

Lab D

Create an EC2 instance running nginx to create a web server.

1. Open the Learner Lab

Choose the 'Learner Lab' course, **not** 'Cloud Foundations Lab'

In the modules click 'Launch AWS Academy Learner Lab'

Click Start Lab – then click AWS link when ready (green)

From AWS console. Change to the region US West Oregon.

2. Create the instance

Open EC2 service.

Launch Instance – Give it a name

Name and tags [Info](#)

Name

Amazon Linux Web Server

Select the Amazon Linux



Leave architecture at 64-bit (x86) and the instance type of t2.micro.

Click the ‘Create new key pair’ link. Even if you already have some, we want to create one that is unique for this server.

▼ Key pair (login) [Info](#)

You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

Key pair name - required

Select ▼ [Create new key pair](#)

Give it a name, leave the key pair type as RSA

Everyone use the .pem OpenSSH (For those with windows less than version 10 you would need to use PuTTY—this is how you do SSH on older windows 7 and 8. See step 3 later.)

Click create key pair and use the key pair name shown

Create key pair X

Key pair name

Key pairs allow you to connect to your instance securely.

amazon_linux_web_kp

The name can include up to 255 ASCII characters. It can't include leading or trailing spaces.

Key pair type

RSA
RSA encrypted private and public key pair

ED25519
ED25519 encrypted private and public key pair

Private key file format

.pem
For use with OpenSSH

.ppk
For use with PuTTY

⚠ When prompted, store the private key in a secure and accessible location on your computer. You will need it later to connect to your instance. [Learn more](#) ↗

A file will download with your key. Open that up in file management. You will want to place it somewhere other than ‘Downloads’. So, move it where you want it. You will navigate to this location later in command line interface.

In the ‘Network settings’ make sure it’s on ‘Create security group’ and click Edit in the upper right of that Network settings section. Give ‘Security group name’ a name (replace what was there if anything). Give it a description.

Firewall (security groups) [Info](#)

A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to your instance.

[Create security group](#) [Select existing security group](#)

Security group name - required

This security group will be added to all network interfaces. The name can't be edited after the security group is created. It must be 255 characters. Valid characters: a-z, A-Z, 0-9, spaces, and ._-:/()#,@[]+=;&!\$*

Description - required [Info](#)

‘Type’ should be ssh and ‘Source type’ is Anywhere. Can give it a description of SSH and leave all other defaults.

Inbound Security Group Rules

▼ Security group rule 1 (TCP, 22, 0.0.0.0/0, SSH)

Type [Info](#) **Protocol [Info](#)** **Port range [Info](#)**

Source type [Info](#) **Source [Info](#)** **Description - optional [Info](#)**

Add another security group rule. This one is HTTP

▼ Security group rule 2 (TCP, 80, 0.0.0.0/0, HTTP)

Type [Info](#) **Protocol [Info](#)** **Port range [Info](#)**

Source type [Info](#) **Source [Info](#)** **Description - optional [Info](#)**

Scroll down to the next section and leave the storage default of 8GB.

Click the expand triangle next to ‘Advanced details’ and enable ‘Termination protection’. This keeps it from being accidentally terminated.



Click Launch instance.

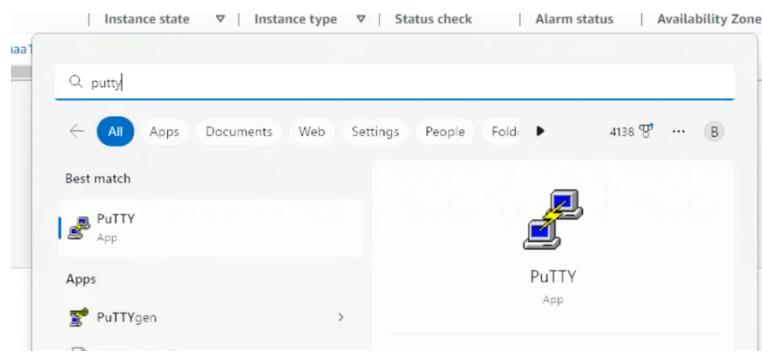
Wait for the Instance state to go from ‘Pending’ to ‘Running’.

