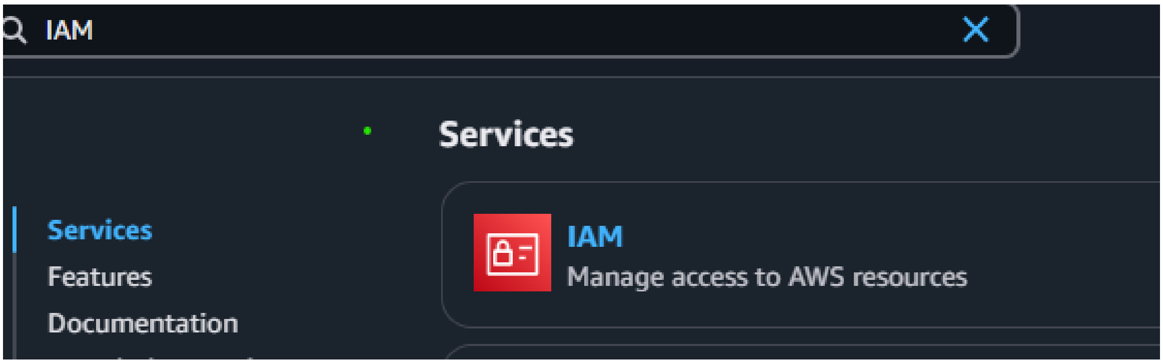
By Steeve Johan MEFIRE

In this project we’re going to set up the foundation of CI/CD pipeline by creating a web app from scratch. We’ll need to launch an EC2 instance and connect to the instance using VScode to generate a web app inside.

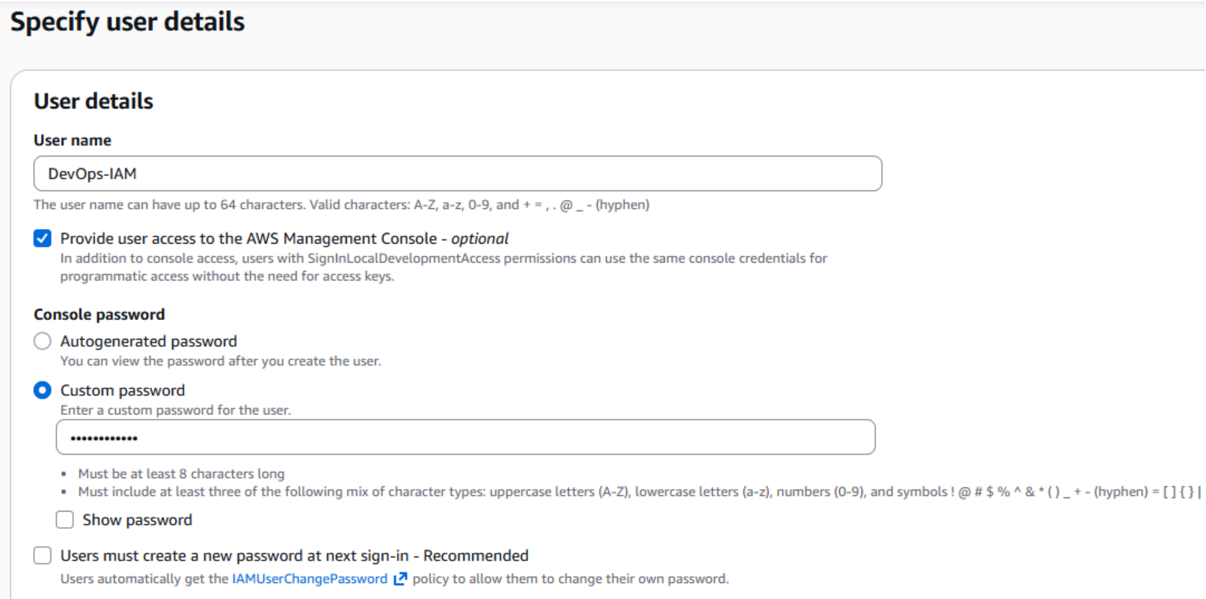
STEP 2# : CREATE AWS IAM AND USER

We need to create an IAM, user to keep you safe when working on AWS about permission to get or do something in your account.

- Type **IAM** in search bar of your AWS console and on the first one



- in left-hand of IAM dashboard click on **user** to create new user



Specify user details

User details

User name

DevOps-IAM

The user name can have up to 64 characters. Valid characters: A-Z, a-z, 0-9, and + = , . @ _ - (hyphen)

☑ Provide user access to the AWS Management Console - *optional*
In addition to console access, users with SignInLocalDevelopmentAccess permissions can use the same console credentials for programmatic access without the need for access keys.

Console password

○ Autogenerated password
You can view the password after you create the user.

● Custom password
Enter a custom password for the user.

