Hello Cloud Gurus,

and welcome to this lesson where we'll create an AMI

using EC2 Image Builder.

And we'll begin by creating an IAM role.

And we'll add the permissions for EC2 Image Builder.

And this is gonna allow Image Builder

to do everything that it needs to run the build

and test processes on EC2 instances.

Next, we'll create an image pipeline,

and this is where we're gonna define

all of our configuration settings.

Then we'll execute our pipeline,

and the output of our pipeline will be an AMI.

And notice the Image Builder launches temporary

build and test EC2 instances

to run the build and test workflows.

And then finally, once the AMI has been created,

we can view it in the Image Builder console,

and also from the EC2 console under AMIs.

So if you're ready to create an AMI using Image Builder,

I'll see you in the console.

So from the console, first of all, search for IAM.

Then select Roles.

And Create Role.

Make sure that your trusted entity type is an AWS service.

Scroll down to Use Case,

and select EC2 under common use cases.

Then hit Next.

And we need to search for a couple of managed policies.

Now the first one is the EC2 instance profile

for Image Builder.

So I'm just gonna search up here for Image Builder.

And this is the one that we are looking for,

EC2 Instance Profile for Image Builder.

So select that one.

And now I need to search

for the Amazon SSM Managed Instance Core Policy.

So clear your filter and then search for SSM.

And this is the one that we are looking for,

the Amazon SSM Managed Instance Core Policy.

And we need this one because Image Builder actually

utilizes SSM under the hood.

So once you've selected those two policies,

you can scroll down to the bottom and hit Next.

We'll give our role a name,

and I'm gonna call it My Image Builder Role.

Scroll down and you can check the permission policies

that we've added.

And then select Create Role.

So now we've created our role,

we are ready to create our image pipeline.

So from the search box, just search for Image Builder.

And there it is.

Then select Create Image Pipeline.

I'm gonna call it My Image Pipeline.

Scrolling down, I'm gonna deselect this option

to collect enhanced metadata.

And this option just uses the Systems Manager inventory

to collect information about the images,

and it requires some additional

systems manager configuration.

So I'm just gonna deselect it so that everything works.

Under Build Schedule,

you can create a schedule to automatically run the pipeline

on a schedule that you define,

but we're just gonna create a pipeline manually.

So select Manual.

And this just means

that the pipeline will run when we initiate it.

Once you've done that, go to the bottom and select Next.

For our configuration options,

we're gonna create a new recipe.

And just remember the image recipe is a document

that defines the components that are going to be applied

to our source image.

So what is going to be included in our source image?

Scrolling down, we can select our image type,

and it's gonna be the Amazon Machine Image,

but you can also use Image Builder

to create Docker images as well.

Down here, we'll provide our recipe name.

And we need to provide a version number as well,

using this format that they define.

So type 1.0.0,

and it must be this format, otherwise it's not gonna work.

Scrolling down to Base Image, this is where

we can select the source image that we'd like to use,

and we'll just stick with the defaults here.

So make sure that you've selected Managed Images.

We'll be using Amazon Linux 2,

so that's gonna be our operating system.

Scrolling down, the image origin is gonna be quick start,

Amazon managed.

Under the image name, we're going to select the X86 version,

so select that one.

And we'll use the latest available operating system version.

Down here, it says the EC2 Image Builder uses the AWS

Systems Manager agent as part of the image build process.

And it's installing it automatically

if it wasn't already included in the base image.

And you can select this box

if you would like Image Builder to remove

the SSM agent after the pipeline has been executed.

Scrolling down to Working Directory,

and this is a temporary working space

that Image Builder uses during the build and test workflows.

And by default, it just uses /tmp.

So we'll stick with that default.

Under Components,

this is where we can select the software or scripts

that define the custom configuration of our image.

And there are loads

of different build components that we can select,

but for our image,

I would like to add the latest security updates.

So if we search for Update.

And then scroll down until you find Update Linux,

and select that.

This is gonna update our operating system

with all of the latest security updates.

And there's also an update kernel as well.

So once you've selected Update Linux,

you can scroll down until you find test components.

And these are the optional tests that we would like

to use to verify our AMI once it's been built.

And there's a few different test components

that they provide for Amazon Linux,

but I just wanna do a simple boot test.

So I'm gonna search for Simple.

And there it is.

So this just executes a simple boot test to verify our AMI.

Once you've selected that, just scroll down.

Under the Storage section,

this is where you can add additional EBS volumes

for your AMI, and we'll just stick with the default.

Then hit Next.

