include::attributes.txt[]

// Attributes

[.topic] [#testing-locally-getting-started] = Getting started with locally testing :info\_titleabbrev: Getting started :keywords: CLI, {aws} CDK, {aws} SAM

== [abstract]

## Getting Started documentation that provides the prerequisites for using {aws} CDK with {aws} SAM, and tutorial for building, locally testing, and deploying a simple {aws} CDK application.

// Content start

This topic describes what you need to use the {aws} SAM CLI with {aws} CDK applications, and it provides instructions for building and locally testing a simple {aws} CDK application.

[#testing-locally-getting-started-prerequisites] == Prerequisites

To test locally, you must install the {aws} SAM CLI. see link:https://docs.aws.amazon.com/serverless-application-model/latest/developerguide/getting\_started.html[Install the {aws} SAM CLI] for installation instructions.

[#testing-locally-getting-started-tutorial] == Creating and locally testing an {aws} CDK application

== [abstract]

## Learn how to use {aws} SAM to download, build, and locally test a {aws} CDK application.

To locally test an {aws} CDK application using the {aws} SAM CLI, you must have an {aws} CDK application that contains a Lambda function. Use the following steps to create a basic {aws} CDK application with a Lambda function. For more information, see link:https://docs.aws.amazon.com/cdk/latest/guide/serverless\_example.html[Creating a serverless application using the {aws} CDK] in the *{aws} Cloud Development Kit ({aws} CDK) Developer Guide*.

[#testing-locally-getting-started-tutorial-init] *Step 1: Create an {aws} CDK application*:: + For this tutorial, initialize an {aws} CDK application that uses TypeScript. + Command to run: + [source,none,subs=“verbatim,attributes”] — $ mkdir cdk-sam-example $ cd cdk-sam-example $ cdk init app –language typescript —

[#testing-locally-getting-started-tutorial-lambda] *Step 2: Add a Lambda function to your application*:: + Replace the code in lib/cdk-sam-example-stack.ts with the following: + [source,typescript,subs=“verbatim,attributes”] — import { Stack, StackProps } from ‘aws-cdk-lib’; import { Construct } from ‘constructs’; import \* as lambda from ‘aws-cdk-lib/aws-lambda’;

export class CdkSamExampleStack extends Stack { constructor(scope: Construct, id: string, props?: StackProps) { super(scope, id, props);

new lambda.Function(this, ‘MyFunction’, { runtime: lambda.Runtime.PYTHON\_3\_12, handler: ‘app.lambda\_handler’, code: lambda.Code.fromAsset(‘./my\_function’), }); } } —-

[#testing-locally-getting-started-tutorial-code] *Step 3: Add your Lambda function code*:: + Create a directory named my\_function. In that directory, create a file named app.py. + Command to run: + ==== [role=“tablist”] macOS / Linux:: + [source,none,subs=“verbatim,attributes”] — $ mkdir my\_function $ cd my\_function $ touch app.py —

Windows:: + [source,none,subs=“verbatim,attributes”] — $ mkdir my\_function $ cd my\_function $ type nul > app.py —

PowerShell:: + [source,none,subs=“verbatim,attributes”] — $ mkdir my\_function $ cd my\_function $ New-Item -Path “app.py" --- + Add the following code toapp.py`:

== [source,none,subs=“verbatim,attributes”]

def lambda\_handler(event, context): return “Hello from SAM and the CDK!” — ====

[#testing-locally-getting-started-tutorial-function] *Step 4: Test your Lambda function*:: + You can use the {aws} SAM CLI to locally invoke a Lambda function that you define in an {aws} CDK application. To do this, you need the function construct identifier and the path to your synthesized {aws} CloudFormation template. + Run the following command to go back to the lib directory: + [source,none,subs=“verbatim,attributes”] — $ cd .. — + *Command to run:* + [source,none,subs=“verbatim,attributes”] — $ cdk synth –no-staging — + [source,none,subs=“verbatim,attributes”] — $ sam local invoke MyFunction –no-event -t ./cdk.out/CdkSamExampleStack.template.json — + *Example output:* + — Invoking app.lambda\_handler (python3.9)

START RequestId: 5434c093-7182-4012-9b06-635011cac4f2 Version: $LATEST “Hello from SAM and the CDK!” END RequestId: 5434c093-7182-4012-9b06-635011cac4f2 REPORT RequestId: 5434c093-7182-4012-9b06-635011cac4f2 Init Duration: 0.32 ms Duration: 177.47 ms Billed Duration: 178 ms Memory Size: 128 MB Max Memory Used: 128 MB —