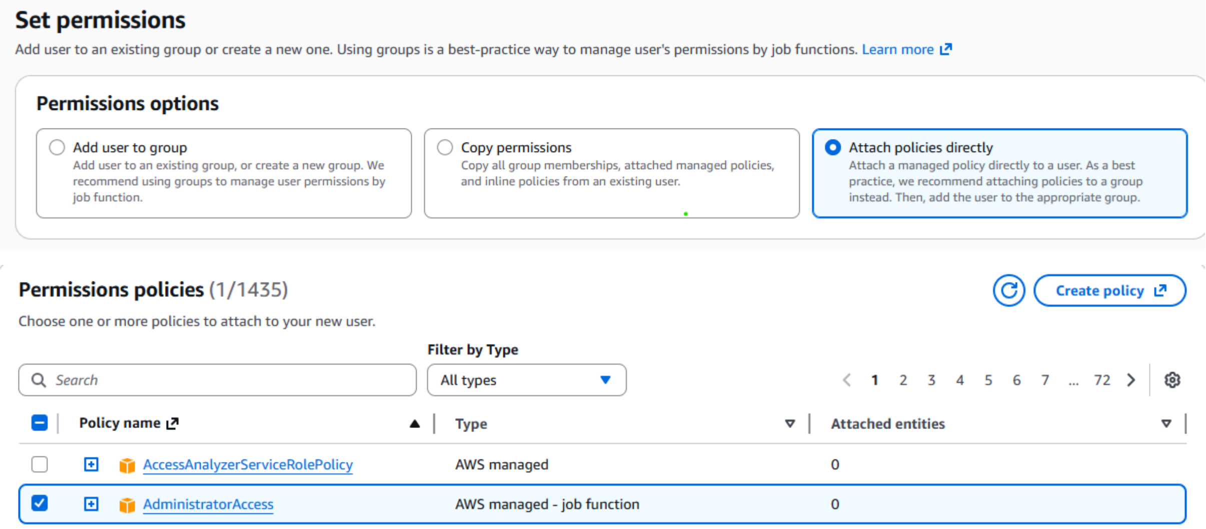
••••••••••••

- Must be at least 8 characters long
- Must include at least three of the following mix of character types: uppercase letters (A-Z), lowercase letters (a-z), numbers (0-9), and symbols ! @ # $ % ^ & * ( ) _ + - (hyphen) = [ ] { } |

☐ Show password

☐ Users must create a new password at next sign-in - Recommended
Users automatically get the IAMUserChangePassword policy to allow them to change their own password.

Then click on **next.**



Set permissions

Add user to an existing group or create a new one. Using groups is a best-practice way to manage user's permissions by job functions. Learn more

Permissions options

○ Add user to group
Add user to an existing group, or create a new group. We recommend using groups to manage user permissions by job function.

○ Copy permissions
Copy all group memberships, attached managed policies, and inline policies from an existing user.

● Attach policies directly
Attach a managed policy directly to a user. As a best practice, we recommend attaching policies to a group instead. Then, add the user to the appropriate group.

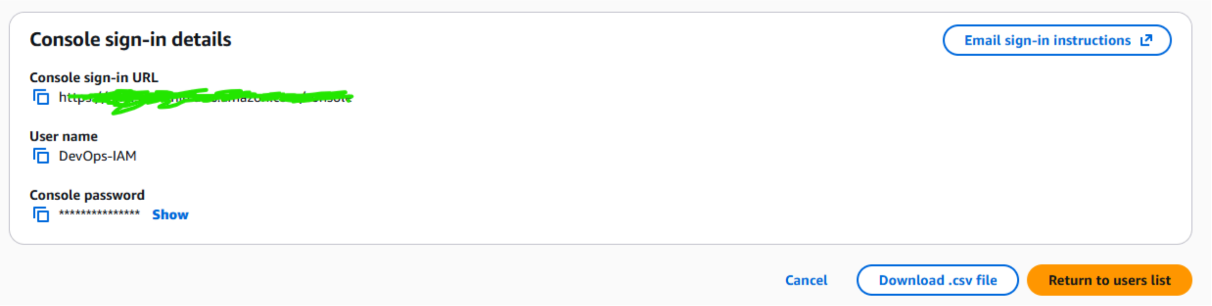
Permissions policies (1/1435)

Choose one or more policies to attach to your new user.

Filter by Type: All types

	Policy name	Type	Attached entities
□	AccessAnalyzerServiceRolePolicy	AWS managed	0
☑	AdministratorAccess	AWS managed - job function	0

- choose **Next,** then click on **create user**



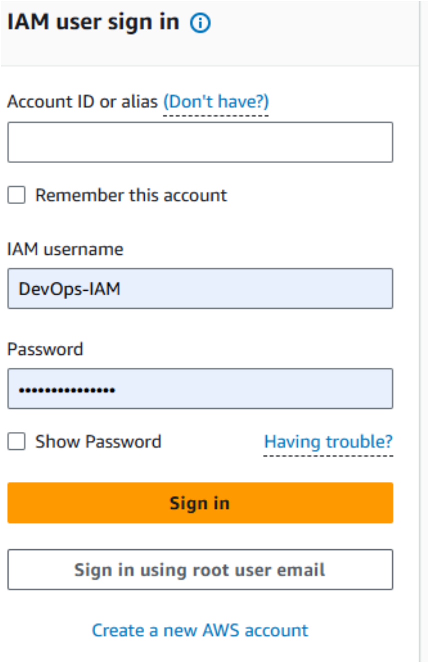
Console sign-in details

Console sign-in URL
[redacted]

User name
DevOps-IAM

Console password
*************** Show

After your IAM user created login with it.



IAM user sign in

Account ID or alias (Don't have?)