Open the instance we just created. It might be pending in the ‘Instance state’ and take a little while for it to get started. Click the check mark next to your instance.

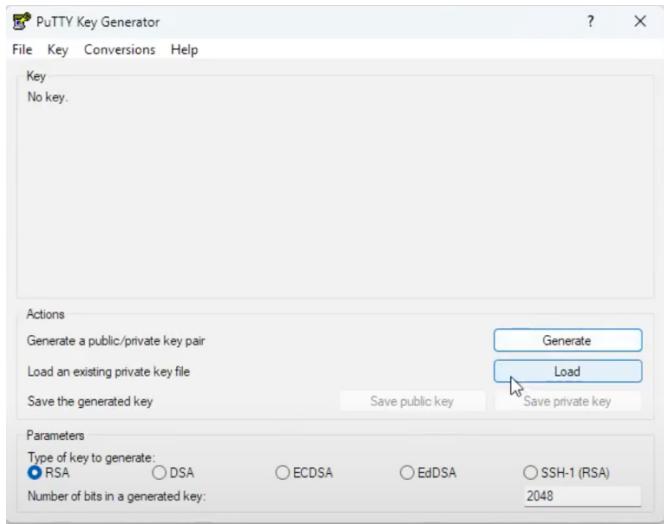
Under the ‘Details’ tab notice the Public IPv4 address available to us we will be viewing it later.

3. Windows Users of version 7 and 8 only – prepare your key file (*skip to step 4 if not using an older version of Windows*)

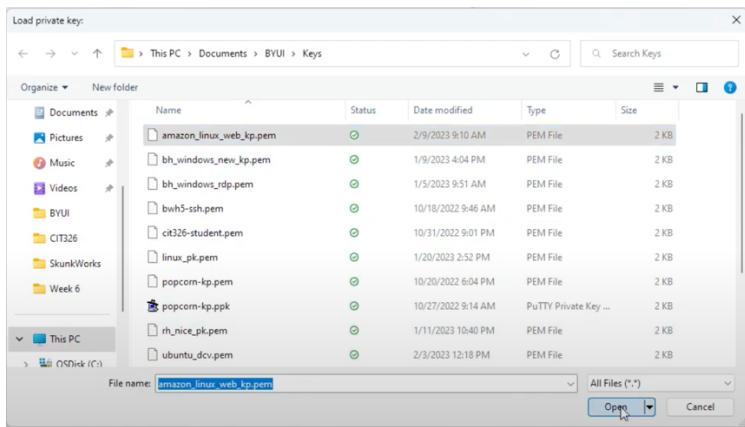
Windows users open up PuTTY (download it if you don’t have it). Windows user will use the PuTTYgen which is the key generator.



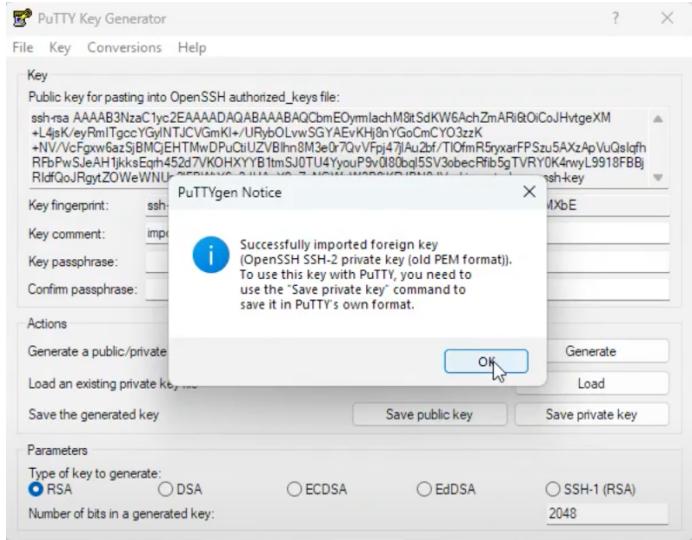
Run the PuTTYgen that comes up. It opens the PuTTY Key Generator



