[ENGLISH]

Hello Cloud Gurus

and welcome to this lecture,

which is going to cover some of the advanced features

of API Gateway.

So the first thing we're going to cover

is that you can import your own APIs

into API Gateway.

And you can do this by importing

your own API definition files.

So you can use the API Gateway Import API feature

to import an API using a definition file.

And the protocol that they support for this

is called OpenAPI,

and it used to be known as Swagger,

so if you hear or see anything about Swagger,

it is now known as OpenAPI.

And Open API is basically an

industry standard description format,

which is used to describe RESTful APIs.

And you can use an OpenAPI definition file

to create a new API

or update an existing API.

And I'm just gonna quickly show you

how this looks in the AWS console.

And you don't need to do this yourself.

I just want to demonstrate

how it looks in the console.

So I'm gonna go to services,

scroll down to Network and Content Delivery,

and select API Gateway,

and then instead of going straight to build an API,

I'm going to select import API.

And this is where we can either

select a definition file from our local machine

or we can paste one in the editor.

And I've just come over to Swagger.io

to find a sample definition file.

So I'm just gonna copy this sample file

come back to API Gateway

and paste my sample definition in here.

And don't worry too much about this sample code,

I'm just using it to show you how easy and quick it is

to import your own OpenAPI files.

And then the next thing you do,

is just create API.

And then as long as you don't have

any syntax errors in your code,

it will go ahead and create an API

based on your OpenAPI definition.

Now I'm not gonna go through this

in an awful lot of detail,

because it is out of scope for the exam,

but the main thing you need to know,

is that if you have already got

an API definition that you want to use,

then you have the ability

to import it into API Gateway.

And it supports the OpenAPI protocol,

formerly known as Swagger.

Moving on to legacy protocols.

For example,

SOAP

or Simple Object Access Protocol,

which returns a response in XML format

instead of JSON.

And SOAP is a legacy protocol

which came out in the 1990s.

And of course the 1990s

was a great decade for music,

but it's maybe not such a great decade

for internet protocols.

And SOAP has long since been super ceded

by newer protocols like REST.

However, with API Gateway,

there are two options

for dealing with legacy applications

which still use SOAP.

So firstly,

you can configure API Gateway

as a web service pass through.

So that means that the response,

is simply passed through as XML,

back to the originating client.

And the finer details of how you do this

are actually out of scope for the exam.

But if you would like to know more

about how to do this,

then one of our awesome students

has actually written a blog post explaining it.

And I've linked to this in the resources section.

And the second option that you have for dealing with SOAP,

is that you can use API Gateway

to transform the XML response

into JSON format.

And once again,

knowing the details of how to do this

is out of scope for the exam.

You just need to know that it's possible

and it is described in more detail

in this AWS blog post

which I have linked to in the resources.

So on to my exam tips.

Just be aware that you can import APIs

using external definition files,

for example in OpenAPI

formerly known as Swagger.

And when dealing with legacy applications which use SOAP,

you can configure API Gateway

as a SOAP web service pass through,

and it will simply pass through the XML response

to the originating client,

or you can use API Gateway

to convert the XML response to JSON format.

So that's it for this lecture.

If you have any questions please let me know.

Otherwise I will see you in the next lecture.

Thank you.

[SPANISH]

Hola gurús de la nube

y bienvenidos a esta conferencia,

que cubrirá algunas de las características avanzadas

de API Gateway.

Así que lo primero que vamos a cubrir

es que puedes importar tus propias APIs

en API Gateway.

Y puedes hacer esto importando

sus propios archivos de definición de API.

Para que pueda usar la función API de importación de API Gateway

para importar una API utilizando un archivo de definición.

Y el protocolo que soportan para esto

se llama OpenAPI,

y solía ser conocido como Swagger,

así que si escuchas o ves algo sobre Swagger,

ahora se conoce como OpenAPI.

Y Open API es básicamente un

formato de descripción estándar de la industria,

que se utiliza para describir las API RESTful.

Y puede usar un archivo de definición de OpenAPI

para crear una nueva API

o actualizar una API existente.

Y te mostraré rápidamente

cómo se ve esto en la consola de AWS.

Y no es necesario que hagas esto tú mismo.

solo quiero demostrar

cómo se ve en la consola.

Así que voy a ir a los servicios,

desplácese hacia abajo hasta Entrega de red y contenido,

y seleccione API Gateway,

y luego, en lugar de ir directamente a construir una API,

Voy a seleccionar Importar API.

Y aquí es donde podemos

seleccione un archivo de definición de nuestra máquina local

o podemos pegar uno en el editor.

Y acabo de llegar a Swagger.io

para encontrar un archivo de definición de muestra.

Así que solo voy a copiar este archivo de muestra

volver a API Gateway

y pegue mi definición de muestra aquí.

Y no se preocupe demasiado por este código de muestra,

Solo lo estoy usando para mostrarte lo fácil y rápido que es.

para importar sus propios archivos OpenAPI.

Y luego lo siguiente que haces,

es solo crear API.

Y luego, mientras no tengas

cualquier error de sintaxis en su código,

seguirá adelante y creará una API

basado en su definición de OpenAPI.

Ahora no voy a pasar por esto

con un montón de detalles,

porque está fuera del alcance del examen,

pero lo principal que necesitas saber,

es que si ya tienes

una definición de API que desea utilizar,

entonces tienes la habilidad

para importarlo a API Gateway.

Y es compatible con el protocolo OpenAPI,

anteriormente conocido como Swagger.

Pasando a los protocolos heredados.

Por ejemplo,

JABÓN

o Protocolo simple de acceso a objetos,

que devuelve una respuesta en formato XML

en lugar de JSON.

Y SOAP es un protocolo heredado

que salió en la década de 1990.

Y por supuesto la década de 1990

fue una gran década para la música,

pero tal vez no sea una gran década

para protocolos de internet.

Y SOAP ha sido reemplazado hace mucho tiempo

por protocolos más nuevos como REST.

Sin embargo, con API Gateway,

hay dos opciones

para tratar con aplicaciones heredadas

que todavía usan SOAP.

Así que en primer lugar,

puede configurar API Gateway

como un paso de servicio web.

Eso significa que la respuesta,

simplemente se pasa como XML,

de vuelta al cliente de origen.

Y los detalles más finos de cómo haces esto

en realidad están fuera del alcance del examen.

Pero si quieres saber más

sobre cómo hacer esto,

entonces uno de nuestros increíbles estudiantes

en realidad ha escrito una publicación de blog explicándolo.

Y he vinculado a esto en la sección de recursos.

Y la segunda opción que tiene para tratar con SOAP,

es que puedes usar API Gateway

para transformar la respuesta XML

en formato JSON.

Una vez mas,

sabiendo los detalles de cómo hacer esto

está fuera del alcance del examen.

Solo necesitas saber que es posible

y se describe con más detalle

en esta publicación de blog de AWS

a la que me he vinculado en los recursos.

Así que en mis consejos de examen.

Solo tenga en cuenta que puede importar API

utilizando archivos de definición externos,

por ejemplo en OpenAPI

anteriormente conocido como Swagger.

Y cuando se trata de aplicaciones heredadas que usan SOAP,

puede configurar API Gateway

como un paso de servicio web SOAP,

y simplemente pasará a través de la respuesta XML

al cliente de origen,

o puede usar API Gateway

para convertir la respuesta XML al formato JSON.

Así que eso es todo por esta conferencia.

Si tiene alguna pregunta, por favor hágamelo saber.

De lo contrario, te veré en la próxima conferencia.

Gracias.