[ENGLISH]

Hello, Cloud Gurus, and welcome to this lesson,

which is going to introduce the Cloud Development Kit,

or CDK.

And we'll begin with what is the CDK?

How does it work?

We'll take a look at some CDK terminology,

review some example code written in TypeScript,

understand the process of using the CDK, and my exam tips.

So what is the Cloud Development Kit, or CDK?

Well, it's an open-source development framework

which allows you to build applications,

define and deploy AWS resources

all using a programming language of your choice.

And the ones that are supported,

for instance, TypeScript, Python, Java, .NET, and Go.

How does it work?

Well, first of all, you model your application

using a supported programming language that you like.

For instance, TypeScript.

Then use the CDK toolkit to transform your code

into a CloudFormation template.

And CDK can then even provision your resources

using CloudFormation.

Now there's some useful terminology

that it is important to understand.

First of all, an app,

which is a container for one or more stacks.

A stack is a unit of deployment

similar to a CloudFormation stack.

And all the AWS resources

defined within the scope of a stack

are provisioned as a single unit.

A stack includes one or more constructs.

And a construct simply defines the AWS resources

that we are deploying.

And for instance, in this example,

we've got SQS and SNS resources,

and we define the app, the stack, and the constructs

using a programming language of our choice.

And the CDK toolkit, also known as the CDK CLI,

is used to create and deploy using the CDK.

So let's review some example code written in TypeScript,

and the app, the stack, and the constructs

are represented as classes or types

using the programming language of our choice.

And this first section defines the stack.

And then this second section defines two constructs.

So we've got the SNS queue called MyappQueue,

and the SNS topic called MyappTopic.

And the queue is subscribed to the topic.

Now you don't need to learn TypeScript

or any other programming language for the exam.

This is just an example

to show you that we can use a coding language of our choice

to define our AWS resources.

And this is great for people

who are already comfortable

with one of the supported programming languages.

So they can easily provision resources using CloudFormation

using a programming language that they like

without having to learn the CloudFormation syntax.

Now, after you've written your code,

the process looks like this.

First, you need to create a new CDK project,

so we run cdk init inside an empty directory.

Next, use npm run build to compile your application.

Then use cdk synth to create a CloudFormation template,

and it's synthesizing a new CloudFormation template

based on the code you've provided.

And then, finally, deploy your stack.

So you use cdk deploy

to deploy the stack using CloudFormation.

And for the exam,

remember that we can use the Cloud Development Kit

to deploy applications and AWS resources

using a programming language of our choice,

like TypeScript, Python, Java, .NET, or Go.

AWS resources are defined

using apps, stacks, and constructs.

And an app is a container for one or more stacks.

A stack is a unit of deployment.

And a construct defines the AWS resources themselves.

The Cloud Development Kit can synthesize

a CloudFormation template.

And to do that, we use the CDK CLI

to create the CloudFormation template from our code.

And we also use the CLI to deploy our stack.

And the process looks like this.

So first of all, we run CDK init inside an empty directory

to create our new CDK project.

Compile the application using,

compile the app using npm run build.

Synthesize or create our CloudFormation template

using CDK synth.

And use CDK deploy to deploy our stack using CloudFormation.

So that is it for this lesson.

Any questions, please let me know.

Otherwise, I'll see you in the next one.

Thank you.

[SPANISH]

Hola, Cloud Gurus, y bienvenidos a esta lección.

que va a presentar el kit de desarrollo en la nube,

o CDK.

Y comenzaremos con ¿Qué es el CDK?

¿Como funciona?

Echaremos un vistazo a la terminología de CDK,

revise un código de ejemplo escrito en TypeScript ,

comprender el proceso de uso del CDK y mis consejos para el examen.

Entonces, ¿qué es el kit de desarrollo en la nube o CDK?

Bueno, es un marco de desarrollo de código abierto.

que le permite construir aplicaciones,

definir e implementar recursos de AWS

todo usando un lenguaje de programación de su elección.

