[ENGLISH]

Hello, Cloud Gurus, and welcome to this lesson

where we'll be creating a code artifact repository.

And we'll start by creating a domain called my-domain.

We'll create a repository in the domain called my-repo,

and this is where the packages are gonna be stored.

We'll also create an upstream repository for my-repo,

and it's gonna be called npm-store.

Next, we'll add an external connection.

So we're going to connect npm-store

to the public npm registry.

Next, we'll associate npm-store

as an upstream repository to my-repo.

And then finally, we're going to use the npm CLI

to install an npm package called Express,

which is just a Node.js web application framework

used to develop mobile and web applications.

And this is going to make Express

and all its associated packages

available to users who are accessing the my-repo repository.

And everything that you need to complete this lesson

is included in the resources,

where I've added a text file

with all the commands that we're going to use.

So we'll be doing everything using the AWS CloudShell.

So after you've logged into the AWS console,

select the CloudShell.

And the first thing we're going to do is create our domain.

And here's the command that we use to do that.

And the domain we create is gonna be called my-domain.

So copy the command and paste it into your CloudShell.

There's the details of my-domain.

And in the next step,

we'll create a repository in our domain.

The first one we're gonna create is my-repo.

So copy that command, paste it into our CloudShell.

So it's creating a repository in my-domain called my-repo.

Next, we're gonna create the upstream repository

and we'll do that using this command.

And our upstream repository is gonna be called npm-store.

In the next step, we'll add our external connection.

So we're gonna associate an external connection in my-domain

to our repository called npm-store,

and it's gonna connect to the public npm registry.

After that, we'll associate the npm-store repository

as an upstream repository to the my-repo repository.

Next, we're going to configure the npm package manager

with our my-repo repository,

and it's going to fetch an authorization token

from CodeArtifact using my AWS credentials.

And if it's worked, it will look like this.

Next, we're going to use the npm CLI

to install an npm package,

and we're going to install the Express package.

And Express is just a Node.js web application framework,

and it's used to develop web and mobile apps.

And if it's all worked, this is what you should see.

So it should have added about 58 packages.

And then finally, we should be able to view the packages

that we just installed in our repository,

and this is the command we're gonna use.

So we're gonna list the packages that are in my-repo.

And it's actually added loads of different packages.

And there they all are.

And if you want to exit and get your command prompt back,

just use the letter Q.

Now, we've done all of this using the command line

just because it's easy,

but you don't need to learn these commands for the exam.

Just be aware of what CodeArtifact is used for

and the high level steps in linking an external repository.

And if I close the CloudShell

and search for CodeArtifact in the search box

and select that.

We can see our repository here and our npm-store.

So if I select my-repo,

here are all the packages that were installed

when we added Express.

And if we head back to repositories and select npm-store,

we can see that it's connected to a public repository.

And if you select Details,

here are the details of the connection.

So for the exam,

just remember that if you want to pull packages

from an external public registry

or external public repository,

then create an upstream repository

with an external connection to the public registry.

For instance, the npm registry for Node.js packages.

And then after that,

you'll be able to pull down packages

from the public registry

and make them available in your own repository.

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.

donde crearemos un repositorio de artefactos de código.

Y comenzaremos creando un dominio llamado mi-dominio.

Crearemos un repositorio en el dominio llamado my-repo,

y aquí es donde se almacenarán los paquetes .

También crearemos un repositorio ascendente para my-repo,

y se llamará npm -store .

A continuación, agregaremos una conexión externa.

Así que vamos a conectar npm -store

al registro público de npm .

A continuación, asociaremos npm -store

como repositorio ascendente de my-repo.

Y finalmente, vamos a usar la CLI de npm

para instalar un paquete npm llamado Express,

que es solo un marco de aplicación web Node.js

Se utiliza para desarrollar aplicaciones móviles y web.

Y esto va a hacer Express

y todos sus paquetes asociados

disponible para los usuarios que acceden al repositorio my-repo.

Y todo lo que necesitas para completar esta lección.

está incluido en los recursos,

donde he agregado un archivo de texto

con todos los comandos que vamos a utilizar.

Así que haremos todo usando AWS CloudShell .

Después de iniciar sesión en la consola de AWS,

seleccione CloudShell . \_

Y lo primero que vamos a hacer es crear nuestro dominio.

Y aquí está el comando que usamos para hacer eso.

Y el dominio que creamos se llamará mi-dominio.

Así que copie el comando y péguelo en su CloudShell .

Ahí están los detalles de mi dominio.

Y en el siguiente paso,

crearemos un repositorio en nuestro dominio.

El primero que vamos a crear es my-repo.

Así que copie ese comando, péguelo en nuestro CloudShell .

Entonces está creando un repositorio en my-domain llamado my-repo.

A continuación, vamos a crear el repositorio ascendente.

y lo haremos usando este comando.

Y nuestro repositorio ascendente se llamará npm -store .

En el siguiente paso, agregaremos nuestra conexión externa.

Así que vamos a asociar una conexión externa en mi dominio

a nuestro repositorio llamado npm -store,

y se conectará al registro público de npm .

Después de eso, asociaremos el repositorio npm -store

como un repositorio ascendente al repositorio my-repo.

A continuación, vamos a configurar el administrador de paquetes npm .

con nuestro repositorio my-repo,

y va a buscar un token de autorización

de CodeArtifact usando mis credenciales de AWS.

Y si ha funcionado, se verá así.

A continuación, vamos a utilizar la CLI de npm

para instalar un paquete npm ,

y vamos a instalar el paquete Express.

Y Express es solo un marco de aplicación web de Node.js,

y se utiliza para desarrollar aplicaciones web y móviles.

Y si todo funcionó, esto es lo que deberías ver.

Entonces debería haber agregado alrededor de 58 paquetes.

Y finalmente, deberíamos poder ver los paquetes.

que acabamos de instalar en nuestro repositorio,

y este es el comando que vamos a usar.

Así que vamos a enumerar los paquetes que están en my-repo.

Y en realidad ha agregado muchos paquetes diferentes.

Y ahí están todos.

Y si desea salir y recuperar su símbolo del sistema,

solo usa la letra Q.

Ahora, hemos hecho todo esto usando la línea de comando

solo porque es fácil,

pero no necesita aprender estos comandos para el examen.

Solo tenga en cuenta para qué se usa CodeArtifact

y los pasos de alto nivel para vincular un repositorio externo.

Y si cierro el CloudShell

y busque CodeArtifact en el cuadro de búsqueda

y seleccione eso.

Podemos ver nuestro repositorio aquí y nuestro npm -store.

Entonces, si selecciono my-repo,

aquí están todos los paquetes que se instalaron

cuando agregamos Express.

Y si volvemos a los repositorios y seleccionamos npm -store,

podemos ver que está conectado a un repositorio público.

Y si selecciona Detalles,

aquí están los detalles de la conexión.

Así que para el examen,

solo recuerda que si quieres sacar paquetes

de un registro público externo

o repositorio público externo,

luego crea un repositorio upstream

con conexión externa al registro público.

Por ejemplo, el registro npm para paquetes de Node.js.

Y luego después de eso,

podrás bajar paquetes

del registro publico

y póngalos a disposición en su propio repositorio.

Eso es todo por esta lección.

Para cualquier duda, por favor hágamelo saber.

De lo contrario, te veré en la próxima.

Gracias.