

## Step-by-step guide to set up a Python environment on an EC2 instance

```
ubuntu@ip-172-31-44-13: ~/$ apt install python3.12-venv
E: Could not open lock file /var/lib/dpkg/lock-frontent - open (13: Permission denied)
E: Unable to acquire the dpkg frontend lock (/var/lib/dpkg/lock-frontent), are you root?
ubuntu@ip-172-31-44-13: ~/$ sudo apt install python3.12-venv
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  python3-pip-whl python3-setuptools-whl
The following NEW packages will be installed:
  python3-pip-whl python3-setuptools-whl python3.12-venv
0 upgraded, 3 newly installed, 0 to remove and 77 not upgraded.
Need to get 2423 kB of archives.
After this operation, 2770 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 python3-pip-whl all 24.0+dfsg-1ubuntu1 [15 kB]
Get:2 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 python3-setuptools-whl all 68.1.2-2ubuntu1 [15 kB]
Get:3 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 python3.12-venv amd64 3.12.3-1 [5676 B]
Fetched 2423 kB in 0s (43.5 MB/s)
Selecting previously unselected package python3-pip-whl.
(Reading database ... 75141 files and directories currently installed.)
Preparing to unpack .../python3-pip-whl_24.0+dfsg-1ubuntu1_all.deb ...
Unpacking python3-pip-whl (24.0+dfsg-1ubuntu1) ...
Selecting previously unselected package python3-setuptools-whl.
Preparing to unpack .../python3-setuptools-whl_68.1.2-2ubuntu1_all.deb ...
Unpacking python3-setuptools-whl (68.1.2-2ubuntu1) ...
Selecting previously unselected package python3.12-venv.
Preparing to unpack .../python3.12-venv_3.12.3-1_amd64.deb ...
```

- **Update Package List:** `sudo apt update`
- **Install python3-pip:** `sudo apt install python3-pip`
- **Install Python Virtual Environment:** `sudo apt install python3.12-venv`
- **Create a Virtual Environment:** `python3 -m venv mlflow_env`
- **Activate the Virtual Environment:** `source mlflow_env/bin/activate`
- **Create a Directory for the Project:** `mkdir mlflow`
- **Change to the Project Directory:** `cd mlflow`
- **Install MLflow:** `pip install mlflow`
- **Install AWS CLI:** `pip install awscli`
- **Install Boto3:** `pip install boto3`
- **Install Setuptools:** `pip install setuptools`
- **List Installed Packages:** `pip list`

Instance summary for i-058409d84e188c901 (Mlflow-Tracking) Info

Connect

Instance state ▼

Actions ▼

Updated 1 minute ago

<div>Instance ID</div> <div> i-058409d84e188c901 (Mlflow-Tracking)</div>	<div>Public IPv4 address</div> <div> 15-206-116-222   <a href="#">open address</a> </div>	<div>Private IPv4 addresses</div> <div> 172-17-0-1</div>
<div>IPv6 address</div> <div>—</div>	<div>Instance state</div> <div> Running</div>	<div>Public IPv4 DNS</div> <div> ec2-15-206-116-222.ap-south-1.compute.amazonaws.com   <a href="#">open address</a> </div>
<div>Hostname type</div> <div>IP name  ec2-15-206-116-222.ap-south-1.compute.internal</div>	<div>Private IP DNS name (IPv4 only)</div> <div> ip-172-17-0-1.ap-south-1.compute.internal</div>	
<div>Answer private resource DNS name</div> <div>IPv4 (A)</div>	<div>Instance type</div> <div>t2.micro</div>	<div>Elastic IP addresses</div> <div>—</div>

aws configure

```
mlflow server -h 0.0.0.0 --backend-store-uri sqlite:///mlflow.db --default-artifact-root s3://mlflow-project-bkt
```

# Code Commit

Developer Tools > CodeCommit > Repositories > ml-classification

ml-classification

Clone URL ▲  
Clone HTTPS  
Clone SSH  
Clone HTTPS (GRC)

▼ Connection steps

HTTPS SSH HTTPS (GRC)

Step 1: Prerequisites

You must use a Git client that supports Git version 1.7.9 or later to connect to an AWS CodeCommit repository. If you do not have a Git client, you can install one from [Git downloads page](#).

