2025-10-30_Set Up a Web App in the Cloud.excalidraw

step 1: Set up IAM user

- __

```
/*** Comments ***

This project sets up foundations of CI/CD pipeline by creating a web
app from scratch

***/
```

we'll need to

- launch ec2
- ssh into ec2 via vscode

1 set up IAM user

create Admin user with Admin rights policy download csv

step 2: launch ec2

- __
  __2 launch instance

```
/*** Comments ***
in this step we are launching an ec2 instance in our aws account, we
be setting up its network settings which will help us find the
instance  and getting the key pair which ensures we have the
permissions
***/
```

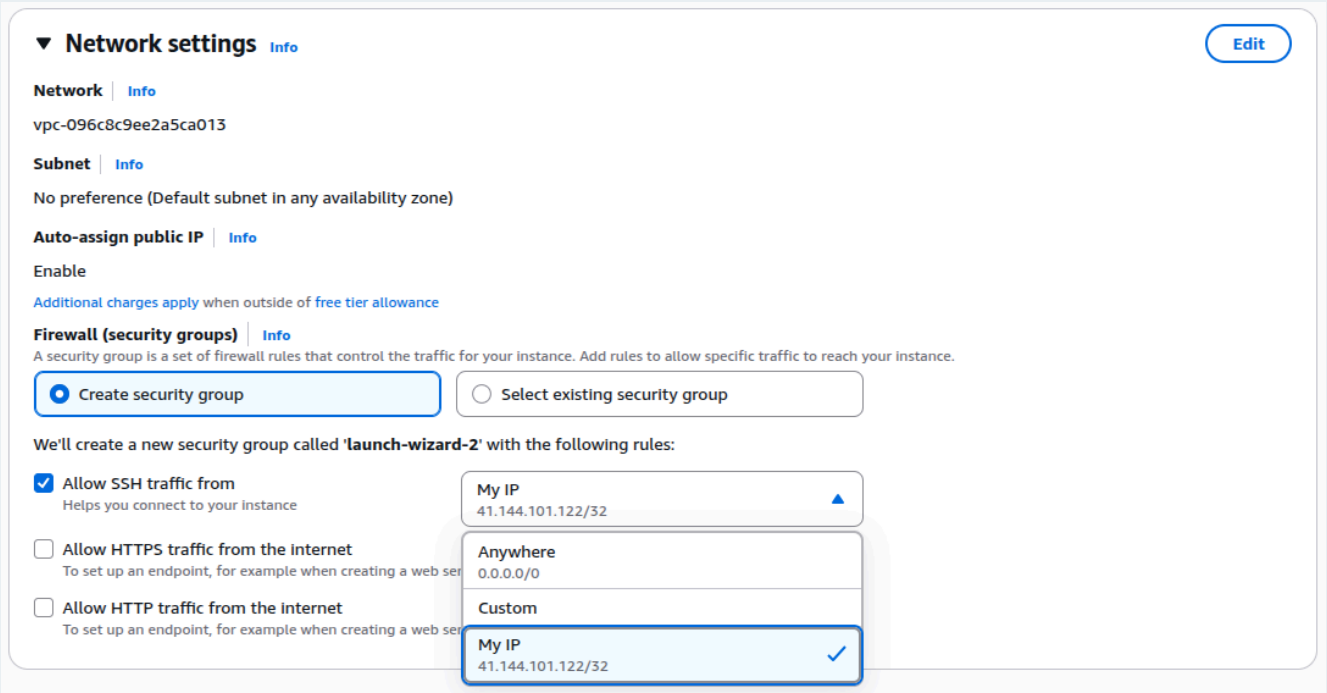
we want web app to live entirely on the cloud, we want to even develop our web apps code in the cloud

ec2 instance name:> nextwork-devops-enter your name'

`nextwork-devops-katleo`

__network settings

💡 network settings #aws/notes/ec2/check-ip-address >



check ip matches here:: https://checkip.amazonaws.com

select key `pair`
public key >> aws keeps
private key >> we keep

hey aws i want access to x ec2 heres my private key >> aws:: ok lets checks if unlocks permission granted if not denied

key pair name: nextwork-keypair

💡 what is a key pair? >

a key pair allows us to securely connect to ec2 we keep a pirvate key and aws keeps the public key aws will authenticate our pirvate key when we want to connect to ec2 later by matching it to the public key

What was downloaded file in your local computer?
aws automatically downloaded private key file .pem for save keeping

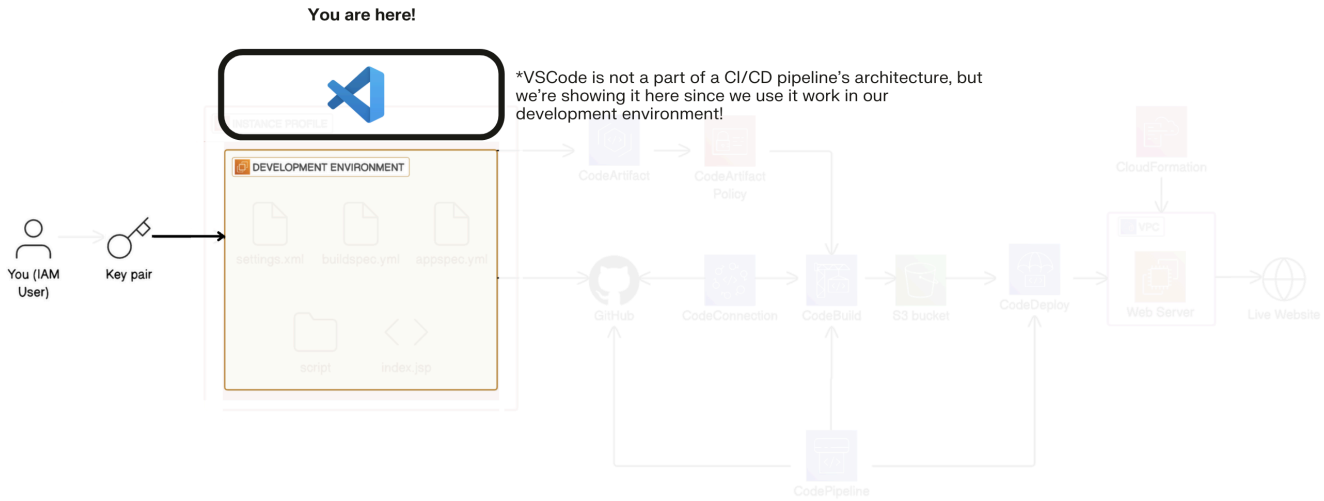
protocol that authorizes users to access remote servers link ec2 instances, its also a type of traffic that lets us transfer data back and forth to ec2 instance once we are connected to it
we enable ssh only form out IP for security

this section
install vs code code editor
setup terminal and update permissions settings of our key

step 3: install vs code

- __

- [ ] Set up a terminal in VS Code, so you can communicate with your EC2 instance.
- [ ] Update your key pair's permission settings, so you can use it to log into your EC2 instance later.



