Hello, Cloud Gurus, and welcome to this lesson.

Now the key to learning AWS

is getting as much hands-on

practical experience as possible.

And in this lesson,

we'll be getting our hands dirty

launching an EC2 instance.

And we'll configure it to run as a web server.

Now you can complete this exercise

using your own AWS Free Tier account.

Or, if you have access to Cloud Playground,

you can just select Cloud Playground

at the top of the screen.

Select AWS Sandbox,

and you will be provided with your own login

to an AWS account that we provide.

So select Open Sandbox,

I'm going to right-click,

and open it in a new incognito window.

Here's our username and password,

so copy the password.

Our username is cloud\_user,

paste in the password,

Sign In, and you are good to go.

But if you don't have access to our Sandbox,

then don't worry,

you can just do everything

within your own Free Tier account.

So from the console,

make sure that you're in the Northern Virginia region.

And this is the main region

where AWS released all their new services.

At the top, we can search for EC2,

so select EC2,

and select Launch Instance.

I'm going to name my instance myWebServer.

And that will just create a tag

with the name that I've provided.

And tags are user-defined key value pairs,

which get associated with your instance.

So they're a really good way

to organize your EC2 instance.

For instance, you might want a tag

to name your system

or describe which team it belongs to,

or which application it's running.

Scrolling down, you'll see that instances

are created using Amazon Machine Images, or AMIs.

And this is basically the operating system image.

So we've got Amazon Linux, we've got Windows,

Red Hat, various versions of Linux,

and even macOS as well.

And there are loads of different ones to choose from.

And if we select this dropdown,

it will show you all the different versions

of Amazon Linux 2.

And there's even some that have been optimized

for deep learning or machine learning workloads.

So these will come pre-configured

with some of the packages

and software that you might need

for machine learning already installed.

But in this course, we are mainly going to be using

the latest version of the Amazon Linux 2 AMI.

So just select that one.

Scroll down, this is where we can select

our chip architecture.

And we'll stick with x86.

Down here, we can select our instance type

based on the requirements of our application.

And this is just going to be a little web server.

So I'm going to select t3.micro.

But if you're working in your own AWS account,

you might want to stick with the free tier instance type.

So here's the specification of our instance type.

And it also tells you the hourly on demand rate as well.

Now it's asking us about key pairs.

So to connect to our instance using SSH,

we will need to create a public and private key pair.

And this is going to allow us

to log in to our instance.

So let's go ahead and create a new one.

So select Create New Key Pair.

We need to give it a name.

I'm just going to call it mynewkeypair.

Stick with the rest of the defaults and Create Key Pair.

And this is going to download

to our local machine

to our Downloads directory.

So this key pair consists of a public key

and a private key.

And it's used to control or restrict

who can log in to our server.

Anyone with the private key

is going to be able to log in.

And if you don't have a private key,

then you will not be able to log in.

Scrolling down to network settings.

Select Edit, and the network

that we install this instance in

is going to be our default VPC.

And don't worry if you don't know what a VPC is--

we're going to be covering that

in much more detail later on in the course.

But for now,

just think of it as your own

virtual data center in the cloud

that is private to you.

So make sure default VPC is selected.

Under subnet, just select one of the subnets

in your default VPC.

And I'm going to select the subnet in us-east-1a.

And we'll be covering subnets

in greater detail later on in the course as well.

Make sure that you auto-assign a public IP address.

So this is going to create the public IP address

that we can use to connect it to our web server.

So that's really, really important.

Next, it's asking us to configure a security group.

And you can think of a security group

as similar to a virtual firewall.

So this is what we use to allow

network traffic to connect into our instance.

And I'm going to change the name of my security group

so that I can identify it more easily later on.

Scrolling down, we can take a look

at the security group rules.

And by default, they give us SSH on port 22.

So this gives us remote login using SSH.

And if you haven't used SSH before,

it's really similar to RDP,

or Remote Desktop Protocol, on Windows.

And it's basically what we use to log in

to have a command line session on our Linux server.

