene creato un nuovo documento di automazione.

Note

Il tuo nuovo documento di automazione in VS Code non viene visualizzato automaticamente in AWS. Prima di poterlo eseguire, è necessario pubblicarlo su AWS.

Apertura di un documento di automazione di Systems Manager esistente

È possibile utilizzare il plugin **AWS Explorer** per individuare i documenti di automazione di Systems Manager esistenti. Quando si apre un documento di automazione esistente, viene visualizzato come file senza titolo in VS Code.

Per aprire il documento Automation

1. Apri VS Code.
2. Nella barra di navigazione a sinistra, scegliere AWS per aprire AWS Explorer.
3. Nella AWS Explorer Systems Manager, scegli l'icona di download sul documento che desideri aprire e quindi scegli la versione del documento. Il file verrà aperto nel formato per quella versione. Altrimenti scegli uno dei due Scarica come JSON o Scarica come file YAML.

Note

Il salvataggio locale di un documento di automazione come file in VS Code non lo fa apparire in AWS. Deve essere pubblicato per AWS prima dell'esecuzione.

Modifica di un documento di automazione di Systems Manager

Se possiedi documenti di Automation, questi vengono visualizzati nella Di proprietà di Me categoria di documenti di Systems Manager nel AWS Explorer. È possibile possedere documenti di automazione già esistenti in AWS e puoi possedere documenti nuovi o aggiornati che hai pubblicato in precedenza AWS da VS Code.

Quando apri un documento di automazione per la modifica in VS Code, puoi farne di più di quanto tu possa fare in AWS Management Console. Ad esempio:

- C'è una convalida dello schema su entrambi JSON e YAML formati.
- Ci sono frammenti disponibili nell'editor di documenti per creare qualsiasi tipo di fase di automazione.
- È disponibile il supporto per il completamento automatico su varie opzioni in JSON e YAML.

Utilizzo delle versioni

I documenti di automazione di Systems Manager utilizzano versioni per la gestione delle modifiche. È possibile scegliere la versione predefinita per un documento di automazione in VS Code.

Per impostare una versione di default

- In AWS Explorer, individua il documento per il quale impostare la versione di default, apri il menu contestuale (facendo clic con il pulsante destro del mouse) del documento e scegli Set default version (Imposta versione predefinita).

Note

Se il documento scelto ha una sola versione, non sarà possibile modificare l'impostazione predefinita.

Creazione di un documento di automazione di Systems Manager

Dopo avere modificato il documento di automazione in VS Code, è possibile pubblicarlo in AWS.

Per pubblicare il documento di automazione

1. Apri il documento di automazione che desideri pubblicare utilizzando la procedura descritta nella sezione [Apertura di un documento di automazione di Systems Manager esistente](#).
2. Apporta le modifiche desiderate per la pubblicazione. Per ulteriori informazioni, consulta la pagina [Modifica di un documento di automazione di Systems Manager](#).
3. In alto a destra del file aperto, scegliere l'icona di caricamento.
4. Nella finestra di dialogo del flusso di lavoro di pubblicazione, scegliere la AWS Regione in cui effettuare la pubblicazione del documento di automazione.
5. Se stai pubblicando un nuovo documento, scegli Creazione rapida. In caso contrario, scegli Aggiornamento rapido per aggiornare un documento di automazione esistente AWS Regione .
6. Immettere il nome per questo documento di automazione.

Quando si pubblica un aggiornamento di un documento di automazione esistente in AWS, viene aggiunta una nuova versione al documento.

Eliminazione di un documento di automazione di Systems Manager

È possibile eliminare i documenti di automazione in VS Code. Se si elimina un documento di automazione, oltre a quest'ultimo vengono eliminate anche tutte le rispettive versioni.

Important

- L'eliminazione è un'azione distruttiva che non può essere annullata.
- L'eliminazione di un documento di automazione già eseguito non comporta l'eliminazione delle risorse che sono state create o modificate all'avvio.

Per eliminare un documento di automazione

1. Apri VS Code.
2. Nella barra di navigazione a sinistra, scegliere AWS per aprire AWS Explorer.
3. Nella AWS Explorer Systems Manager, apri il menu contestuale (pulsante destro del mouse) del documento che desideri eliminare, quindi scegli Eliminazione di documento.

Esecuzione di un documento di automazione di Systems Manager

Una volta pubblicato il documento Automation su AWS, puoi eseguirlo affinché effettui operazioni per conto tuo nel tuo AWS account. Puoi eseguire un documento di automazione utilizzando la AWS Management Console, le API di Systems Manager, la AWS CLI o AWS Tools for PowerShell. Per istruzioni sull'esecuzione di un documento di automazione, consulta [Esecuzione di un'automazione semplice](#) nella AWS Systems Manager Guida per l'utente di.

In alternativa, se desideri utilizzare uno degli SDK AWS con le API di Systems Manager per eseguire il documento di automazione, consulta i [riferimenti sugli SDK AWS](#).

Note

L'esecuzione di un documento di automazione può creare nuove risorse in AWS e possono incorrere in costi di fatturazione. Prima di avviare il documento di automazione, ti consigliamo vivamente di verificare quali risorse creerà nell'account.

Risoluzione dei problemi dei documenti di Systems Manager Automation in Toolkit for VS Code

Ho salvato il documento di automazione in VS Code, ma non lo vedo nella AWS Management Console.

Il salvataggio di un documento di automazione in VS Code non pubblica il documento di automazione in AWS. Per ulteriori informazioni sulla pubblicazione di un documento di automazione, consulta la sezione [Creazione di un documento di automazione di Systems Manager](#).

La pubblicazione del documento di automazione non è riuscita per un errore di autorizzazione.

Assicurati che il profilo delle credenziali AWS disponga delle autorizzazioni necessarie per pubblicare documenti di automazione. Per un esempio di policy delle autorizzazioni, consulta [Autorizzazioni IAM per i documenti di automazione di Systems Manager](#).

Ho pubblicato il mio documento Automation su AWS, ma non lo vedo nella AWS Management Console.

Assicurati di avere pubblicato il documento nello stesso documento AWS Regione in cui stai navigando AWS Management Console.

Ho eliminato il documento di automazione, ma mi vengono comunque addebitati i costi delle risorse create.

L'eliminazione di un documento di automazione non elimina le risorse create o modificate. Puoi identificare le risorse AWS create, consultare gli addebiti e scegliere quali risorse eliminare nella [Console di gestione della fatturazione di AWS](#).

Lavorare con AWS Step Functions

AWS Toolkit for Visual Studio Code (VS Code) fornisce supporto per [AWS Step Functions](#).

Utilizzando il Toolkit for VS Code, è possibile creare, aggiornare ed eseguire macchine a stati Step Functions.

Argomenti

- [Lavorare con AWS Step Functions](#)

Lavorare con AWS Step Functions

È possibile utilizzare AWS Toolkit for Visual Studio Code (VS Code) per eseguire varie operazioni con [macchine a stati](#).

Argomenti

- [Prerequisiti](#)
- [Lavora con macchine a stati in VS Code](#)
- [Modelli di macchine a stati](#)
- [Visualizzazione del grafico di State Machine](#)
- [Frammenti di codice](#)
- [Completamento e convalida del codice](#)

Prerequisiti

- Assicurati che il tuo sistema soddisfi i prerequisiti specificati in [Installazione del Toolkit for VS Code](#), quindi installa il toolkit.
- Assicurati di aver configurato le tue credenziali prima di aprire Explorer.AWS