Y los que son compatibles,

por ejemplo, TypeScript , Python, Java, .NET y Go.

¿Como funciona?

Bueno, antes que nada, modelas tu aplicación

usando un lenguaje de programación compatible que te guste.

Por ejemplo, TypeScript .

Luego use el kit de herramientas CDK para transformar su código

en una plantilla de CloudFormation .

Y CDK puede incluso aprovisionar sus recursos

usando Formación de nubes .

Ahora hay algo de terminología útil

que es importante entender.

En primer lugar, una aplicación,

que es un contenedor para una o más pilas.

Una pila es una unidad de implementación.

similar a una pila de CloudFormation .

Y todos los recursos de AWS

definido dentro del alcance de una pila

se aprovisionan como una sola unidad.

Una pila incluye una o más construcciones.

Y una construcción simplemente define los recursos de AWS

que estamos desplegando.

Y por ejemplo, en este ejemplo,

tenemos recursos SQS y SNS,

y definimos la aplicación, la pila y las construcciones

utilizando un lenguaje de programación de nuestra elección.

Y el kit de herramientas de CDK, también conocido como CDK CLI,

se utiliza para crear e implementar utilizando el CDK.

Entonces, revisemos un código de ejemplo escrito en TypeScript ,

y la aplicación, la pila y las construcciones

se representan como clases o tipos

utilizando el lenguaje de programación de nuestra elección.

Y esta primera sección define la pila.

Y luego esta segunda sección define dos constructos.

Así que tenemos la cola de SNS llamada MyappQueue ,

y el tema de SNS llamado MyappTopic .

Y la cola está suscrita al tema.

Ahora no necesitas aprender TypeScript

o cualquier otro lenguaje de programación para el examen.

esto es solo un ejemplo

para mostrarle que podemos usar un lenguaje de codificación de nuestra elección

para definir nuestros recursos de AWS.

Y esto es genial para la gente.

que ya están cómodos

con uno de los lenguajes de programación soportados.

Para que puedan aprovisionar recursos fácilmente usando CloudFormation

usando un lenguaje de programación que les guste

sin tener que aprender la sintaxis de CloudFormation .

Ahora, después de haber escrito su código,

el proceso se parece a esto.

Primero, necesita crear un nuevo proyecto CDK,

entonces ejecutamos cdk init dentro de un directorio vacío.

Luego, use npm run build para compilar su aplicación.

Luego use cdk synth para crear una plantilla de CloudFormation ,

y está sintetizando una nueva plantilla de CloudFormation

basado en el código que ha proporcionado.

Y luego, finalmente, implemente su pila.

Entonces usas cdk deployment

para implementar la pila usando CloudFormation .

Y para el examen,

recuerda que podemos usar el Cloud Development Kit

para implementar aplicaciones y recursos de AWS

usando un lenguaje de programación de nuestra elección,

como TypeScript , Python, Java, .NET o Go.

Los recursos de AWS están definidos

usando aplicaciones, pilas y construcciones.

Y una aplicación es un contenedor para una o más pilas.

Una pila es una unidad de despliegue.

Y una construcción define los propios recursos de AWS.

El kit de desarrollo de la nube puede sintetizar

a Plantilla de formación en la nube .

Y para hacer eso, usamos CDK CLI

para crear la plantilla de CloudFormation a partir de nuestro código.

Y también usamos la CLI para implementar nuestra pila.

Y el proceso se ve así.

Entonces, antes que nada, ejecutamos CDK init dentro de un directorio vacío

para crear nuestro nuevo proyecto CDK.

Compile la aplicación usando,

compila la aplicación usando npm run build.

Sintetice o cree nuestra plantilla de CloudFormation

utilizando el sintetizador CDK.

Y use la implementación de CDK para implementar nuestra pila usando CloudFormation .

Eso es todo por esta lección.

Para cualquier duda, por favor hágamelo saber.

De lo contrario, te veré en la próxima.

Gracias.