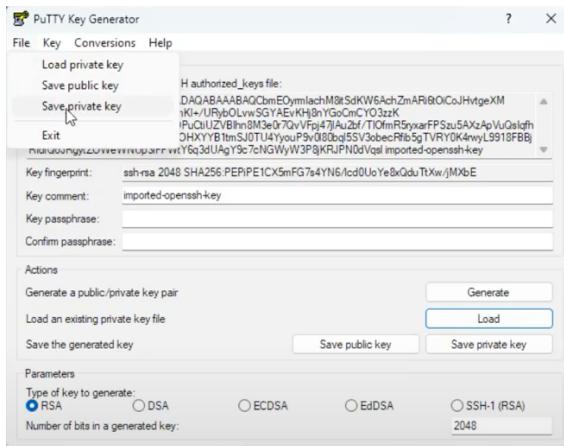
Click ‘Load’ and go to where you stored your .pem key file earlier. If it doesn’t show show All Files. Select that file and open.



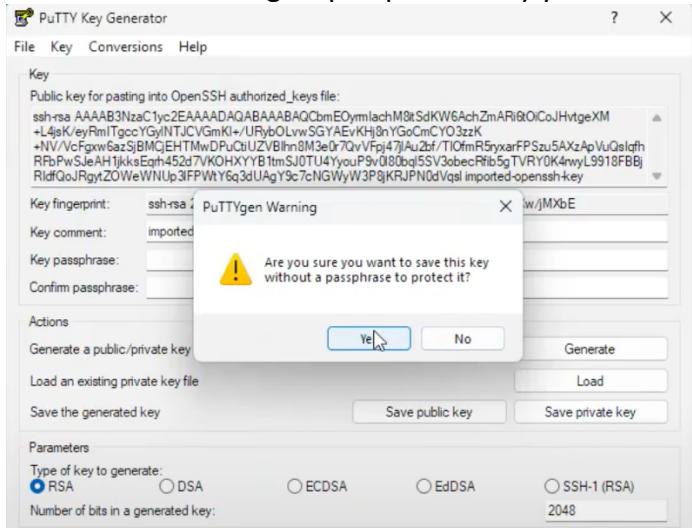
It imports that in by clicking OK



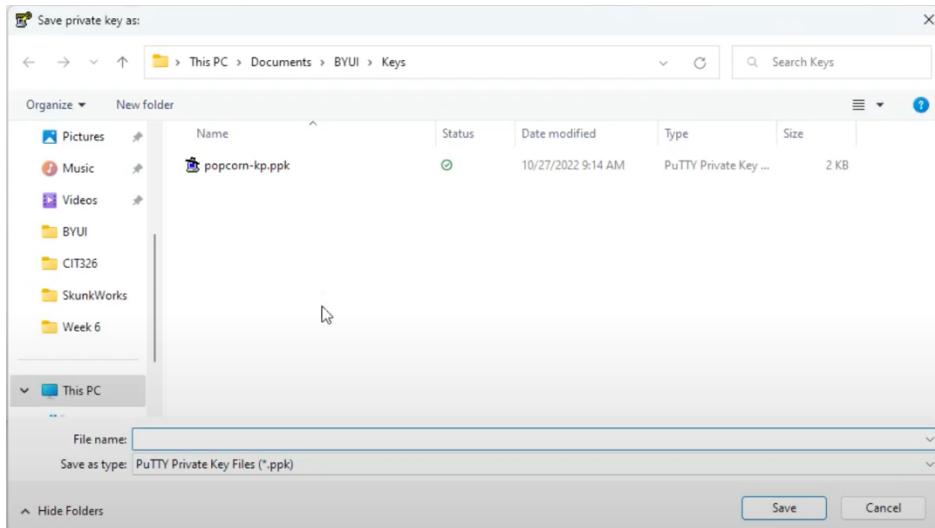
Under the file menu Save a private key.