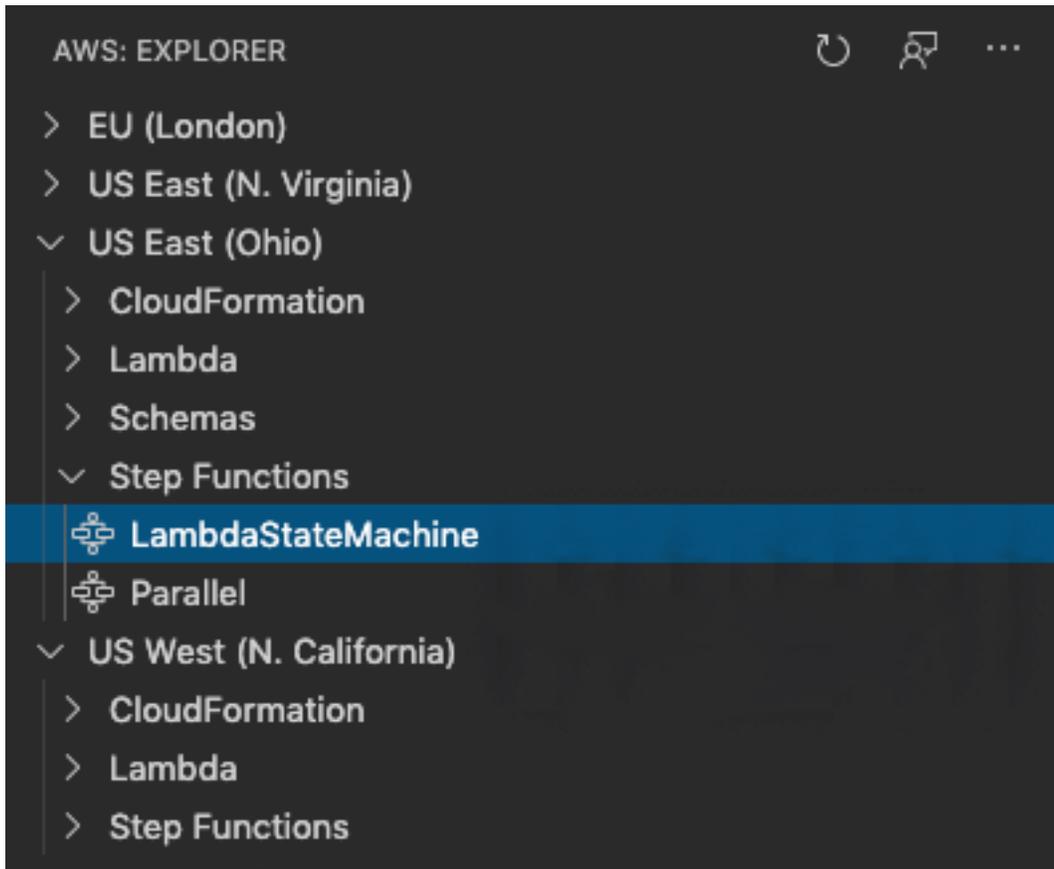
Lavora con macchine a stati in VS Code

È possibile utilizzare VS Code per interagire con macchine a stati remoti e sviluppare macchine a stati localmente in formato JSON o YAML. È possibile creare o aggiornare macchine a stati, elencare le macchine a stati esistenti, eseguirle e scaricarle. VS Code consente inoltre di creare nuove macchine a stati da modelli, osservare una visualizzazione della macchina a stati e fornire frammenti, completamento e convalida del codice.

Elenca le macchine a stati esistenti

Se sono già state create macchine a stato, è possibile visualizzarne un elenco:

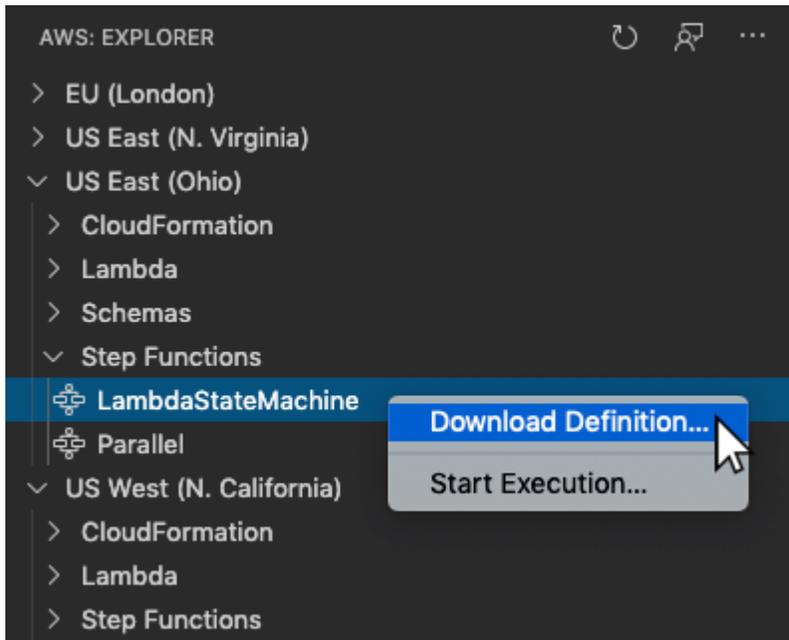
1. Apri AWS Explorer.
2. Seleziona Step Functions
3. Verificare che siano elencate tutte le macchine a stato nell'account.



Scarica una macchina a stati

Per scaricare una macchina a stati:

1. In AWS Explorer, fai clic con il pulsante destro del mouse sulla macchina a stati che desideri scaricare.
2. Selezionare Download, quindi selezionare la posizione in cui si desidera scaricare la macchina a stati.
3. Verificare che sia stata scaricata correttamente.



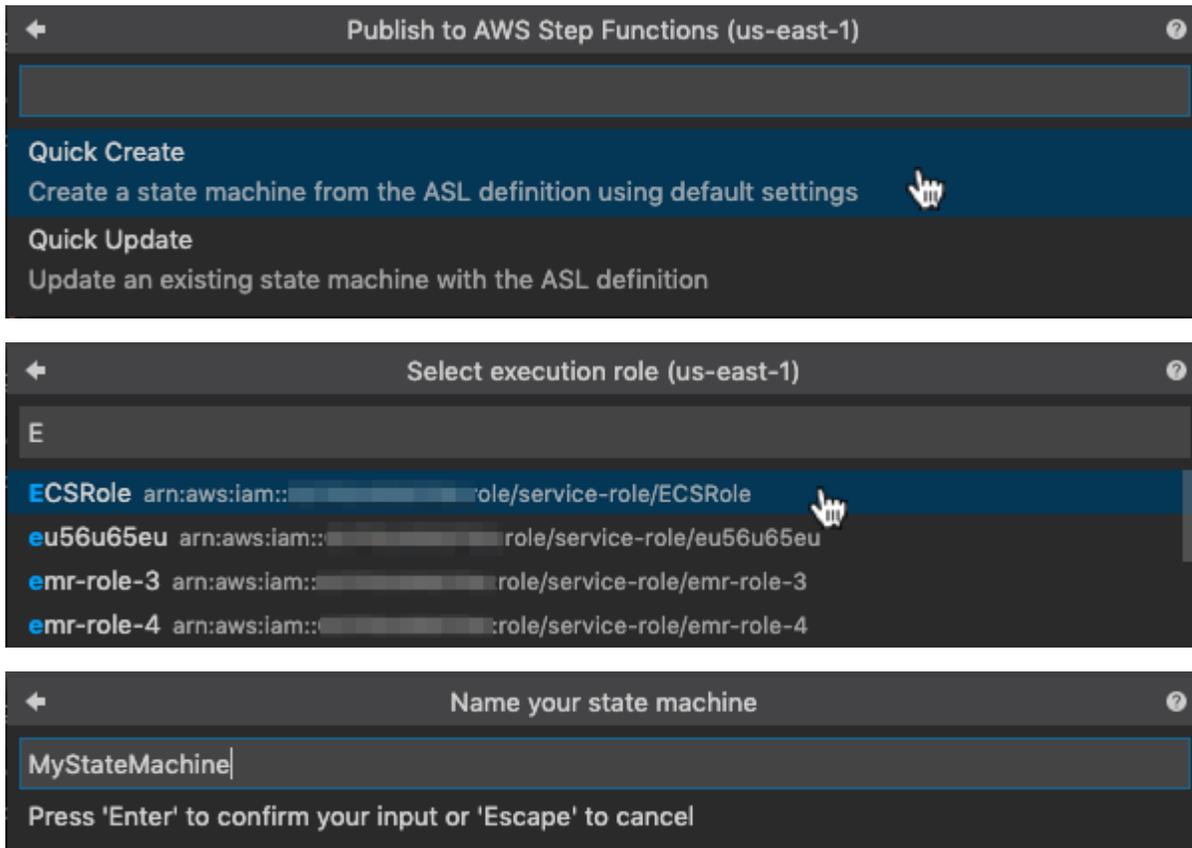
Creazione di una macchina a stati

È possibile creare autonomamente una nuova macchina a stati oppure è possibile utilizzare un modello. Per ulteriori informazioni sulla creazione di una macchina a stati da un modello, vedere la sezione [Modelli di macchina di stati](#) . Per creare una nuova macchina a stati:

1. Crea un nuovo file [Amazon States Language](#) (ASL) con la definizione della macchina a stati. Usa il menu in basso a destra per impostarlo come Amazon States Language.
2. Selezionare Pubblica in Step Functions.

