[#hello-world-configure] == Step 2: Configure your {aws} environment

In this step, you configure the {aws} environment for your CDK stack. By doing this, you specify which environment your CDK stack will be deployed to.

First, determine the {aws} environment that you want to use. An {aws} environment consists of an {aws} account and {aws} Region.

When you use the {aws} CLI to configure security credentials on your local machine, you can then use the {aws} CLI to obtain {aws} environment information for a specific profile.

*To use the {aws} CLI to obtain your {aws} account ID*:: + . Run the following {aws} CLI command to get the {aws} account ID for your default profile: + [source,bash,subs=“verbatim,attributes”] — $ aws sts get-caller-identity –query “Account” –output text — + . If you prefer to use a named profile, provide the name of your profile using the --profile option: + [source,bash,subs=“verbatim,attributes”] — $ aws sts get-caller-identity –profile your-profile-name –query “Account” –output text —

*To use the {aws} CLI to obtain your {aws} Region*:: + . Run the following {aws} CLI command to get the Region that you configured for your default profile: + [source,bash,subs=“verbatim,attributes”] — $ aws configure get region — + . If you prefer to use a named profile, provide the name of your profile using the --profile option: + [source,bash,subs=“verbatim,attributes”] — $ aws configure get region –profile your-profile-name —

Next, you will configure the {aws} environment for your CDK stack by modifying the HelloCdkStack instance in your *application file*. For this tutorial, you will hard code your {aws} environment information. This is recommended for production environments. For information on other ways to configure environments, see xref:configure-env[Configure environments to use with the {aws} CDK].

*To configure the environment for your CDK stack*:: + . In your *application file*, use the env property of the Stack construct to configure your environment. The following is an example: + ==== [role=“tablist”] TypeScript::: Located in bin/hello-cdk.ts: + [source,javascript,subs=“verbatim,attributes”] — #!/usr/bin/env node import ‘source-map-support/register’; import \* as cdk from ‘aws-cdk-lib’; import { HelloCdkStack } from ‘../lib/hello-cdk-stack’;

const app = new cdk.App(); new HelloCdkStack(app, ‘HelloCdkStack’, { env: { account: ‘123456789012’, region: ‘us-east-1’ }, }); —

JavaScript::: Located in bin/hello-cdk.js: + [source,javascript,subs=“verbatim,attributes”] — #!/usr/bin/env node

const cdk = require(‘aws-cdk-lib’); const { HelloCdkStack } = require(‘../lib/hello-cdk-stack’);

const app = new cdk.App(); new HelloCdkStack(app, ‘HelloCdkStack’, { env: { account: ‘123456789012’, region: ‘us-east-1’ }, }); —

Python::: Located in app.py: + [source,python,subs=“verbatim,attributes”] — #!/usr/bin/env python3 import os

import aws\_cdk as cdk

from hello\_cdk.hello\_cdk\_stack import HelloCdkStack

app = cdk.App() HelloCdkStack(app, “HelloCdkStack”, env=cdk.Environment(account=‘123456789012’, region=‘us-east-1’), )

== app.synth()

Java::: Located in +src/main/java/.../HelloCdkApp.java+: + [source,java,subs=“verbatim,attributes”] — package com.myorg;

import software.amazon.awscdk.App; import software.amazon.awscdk.Environment; import software.amazon.awscdk.StackProps;

import java.util.Arrays;

public class HelloCdkApp { public static void main(final String[] args) { App app = new App();

…. new HelloCdkStack(app, “HelloCdkStack”, StackProps.builder() .env(Environment.builder() .account(“123456789012”) .region(“us-east-1”) .build())

.build());  
  
app.synth();

} } —- ….

C#::: Located in src/HelloCdk/Program.cs: + [source,csharp,subs=“verbatim,attributes”] — using Amazon.CDK; using System; using System.Collections.Generic; using System.Linq;

namespace HelloCdk { sealed class Program { public static void Main(string[] args) { var app = new App(); new HelloCdkStack(app, “HelloCdkStack”, new StackProps { Env = new Amazon.CDK.Environment { Account = “123456789012”, Region = “us-east-1”, } }); app.Synth(); } } } —

Go::: Located in hello-cdk.go: + [source,go,subs=“verbatim,attributes”] — package main

import ( “github.com/aws/aws-cdk-go/awscdk/v2” “github.com/aws/constructs-go/constructs/v10” “github.com/aws/jsii-runtime-go” )

// …

func main() { defer jsii.Close()

app := awscdk.NewApp(nil)

NewHelloCdkStack(app, “HelloCdkStack”, &HelloCdkStackProps{ awscdk.StackProps{ Env: env(), }, })

app.Synth(nil) }

func env() \*awscdk.Environment { return &awscdk.Environment{ Account: jsii.String(“123456789012”), Region: jsii.String(“us-east-1”), } } — ====