Now in terms of this SSH rule,

we can leave it as set to anywhere,

or we can set it to a custom IP or range of IPs,

or I can set it to my own IP address as well.

And if this was your production system,

then you'd want to lock it down to a specific IP address

or IP address range.

So you might want to select My IP or a Custom IP.

But for now, let's just keep things simple, and

we're just going to keep that to anywhere.

Now in addition to SSH,

we also need to configure another rule,

this time for HTTP.

So select Add Security Group Rule,

the type is going to be HTTP using port 80,

and the source is going to be anywhere

because this is going to be a web server,

so it means that any IP address on the internet

needs to be able to connect it to our instance

on port 80, the HTTP port.

Scrolling down, we can configure storage.

And this is where we can add additional storage volumes,

or EBS volumes.

And by default, you get a single 8 gig volume.

And this is where the operating system

of your instance is going to be installed.

And the current recommendation

is to go with a general purpose,

gp2-type volume for most applications.

But then, if he had a very high performance application

which needed additional IOPS,

then you could select a provisioned IOPS volume instead.

Scrolling down to advanced details,

I'm just going to open that up.

And there's a lot of things under here,

which we'll be covering later on in the course.

And we don't actually need to change

any of these defaults in here for now.

But there's a few that I'd like to show you

that are interesting.

So scrolling down the screen,

we can select Termination Protection.

And this is what you can use

to protect against accidental termination

of your instance.

And by enabling this,

it means that if you want to terminate this instance,

then you will have to disable termination protection.

And this is great for protecting

a production system

from accidentally being terminated.

Moving down to monitoring,

all instances are monitored

by default using CloudWatch.

And by default, the basic monitoring

that they give you is free of charge.

And they collect the monitoring metrics

at 5-minute intervals.

And that's what you get for free.

But if you want to enable detailed monitoring,

then you can enable it here.

And then CloudWatch will collect the data

at 1-minute intervals instead,

but they do charge you extra for it.

So let's just stick to the default.

Scrolling down, placement groups

are used when you need to control the placement

of your instances.

So maybe you need to spread

them across the underlying hardware.

Or maybe you want to keep them

on the same underlying hardware.

Well, you can do that using placement groups.

EBS optimization provides dedicated I/O capacity

for your EBS volumes.

And this gives you the best performance

for your EBS volumes,

so it's really good

for I/O-intensive workloads, like databases.

And then down here, towards the bottom,

you will find a text box under user data.

And this is where you can add commands

that you want to run when the system first boots up.

So you can add the commands in here

as a bootstrap script.

And this is something that we're going to cover properly

later on in the course.

So now once you've selected all of your options,

you can select Launch Instance.

So our instance has successfully been initiated.

Here's our instance ID.

So if you click on the instance ID,

it's going to take you to the EC2 dashboard.

So here's our instance, it's running,

but it's still initializing.

And in a few moments,

it should be ready to use.

And you can just refresh this screen using this button.

So once your instance has passed its status checks,

just select your instance ID here,

and here is our public IP address.

So the next thing we're going to do

is connect to our instance

using this public IP address.

Now if you're working on a Mac,

you can follow along with me.

But if your local machine

that you're working on is a Windows machine,

then you'll have a slightly different process

for logging in.

So if you're working on a Windows machine,

then please stop this video now

and go to the very next video in the course,

which is going to show you

how to log in using a Windows machine

and using a tool that you can download, called PuTTY.

But if you're a Mac user like me,

then you can follow along and do exactly what I do.

So first of all, open up a terminal window,

and make sure you're in your Downloads directory.

I'm going to check that my private key

has successfully downloaded,

and there it is.

And the next thing we need to do

is set the permissions.

So run chmod400

and the name of your key pair.

And this is just going to set the permissions on this file.

And if you forget to do this,

it will not work.

And now we are ready to SSH on to our EC2 instance.

So just type ssh ec2-user@,

then paste in the public IP address of your instance,

then -i followed by the name of your key pair

and hit Enter.