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

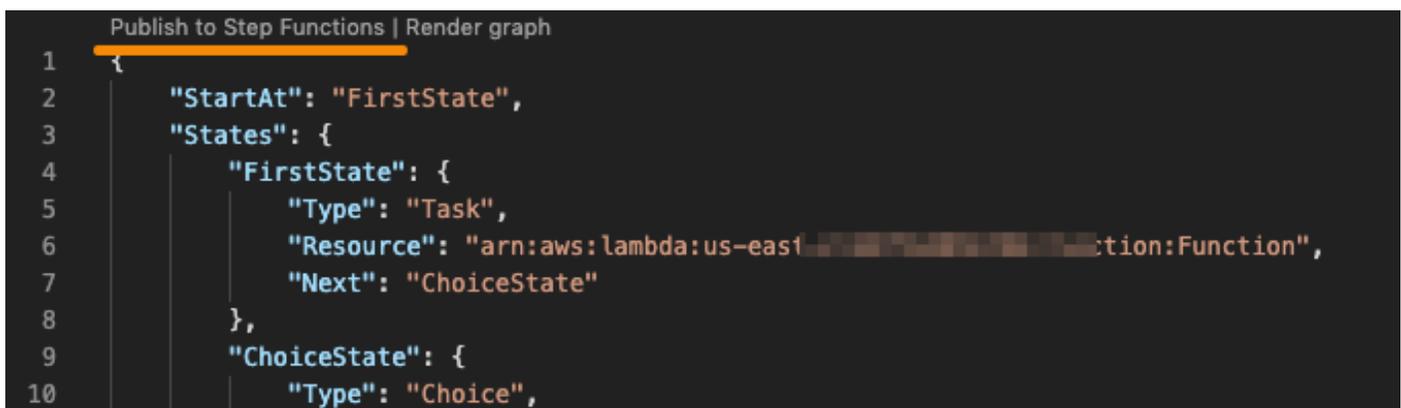
3. Selezionare Quick Create (Creazione rapida), scegliere un ruolo e assegnare un nome alla macchina a stati.



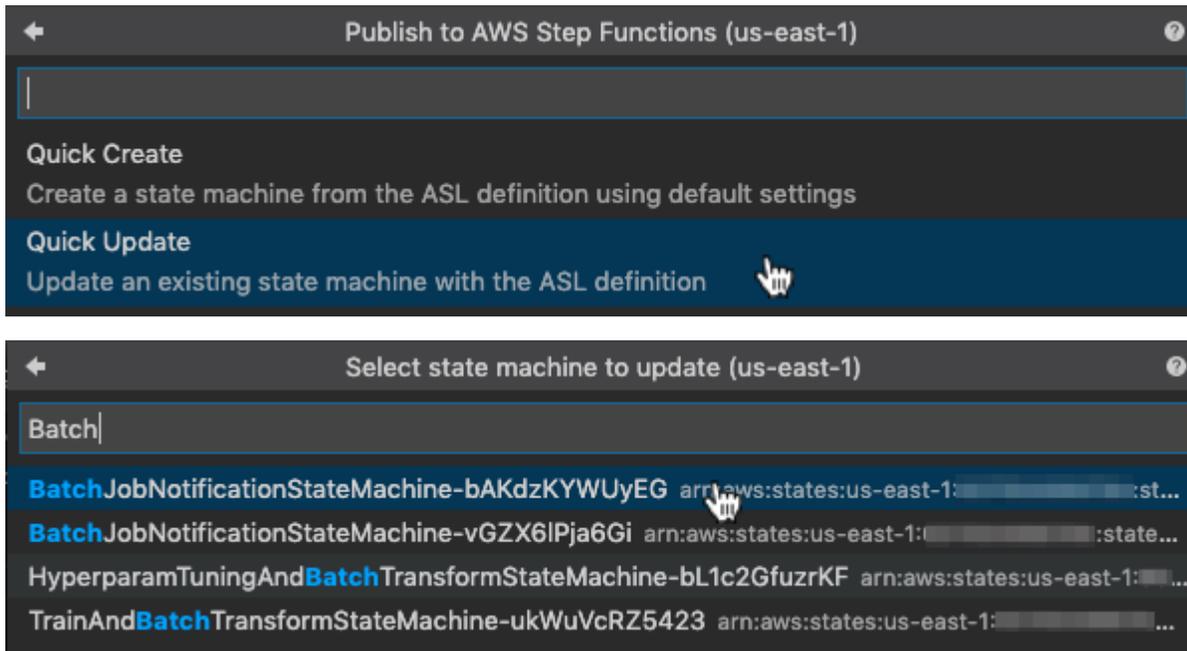
Aggiornamento di una macchina a stati

Per aggiornare una macchina a stati:

1. Modificare il file ASL con la definizione della macchina a stati.
2. Selezionare Pubblica in Step Functions.



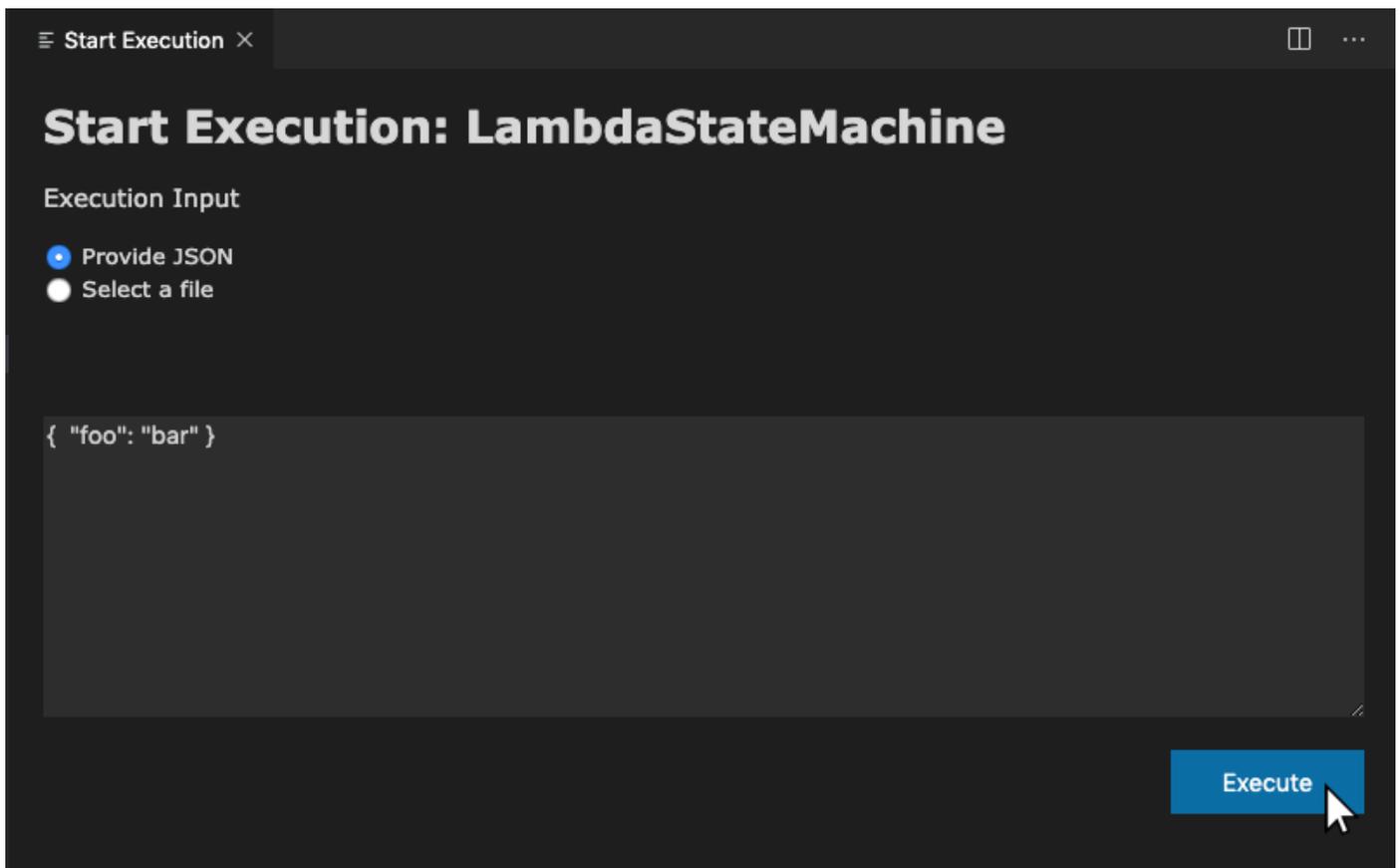
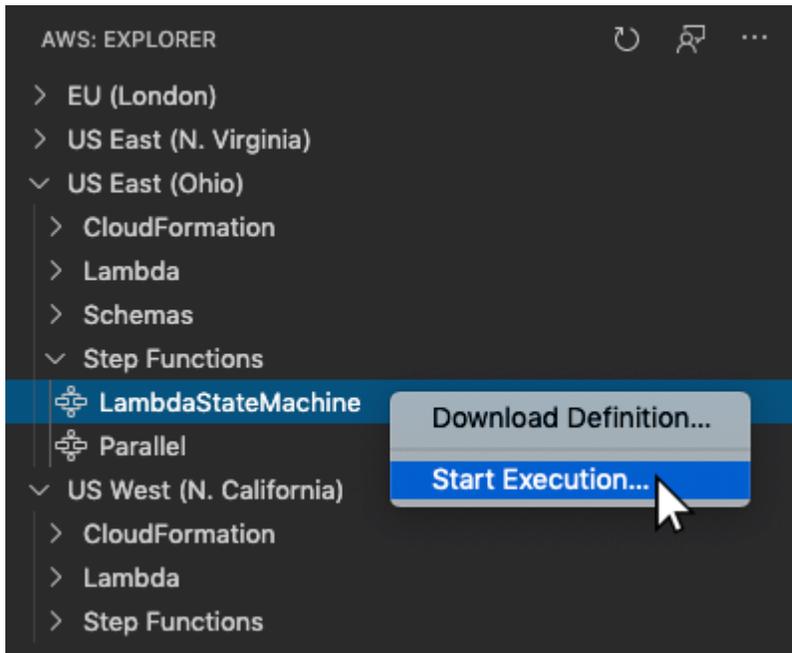
3. Selezionare Quick Update (Aggiornamento rapido), quindi selezionare la macchina a stati che si desidera aggiornare.



Esegui una macchina a stati

Per eseguire una macchina a stati:

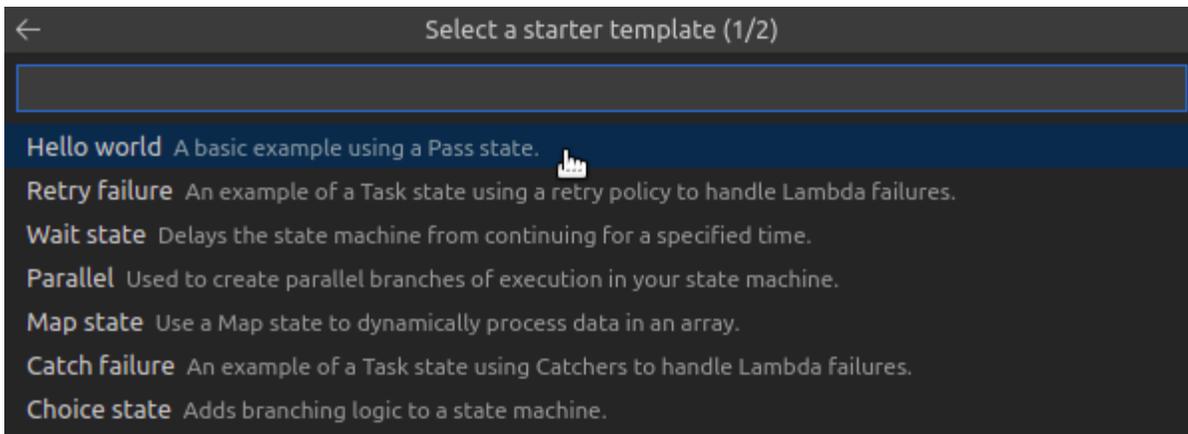
1. In AWS Explorer, fai clic con il pulsante destro del mouse sulla macchina a stati che desideri eseguire.
2. Fornisci input per la macchina a stati. È possibile provare sia l'input da un file che l'input in una casella di testo.
3. Avvia la macchina a stati e verifica che funzioni correttamente.



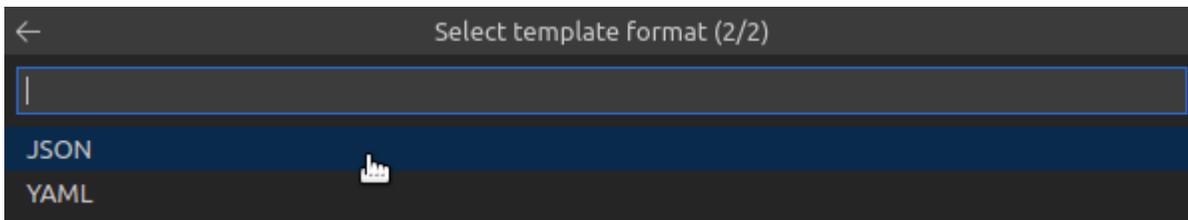
Modelli di macchine a stati

Quando si crea una macchina a stati, è possibile crearla da un modello. Il modello contiene una definizione di macchina a stati di esempio con diversi stati di uso comune e fornisce un punto di partenza. Per utilizzare i modelli di macchina a stati:

1. Apri la palette dei comandi in VS Code.
2. Seleziona AWS Toolkit Crea una nuova macchina a stati Step Functions.
3. Scegliere il modello che si desidera utilizzare.



4. Scegli se vuoi usare il formato modello JSON o YAML.



Visualizzazione del grafico di State Machine

Le visualizzazioni dei grafici consentono di vedere l'aspetto della macchina a stati in formato grafico. Quando si crea una visualizzazione grafica, si apre un'altra scheda che mostra una visualizzazione della macchina a stati JSON o YAML. È quindi possibile confrontare la definizione della macchina a stati che si sta scrivendo contemporaneamente alla sua visualizzazione. Quando si modifica la definizione della macchina a stati, la visualizzazione verrà aggiornata.

Note

Per creare una visualizzazione di una definizione di macchina a stati, la definizione deve essere aperta nell'editor attivo. Se si chiude o viene rinominato il file di definizione, la visualizzazione verrà chiusa.

Per creare una visualizzazione del grafico della macchina a stati:

1. Definire la macchina a stati.
2. Apri la palette dei comandi in VS Code.
3. Per creare una visualizzazione, usa il pulsante di visualizzazione nell'angolo in alto a destra o scegli AWS Renderizza grafico.