__open vscode in devops folder

> 💡 **what is a terminal** ›
>
> A terminal is where you send instructions to your computer using text instead of clicks. For example, instead of right-clicking on your desktop to create a new folder, you can type a simple text command in your terminal instead. It's like sending text messages to your computer's operating system to tell it what to do. Every computer has a terminal. On Windows, it's often called Command Prompt or PowerShell, while macOS and Linux systems use Terminal

Navigate your terminal to the DevOps folder: `cd ~/Desktop/DevOps`

💡 `cd` , `~` `/desktop/devops` ... what do these commands do? ›

`cd` (Change Directory): This command is used to move between folders in your file system.

`~` represents your home directory i.e. the starting point of a computer's file system. For most users, this is where you begin when you open the terminal.

`/Desktop/DevOps` tells your terminal how to get from your home directory to the DevOps folder. The forward slashes (/) are used to separate each directory in the path.

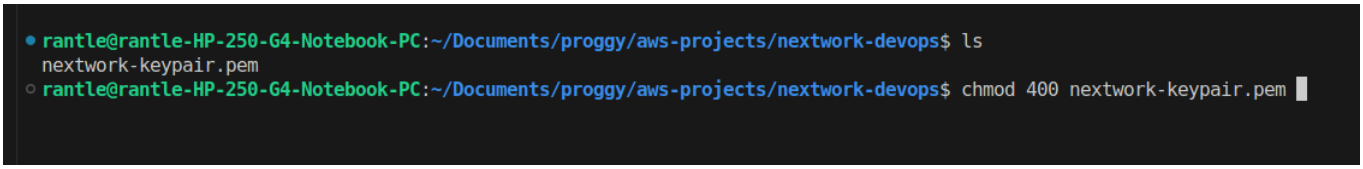
__Once you're in the DevOps folder, you might want to check if your .pem file is there.

```
ls

cat ne<tab to autocomplete> <enter>
```

Use `ls` (Mac/Linux) or `dir` (Windows) :: ls or dir lists all the files and directories in your current location. If your .pem file is in the DevOps folder, you should see it listed 👀

__Change the permissions of your .pem file:




```
chmod 400 nextwork-keypair.pem
```

What is chmod? ›

This command stands for "change mode", and it changes the permissions of your .pem file. Using 400 makes it readable only by you (the owner) and restricts access

for everyone else.

We're changing the permissions of your .pem file so that you have access to it when you connect to your EC2 instance later. Blocking out everyone else keeps your .pem file i.e. your secret key secure.

by default private key is set to have max security ie block everyone from reading it

__for windows only

```
icacls "nextwork-keypair.pem" /reset
icacls "nextwork-keypair.pem" /grant:r "Enter your username:R"
icacls "nextwork-keypair.pem" /inheritance:r
```

Make sure to enter your Windows username. If you don't know your username, run `whoami` in your terminal to find out.

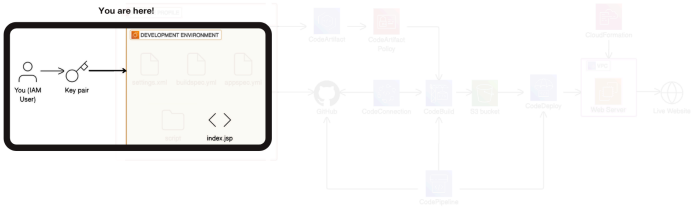
> 🙋 **Running into errors?**
> You're not alone - this happens all the time. Share any errors/questions with the NextWork community!
>
> In the meantime, here are a few troubleshooting steps you can try:
>
> - Double check that the file name in your command i.e. **nextwork-keypair.pem** matches the file in your DevOps folder.
> - Make sure you're running the command in the **DevOps** folder - run `ls` (Linux) or `dir` (Windows) to check you're in the right place.
> - Check your EC2 instance's security group settings - is it allowing SSH traffic (port 22) from **your** IP address?
> - Check that your IP address is listed the same in AWS's tool and in an official IP address checker tool like this one. If it's not, try adding the IP address from the official IP address checker tool to the security group settings.

Nice work! Now that your .pem file is secure, let's connect to your EC2 instance.

step 4: connect ec2

- __

- [ ] Connect to your EC2 instance.



Use the following command to connect to your EC2 instance:

```
ssh -i "PATH TO YOUR .PEM FILE" "YOUR PUBLIC IPV4 DNS" ec2-user@
```

E.g.,

```
ssh -i ~/Desktop/DevOps/nextwork-keypair.pem ec2-user@ec2-13-239-113-205.ap-southeast-2.compute.amazonaws.com
```

💡 What does above commands do? >

`ssh` starts a secure shell connection to your EC2 instance.
`-i` specifies the identity file (your .pem file) you're using to authenticate a connection.
`ec2-user@` `Enter your IPV4 address`