□ Remember this account

IAM username

DevOps-IAM

Password

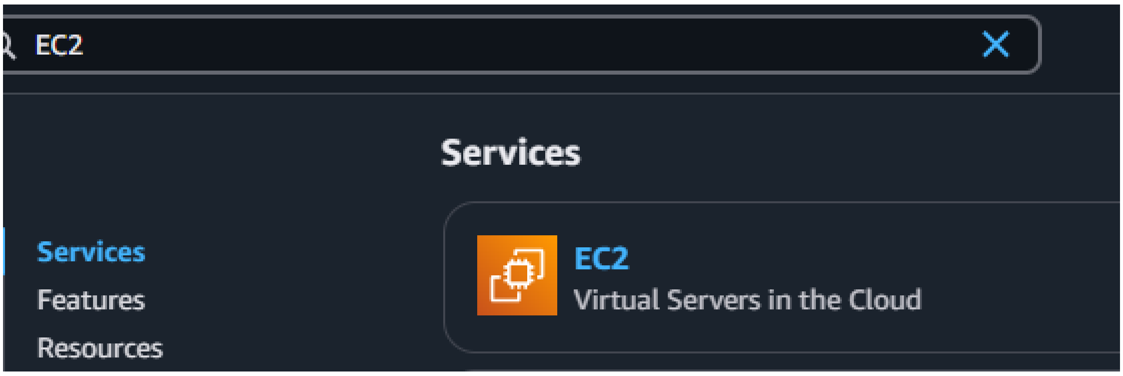
•••••••••••••••

□ Show Password

STEP #2 : LAUNCH INSTANCE EC2

We want your web to be entirely created and run on the cloud, we'll use the server (EC2 instance) to house our development work.

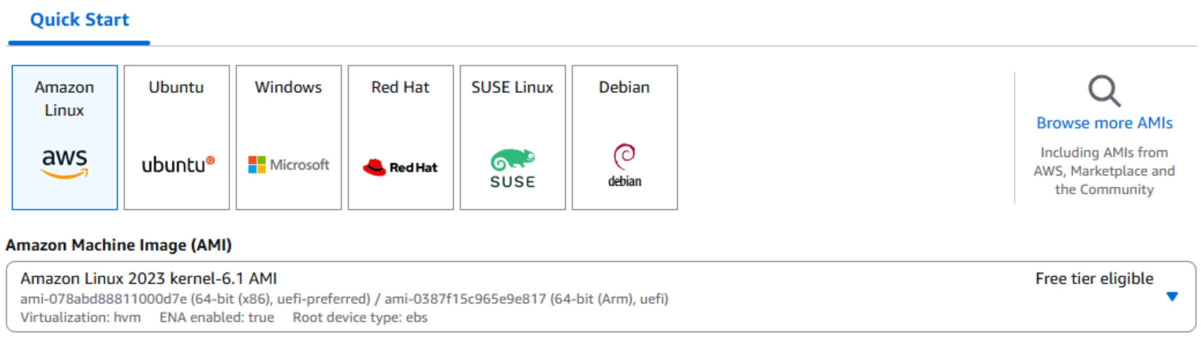
- Head to Amazon EC2 in your AWS Management Console.



EC2

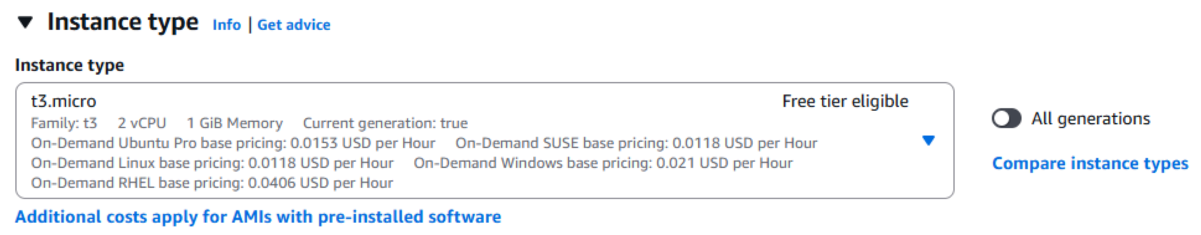
- In your EC2 console, select from right-hand **Launch instance**
- Enter the of your instance

Make sure you **select free tier eligible**



Amazon Machine Image (AMI)

Amazon Linux 2023 kernel-6.1 AMI — Free tier eligible
ami-078abd88811000d7e (64-bit (x86), uefi-preferred) / ami-0387f15c965e9e817 (64-bit (Arm), uefi)
Virtualization: hvm ENA enabled: true Root device type: ebs



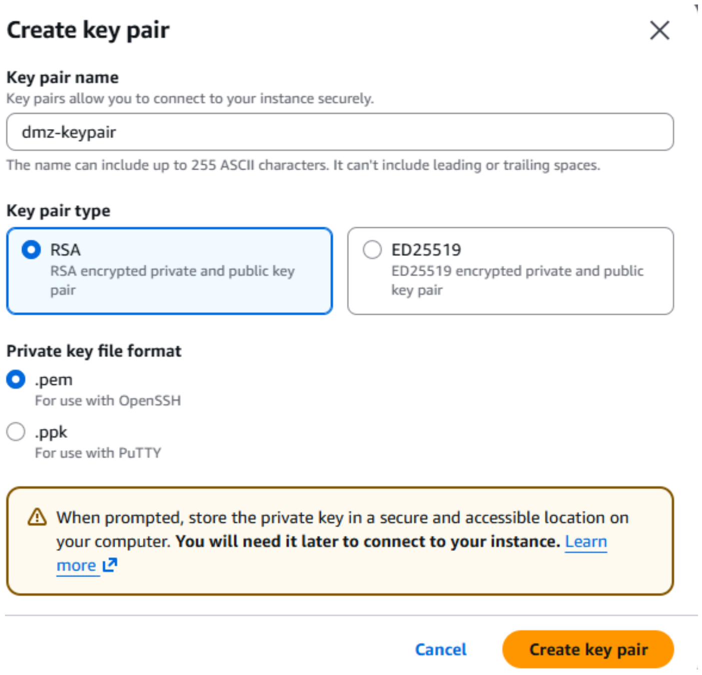
Instance type

t3.micro — Free tier eligible
Family: t3 2 vCPU 1 GiB Memory Current generation: true
On-Demand Ubuntu Pro base pricing: 0.0153 USD per Hour On-Demand SUSE base pricing: 0.0118 USD per Hour
On-Demand Linux base pricing: 0.0118 USD per Hour On-Demand Windows base pricing: 0.021 USD per Hour
On-Demand RHEL base pricing: 0.0406 USD per Hour