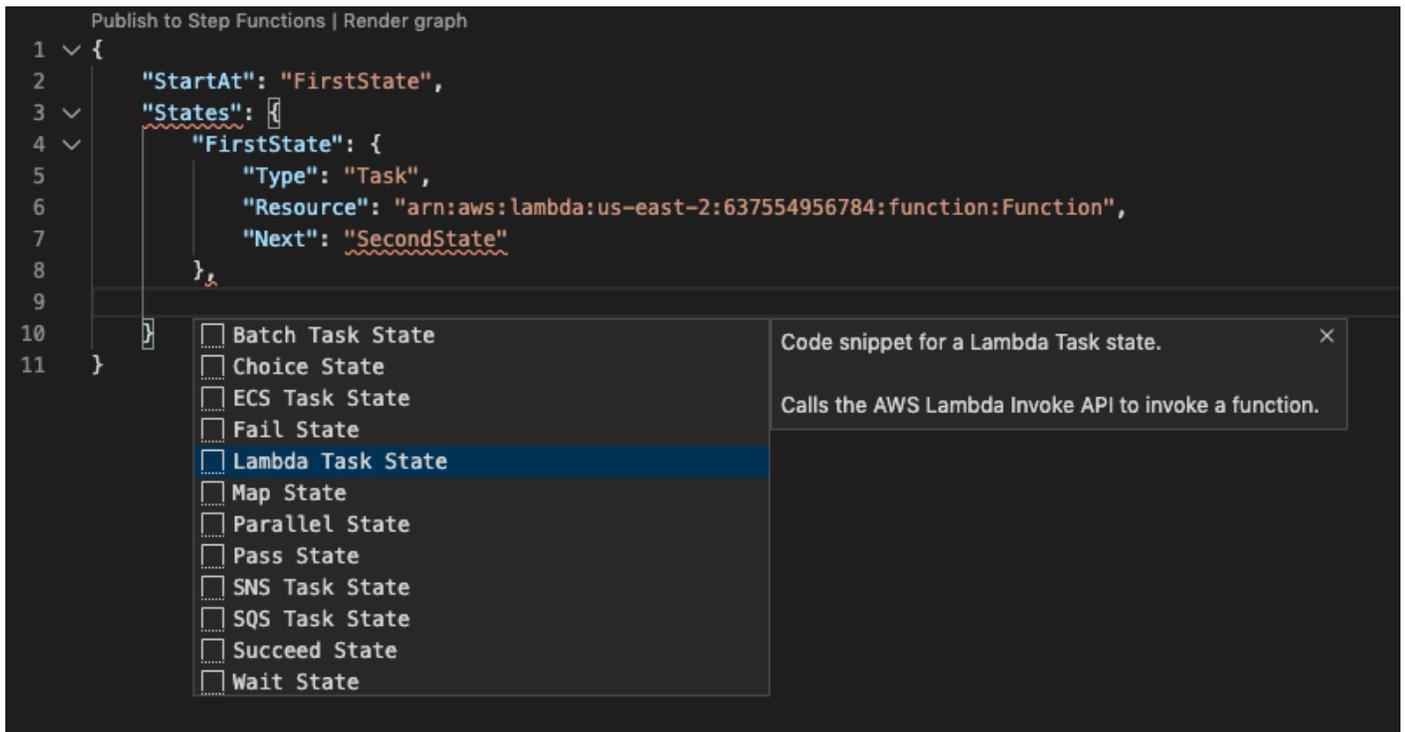
```
1  {
2    "Comment": "An example of the Amazon States Language using a choice
3    state.",
4    "StartAt": "FirstState",
5    "States": {
6      "FirstState": {
7        "Type": "Task",
8        "Resource":
9        "arn:aws:lambda:us-east-2:637554956784:function:Function",
10       "Next": "ChoiceState"
11     },
12     "ChoiceState": {
13       "Type": "Choice",
14       "Choices": [
15         {
16           "Variable": "$.foo",
17           "NumericEquals": 1,
18           "Next": "FirstMatchState"
19         },
20         {
21           "Variable": "$.foo",
22           "NumericEquals": 2,
23           "Next": "SecondMatchState"
24         }
25       ],
26       "Default": "DefaultState"
27     },
28     "FirstMatchState": {
29       "Type": "Task",
30       "Resource":
31       "arn:aws:lambda:us-east-2:637554956784:function:Function",
32       "Next": "NextState"
33     },
34     "SecondMatchState": {
35       "Type": "Task",
36       "Resource":
37       "arn:aws:lambda:us-east-2:637554956784:function:Function",
38       "Next": "NextState"
39     },
40     "NextState": {
41       "Type": "Task",
42       "Resource":
43       "arn:aws:lambda:us-east-2:637554956784:function:Function",
44       "Next": "End"
45     },
46     "DefaultState": {
47       "Type": "Task",
48       "Resource":
49       "arn:aws:lambda:us-east-2:637554956784:function:Function",
50       "Next": "End"
51     },
52     "End": {}
53   }
54 }
```

Ln 2, Col 81 Spaces: 4 UTF-8 LF Amazon States Language AWS Credentials: profile:default

Frammenti di codice

I frammenti di codice consentono di inserire brevi sezioni di codice. Per utilizzare frammenti di codice:

1. Apri un file e salvalo con l'estensione `.asl.json` per il formato JSON o `.asl.yaml` per il formato YAML.
2. Creare una nuova macchina a stati con la proprietà States (Stati).
3. Posizionare il cursore all'interno di States (Stati).
4. Utilizzare la combinazione di tasti `Control + Space` e selezionare il frammento di codice preferito.
5. Utilizzare `Tab` per passare la variabile e i parametri nel frammento di codice.
6. Provare i frammenti Retry (Riprova) e Catch (Cattura) posizionando il cursore all'interno dello stato relativo.



Completamento e convalida del codice

Per vedere come funziona il completamento del codice:

1. Creare diversi stati.
2. Posiziona il cursore dopo una proprietà Next o Default. StartAt
3. Utilizzare la combinazione di tasti `Control + Space` per elencare i completamenti disponibili. Ulteriori proprietà possono essere accessibili utilizzando di nuovo `Control + Space` e saranno basate sul valore Type di State.

4. Durante l'utilizzo, la convalida del codice avverrà per:

- Proprietà mancanti
- Valori errati
- Nessuno stato terminale
- Stati inesistenti verso i quali è stato eseguito il puntamento

```

"FirstMatchState": {
  "Type": "Task",
  "Resource": "arn:aws:lambda:us-east-2:637554956784:function:Function",
  "Next": ""
},
"SecondMatchState": {
  "Type": "Task",
  "Resource": "arn:aws:lambda:us-east-2:637554956784:function:Function",
  "Next": "SecondMatchState"
},
"DefaultState": {
  "Type": "Fail",
  "Error": "DefaultStateError",
  "Cause": "No Matches!"
},
"NextState": {
  "Type": "Task",
  "Resource": "arn:aws:lambda:us-east-2:637554956784:function:Function",
  "End": true
}

```

```

"FirstMatchState": {
  "Type": "Task",
  "Resource": "arn:aws:lambda:us-east-2:637554956784:function:Function",
  "Catch": [
    {
      "Error": "DefaultStateError",
      "Cause": "No Matches!"
    }
  ]
},

```

An array of objects, called Catchers, that define a fallback state. This state is executed if the state encounters runtime errors and its retry policy is exhausted or isn't defined.

Utilizzo di Threat Composer

È possibile utilizzarlo AWS Toolkit for Visual Studio Code per lavorare con lo strumento Threat Composer. Threat Composer è uno strumento di modellazione delle minacce che può semplificare il processo di modellazione delle minacce.

[Per informazioni dettagliate sullo strumento Threat Composer, consulta l'archivio Threat Composer. GitHub](#)

I seguenti argomenti descrivono come utilizzare Threat Composer in. AWS Toolkit for Visual Studio Code

Argomenti

- [Utilizzo di Threat Composer dal Toolkit](#)

Utilizzo di Threat Composer dal Toolkit

Con Threat Composer puoi creare, visualizzare e modificare i modelli di minaccia di Threat Composer direttamente in VS Code. Per informazioni dettagliate sullo strumento Threat Composer, consulta il repository [Threat Composer. GitHub](#)

Le seguenti sezioni descrivono come accedere agli strumenti Threat Composer in. AWS Toolkit for Visual Studio Code

Accesso a Threat Composer dal Toolkit

Esistono 3 modi principali per accedere a Threat Composer dal Toolkit.

Accesso a Threat Composer tramite un modello di minaccia esistente

Per aprire Threat Composer, apri un file di modello di minaccia esistente (estensione `.tc.json`) in VS Code. Threat Composer apre e rende automaticamente una visualizzazione del file del modello di minaccia nella finestra dell'editor di VS Code.

Creazione di un nuovo modello di minaccia Threat Composer

1. Dal menu principale di VS Code, espandi File, quindi scegli Nuovo file.
2. Dalla finestra di dialogo Nuovo file, scegli Threat Composer File... .