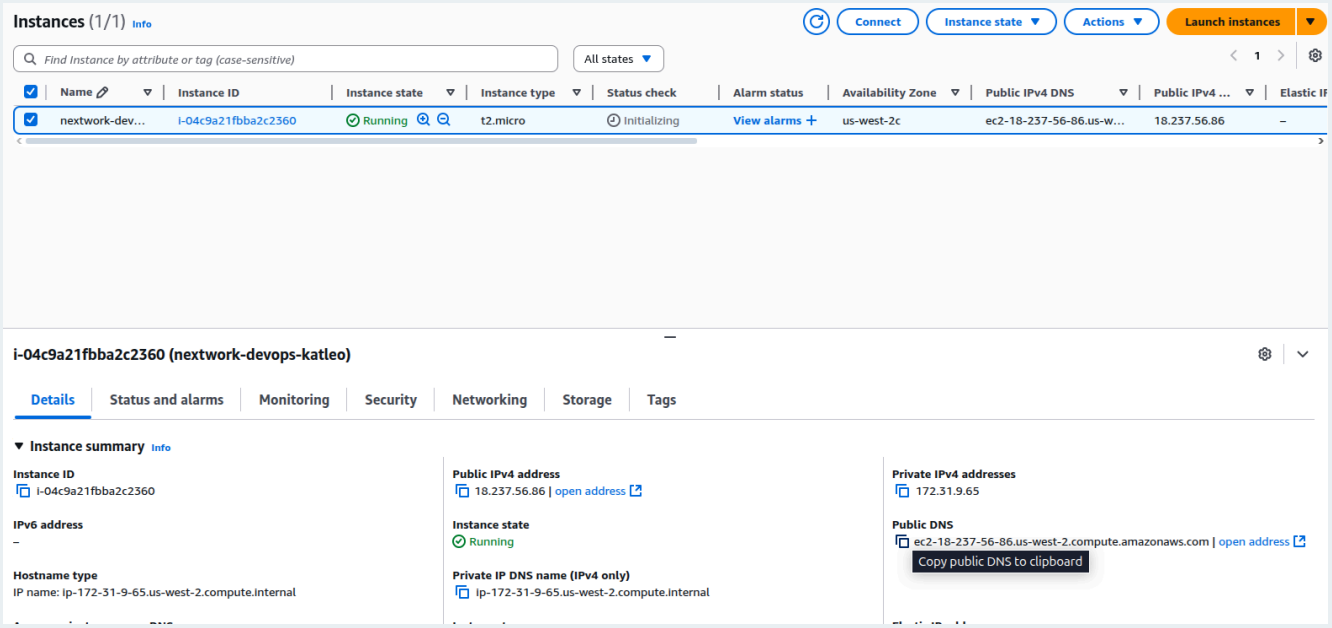
- specifies the username (ec2-user) and
- the address of the EC2 instance (Public DNS) to connect to.

💡 What is a Public IPv4 DNS? >

A Public IPv4 DNS (which stands for Domain Name System) is the public address for your EC2 server that the internet uses to find and connect to it. The local computer you're using to do this project will find and connect to your EC2 instance through this IPv4 DNS.

💡 public DNS of instance >

ec2-18-237-56-86.us-west-2.compute.amazonaws.com



since i am already in the dir with the .pem file

```
ssh -i nextwork-keypair.pem ec2-user@ec2-18-237-56-86.us-west-2.compute.amazonaws.com
```

/home/rantle/Documents/proggy/aws-projects/nextwork-devops
~/Documents/proggy/aws-projects/nextwork-devops/

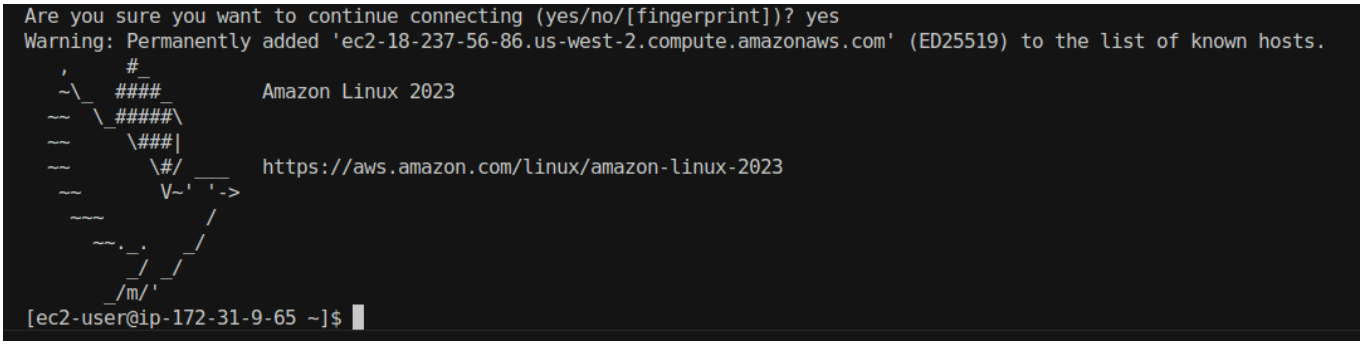
```
ssh -i ~/Documents/proggy/aws-projects/nextwork-devops/nextwork-keypair.pem ec2-user@ec2-18-237-56-86.us-west-2.compute.amazonaws.com
```

relative path >> open terminal / vscode from .pem folder

```
ssh -i "nextwork-keypair.pem" ec2-user@ec2-18-237-56-86.us-west-2.compute.amazonaws.com
```

ssh -i "nextwork-keypair.pem" ec2user@ec2-44-248-98-119.us-west-2.compute.amazonaws.com

success!!



~~step 4 ✅ 2025-10-31~~

step 5: Install Apache Maven and Amazon Corretto 8

- __

Connection DONE 😎

This means your terminal has now entered into your EC2 instance and can use it like a computer that's right in front of you!

Our goal today is to set up a web app inside this instance, so let's install two tools that are going to help us build Java web apps.

💡 what is apache maven? ›

Apache Maven is a powerful tool that automates the building of software.

💡 what does building mean? ›

Building is a process that turns your code into a final product (like web apps) that is ready for virtual machines (like EC2 instances) to host.

The key steps include:

- Compiling: converting your code, written in programming languages like Java, into machine code that a computer can run.
- Linking: combining the compiled machine code with additional dependencies (which are external resources your application needs to work) to create a single program for the computer to run all together.