Answer yes to this question.

And there we are,

we have logged into our EC2 instance.

And if it's worked,

you should see something like this.

So we've established an SSH connection

over port 22 to our EC2 instance.

So now, I'm just going to clear the screen,

elevate my privileges.

So type sudo su.

So now I'm effectively the root user,

and I've got full admin access on this instance.

So the next thing we're going to do

is turn our Linux server into a web server.

So first of all,

we'll upgrade our operating system

to all the latest packages.

And to do that, just type yum update -y and hit Enter.

And this just updates to all the latest packages.

And it will update our operating system kernel as well.

So it just means we're fully up to date

with any security features and kernel packages.

Once we're done,

we're ready to install the Apache web server.

So type yum install httpd -y and hit Enter.

And then to start Apache,

here's the command we use,

systemctl start httpd, and hit Enter.

And then we want our web server

to start automatically at boot time.

So there's another command

that we need to run for that,

systemctl enable httpd, and hit Enter.

And then there's one last thing

that we can do just to check

that everything started correctly.

Run systemctl status httpd.

And we can see that the process is active.

And it started running about a minute ago.

So the next thing we'll do

is we're going to create a little webpage.

So I'm going to change directory to /var/www/html, type ls,

and there's no web pages in here yet.

So this is just the default directory structure

that gets installed when you install httpd.

So let's just add our first webpage.

And I'm going to do this using the basic text editor

that they provide in the EC2 instance.

And it's called Nano.

So just type nano, space,

followed by the name of the file

that you're creating.

And the name of the file is going to be index.html.

Now if you've never used Nano before,

then it might be a little bit scary,

but it's actually pretty easy

once you get used to it.

And we're just going to type in the contents of our file.

So it's going to be just the most basic possible HTML file.

So we start off with &lt;html&gt;,

we'll start a body section,

then create a heading.

And then this is the text that's going to appear

on our website.

Then I'll end my heading and the body.

And this indicates the end of the HTML file.

And then to save the file,

just hit Ctrl and the letter X at the same time.

We answer yes to this question.

And the file to write is index.html,

so hit Enter, then type ls.

And there is our file.

So now, we should be able to check

if we can access our webpage.

So if you head back to your browser window,

I'm going to copy my public IP address

of my EC2 instance,

open a new browser tab,

paste in the IP address, and hit Enter.

And there we go.

And if it's all worked,

then this is what you should see.

And if you're not seeing this,

there's a few things that you should go back and check.

First of all, you can go back

and make sure that you created the index.html correctly,

make sure that the httpd service is running.

And also make sure that you configured

the security group port

to allow access for everyone on port 80,

which is the HTTP port.

So there we go.

We've created an EC2 instance,

we've installed the Apache web server,

and we've created a very simple website

that we can access over the internet

using the public IP address of this instance.

So that's the end of this lesson.

And if you created this in your own AWS account,

then do remember to go in

and terminate your instance

as soon as you're finished

to avoid any unnecessary charges.

And you can do that

by just heading to the EC2 dashboard,

selecting your instance,

and selecting Instance State,

and Terminate Instance.

So hopefully all of that worked for you.

If you have any questions,

please let me know.

Otherwise, feel free to move on to the next lecture.

Thank you.

Hola, Cloud Gurus, y bienvenidos a esta lección.

Ahora la clave para aprender AWS

se está poniendo tan práctico

experiencia práctica posible.

Y en esta lección,

nos estaremos ensuciando las manos

lanzamiento de una instancia EC2.

Y lo configuraremos para que se ejecute como un servidor web.

Ahora puedes completar este ejercicio.

utilizando su propia cuenta de capa gratuita de AWS.

O, si tiene acceso a Cloud Playground,

solo puede seleccionar Cloud Playground

en la parte superior de la pantalla.

Seleccione AWS Sandbox,

y se le proporcionará su propio inicio de sesión

a una cuenta de AWS que proporcionamos.