3. Quando richiesto, inserisci un `file` name, quindi premi il **enter** tasto per aprire Threat Composer e creare una visualizzazione del file di modello di minaccia vuoto in una nuova finestra dell'editor di VS Code.

Creazione di un nuovo modello di minaccia Threat Composer dalla Command Palette

1. Da VS Code, apri la palette dei comandi premendo **Cmd + Shift + P** o **Ctrl + Shift + P** (Windows).
2. Nel campo di ricerca, inserisci **Threat Composer** e scegli Crea nuovo file Threat Composer quando viene inserito nei risultati.
3. Quando richiesto, inserisci un `file` name, quindi premi il **enter** tasto per aprire Threat Composer e creare una visualizzazione del tuo file di modello di minaccia vuoto in una nuova finestra dell'editor di VS Code.

Utilizzo delle risorse

Oltre ad accedere ai AWS servizi elencati per impostazione predefinita in AWS Explorer, puoi anche andare su Risorse e scegliere tra centinaia di risorse da aggiungere all'interfaccia. In AWS, una risorsa è un'entità con cui puoi lavorare. Alcune delle risorse che possono essere aggiunte includono Amazon AppFlow, Amazon Kinesis Data AWS IAM Streams, ruoli, VPC Amazon e CloudFront le distribuzioni Amazon.

Dopo aver effettuato la selezione, puoi andare a Risorse ed espandere il tipo di risorsa per elencare le risorse disponibili per quel tipo. Ad esempio, se selezioni il tipo di risorsa `AWS Toolkit::Lambda::Function`, potrai accedere alle risorse che definiscono funzioni diverse, le loro proprietà e i relativi attributi.

Dopo aver aggiunto un tipo di risorsa a Resources (Risorse), potrai interagire con quel tipo e le relative risorse nei seguenti modi:

- Visualizza un elenco di risorse esistenti disponibili nella AWS regione corrente per questo tipo di risorsa.
- Visualizza una versione di sola lettura del JSON file che descrive una risorsa.
- Potrai copiare l'identificatore della risorsa per la risorsa.
- Visualizza la AWS documentazione che spiega lo scopo del tipo di risorsa e lo schema (in JSON e YAML formati) per la modellazione di una risorsa.

- Crea una nuova risorsa modificando e salvando un modello in JSON formato conforme a uno schema. *
- Aggiorna o elimina una risorsa esistente.*

Important

* Nella versione corrente AWS Toolkit for Visual Studio Code dell'opzione per creare, modificare ed eliminare risorse è una funzionalità sperimentale. Dato che le funzionalità sperimentali vengono continuamente testate e aggiornate, potrebbero presentare problemi di usabilità. E le funzionalità sperimentali possono essere rimosse AWS Toolkit for Visual Studio Code senza preavviso.

Per consentire l'uso di funzionalità sperimentali per le risorse, apri il riquadro Impostazioni nel tuo VS Code IDE, espandi Estensioni e scegli AWS Toolkit.

In AWS Toolkit Experiments, seleziona questa opzione `jsonResourceModification` per consentirti di creare, aggiornare ed eliminare risorse.

Per ulteriori informazioni, consulta [Utilizzo delle caratteristiche sperimentali](#).

IAM autorizzazioni per l'accesso alle risorse

Sono necessarie AWS Identity and Access Management autorizzazioni specifiche per accedere alle risorse associate AWS ai servizi. Ad esempio, un'IAM identità, come un utente o un ruolo, richiede le autorizzazioni `Lambda::Function` per accedere alle risorse.

Oltre alle autorizzazioni per le risorse di servizio, un'IAM identità richiede le autorizzazioni per consentire a Toolkit for VS Code di AWS richiamare le operazioni di Cloud API Control per suo conto. API Le operazioni di Cloud Control consentono all'IAM utente o al ruolo di accedere e aggiornare le risorse remote.

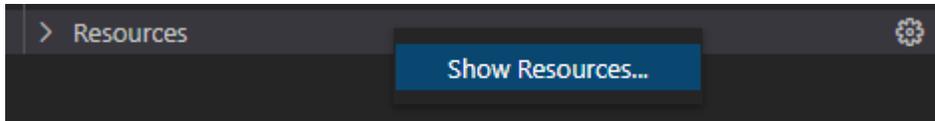
Il modo più semplice per concedere le autorizzazioni consiste nell'allegare la policy AWS gestita all'IAM identità che chiama queste API operazioni utilizzando l'interfaccia Toolkit. `PowerUserAccess` Questa [policy gestita](#) concede una serie di autorizzazioni per l'esecuzione di attività di sviluppo delle applicazioni, incluse le operazioni di chiamata API.

Per le autorizzazioni specifiche che definiscono le API operazioni consentite su risorse remote, consulta la Guida per l'utente di [AWS Cloud Control](#). API

Aggiunta di risorse e interazione con le risorse esistenti

1. In AWS Explorer, fai clic con il pulsante destro del mouse su Risorse e scegli Mostra risorse.

Un riquadro mostra un elenco di tipi di risorse disponibili per la selezione.



2. Nel riquadro di selezione, seleziona i tipi di risorse da aggiungere all'AWS Explorer e premi Invio o scegli OK per confermare.

I tipi di risorse selezionati sono elencati sotto Risorse.

Note

Se hai già aggiunto un tipo di risorsa all'Esploratore AWS e hai deselezionato la relativa casella di controllo, tale tipo non è più elencato sotto Risorse dopo aver scelto OK. Solo i tipi di risorse attualmente selezionati sono visibili nell'Esploratore AWS .

3. Per visualizzare le risorse già esistenti per un tipo di risorsa, espandi la voce per quel tipo.

Un elenco di risorse disponibili viene visualizzato sotto il tipo di risorse.

4. Per interagire con una risorsa specifica, fai clic con il pulsante destro del mouse sul suo nome e quindi seleziona una delle opzioni seguenti:
 - Copia l'identificatore della risorsa: copia l'identificatore della risorsa specifica negli appunti. Ad esempio, la risorsa `AWS Toolkit:DynamoDB::Table` può essere identificata utilizzando la proprietà `TableName`.
 - Anteprima: visualizza una versione di sola lettura del modello in formato elettronico che descrive la risorsaJSON.

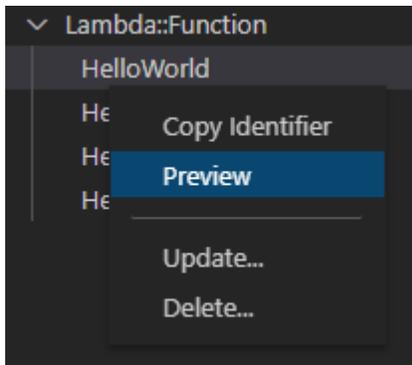
Una volta visualizzato il modello di risorsa, puoi modificarlo scegliendo l'icona Aggiorna a destra della scheda dell'editor. Per aggiornare una risorsa, è necessario che la risorsa richieda [???](#) sia abilitata.

 - Aggiornamento: modifica il modello JSON formattato per la risorsa in un editor VS Code. Per ulteriori informazioni, consulta [Creazione e modifica delle risorse](#).

- Elimina: elimina la risorsa confermando l'eliminazione in una finestra di dialogo visualizzata. (L'eliminazione delle risorse è attualmente disponibile [???](#) in questa versione di AWS Toolkit for Visual Studio Code.)

Warning

Se si elimina una risorsa, qualsiasi AWS CloudFormation stack che utilizza tale risorsa non verrà aggiornato. Per correggere questo errore di aggiornamento, è necessario ricreare la risorsa o rimuovere il riferimento ad essa nel modello dello stack. AWS CloudFormation Per ulteriori informazioni, consulta [questo articolo nel Knowledge Center](#).



Creazione e modifica delle risorse

Important

La creazione e l'aggiornamento delle risorse sono attualmente una funzionalità [???](#) in questa versione del AWS Toolkit for Visual Studio Code.