- Packaging: assembling your file into a package. For Java projects (which is what we're doing), this would be a JAR (Java Archive Resource) file that you'll learn more about later in this DevOps Series. This JAR file makes your program easily distributed or deployed in other environments.
- Testing: running automated tests to make sure your software works without bugs or issues.

  These steps are sooooo important for producing a functioning software application. So helpful to have tools like Apache Maven automate these processes for you.

#aws/notes/ec2/installMaven-installJava

__install maven inside our ec2 instance

download setup file that tells your computer where to find Apache Maven.

```
wget https://archive.apache.org/dist/maven/maven-
3/3.5.2/binaries/apache-maven-3.5.2-bin.tar.gz
```

extracts the downloaded package and saves it in a folder called opt

```
sudo tar -xzf apache-maven-3.5.2-bin.tar.gz -C /opt
```

save a path to the extracted package, so that we can run Maven commands from any directory after this!

```
echo "export PATH=/opt/apache-maven-3.5.2/bin:$PATH" >> ~/.bashrc

source ~/.bashrc
```

```
wget https://archive.apache.org/dist/maven/maven-
3/3.5.2/binaries/apache-maven-3.5.2-bin.tar.gz
sudo tar -xzf apache-maven-3.5.2-bin.tar.gz -C /opt
echo "export PATH=/opt/apache-maven-3.5.2/bin:$PATH" >> ~/.bashrc
source ~/.bashrc
```

**install Java 8, or more specifically, Amazon Correto 8:
tell instance where to find java
saving javas location so we can run from any dir in our instance

```
sudo dnf install -y java-1.8.0-amazon-corretto-devel
export JAVA_HOME=/usr/lib/jvm/java-1.8.0-amazon-corretto.x86_64
export PATH=/usr/lib/jvm/java-1.8.0-amazon-
corretto.x86_64/jre/bin/:$PATH
```

💡 **What is Java? What is Amazon Correto 8? ›**

Java is a popular programming language used to build different types of applications, from mobile apps to large enterprise systems.

Amazon Corretto 8 is a version of Java that we're using for this project. It's free, reliable and provided by Amazon.

https://docs.aws.amazon.com/corretto/latest/corretto-8-ug/what-is-corretto-8.html

To verify that Maven is installed correctly, run the following command next:

```
mvn -v
```

```
Last login: Fri Oct 31 07:41:44 2025 from 41.144.101.122
[ec2-user@ip-172-31-9-65 ~]$ mvn -v
Apache Maven 3.5.2 (138edd61fd100ec658bfa2d307c43b76940a5d7d; 2017-10-18T07:58:13Z)
Maven home: /opt/apache-maven-3.5.2
Java version: 1.8.0_472, vendor: Amazon.com Inc.
Java home: /usr/lib/jvm/java-1.8.0-amazon-corretto.x86_64/jre
Default locale: en, platform encoding: UTF-8
OS name: "linux", version: "6.1.156-177.286.amzn2023.x86_64", arch: "amd64", family: "unix"
[ec2-user@ip-172-31-9-65 ~]$
```

To verify that you've installed Java 8 correctly, run this next:

```
java -version
```

```
[ec2-user@ip-172-31-9-65 ~]$ java -version
openjdk version "1.8.0_472"
OpenJDK Runtime Environment Corretto-8.472.08.1 (build 1.8.0_472-b08)
OpenJDK 64-Bit Server VM Corretto-8.472.08.1 (build 25.472-b08, mixed mode)
[ec2-user@ip-172-31-9-65 ~]$
```

- [x] ~~step 5:Install Apache Maven and Amazon Corretto 8 ✅ 2025-10-31~~
  - [x] ~~Install Apache Maven on your EC2 instance. ✅ 2025-10-31~~
  - [x] ~~Install Amazon Corretto 8, a version of Java. ✅ 2025-10-31~~
  - [x] ~~Verify the installations. ✅ 2025-10-31~~

step 6 create the application

- __

#aws/notes/ec2/UseMvnToGenerateAJavaWebApp . To do this, use these commands:

```
mvn archetype:generate \
    -DgroupId=com.nextwork.app \
    -DartifactId=nextwork-web-project \
    -DarchetypeArtifactId=maven-archetype-webapp \
    -DinteractiveMode=false
```

💡 what is mvn >

When you run mvn commands, you're asking Maven to perform tasks (like creating a new project or building an existing one).

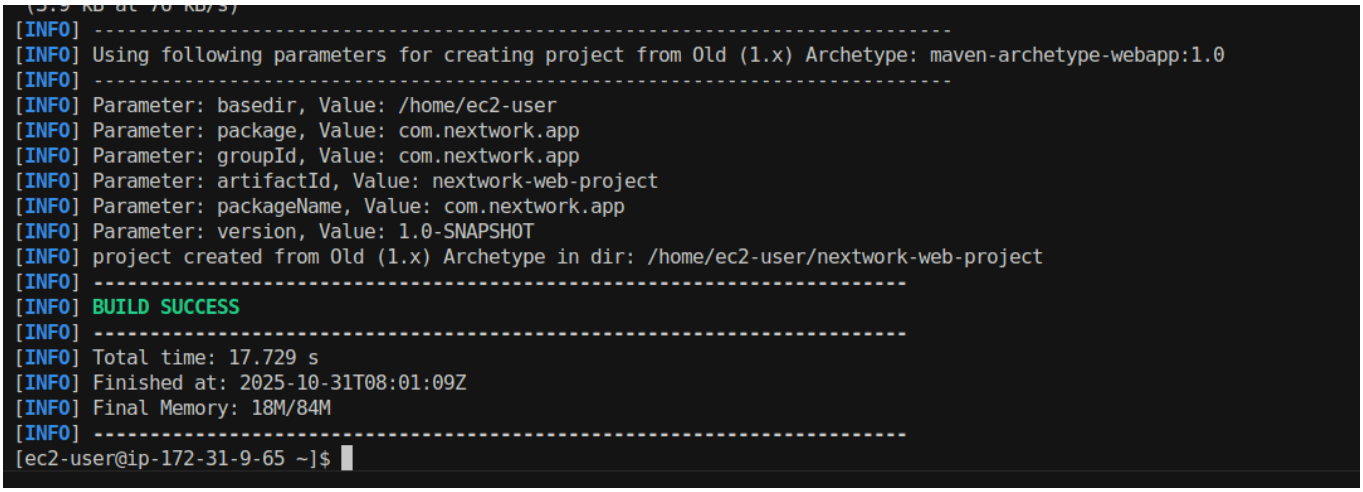
The mvn archetype:generate command specifically tells Maven to create a new project from a template (which Maven calls an archetype). This command sets up a basic structure for your project, so you don't have to start from scratch.