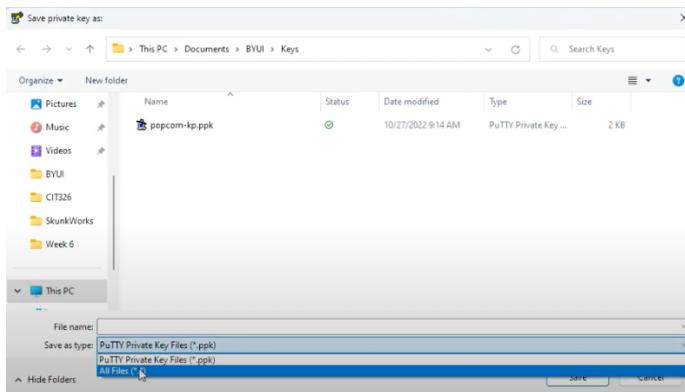
You don't need to sign a passphrase. Say yes



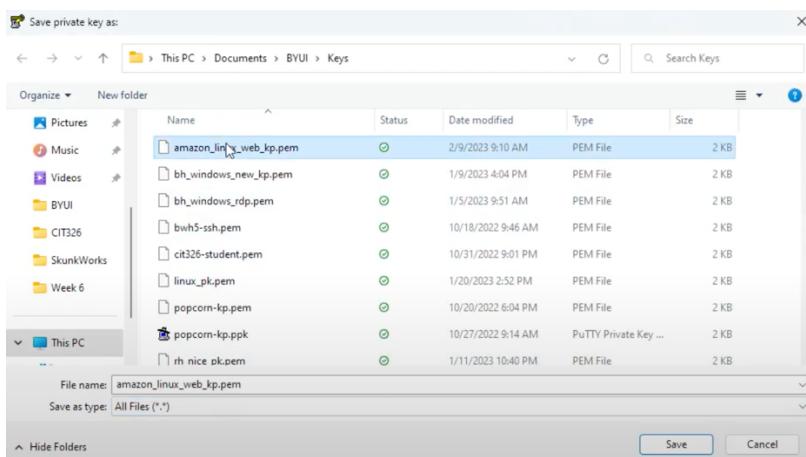
It will open up the Save dialog window and the key will be in putty form.



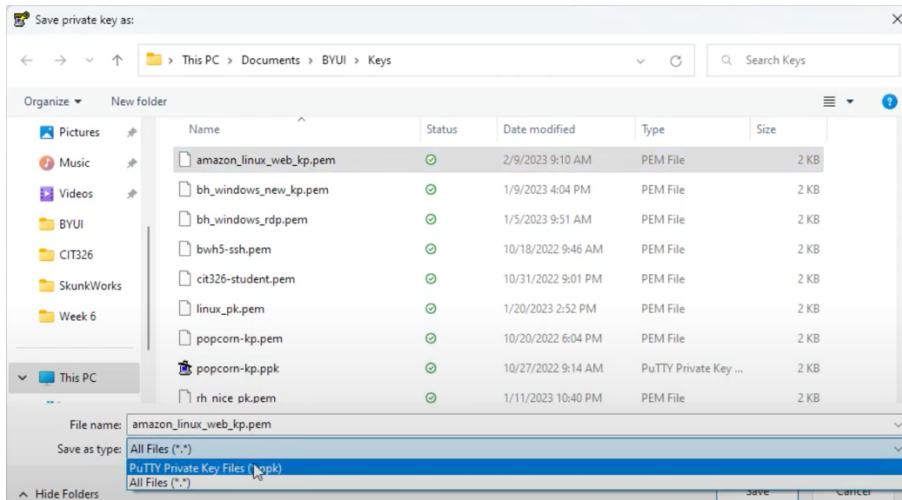
You can select all files again.



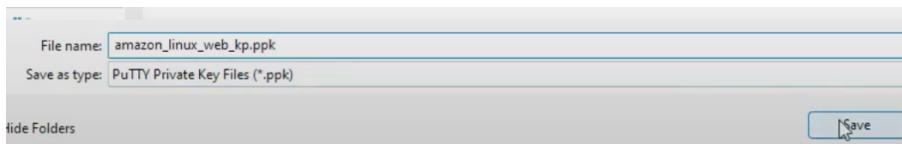
and select the .pem file again.



Change the Save as type to 'PuTTY PrivateKeyFiles(*.ppk)



Take off the .pem of the File name so it's a .ppk file and click Save



So now where you've saved your keys you should have the open ssh format and the putty format.