Así que seleccione Abrir Sandbox,

Voy a hacer clic derecho,

y ábrelo en una nueva ventana de incógnito.

Aquí está nuestro nombre de usuario y contraseña,

así que copia la contraseña.

Nuestro nombre de usuario es cloud\_user,

pega la contraseña,

Inicie sesión y estará listo para comenzar.

Pero si no tiene acceso a nuestro Sandbox,

entonces no te preocupes,

puedes hacer todo

dentro de su propia cuenta de nivel gratuito.

Así que desde la consola,

asegúrese de estar en la región del norte de Virginia.

Y esta es la región principal.

donde AWS lanzó todos sus nuevos servicios.

En la parte superior, podemos buscar EC2,

así que seleccione EC2,

y seleccione Iniciar instancia.

Voy a nombrar mi instancia myWebServer.

Y eso solo creará una etiqueta.

con el nombre que he proporcionado.

Y las etiquetas son pares de valores clave definidos por el usuario,

que se asocian con su instancia.

Así que son una muy buena manera

para organizar su instancia EC2.

Por ejemplo, es posible que desee una etiqueta

para nombrar su sistema

o describir a qué equipo pertenece,

o qué aplicación está ejecutando.

Si se desplaza hacia abajo, verá que las instancias

se crean utilizando imágenes de máquina de Amazon o AMI.

Y esta es básicamente la imagen del sistema operativo.

Así que tenemos Amazon Linux, tenemos Windows,

Red Hat, varias versiones de Linux,

e incluso macOS también.

Y hay un montón de diferentes para elegir.

Y si seleccionamos este desplegable,

te mostrará todas las diferentes versiones

de AmazonLinux 2.

E incluso hay algunos que han sido optimizados.

para cargas de trabajo de aprendizaje profundo o aprendizaje automático.

Entonces estos vendrán preconfigurados

con algunos de los paquetes

y software que pueda necesitar

para el aprendizaje automático ya instalado.

Pero en este curso, vamos a usar principalmente

la última versión de la AMI de Amazon Linux 2.

Así que solo selecciona ese.

Desplácese hacia abajo, aquí es donde podemos seleccionar

nuestra arquitectura de chips.

Y nos quedaremos con x86.

Aquí abajo, podemos seleccionar nuestro tipo de instancia

en función de los requisitos de nuestra aplicación.

Y esto solo va a ser un pequeño servidor web.

Así que voy a seleccionar t3.micro.

Pero si está trabajando en su propia cuenta de AWS,

es posible que desee seguir con el tipo de instancia de nivel gratuito.

Así que aquí está la especificación de nuestro tipo de instancia.

Y también le dice la tarifa por hora bajo demanda.

Ahora nos está preguntando sobre pares de claves.

Entonces, para conectarnos a nuestra instancia usando SSH,

necesitaremos crear un par de claves pública y privada.

Y esto nos va a permitir

para iniciar sesión en nuestra instancia.

Así que sigamos adelante y creemos uno nuevo.

Así que seleccione Crear nuevo par de claves.

Tenemos que darle un nombre.

Voy a llamarlo mi nuevo par de llaves.

Siga con el resto de los valores predeterminados y cree un par de claves.

Y esto se va a descargar

a nuestra máquina local

a nuestro directorio de Descargas.

Así que este par de claves consiste en una clave pública

y una clave privada.

Y se usa para controlar o restringir

quién puede iniciar sesión en nuestro servidor.

Cualquiera con la clave privada.

va a poder iniciar sesión.

Y si no tienes una clave privada,

entonces no podrá iniciar sesión.

Desplácese hacia abajo hasta la configuración de red.

Seleccione Editar y la red

que instalamos esta instancia en

será nuestra VPC predeterminada.

Y no se preocupe si no sabe qué es una VPC:

vamos a estar cubriendo eso

con mucho más detalle más adelante en el curso.

Pero por ahora,

solo piensa en ello como tuyo

centro de datos virtual en la nube

eso es privado para ti.