💡 extra details >

`-DartifactId=nextwork-web-project` names your project

`-DarchetypeArtifactId=maven-archetype-webapp` specifies that you're creating a web application.

`-DinteractiveMode=false` runs the command without pausing for user input, so Maven will go ahead and install everything without waiting for your confirmation.



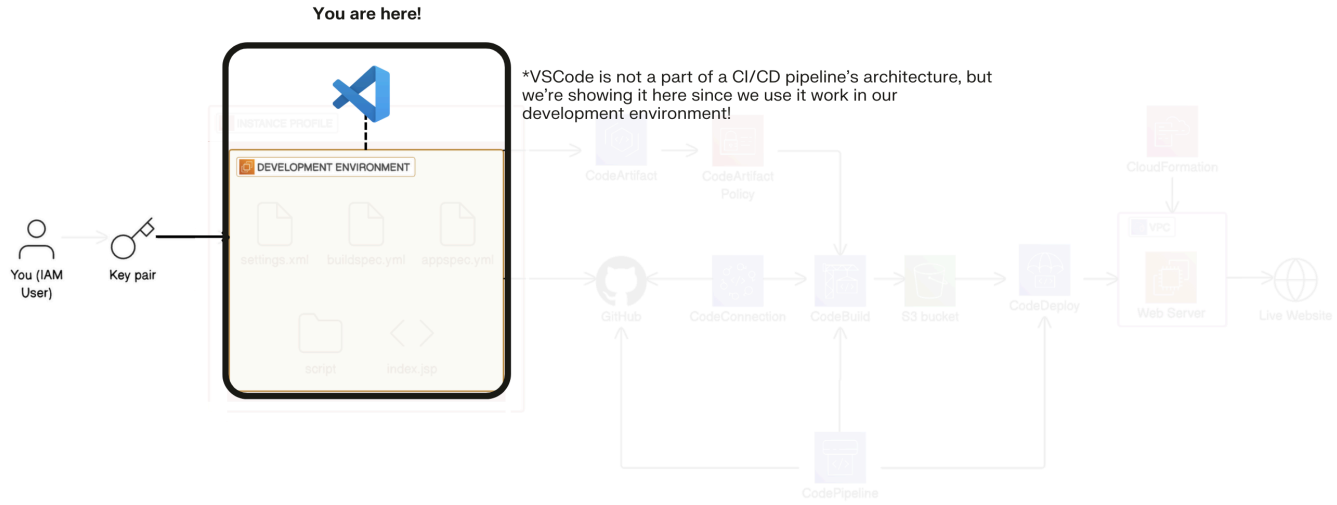
- [ ] step 6
  - [x] ~~Run Maven commands in your terminal to generate a Java web app. ✅ 2025-10-31~~

💡 **review questions** >

How did you create a Java web app?

step 7 Connect VS Code with your EC2 Instance

- __
  In this step, we'll connect VS Code to your EC2 instance so we can see and edit the web app we've just created.



💡 Wait...aren't I already connected to my EC2 instance with SSH? ›

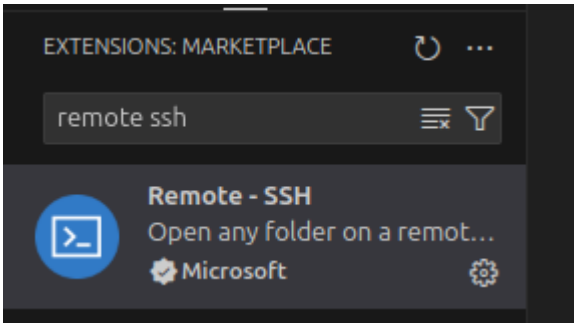
Yes, connecting with SSH in the terminal lets you send text commands to your EC2 instance, but you don't get all the benefits of having an IDE like VS Code.

When you connect VS Code itself to your EC2 instance (not just your terminal), you unlock VS Code's IDE features (like file navigation and code editing) directly on your EC2 instance. This will make it so much easier for you to edit and manage your web app for the rest of this DevOps series.

__In VS Code, install the Remote - SSH extension.

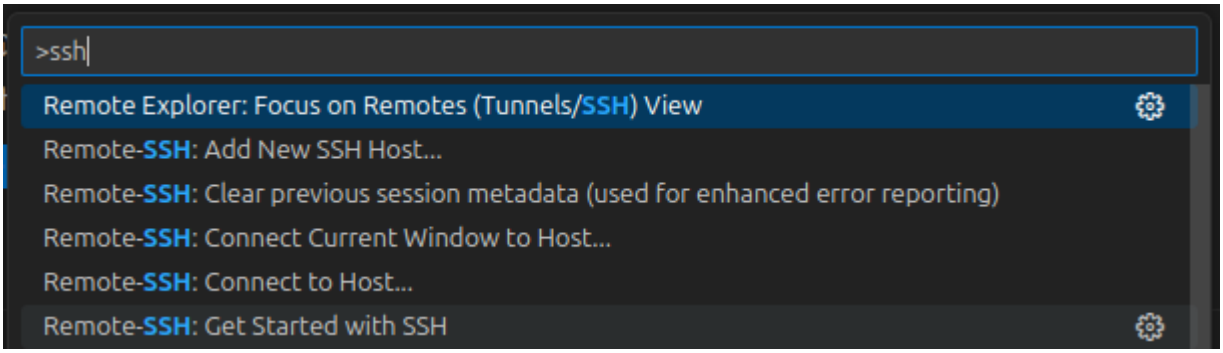
💡 why are we installing remote ssh? ›

The Remote - SSH extension in VS Code lets you connect directly via SSH to another computer securely over the internet. This lets you use VS Code to work on files or run programs on that server as if you were doing it on your own computer, which will come in handy when we edit the web app in your EC2 instance!