4. Connect to your instance.

Go back to your Instance and check the box next to the instance. Click 'Connect'.

Instances (1/1) Info		C	Connect	Instance state ▾	Actions ▾
<input type="text"/> Find Instance by attribute or tag (case-sensitive)					
<input checked="" type="checkbox"/>	Name ✍	Instance ID	Instance state	Instance type	Launch time
<input checked="" type="checkbox"/>	Amazon Linux Web Server	i-09d22341175de57bf	Running Details Stop	t2.micro	2023-11-11 10:40:00

Click on the SSH client tab

The screenshot shows a web-based interface for connecting to an Amazon Linux Web Server instance. At the top, there's a header with the title 'Connect to instance' and a 'Info' link. Below the header, a sub-header says 'Connect to your instance i-09d22341175de57bf (Amazon Linux Web Server) using any of these options'. There are four tabs at the top of the main content area: 'EC2 Instance Connect', 'Session Manager', 'SSH client' (which is underlined, indicating it's selected), and 'EC2 serial console'. Under the 'SSH client' tab, there's a section titled 'Instance ID' with a dropdown menu showing 'i-09d22341175de57bf (Amazon Linux Web Server)'. Below this, a numbered list of steps is provided: 1. Open an SSH client. 2. Locate your private key file. The key used to launch this instance is `amazon_linux_web_kp.pem`. 3. Run this command, if necessary, to ensure your key is not publicly viewable.
`chmod 400 amazon_linux_web_kp.pem`. 4. Connect to your instance using its Public DNS:
`ec2-3-87-154-9.compute-1.amazonaws.com`. An example command is shown at the bottom: `ssh -i "amazon_linux_web_kp.pem" ec2-user@ec2-3-87-154-9.compute-1.amazonaws.com`.

Copy the Example command line command given near the bottom that will allow you to connect to this instance. We will use it a bit later.

Open up your terminal. Not cmd for windows users, windows users use PowerShell.

Navigate your prompt to the directory where your keys are stored.

Copy your .pem file with cp.

```
/Keys$ cp amazon_linux_web_kp.pem ~/keys
```

Paste in the command we copied from our instance:

```
/keys$ ssh -i "amazon_linux_web_kp.pem" ec2-user@ec2-34-208-167-1.us-west-2.compute.amazonaws.com
```

Say yes:

```
The authenticity of host 'ec2-34-208-167-1.us-west-2.compute.amazonaws.com (34.208.167.1)' can't be established.
ED25519 key fingerprint is SHA256:0ukm86aeF6G+bDPNr/XX5jAIkG0dD08Mmufj01Augkk.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
```

Then you should get connected. It will look like this:

```
Warning: Permanently added 'ec2-54-162-81-82.compute-1.amazonaws.com,54.162.81.8' (ECDSA) to the list of known hosts.  
#  
~\ _ ##### Amazon Linux 2023  
~~ \#####  
~~ \###|  
~~ \#/ V~' ->  
~~ /_/_/  
~/m/  
[ec2-user@ip-172-31-37-15 ~]$
```

If you get this:

```
Warning: Permanently added 'ec2-35-161-82-30.us-west-2.compute.amazonaws.com' (ED25519) to the list of known hosts.  
@#####  
@ WARNING: UNPROTECTED PRIVATE KEY FILE! @  
@#####  
Permissions 0644 for 'amazon_linux_web_kp.pem' are too open.  
It is required that your private key files are NOT accessible by others.  
This private key will be ignored.  
Load key "amazon_linux_web_kp.pem": bad permissions  
ec2-user@ec2-35-161-82-30.us-west-2.compute.amazonaws.com: Permission denied (publickey,gssapi-keyex,gssapi-with-mic).
```

Then copy and run the chmod command from AWS:

3. Run this command, if necessary, to ensure your key is not publicly viewable.
 - ❑ chmod 400 amazon_linux_web_kp.pem

5. Install nginx

Install nginx with this command: (that last character is a 1) not an l

```
]$ sudo amazon-linux-extras install -y nginx1
```

Or

```
sudo yum install nginx
```

```
y
```

Start the engine with this command: (that is systemctl, not a number 1)

```
]$ sudo systemctl start nginx
```

To verify everything is up and running:

```
$ wget http://localhost
```

And we have an index.html file there.

```
2023-02-09 16:28:20 (119 MB/s) - 'index.html' saved [615/615]
```

Go back to our instance get out of the 'Connect to instance' window by clicking cancel

And check the instance again.