Así que asegúrese de que esté seleccionada la VPC predeterminada.

En subred, simplemente seleccione una de las subredes

en su VPC predeterminada.

Y voy a seleccionar la subred en us-east-1a.

Y estaremos cubriendo subredes

también con mayor detalle más adelante en el curso.

Asegúrese de asignar automáticamente una dirección IP pública.

Así que esto va a crear la dirección IP pública

que podemos utilizar para conectarlo a nuestro servidor web.

Así que eso es muy, muy importante.

A continuación, nos pide que configuremos un grupo de seguridad.

Y puedes pensar en un grupo de seguridad

como similar a un cortafuegos virtual.

Así que esto es lo que usamos para permitir

tráfico de red para conectarse a nuestra instancia.

Y voy a cambiar el nombre de mi grupo de seguridad

para poder identificarlo más fácilmente más adelante.

Desplazándonos hacia abajo, podemos echar un vistazo

en las reglas del grupo de seguridad.

Y por defecto nos dan SSH en el puerto 22.

Entonces esto nos da un inicio de sesión remoto usando SSH.

Y si no ha usado SSH antes,

es muy similar a RDP,

o Protocolo de escritorio remoto, en Windows.

Y es básicamente lo que usamos para iniciar sesión

tener una sesión de línea de comando en nuestro servidor Linux.

Ahora, en términos de esta regla SSH,

podemos dejarlo como se establece en cualquier lugar,

o podemos establecerlo en una IP personalizada o rango de IP,

o también puedo configurarlo en mi propia dirección IP.

Y si este fuera tu sistema de producción,

entonces querrá bloquearlo en una dirección IP específica

o rango de direcciones IP.

Por lo tanto, es posible que desee seleccionar Mi IP o una IP personalizada.

Pero por ahora, mantengamos las cosas simples y

solo vamos a mantener eso en cualquier lugar.

Ahora, además de SSH,

también necesitamos configurar otra regla,

esta vez para HTTP.

Así que seleccione Agregar regla de grupo de seguridad,

el tipo va a ser HTTP usando el puerto 80,

y la fuente va a estar en cualquier parte

porque esto va a ser un servidor web,

por lo que significa que cualquier dirección IP en Internet

necesita poder conectarlo a nuestra instancia

en el puerto 80, el puerto HTTP.

Desplazándonos hacia abajo, podemos configurar el almacenamiento.

Y aquí es donde podemos agregar volúmenes de almacenamiento adicionales,

o volúmenes de EBS.

Y por defecto, obtienes un solo volumen de 8 gigas.

Y aquí es donde el sistema operativo

de su instancia se va a instalar.

Y la recomendación actual

es ir con un propósito general,

Volumen tipo gp2 para la mayoría de las aplicaciones.

Pero entonces, si tuviera una aplicación de muy alto rendimiento

que necesitaba IOPS adicionales,

entonces podría seleccionar un volumen de IOPS aprovisionado en su lugar.

Desplazándose hacia abajo a los detalles avanzados,

Solo voy a abrir eso.

Y hay muchas cosas aquí debajo,

que cubriremos más adelante en el curso.

Y en realidad no necesitamos cambiar

cualquiera de estos valores predeterminados aquí por ahora.

Pero hay algunos que me gustaría mostrarte

que son interesantes

Entonces, desplazándose hacia abajo en la pantalla,

podemos seleccionar Protección de Terminación.

Y esto es lo que puedes usar

para proteger contra la terminación accidental

de tu instancia.

Y al habilitar esto,

significa que si desea terminar esta instancia,

entonces tendrá que deshabilitar la protección de terminación.

Y esto es genial para proteger

un sistema de producción

de ser rescindido accidentalmente.

Bajando al monitoreo,

todas las instancias son monitoreadas

por defecto usando CloudWatch.

Y por defecto, el monitoreo básico

que te dan es gratis.

Y recopilan las métricas de seguimiento.

a intervalos de 5 minutos.

Y eso es lo que obtienes gratis.