Find the option in VS Code to a new SSH Host.

ctrl+shift+p



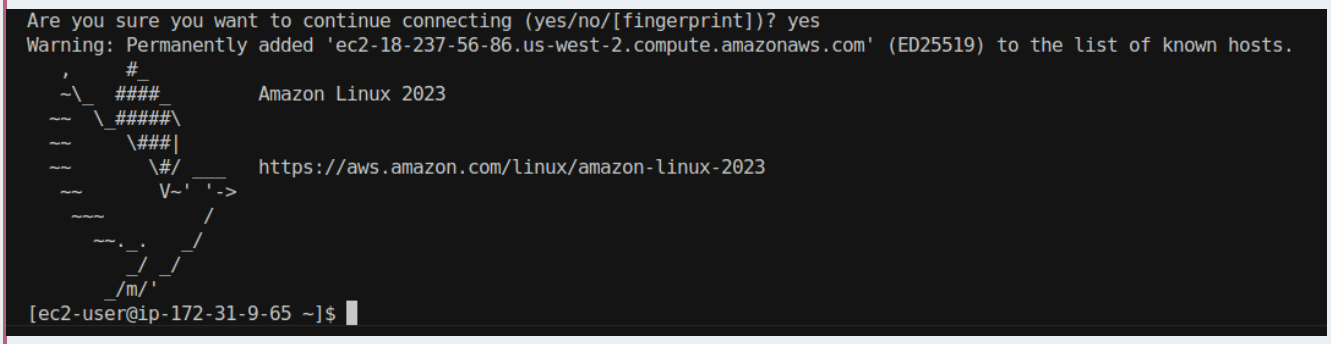
💡 copy paste ssh command >

relative path >> open terminal / vscode from .pem folder

```
ssh -i "nextwork-keypair.pem" ec2-user@ec2-18-237-56-86.us-west-2.compute.amazonaws.com
```

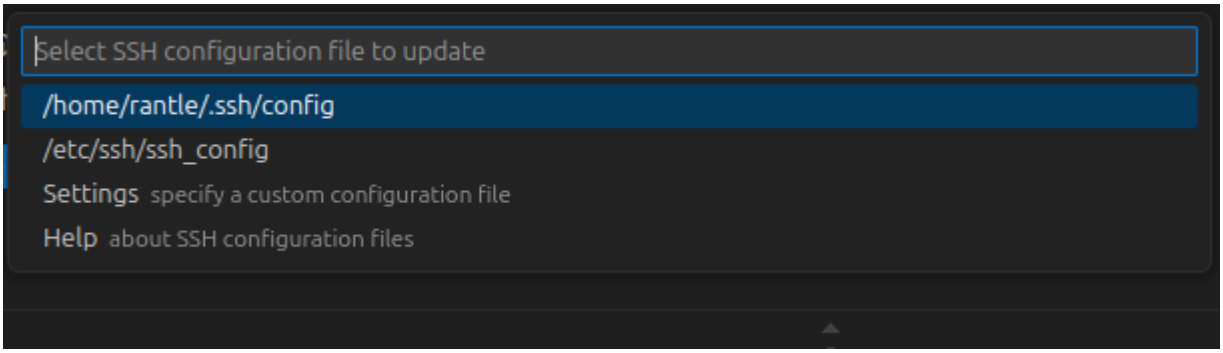
ssh -i "nextwork-keypair.pem" ec2user@ec2-44-248-98-119.us-west-2.compute.amazonaws.com

success!!