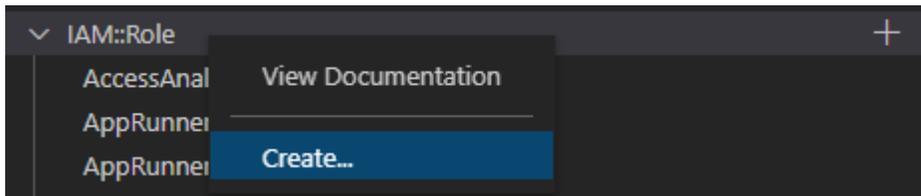
La creazione di una nuova risorsa comporta l'aggiunta di un tipo di risorsa all'elenco Risorse e quindi la modifica JSON di un modello in formato che definisce la risorsa, le sue proprietà e i suoi attributi.

Ad esempio, una risorsa che appartiene al tipo di `AWS Toolkit:SageMaker::UserProfile` risorsa viene definita con un modello che crea un profilo utente per Amazon SageMaker Studio. Il modello che definisce questa risorsa del profilo utente deve essere conforme allo schema del tipo di

risorsa per `AWS Toolkit:SageMaker::UserProfile`. Se il modello non è conforme allo schema, ad esempio a causa di proprietà mancanti o errate, la risorsa non può essere creata o aggiornata.

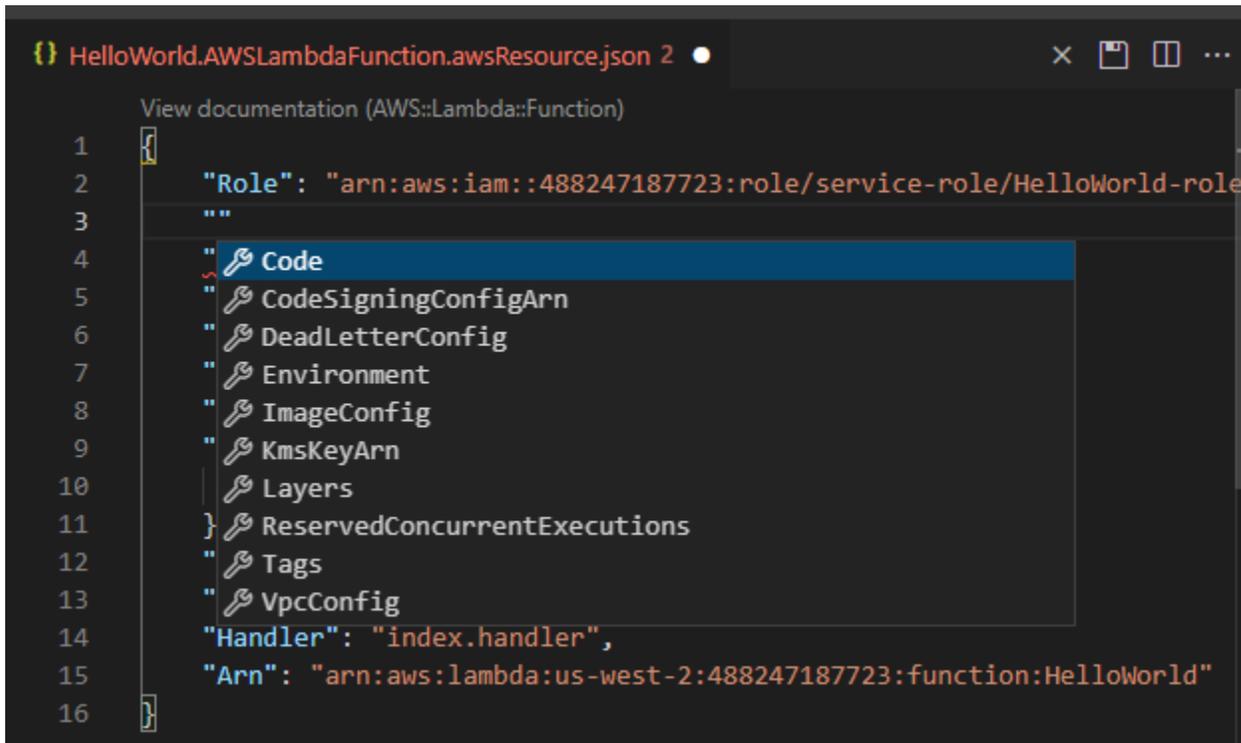
1. Aggiungi il tipo di risorsa per la risorsa che desideri creare facendo clic con il pulsante destro del mouse su Risorse e scegliendo Mostra risorse.
2. Dopo aver aggiunto il tipo di risorsa in Risorse, scegli l'icona più («+») per aprire il file modello in un nuovo editor.

In alternativa, puoi fare clic con il pulsante destro del mouse sul nome del tipo di risorsa e scegliere Crea. Puoi anche accedere alle informazioni su come modellare la risorsa scegliendo Visualizza documentazione.



3. Nell'editor, inizia a definire le proprietà che costituiscono il modello di risorsa. La funzionalità di completamento automatico suggerisce nomi di proprietà conformi allo schema del modello. Quando passi il mouse su un tipo di proprietà, un riquadro mostra una descrizione dello scopo per cui viene utilizzato. Per informazioni dettagliate sullo schema, scegli Visualizza documentazione.

Qualsiasi testo non conforme allo schema delle risorse è indicato da una sottolineatura rossa ondulata.



```
1 {
2   "Role": "arn:aws:iam::488247187723:role/service-role/HelloWorld-role",
3   ""
4   "Code"
5   "CodeSigningConfigArn"
6   "DeadLetterConfig"
7   "Environment"
8   "ImageConfig"
9   "KmsKeyArn"
10  "Layers"
11  "ReservedConcurrentExecutions"
12  "Tags"
13  "VpcConfig"
14  "Handler": "index.handler",
15  "Arn": "arn:aws:lambda:us-west-2:488247187723:function:HelloWorld"
16 }
```

4. Dopo aver dichiarato la risorsa, scegli l'icona Salva per convalidare il modello e salvare la risorsa nel Cloud remoto. AWS

Se il modello definisce la risorsa in base al relativo schema, viene visualizzato un messaggio per confermare che la risorsa è stata creata. Se la risorsa esiste già, il messaggio conferma che la risorsa è stata aggiornata.

Dopo la creazione, la risorsa viene aggiunta all'elenco sotto l'intestazione del tipo di risorse.

5. Se il file contiene errori, viene visualizzato un messaggio che spiega che la risorsa non può essere creata o aggiornata. Scegli Visualizza registri per identificare gli elementi del modello che devi correggere.

Sicurezza per AWS Toolkit for Visual Studio Code

Argomenti

- [Protezione dei dati in AWS Toolkit for Visual Studio Code](#)

Protezione dei dati in AWS Toolkit for Visual Studio Code

Il modello di [responsabilità AWS condivisa modello](#) di si applica alla protezione dei dati in AWS Toolkit for Visual Studio Code. Come descritto in questo modello, AWS è responsabile della protezione dell'infrastruttura globale che gestisce tutti i Cloud AWS. L'utente è responsabile del controllo dei contenuti ospitati su questa infrastruttura. L'utente è inoltre responsabile della configurazione della protezione e delle attività di gestione per i AWS servizi utilizzati. Per ulteriori informazioni sulla privacy dei dati, consulta la sezione [Privacy dei dati FAQ](#). Per informazioni sulla protezione dei dati in Europa, consulta il [Modello di responsabilitàAWS condivisa e GDPR](#) il post sul blog sulla AWS sicurezza.

Ai fini della protezione dei dati, ti consigliamo di proteggere Account AWS le credenziali e di configurare i singoli utenti con AWS IAM Identity Center o AWS Identity and Access Management (IAM). In tal modo, a ogni utente verranno assegnate solo le autorizzazioni necessarie per svolgere i suoi compiti. Ti suggeriamo, inoltre, di proteggere i dati nei seguenti modi:

- Utilizza l'autenticazione a più fattori (MFA) con ogni account.
- UsaSSL/TLSper comunicare con AWS le risorse. Richiediamo TLS 1.2 e consigliamo TLS 1.3.
- Configurazione API e registrazione delle attività degli utenti con AWS CloudTrail.
- Utilizza soluzioni di AWS crittografia, insieme a tutti i controlli di sicurezza predefiniti all'interno AWS servizi.
- Utilizza i servizi di sicurezza gestiti avanzati, come Amazon Macie, che aiutano a individuare e proteggere i dati sensibili archiviati in Amazon S3.
- Se hai bisogno di FIPS 140-3 moduli crittografici convalidati per accedere AWS tramite un'interfaccia a riga di comando o unAPI, usa un endpoint. FIPS Per ulteriori informazioni sugli FIPS endpoint disponibili, vedere [Federal Information Processing Standard \(\) 140-3. FIPS](#)

Ti consigliamo vivamente di non inserire mai informazioni riservate o sensibili, ad esempio gli indirizzi e-mail dei clienti, nei tag o nei campi di testo in formato libero, ad esempio nel campo Nome. Ciò

include quando lavori con AWS Toolkit for Visual Studio Code o AWS servizi altro utilizzando la console API, AWS CLI, AWS SDKs o. I dati inseriti nei tag o nei campi di testo in formato libero utilizzati per i nomi possono essere utilizzati per la fatturazione o i log di diagnostica. Se fornisci un URL a un server esterno, ti consigliamo vivamente di non includere le informazioni sulle credenziali URL per convalidare la tua richiesta a quel server.