Additional costs apply for AMIs with pre-installed software

On **Key pair, select create new key pair**

a Key pair is using to secure access to your EC2 instance



Key pair name

Key pairs allow you to connect to your instance securely.

dmz-keypair

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

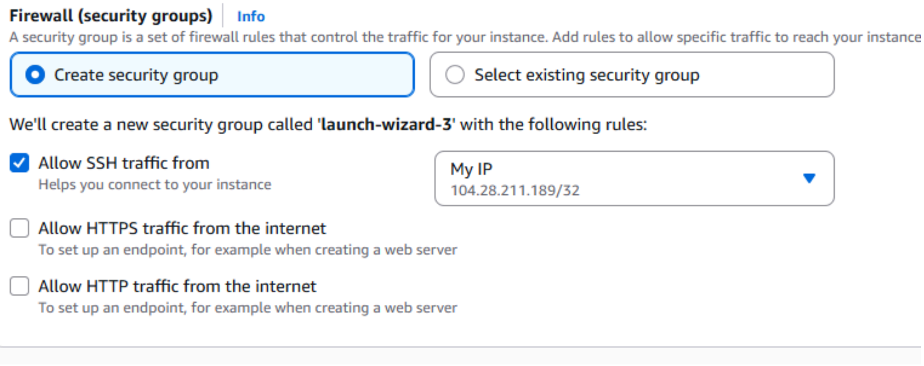
Private key file format

● .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

For **Allow SSH traffic from**, select the dropdown and choose **My IP**. This makes sure only you can access your EC2 instance.



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 reach your instance

● Create security group ○ Select existing security group

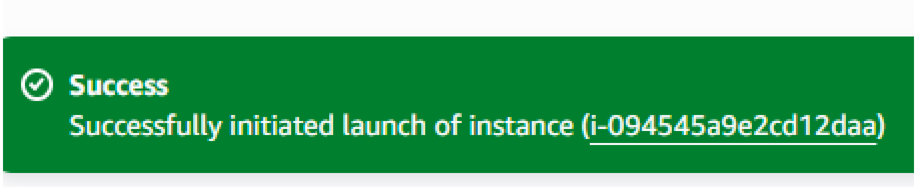
We'll create a new security group called '**launch-wizard-3**' with the following rules:

☑ Allow SSH traffic from — My IP 104.28.211.189/32
Helps you connect to your instance

□ Allow HTTPS traffic from the internet
To set up an endpoint, for example when creating a web server

□ Allow HTTP traffic from the internet
To set up an endpoint, for example when creating a web server

After that, click on launch instance to end.

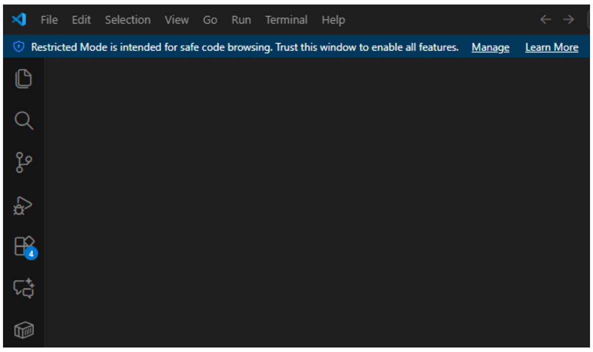


Success
Successfully initiated launch of instance (i-094545a9e2cd12daa)

STEP 3# : INSTALL VS CODE

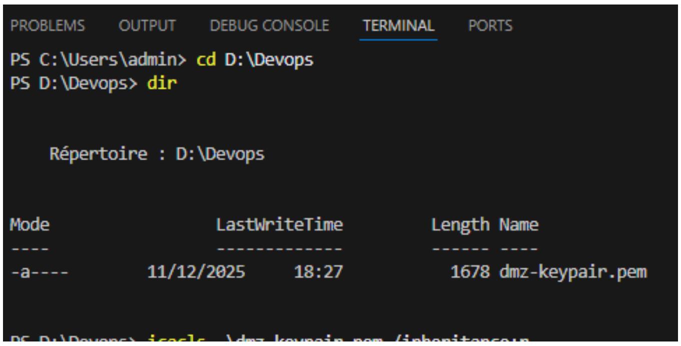
We'll use Visual Studio Code (VS Code) to connect with your instance, so you can create and edit your web app's code.