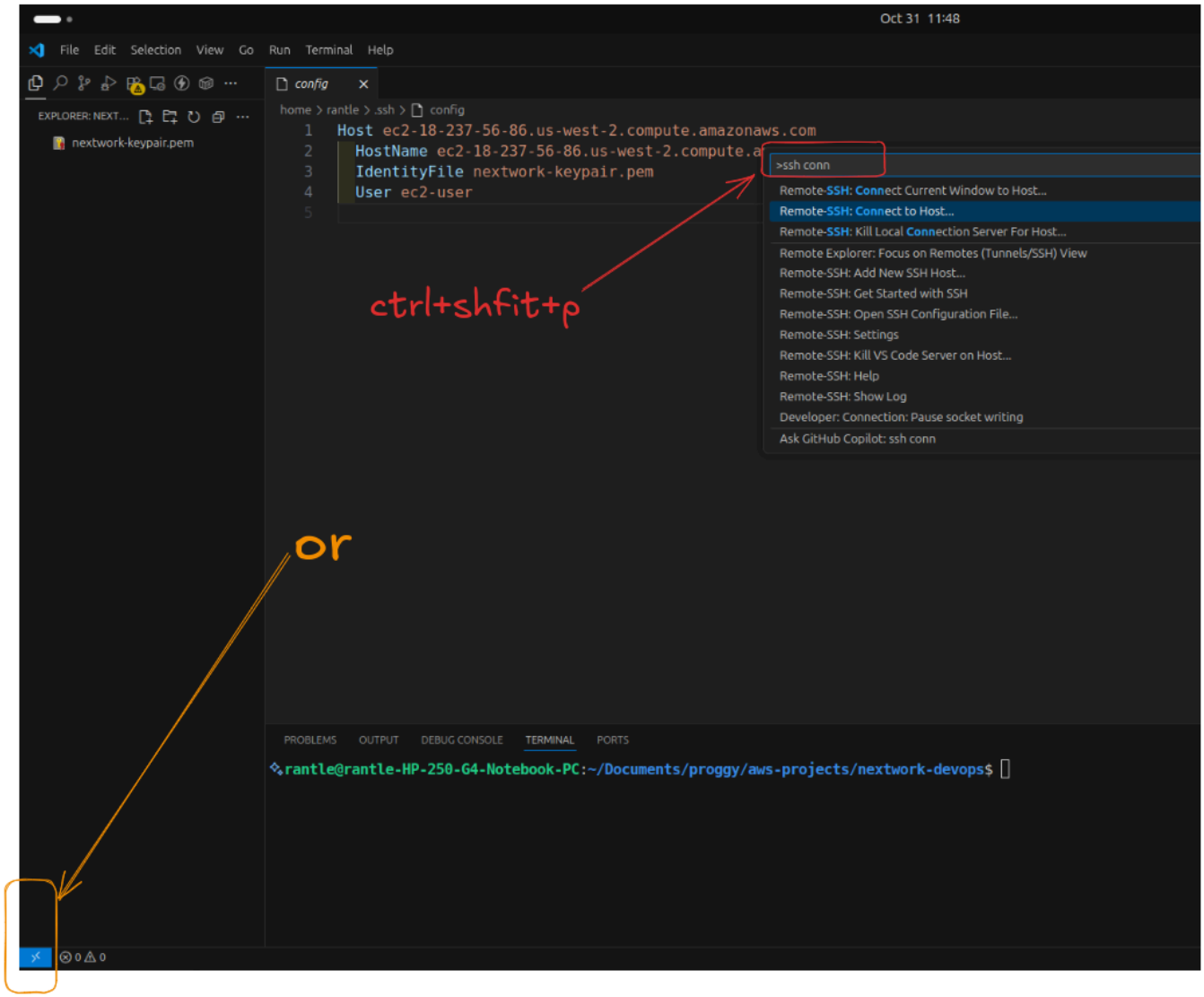
- [x] ~~step 4 ✅ 2025-10-31~~

what file to update ie... where to save these settings >> select firsr

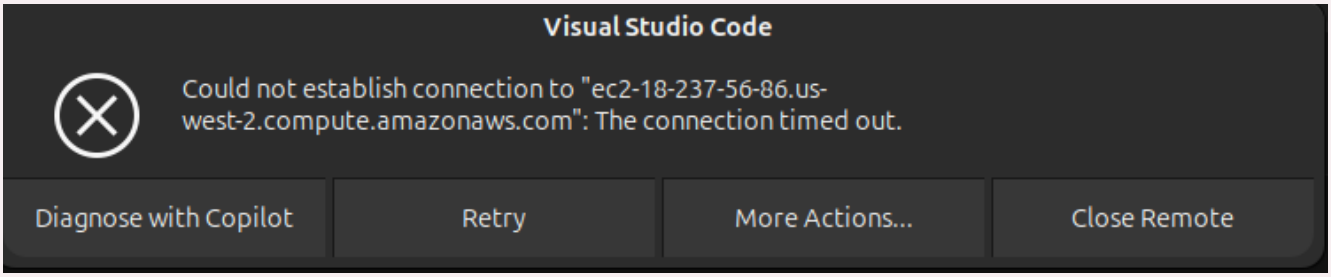


Confirm that the generated configuration file has the right details.

config file is required for remote connection
host is ec2 instance address
identity file is the location of our private key
user specifies the default user for accessing ec2