6. To view the index.html of our new web server.

Grab the IP address

The screenshot shows the AWS CloudWatch Instances console. At the top, there's a search bar with placeholder text 'Find instance by attribute or tag (case-sensitive)'. Below it is a table with columns: Name, Instance ID, Instance state, Instance type, and Status check. There are two rows: one for a 'Linux server' (stopped) and one for an 'Amazon Linux ...' instance (running). The 'Amazon Linux ...' row has a checked checkbox. Below the table, the instance details for 'Amazon Linux ...' are expanded. It shows tabs for Details, Security, Networking, Storage, Status checks, Monitoring, and Tags. Under the Details tab, there's a section for 'Instance summary' with a link to 'Info'. A green box highlights the 'Public IPv4 address copied' link, which is located next to the instance ID 'i-02e523cba9eaaa1c1 (Amazon Linux Web Server)'. To the right of the copied link, there's a button labeled 'View in browser' with the IP address '34.208.167.1' and a link to 'open address'.

And paste it into a browser tab <http://xx.xxx.xxx.1>

And you should be able to get into your nginx page:

Welcome to nginx!

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to nginx.org. Commercial support is available at nginx.com.

Thank you for using nginx.

Let's find where that file is. Go to where nginx was installed

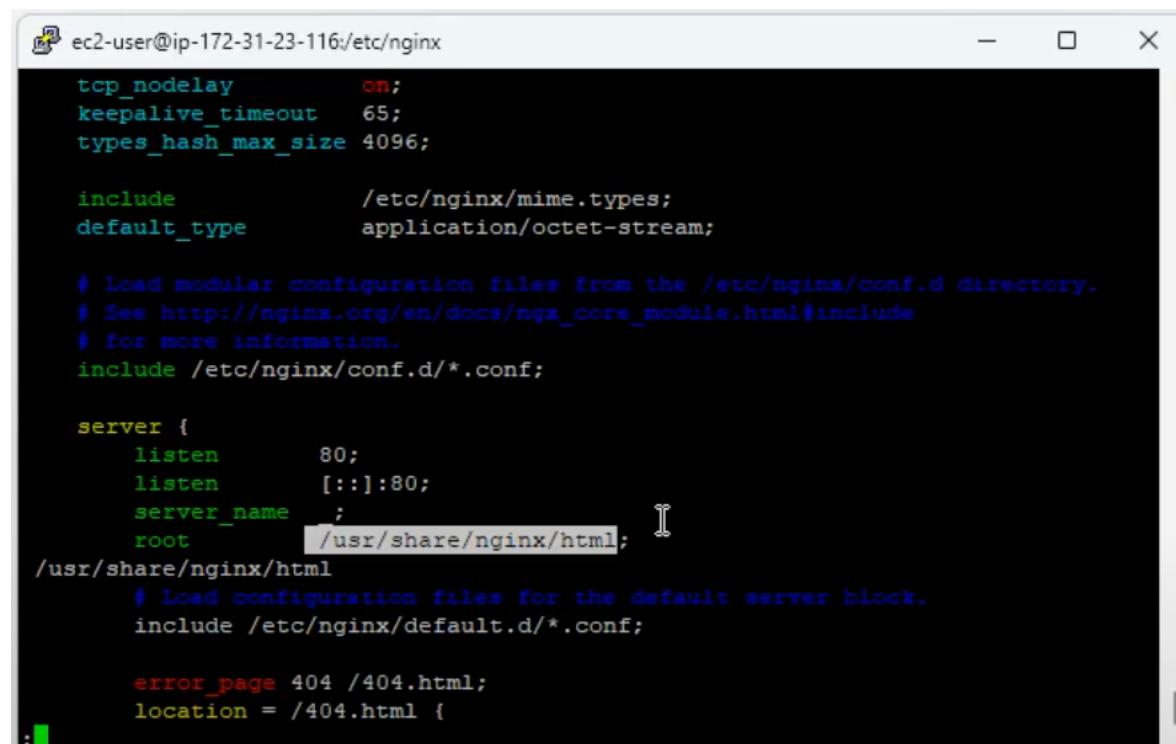
```
| $ cd /etc/nginx/
```

The ls command there will show the files

Edit the configuration file with:

```
| $ vi nginx.conf
```

Arrow down to the server root and copy this path:



```
ec2-user@ip-172-31-23-116:/etc/nginx
tcp_nodelay          on;
keepalive_timeout    65;
types_hash_max_size 4096;