You must have an AWS CodeCommit managed policy attached to your IAM user, belong to a CodeStar project team, or have the equivalent permissions. [Learn how to create and configure an IAM user for accessing AWS CodeCommit.](#) | [Learn how to add team members to an AWS CodeStar Project.](#)

Step 2: Git credentials

## AWS CodeCommit credentials

Identity and Access Management (IAM)

Search IAM

Dashboard

▼ Access management

User groups

Users

Roles

Policies

Identity providers

Account settings

▼ Access reports

Access Analyzer

AWS account ID

Canonical user ID

AWS IAM credentials

AWS CodeCommit credentials

Amazon Keyspaces credentials

SSH public keys for AWS CodeCommit (0)

Actions

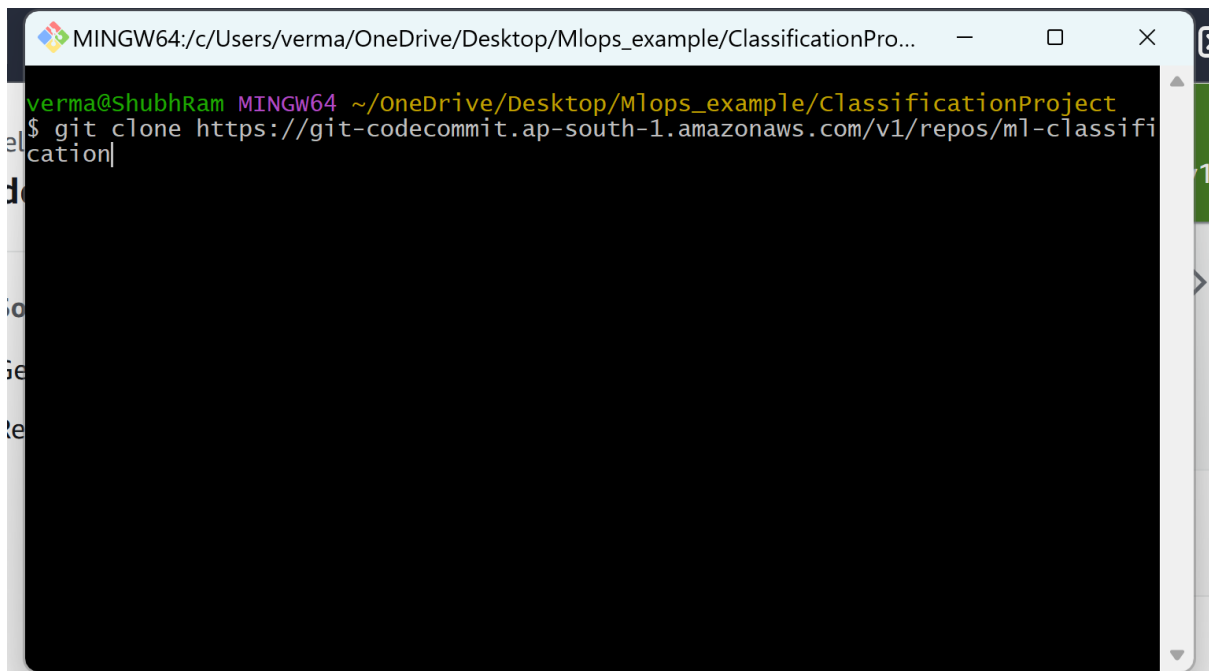
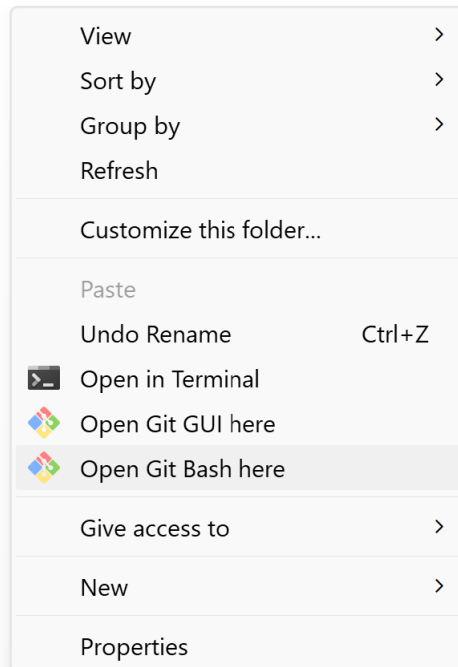
Upload SSH public key

User SSH public keys to authenticate access to AWS CodeCommit repositories. You can have a maximum of five SSH public keys (active or inactive) at a time. [Learn more](#)

SSH Key ID	Uploaded	Status
No SSH public keys		
Upload SSH public key		

## Git bash

This



## Git Credential Manager



## Git Credential Manager

Enter your credentials for 'https://git-codecommit.ap-south-1.amazonaws.com/'

Continue