error >

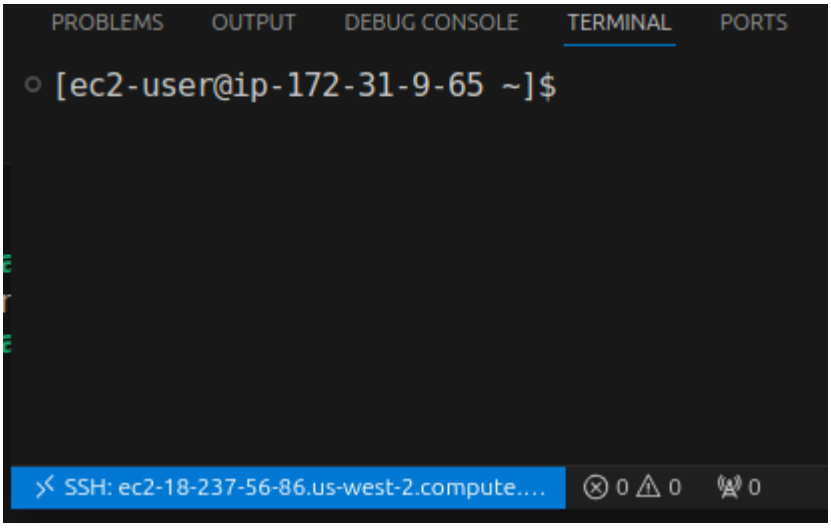


Fix error

fileidentity >> was pointing to incorrect path

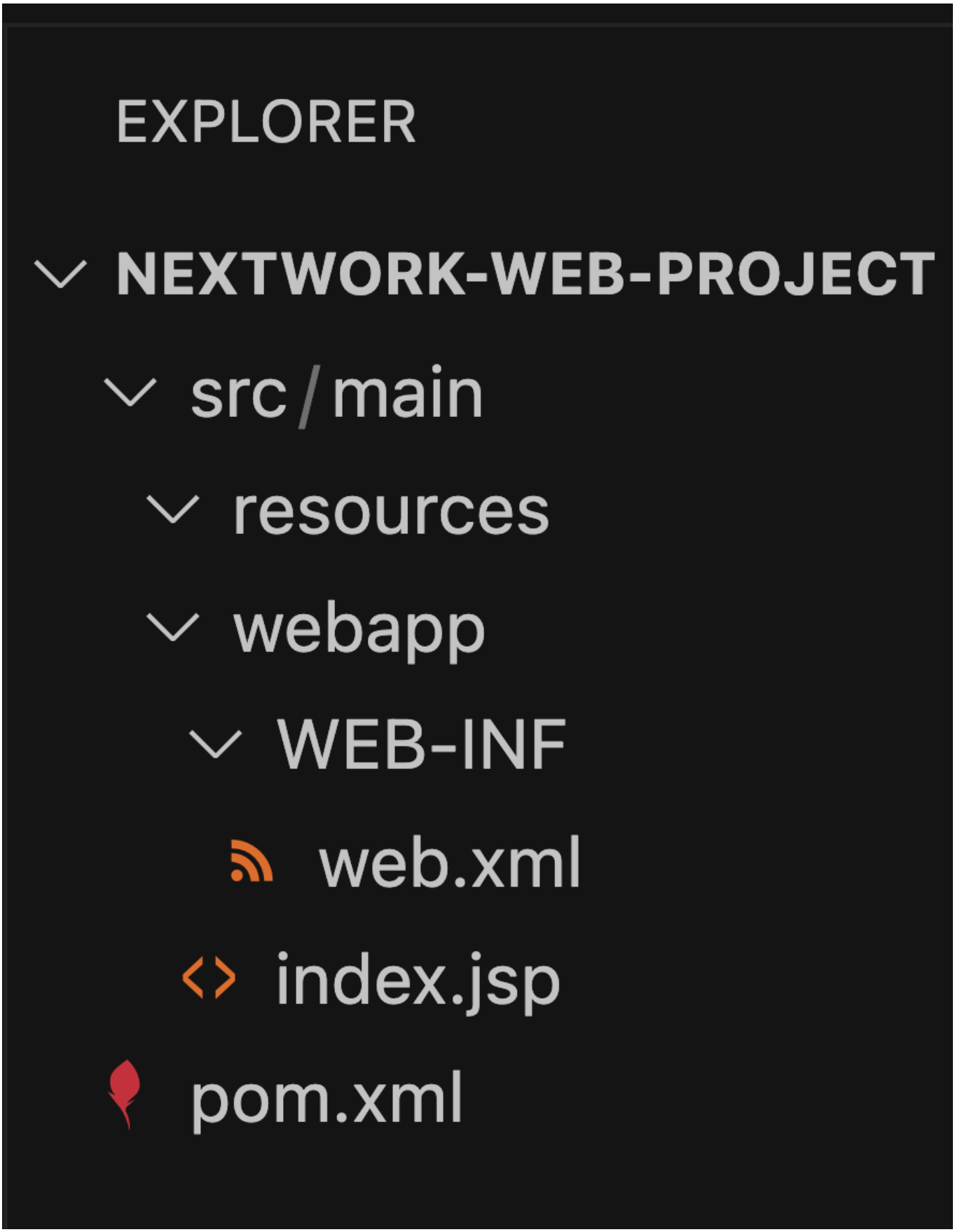
```
Host ec2-18-237-56-86.us-west-2.compute.amazonaws.com
  HostName ec2-18-237-56-86.us-west-2.compute.amazonaws.com
  IdentityFile ~/Documents/proggy/aws-projects/nextwork-
```

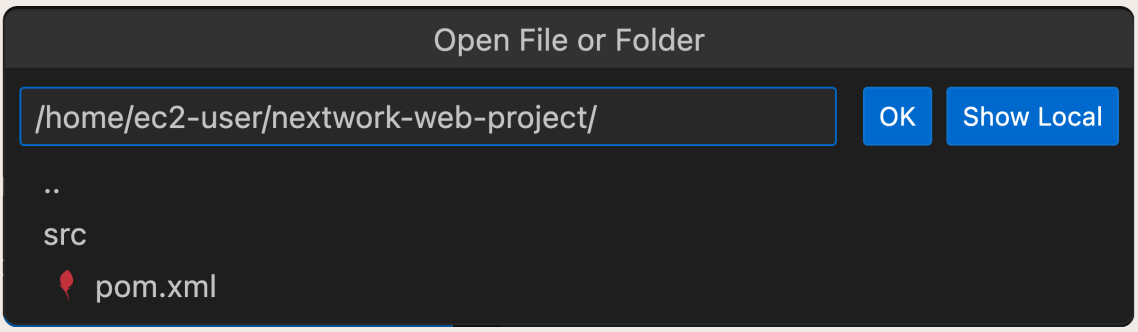
```
devops/nextwork-keypair.pem
  User ec2-user
```



We can explore your web app's files next.

From your new VS Code window, open the nextwork-web-project folder.


EXPLORER
NEXTWORK-WEB-PROJECT
src / main
resources
webapp
WEB-INF
web.xml
index.jsp
pom.xml



💡 about files >

The src (source) folder holds all the source code files that define how your web app looks and works.

src is further divided into webapp, which are the web app's files e.g. HTML, CSS, JavaScript, and JSP files, and resources, which are the configuration files a web app might need e.g. connection settings to a database.

pom.xml is a Maven Project Object Model file. It stores information and configuration details that Maven will use to build the project. We'll use pom.xml later in this project series!

index.jsp defines both static and dynamic content

💡 more on index.jsp >

index.jsp is a file used in Java web apps. It's similar to an HTML file because it contains markup to display web pages. However, index.jsp can also include Java code, which lets it generate dynamic content.

This means content can change depending on things like user input or data from a database. Social media apps are great examples of web apps because the content you see is always changing, updating and personalised to you. HTML files are static and can't include Java code. That's why it's so important to install Java in your EC2 instance - so you can run the Java code in your web app!

- [ ] step 7 Connect VS Code with your EC2 Instance
  - [x] ~~Install an extension in VS Code. ✅ 2025-10-31~~
  - [x] ~~Use the extension to set up a connection between VS Code and your EC2 instance. ✅ 2025-10-31~~
  - [x] ~~Explore and edit your Java web app's files using VS Code. ✅ 2025-10-31~~

> 💡 **review questions** >
>
> What are we doing in this step?
> why did we install remote ssh?
> What was in the configuration file?

what we learn
how to set up java web app in Ec2 instance and cconnect it
In the next project, you'll set up a Github repository to store your web app's code.

secret mission

edit code without IDE

delete resources

in this project we covered

💂 Set up an IAM user: You created a new IAM user with admin permissions to provide a safer alternative to using the AWS root account for ongoing projects.

☁️ Set up VS Code: You set up a new IDE environment using VS Code to write, run, and debug code. You also learnt how to connect VS Code to your EC2 instance to use it as an IDE.

⬇️ Install Maven & Java: You installed Apache Maven and Amazon Corretto 8 in your EC2 instance to manage your project's dependencies for building a Java web app.

🏗️ Create the application: Using Maven, you generated a new Java web app from a template, creating a basic project structure and environment for further development.

💎 Edit your code without VS Code: You edited index.jsp again, this time using nano instead of an IDE. It's a completely different experience!