include              /etc/nginx/mime.types;
default_type         application/octet-stream;

# Load modular configuration files from the /etc/nginx/conf.d directory.
# See http://nginx.org/en/docs/ngx_core_module.html#include
# for more information.
include /etc/nginx/conf.d/*.conf;

server {
    listen      80;
    listen      [::]:80;
    server_name ;
    root       /usr/share/nginx/html;  █
/usr/share/nginx/html
    # Load configuration files for the default server block.
    include /etc/nginx/default.d/*.conf;

    error_page 404 /404.html;
    location = /404.html {
```

Don't save it and get out of the editor Shift+Q+Z and go to that directory.

```
| $ cd /usr/share/nginx/html
```

The ls command will now show the index.html file.

Type this to edit that file:

```
$ sudo nano index.html
```

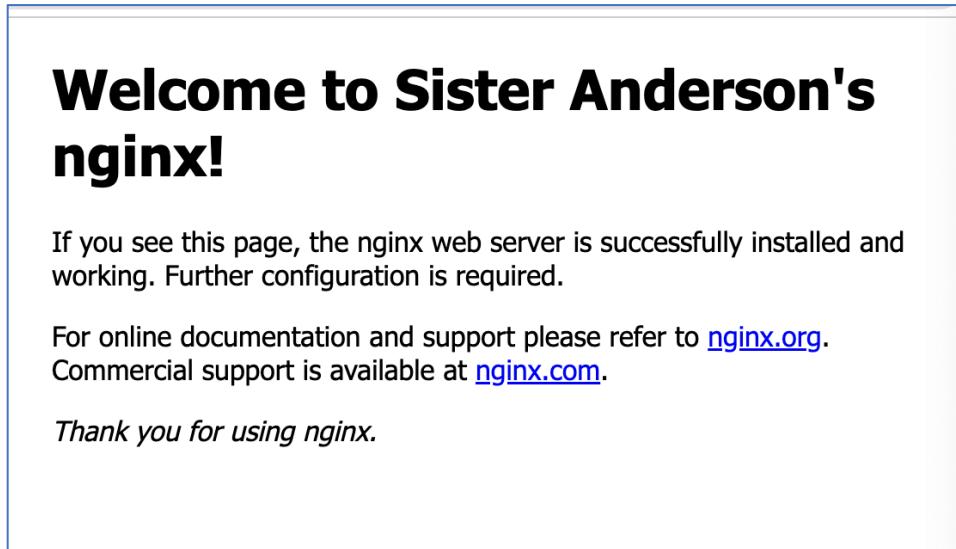
Use the arrow keys to edit the <h1> for example. Use your name, not mine.

```
<!DOCTYPE html>
<html>
<head>
<title>Welcome to nginx!</title>
<style>
html { color-scheme: light dark; }
body { width: 35em; margin: 0 auto;
font-family: Tahoma, Verdana, Arial, sans-serif; }
</style>
</head>
<body>
<h1>Welcome to Sister Anderson's nginx!</h1>
<p>If you see this page, the nginx web server is successfully installed and
working. Further configuration is required.</p>
<p>For online documentation and support please refer to
<a href="http://nginx.org">nginx.org</a>.<br/>
Commercial support is available at
<a href="http://nginx.com">nginx.com</a>.</p>
```

Ctrl-O then enter, to write. Should see something like 'Wrote out 23 lines'

Ctrl-X to exit

Refresh your browser page and it should show the changes.



Screenshot this index.html page for submission.