- Download and install VS code on your computer

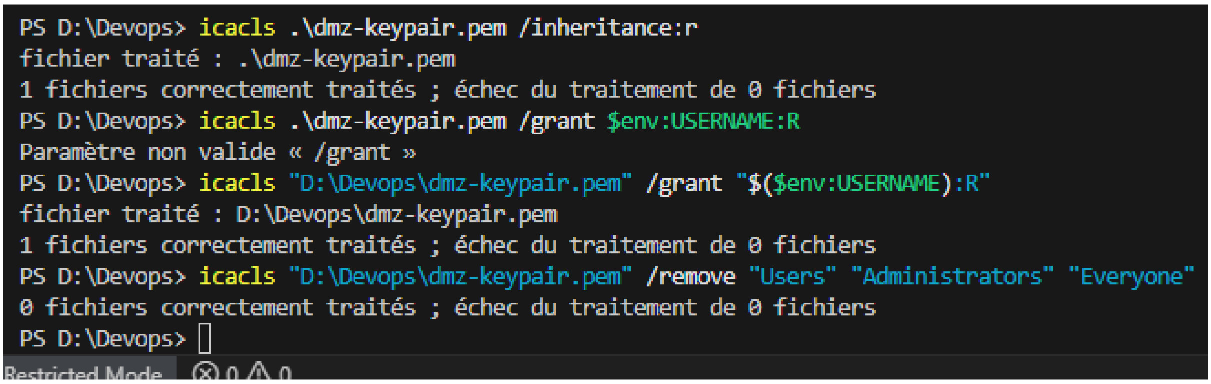


Restricted Mode is intended for safe code browsing. Trust this window to enable all features. Manage Learn More

Click on Terminal and choose a new terminal and enter the access path where you have saved your keypair to give permission to users.



```
PS C:\Users\admin> cd D:\Devops
PS D:\Devops> dir


    Répertoire : D:\Devops


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
-a----        11/12/2025     18:27           1678 dmz-keypair.pem


PS D:\Devops> icacls .\dmz-keypair.pem /inheritance:r
```

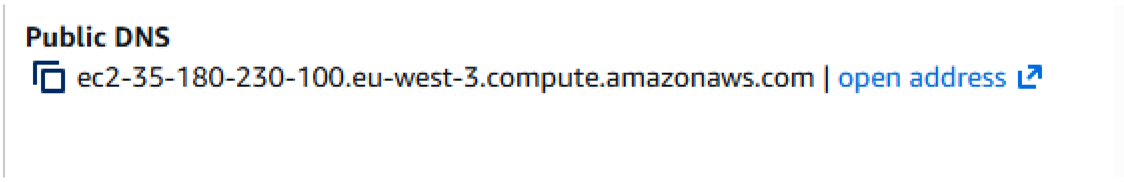


```
PS D:\Devops> icacls .\dmz-keypair.pem /inheritance:r
fichier traité : .\dmz-keypair.pem
1 fichiers correctement traités ; échec du traitement de 0 fichiers
PS D:\Devops> icacls .\dmz-keypair.pem /grant $env:USERNAME:R
Paramètre non valide « /grant »
PS D:\Devops> icacls "D:\Devops\dmz-keypair.pem" /grant "$($env:USERNAME):R"
fichier traité : D:\Devops\dmz-keypair.pem
1 fichiers correctement traités ; échec du traitement de 0 fichiers
PS D:\Devops> icacls "D:\Devops\dmz-keypair.pem" /remove "Users" "Administrators" "Everyone"
0 fichiers correctement traités ; échec du traitement de 0 fichiers
PS D:\Devops> 
```

STEP 2 # : CONNECT INSTANCE EC2

Now over the terminal we're going to try to connect to our EC2 instance. Once we've connecté we can start setting up our web App inside.

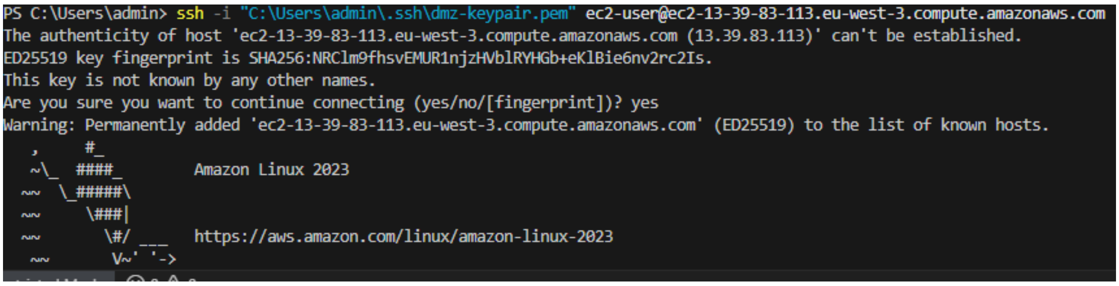
- Go back in your EC2 intance
- Click on the checkbox next to your EC2 instance to view its details.
- Under the Details tab, look for Public **IPv4 DNS.** You'll need this in the next instruction!



Public DNS

ec2-35-180-230-100.eu-west-3.compute.amazonaws.com | open address

Now we're going to establish ssh connection between our VS code's terminal and EC2 instance



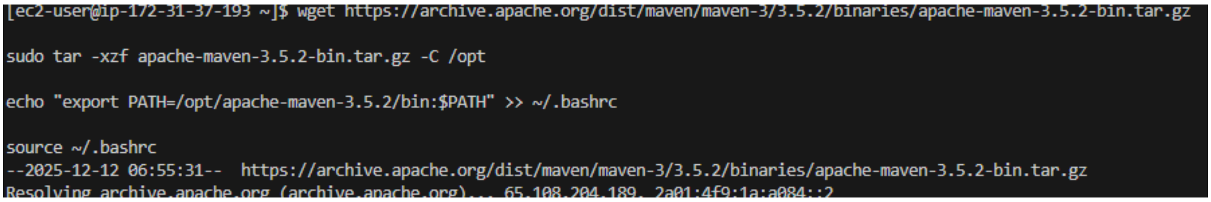
```
PS C:\Users\admin> ssh -i "C:\Users\admin\.ssh\dmz-keypair.pem" ec2-user@ec2-13-39-83-113.eu-west-3.compute.amazonaws.com
The authenticity of host 'ec2-13-39-83-113.eu-west-3.compute.amazonaws.com (13.39.83.113)' can't be established.
ED25519 key fingerprint is SHA256:NRClm9fhsvEMUR1njzHVblRYHGb+eKlBie6nv2rc2Is.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'ec2-13-39-83-113.eu-west-3.compute.amazonaws.com' (ED25519) to the list of known hosts.
   ,     #_
   ~\_  ####_        Amazon Linux 2023
  ~~  \_#####\
  ~~     \###|
  ~~       \#/ ___   https://aws.amazon.com/linux/amazon-linux-2023
   ~~       V~' '->
```

STEP 5 # : INSTALL APACHE MAVEN AND AMAZON CORRETTO 8

Apache Maven is a tool that helps developers build and organize Java software projects. It's also a package manager, which means it automatically download any external pieces of code your project depends on to work.