Pero si desea habilitar la supervisión detallada,

entonces puedes habilitarlo aquí.

Y luego CloudWatch recopilará los datos.

en intervalos de 1 minuto en su lugar,

pero te cobran extra por ello.

Así que sigamos con el valor predeterminado.

Desplazamiento hacia abajo, grupos de ubicación

se utilizan cuando es necesario controlar la colocación

de tus instancias.

Así que tal vez necesites difundir

a través del hardware subyacente.

O tal vez quieras conservarlos

en el mismo hardware subyacente.

Bueno, puede hacerlo usando grupos de ubicación.

La optimización de EBS proporciona capacidad de E/S dedicada

para sus volúmenes de EBS.

Y esto te da el mejor rendimiento.

para sus volúmenes de EBS,

así que es muy bueno

para cargas de trabajo intensivas de E/S, como bases de datos.

Y luego aquí abajo, hacia el fondo,

encontrará un cuadro de texto debajo de los datos del usuario.

Y aquí es donde puedes agregar comandos.

que desea ejecutar cuando el sistema se inicia por primera vez.

Entonces puedes agregar los comandos aquí

como un script de arranque.

Y esto es algo que vamos a cubrir adecuadamente

más adelante en el curso.

Así que ahora, una vez que haya seleccionado todas sus opciones,

puede seleccionar Iniciar instancia.

Así que nuestra instancia se ha iniciado con éxito.

Aquí está nuestra ID de instancia.

Entonces, si hace clic en la ID de la instancia,

lo llevará al tablero de EC2.

Así que aquí está nuestra instancia, se está ejecutando,

pero todavía se está inicializando.

Y en unos momentos,

debe estar listo para usar.

Y puede actualizar esta pantalla usando este botón.

Entonces, una vez que su instancia haya pasado sus controles de estado,

simplemente seleccione su ID de instancia aquí,

y aquí está nuestra dirección IP pública.

Así que lo siguiente que haremos

es conectarse a nuestra instancia

utilizando esta dirección IP pública.

Ahora bien, si está trabajando en una Mac,

Puedes seguirme.

Pero si su máquina local

en la que está trabajando es una máquina con Windows,

entonces tendrás un proceso ligeramente diferente

para iniciar sesión.

Entonces, si está trabajando en una máquina con Windows,

entonces por favor detén este video ahora

e ir al siguiente video del curso,

que te va a mostrar

cómo iniciar sesión usando una máquina con Windows

y usando una herramienta que puedes descargar, llamada PuTTY.

Pero si eres un usuario de Mac como yo,

entonces puedes seguirme y hacer exactamente lo que yo hago.

Entonces, antes que nada, abra una ventana de terminal,

y asegúrese de estar en su directorio de Descargas.

Voy a comprobar que mi clave privada

se ha descargado con éxito,

y ahí está

Y lo siguiente que tenemos que hacer

se establecen los permisos.

Así que ejecuta chmod400

y el nombre de su par de claves.

Y esto solo va a establecer los permisos en este archivo.

Y si te olvidas de hacer esto,

no funcionará.

Y ahora estamos listos para usar SSH en nuestra instancia EC2.

Simplemente escriba ssh ec2-user@,

luego pegue la dirección IP pública de su instancia,

luego -i seguido del nombre de su par de claves

y presione Entrar.

Responda sí a esta pregunta.

Y ahí estamos,

hemos iniciado sesión en nuestra instancia EC2.

Y si ha funcionado,

Debería ver algo como esto.

Así que hemos establecido una conexión SSH

a través del puerto 22 a nuestra instancia EC2.

Así que ahora, solo voy a limpiar la pantalla,

elevar mis privilegios.

Así que escribe sudo su.

Así que ahora soy efectivamente el usuario raíz,

y tengo acceso completo de administrador en esta instancia.

Así que lo siguiente que haremos

es convertir nuestro servidor Linux en un servidor web.

Entonces, en primer lugar,

actualizaremos nuestro sistema operativo

a todos los últimos paquetes.