Under Infrastructure Configuration,

this is where we can define the infrastructure

that Image Builder will use to run the build

and test processes.

And if you remember Image Builder launches EC2 instances

in your account to customize the images

and run the validation tests.

But if we stick with the defaults,

then Image Builder is actually gonna launch

quite large EC2 instances, which will not only take longer

to configure, they also cost you more money.

So let's create our own infrastructure configuration.

So select Create a New Infrastructure Configuration,

and this is gonna give us control

over the type of EC2 instances that it's gonna launch.

So first of all, we'll give it a name.

Under IAM Role, we're gonna select the role that we created

in the beginning.

Under AWS Infrastructure,

this is where we can select the instance type

that Image Builder is gonna use

for the build and test instances.

So I'm gonna search for Micro.

And select T3 Micro.

Down here, we can select an SNS topic

to receive notifications

but I'm just gonna skip that.

We can also define which VPC, subnet,

and security groups we want to use, but I'm just gonna go

with the defaults and launch these in the default VPC.

There's also some troubleshooting settings,

so you can select whether you want to terminate

your instance on failure

or keep it alive so that you can troubleshoot.

And some instance metadata settings as well,

but these are beyond the scope for the exam.

So for everything else, we're gonna stick with the defaults

and hit Next.

Under Distribution Settings, this is where you can define

which regions we're gonna create AMIs for,

and which regions will our AMI be distributed to.

And by default, the AMI will be distributed

to the current region, so US East 1.

So you should see under the region settings, US East 1.

At this point we can just hit Next.

Review all of our settings, and if you're happy with

all of your options, select Create Pipeline.

And there we go.

It should have successfully created everything.

So it's created my image pipeline, my recipe,

the infrastructure configuration,

and my distribution settings as well.

So I'll just close all of those messages down.

There's my pipeline.

I'm just gonna select the pipeline, select Actions,

and Run Pipeline.

And if you select View Details up here,

this is where you can see the status.

Now this is gonna take a lot longer

than you might be expecting.

We really just have to wait an eternity for it to finish.

Last time I used Image Builder, it took about 20 minutes

or so to complete the pipeline.

So just be prepared.

It's not quick.

But while this is running,

we can go in and check our EC2 instances.

So up here, I'm gonna search for EC2.

Open it up in a new tab.

And if we select Instances, you should see here

that it's already launched our build instance.

So this is the EC2 instance that Image Builder is using

to build our new image.

And then in a few minutes,

if everything's working correctly,

we should see a test image being created as well.

So it's gonna create another EC2 instance

that it's going to use to do

our simple boot test on our AMI.

So just be patient and in the next few minutes,

you should see that instance appearing.

But as I said,

it's gonna take about 20 minutes to complete.

So now is a great time to step away from the screen,

have a quick break, and come back to it in a few minutes.

After a few minutes, I'm just gonna refresh my screen.

And there is my test instance, which is running.

My build instance has now been terminated

because the build phase is over, we're into the test phase.

And if I select my Image Builder tab,

I'm gonna refresh the screen.

And we can see the status is that it's in the test phase.

So probably a few more minutes until that will be completed.

A few more minutes later, I'm gonna refresh my screen here.

And the status has now changed to distributing,

and it's been about 25 minutes

since I originally started the build pipeline.

So it's still not completed.

I'm just gonna refresh again.

And finally it's got a status of available.

So if everything has completed successfully,

it should show you a status of available,

and it's taken approximately 26 or 27 minutes.

So this is our output image, and here's the ARN.

So this is the output image,

and if we select Images on the left.

Here's our image name and the version.

And the ARN of our image.

But you can also view this AMI from the EC2 console.

So head over to the EC2 console.

On the left hand side, scroll down.

Then under Images, you'll find AMI.

And there is your AMI.

So that is everything that I wanted to show you

for EC2 Image Builder.

And for the exam, just remember that in the first step,

we provide a base operating system image.

For example, the Amazon Linux 2 AMI.

Next, we defined the software to install.

And that could be anything like .NET, Node.js, Python,

or the latest security updates, like we did in this lesson.

Or you could even add the latest kernel updates

or security settings, et cetera.

The third step is that EC2 Image Builder

will run tests on the new image,

for instance, to check that the image boots correctly.

And then the fourth step is to distribute the image

to the region of your choice.

And by default, it's gonna appear

in the region that you are operating in.

And for us, that was US East 1.

So that is it for this lesson.

Any questions, please let me know.

Otherwise, please join me in the next lesson.

Thank you.