Install Apache Maven using the commands below:

- **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**

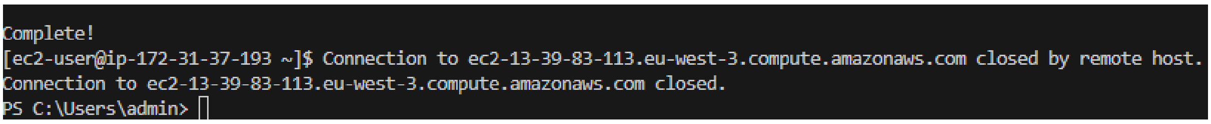


```
[ec2-user@ip-172-31-37-193 ~]$ wget https://archive.apache.org/dist/maven/maven-3/3.5.2/binaries/apache-maven-3.5.2-bin.tar.gz

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
--2025-12-12 06:55:31--  https://archive.apache.org/dist/maven/maven-3/3.5.2/binaries/apache-maven-3.5.2-bin.tar.gz
Resolving archive.apache.org (archive.apache.org)... 65.108.204.189, 2a01:4f9:1a:a084::2
```

Now we're going to install Java 8, or more specifically, Amazon Correto 8.

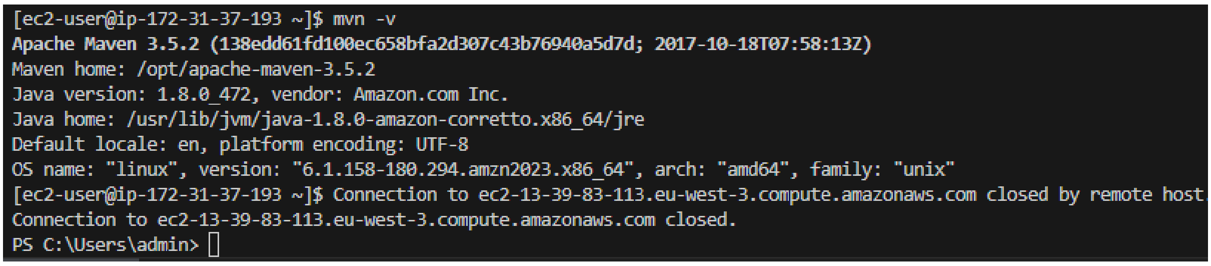
Run these commands:

- **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**



```
Complete!
[ec2-user@ip-172-31-37-193 ~]$ Connection to ec2-13-39-83-113.eu-west-3.compute.amazonaws.com closed by remote host.
Connection to ec2-13-39-83-113.eu-west-3.compute.amazonaws.com closed.
PS C:\Users\admin> 
```

To verify that Maven is installed correctly, type **maven -v**



```
[ec2-user@ip-172-31-37-193 ~]$ 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: "6.1.158-180.294.amzn2023.x86_64", arch: "amd64", family: "unix"
[ec2-user@ip-172-31-37-193 ~]$ Connection to ec2-13-39-83-113.eu-west-3.compute.amazonaws.com closed by remote host.
Connection to ec2-13-39-83-113.eu-west-3.compute.amazonaws.com closed.
PS C:\Users\admin> 
```

STEP #6 : CREATE A WEB APP

We've assembled both Maven and Java into our EC2 instance. Now let's cut straight to generating the web app!

Run this code :

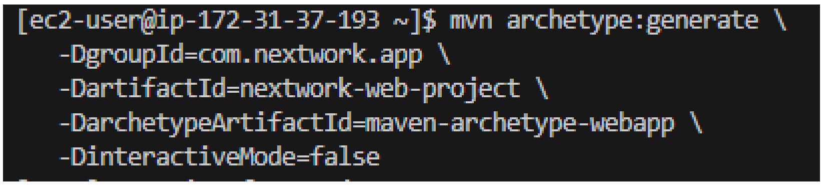
**mvn archetype:generate **

  **-DgroupId=com.nextwork.app **

  **-DartifactId=nextwork-web-project **

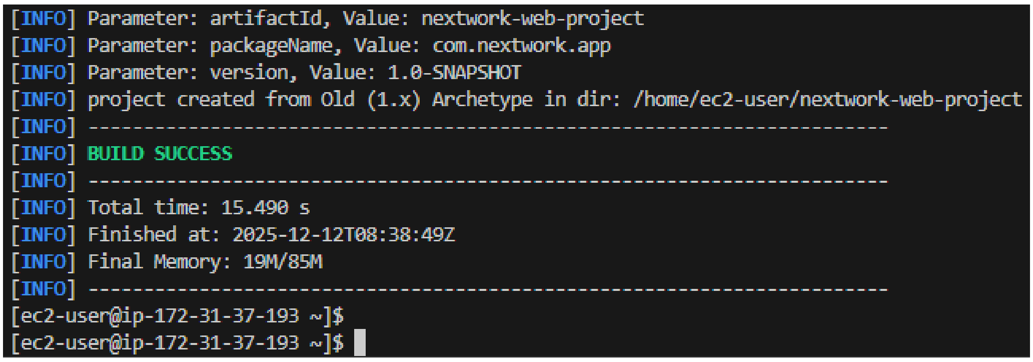
  **-DarchetypeArtifactId=maven-archetype-webapp **