Y para hacer eso, solo escriba yum update -y y presione Enter.

Y esto solo se actualiza a todos los paquetes más recientes.

Y también actualizará el kernel de nuestro sistema operativo.

Entonces solo significa que estamos completamente actualizados

con cualquier característica de seguridad y paquetes de kernel.

Una vez que hayamos terminado,

estamos listos para instalar el servidor web Apache.

Así que escribe yum install httpd -y y presiona Enter.

Y luego para iniciar Apache,

aquí está el comando que usamos,

systemctl inicia httpd y presiona Enter.

Y luego queremos nuestro servidor web

para iniciarse automáticamente en el momento del arranque.

Así que hay otro comando

que tenemos que correr para eso,

systemctl habilite httpd y presione Enter.

Y luego hay una última cosa

que podemos hacer solo para comprobar

que todo empezó correctamente.

Ejecute systemctl status httpd.

Y podemos ver que el proceso está activo.

Y empezó a funcionar hace un minuto.

Así que lo siguiente que haremos

es que vamos a crear una pequeña página web.

Voy a cambiar el directorio a /var/www/html, escribir ls,

y no hay páginas web aquí todavía.

Así que esta es solo la estructura de directorios predeterminada

que se instala cuando instalas httpd.

Así que agreguemos nuestra primera página web.

Y voy a hacer esto usando el editor de texto básico.

que proporcionan en la instancia EC2.

Y se llama Nano.

Así que solo escribe nano, espacio,

seguido del nombre del archivo

que estás creando.

Y el nombre del archivo será index.html.

Ahora, si nunca has usado Nano antes,

entonces podría dar un poco de miedo,

pero en realidad es bastante fácil

una vez que te acostumbras.

Y solo vamos a escribir el contenido de nuestro archivo.

Así que será el archivo HTML más básico posible.

Así que comenzamos con &lt;html&gt;,

vamos a empezar una sección del cuerpo,

luego crea un encabezado.

Y luego este es el texto que va a aparecer.

en nuestra página web.

Luego terminaré mi encabezado y el cuerpo.

Y esto indica el final del archivo HTML.

Y luego para guardar el archivo,

simplemente presione Ctrl y la letra X al mismo tiempo.

Respondemos afirmativamente a esta pregunta.

Y el archivo a escribir es index.html,

así que presiona Enter, luego escribe ls.

Y ahí está nuestro archivo.

Así que ahora, deberíamos poder comprobar

si podemos acceder a nuestra página web.

Entonces, si regresa a la ventana de su navegador,

Voy a copiar mi dirección IP pública

de mi instancia EC2,

abrir una nueva pestaña del navegador,

pegue la dirección IP y presione Entrar.

Y ahí vamos.

Y si todo ha funcionado,

entonces esto es lo que deberías ver.

Y si no estás viendo esto,

hay algunas cosas que debe volver atrás y comprobar.

En primer lugar, puedes volver

y asegúrese de haber creado el index.html correctamente,

asegúrese de que el servicio httpd se esté ejecutando.

Y también asegúrese de haber configurado

el puerto del grupo de seguridad

para permitir el acceso de todos en el puerto 80,

que es el puerto HTTP.

Así que ahí vamos.

Hemos creado una instancia EC2,

hemos instalado el servidor web Apache,

y hemos creado un sitio web muy simple

que podemos acceder a través de internet

utilizando la dirección IP pública de esta instancia.

Así que ese es el final de esta lección.

Y si creó esto en su propia cuenta de AWS,

entonces recuerda entrar

y termina tu instancia

tan pronto como termines

para evitar cargos innecesarios.

Y puedes hacer eso

con solo dirigirse al tablero de EC2,

seleccionando su instancia,

y seleccionando Estado de instancia,

y Terminar instancia.

Así que espero que todo eso haya funcionado para ti.

Si tienes alguna pregunta,

por favor hágamelo saber.

De lo contrario, siéntase libre de pasar a la siguiente lección.

Gracias.