Cronologia dei documenti per la Guida per AWS Toolkit for Visual Studio Code l'utente

La tabella che segue descrive le modifiche importanti apportate in ogni versione di AWS Toolkit for Visual Studio Code. Per ricevere notifiche sugli aggiornamenti di questa documentazione, puoi iscriverti a un [RSSfeed](#).

Modifica	Descrizione	Data
AWS Aggiornamenti di Identity and Access Management (IAM) Access Analyzer	Contenuto di IAM Access Analyzer aggiornato per includere nuovi API riferimenti.	10 luglio 2024
AWS Identity and Access Management (IAM) - Analizzate le ore degli accessi	È stato aggiunto un nuovo argomento della guida utente per IAM Access Analyzer.	23 maggio 2024
Connect al flusso di AWS autorizzazione aggiornato	Il flusso di autorizzazione è stato aggiornato per riflettere le modifiche al processo di autenticazione e la separazione di Amazon Q da AWS Toolkit for Visual Studio Code	30 aprile 2024
Estensione Amazon Q per VS Code	A partire dal 30 aprile 2024, CodeWhisperer fa ora parte di Amazon Q e Amazon Q è disponibile come estensione per VS Code.	30 aprile 2024
Supporto per il cloud privato virtuale negli ambienti di sviluppo	Contenuti aggiornati che coprono le modifiche all'interfaccia utente per il supporto VPC negli ambienti di sviluppo.	21 gennaio 2024

Composer dell'applicazione	È stato aggiunto un nuovo argomento Application Composer alla Guida per l'AWS Toolkit for Visual Studio Code utente.	28 novembre 2023
SSO supporto per CodeCatalyst	Contenuti aggiornati per coprire il supporto di IAM Identity Center CodeCatalyst e gli ambienti di sviluppo.	17 novembre 2023
Aggiunti link per il download di VS Code e Toolkit	Contenuto aggiornato con link per il download di VS Code e. AWS Toolkit for Visual Studio Code	1 novembre 2023
Argomento di Amazon Redshift	È stato aggiunto un nuovo argomento di Amazon Redshift alla Guida per l'AWS Toolkit for Visual Studio Code utente.	17 ottobre 2023
Connect al flusso di AWS autorizzazione aggiornato	Flusso di autorizzazione aggiornato per concentrarsi sui metodi di autenticazione specifici del servizio.	29 settembre 2023
Guida utente creata: crea un modello CloudFormation	Crea una nuova guida utente che descrive come creare un CloudFormation modello utilizzando il Toolkit for VS Code	17 dicembre 2021
Aggiornamento minore dell'interfaccia utente	È stato aggiornato il testo esistente da «Preview Machine State» a «Render graph» per adattarlo meglio all'interfaccia utente.	14 dicembre 2021

<u>Creata la Guida per l'utente per Amazon Elastic Container Service Exec</u>	Questa è una panoramica di Amazon ECS Exec.	13 dicembre 2021
<u>Guida utente creata per il servizio AWS IoT Toolkit for VS Code</u>	Questa guida per l'utente ha lo scopo di aiutarti a iniziare a utilizzare il AWS IoT servizio Toolkit for VS Code.	22 novembre 2021
<u>Supporto per funzionalità sperimentali</u>	È stato aggiunto il supporto per l'attivazione di funzionalità sperimentali per AWS i servizi.	14 ottobre 2021
<u>Support per AWS le risorse</u>	È stato aggiunto il supporto per accedere ai tipi di risorse e alle opzioni di interfaccia per creare, modificare e cancellare risorse.	14 ottobre 2021
<u>Panoramica del ECR servizio Amazon per AWS Toolkit for Visual Studio Code</u>	Aggiunta una panoramica e una procedura dettagliata per le caratteristiche e le funzioni del ECR servizio Amazon accessibili in VS Code	14 ottobre 2021
<u>Support per ARM64 ambienti</u>	È ora possibile eseguire applicazioni serverless in ambienti emulati ARM64 basati su base nonché in ambienti basati su x86_64.	1° ottobre 2021
<u>Applicazione Serverless AWS</u>	È stato aggiunto il supporto per l'esecuzione di applicazioni sulla piattaforma AWS SAM ARM64	30 settembre 2021

Sezione Node.js per l'aggiornamento del formato	In base al feedback dei clienti, formattazione aggiornata per TypeScript Node.js/.	12 agosto 2021
Supporto per App Runner	È AWS Toolkit for Visual Studio Code stato aggiunto il supporto per. AWS App Runner	11 agosto 2021
Funzioni di debug di Go	È stato aggiunto il supporto per il debug delle funzioni Go locali.	10 maggio 2021
Debug delle funzioni Java	È stato aggiunto il supporto per il debug delle funzioni Java locali.	22 Aprile 2021
YAMLsupporto per AWS Step Functions	È stato aggiunto YAML il supporto per AWS Step Functions.	4 marzo 2021
Eseguire il debug delle risorse di Amazon Gateway API	È stato aggiunto il supporto per il debug delle risorse locali di Amazon API Gateway.	1 dicembre 2020
Amazon API Gateway	È stato aggiunto il supporto per Amazon API Gateway.	1 dicembre 2020
Applicazione Serverless AWS	È stato aggiunto il supporto per le immagini dei container Lambda con applicazioni serverless.	1 dicembre 2020
AWS Systems Manager supporto	È stato aggiunto il supporto per i documenti di Systems Manager Automation.	30 settembre 2020
CloudWatch Registri	È stato aggiunto il supporto per i CloudWatch registri.	24 agosto 2020

Amazon S3	È stato aggiunto il supporto per Amazon S3.	30 luglio 2020
AWS Step Functions supporto	È stato aggiunto il supporto per AWS Step Functions.	31 marzo 2020
Contenuti di sicurezza	Aggiunti contenuti di sicurezza .	6 febbraio 2020
Lavorare con Amazon EventBridge Schemas	Aggiunto supporto per Amazon EventBridge Schemas	1 dicembre 2019
AWS CDK	Versione di anteprima del AWS CDK servizio.	25 novembre 2019
Utilizzo di un processo di creazione di credenziali esterno	Sono state aggiunte informazioni sull'utilizzo di un processo di credenziali esterno per ottenere AWS le credenziali.	25 settembre 2019
Utilizzo IntelliSense per i file di definizione delle attività	IntelliSense è stato aggiunto il supporto per lavorare con i file di Amazon ECS task Definition.	24 settembre 2019
Guida per l'utente per AWS Toolkit for Visual Studio Code	Rilasci per la disponibilità generale	11 luglio 2019
Guida per l'utente per AWS Toolkit for Visual Studio Code	Aggiornata struttura del documento per chiarezza e facilità d'uso.	3 luglio 2019
Installazione di AWS Toolkit for Visual Studio Code	Sono state aggiunte informazioni sull'installazione del linguaggio SDKs per supportar e varie toolchain.	12 giugno 2019

[Configura la tua toolchain](#)

Aggiunte informazioni sulla configurazione di varie toolchain.

12 giugno 2019

[Versione iniziale](#)

Versione iniziale della Guida per l'utente di AWS Toolkit for Visual Studio Code.

28 marzo 2019

Le traduzioni sono generate tramite traduzione automatica. In caso di conflitto tra il contenuto di una traduzione e la versione originale in Inglese, quest'ultima prevarrà.