  -DinteractiveMode=false



```
[ec2-user@ip-172-31-37-193 ~]$ mvn archetype:generate \
   -DgroupId=com.nextwork.app \
   -DartifactId=nextwork-web-project \
   -DarchetypeArtifactId=maven-archetype-webapp \
   -DinteractiveMode=false
```

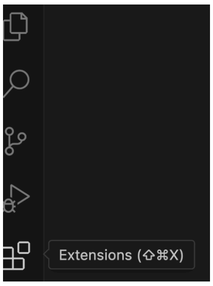
You'll see this message once your application determined



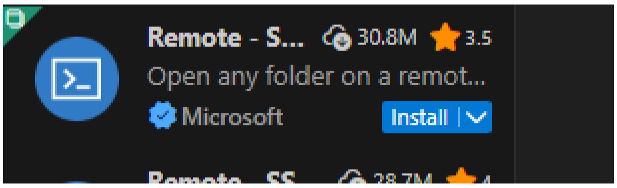
```
[INFO] Parameter: artifactId, Value: nextwork-web-project
[INFO] Parameter: packageName, Value: com.nextwork.app
[INFO] Parameter: version, Value: 1.0-SNAPSHOT
[INFO] project created from Old (1.x) Archetype in dir: /home/ec2-user/nextwork-web-project
[INFO] ------------------------------------------------------------------------
[INFO] BUILD SUCCESS
[INFO] ------------------------------------------------------------------------
[INFO] Total time: 15.490 s
[INFO] Finished at: 2025-12-12T08:38:49Z
[INFO] Final Memory: 19M/85M
[INFO] ------------------------------------------------------------------------
[ec2-user@ip-172-31-37-193 ~]$
[ec2-user@ip-172-31-37-193 ~]$
```

CONNECT VS CODE WITH OUR EC2 INSTANCE

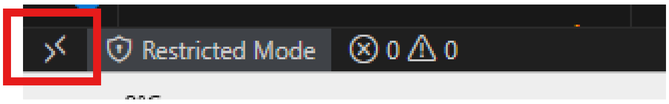
Click on the Extensions icon at the side of your VS Code window. Then, type **Remote - SSH** in the search



click on **install,** to install extension



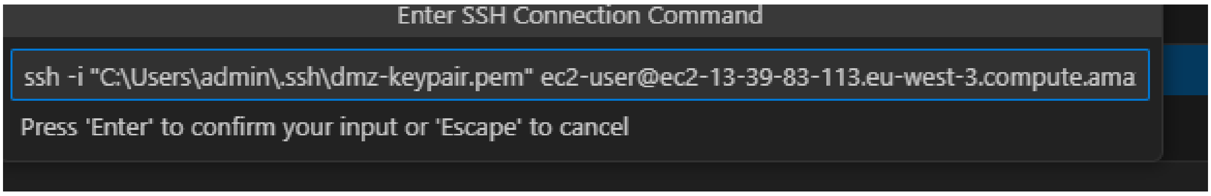
Click on the double arrow icon at the bottom left corner of your VS Code window. This button is a shortcut to use Remote - SSH.



- Select Connect to Host...
- Select + Add New SSH Host…

Enter the SSH command you used to connect to your EC2 instance : in my case was :

ssh -i "C:\Users\admin\.ssh\dmz-keypair.pem" ec2-user@ec2-13-39-83-113.eu-west-3.compute.amazonaws.com

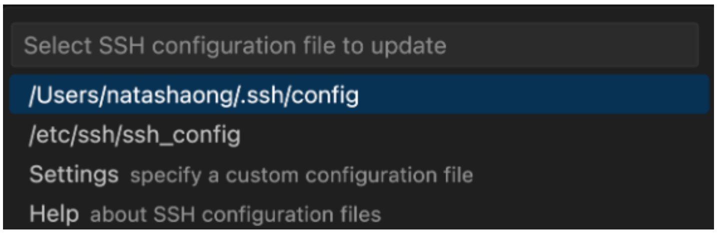


Enter SSH Connection Command

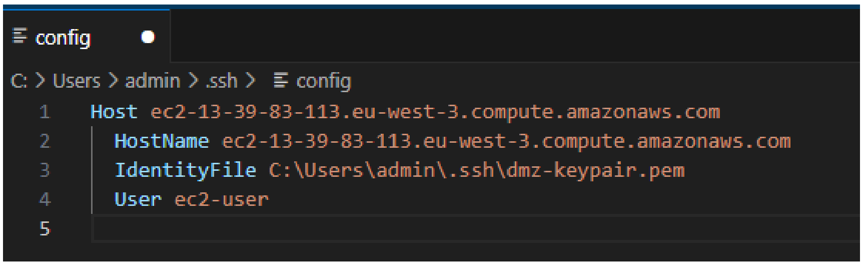
```
ssh -i "C:\Users\admin\.ssh\dmz-keypair.pem" ec2-user@ec2-13-39-83-113.eu-west-3.compute.ama
```

Press 'Enter' to confirm your input or 'Escape' to cancel

- Select the configuration file at the top of your window. It should look similar to **/Users//.ssh/config**


Select SSH configuration file to update
/Users/natashaong/.ssh/config
/etc/ssh/ssh_config
Settings specify a custom configuration file
Help about SSH configuration files

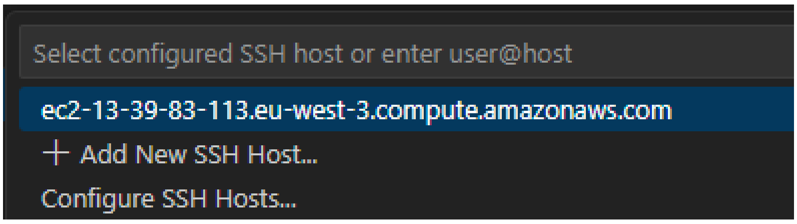
After connection click on pop pup show config



```
Host ec2-13-39-83-113.eu-west-3.compute.amazonaws.com
  HostName ec2-13-39-83-113.eu-west-3.compute.amazonaws.com
  IdentityFile C:\Users\admin\.ssh\dmz-keypair.pem
  User ec2-user

```

- Now you’re ready to connect VS Code with your EC2 instance!

- Click on the double arrow button on the bottom left corner and select Connect to Host again.
- You should now see your EC2 instance listed at the top.


Select configured SSH host or enter user@host
ec2-13-39-83-113.eu-west-3.compute.amazonaws.com
+ Add New SSH Host...
Configure SSH Hosts...

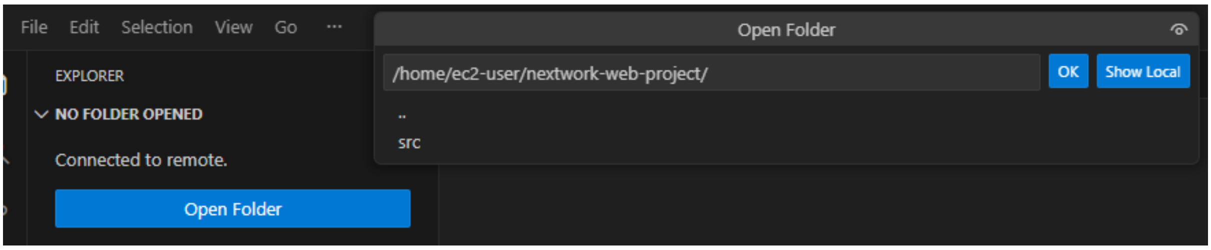
- Select the EC2 instance and off we gooooooooooo to a new VS Code window
- Check the bottom right hand corner of your new VS Code window - it should show your EC2 instance's IPV4 DNS.

Select Open folder.

At the top of your VS Code window, you should see a drop down of different file and folder names. Ooooo, this is VS Code asking you which specific file/folder you'd like to open!

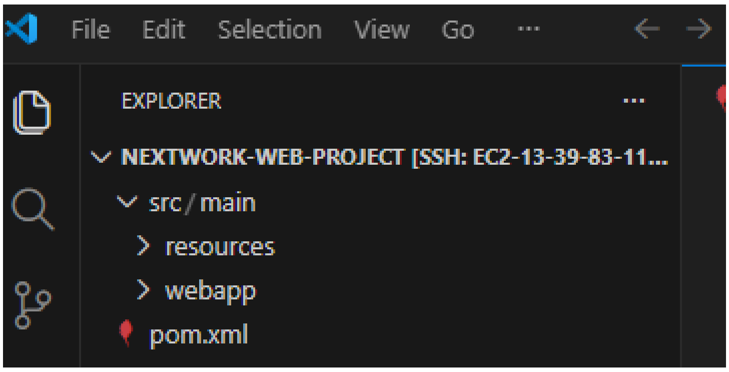
Enter /home/ec2-user/nextwork-web-project.

Press OK.

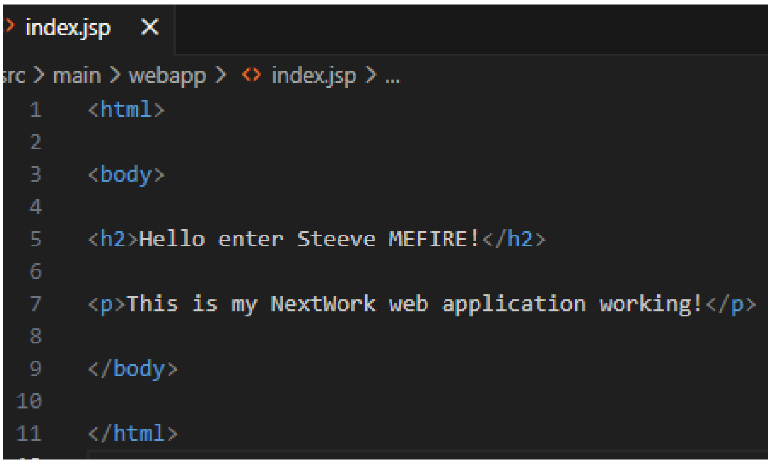


/home/ec2-user/nextwork-web-project/

Here we got our web App generate with Maven


File Edit Selection View Go ···
EXPLORER ···
NEXTWORK-WEB-PROJECT [SSH: EC2-13-39-83-11...
src / main
resources
webapp
pom.xml

Go inside index.jsp and write that code



```
<html>

<body>

<h2>Hello enter Steeve MEFIRE!</h2>

<p>This is my NextWork web application working!</p>

</body>

</html>
```

you've just learnt how to set up a web app on an EC2 instance AND connect it with VS Code, one of the most popular and practical IDEs out there.