

## **Placement Empowerment Program**

### ***Cloud Computing and DevOps Centre***

**Set Up a Virtual Machine in the Cloud**  
**Create a free-tier AWS account. Launch a**  
**virtual machine and SSH into it.**

Name : Oviya G

Department : IT

# Introduction and Overview

In this POC, we will learn how to Set up a virtual machine in the cloud is essential for running applications, hosting websites, or learning cloud computing. AWS (Amazon Web Services) provides a **Free Tier**, allowing users to launch and use an EC2 (Elastic Compute Cloud) instance at no cost for the first 12 months. This project will guide you through **creating an AWS account, launching a virtual machine (EC2 instance), and connecting to it via SSH.**

## Objective

### What You Will Learn

- How to create an AWS Free Tier account
- How to launch an EC2 instance (Virtual Machine) using AWS
- How to connect to the instance via SSH
- Basic management of an EC2 instance

### Requirements

- A valid email address and phone number (for AWS signup)
- A computer with an SSH client (Linux/macOS terminal or Windows PowerShell/PuTTY)
- A stable internet connection

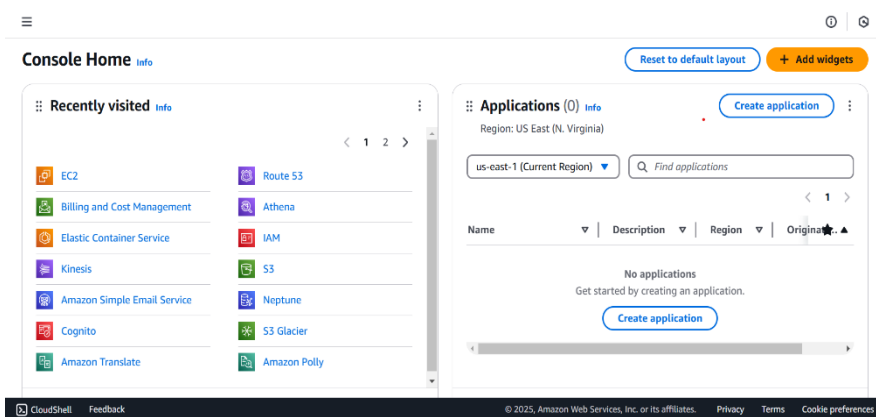
## Step-by-Step Overview

Step1:

### Create an AWS Free Tier Account

1. Go to [AWS Free Tier](#).

2. Click "**Create an AWS Account**".
3. Fill in the required details (email, password, account name).
4. Provide your **billing information** (Credit/Debit card is required for identity verification, but you won't be charged for Free Tier usage).
5. Verify your **phone number**.
6. Choose a **support plan** (select "Basic" for free).
7. Log in to the **AWS Management Console**.



## Step 2 :

### Launch an EC2 Instance (Virtual Machine)

#### 1. Open the EC2 Dashboard:

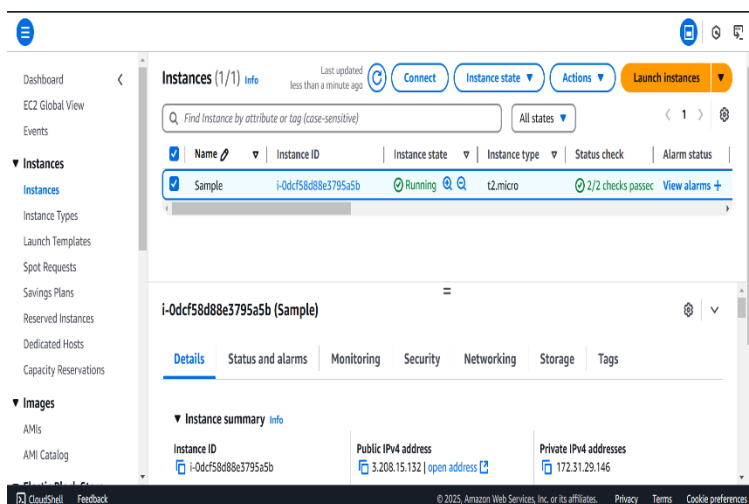
- In AWS Console, search for **EC2** and open the **EC2 Dashboard**.

#### 2. Click "Launch Instance".

#### 3. Configure the EC2 Instance:

- **Name:** Choose a name (e.g., "MyEC2Instance").
- **Amazon Machine Image (AMI):** Select **Amazon Linux 2023** or **Ubuntu 22.04** (both are Free Tier eligible).
- **Instance Type:** Choose **t2.micro** (1 vCPU, 1GB RAM - Free Tier eligible).
- **Key Pair:**
  - Click **Create new key pair**.

- Name it (e.g., "my-key").
  - Select **.pem format** and **Download Key Pair** (store it securely).
  - **Security Group:**
    - Allow **SSH (port 22)**.
    - Restrict access to **Your IP** for security.
  - **Storage:** Keep default (8GB SSD).
4. Click **Launch Instance** and wait for initialization.



5.

## Step 3:

### Connect to the EC2 Instance via SSH

#### For Linux/macOS Users

1. Open a terminal.
2. Navigate to the directory where the .pem key is stored:  
`cd /path/to/key`
3. Set the correct permissions for the key file:  
`chmod 400 my-key.pem`
4. Connect to the instance (replace your-ec2-ip with the actual IP):
5. `ssh -i my-key.pem ec2-user@your-ec2-ip # Amazon Linux`  
`ssh -i my-key.pem ubuntu@your-ec2-ip # Ubuntu`

#### For Windows Users (PowerShell)

1. Open **PowerShell** as Administrator.

2. Navigate to the .pem file location:

```
cd C:\Users\YourUser\Downloads
```

3. Connect using SSH:

```
ssh -i my-key.pem ec2-user@your-ec2-ip
```

## For Windows Users (Using PuTTY)

1. Convert .pem to .ppk using **PuTTYgen**:

- Open **PuTTYgen** → Click **Load** → Select my-key.pem → Click **Save Private Key**.

2. Connect using **PuTTY**:

- Open **PuTTY**.
- Enter **Host Name**: your-ec2-ip.
- Under **Connection** → **SSH** → **Auth**, load the .ppk key.
- Click **Open**.

EC2 > Instances > i-0dcf58d88e3795a5b > Connect to instance

### Connect to instance Info

Connect to your instance i-0dcf58d88e3795a5b (Sample) using any of these options

EC2 Instance Connect | Session Manager | **SSH client** | EC2 serial console

**Instance ID**  
i-0dcf58d88e3795a5b (Sample)

1. Open an SSH client.
2. Locate your private key file. The key used to launch this instance is Sample123.pem
3. Run this command, if necessary, to ensure your key is not publicly viewable.  
`chmod 400 "Sample123.pem"`
4. Connect to your instance using its Public DNS:  
`ec2-3-208-15-132.compute-1.amazonaws.com`

✔ Command copied

```
ssh -i "Sample123.pem" ubuntu@ec2-3-208-15-132.compute-1.amazonaws.com
```

**Note:** In most cases, the guessed username is correct. However, read your AMI usage instructions to check if the AMI owner has changed the default AMI username.

CloudShell Feedback © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

## Step 4: Verify Connection & Manage Instance

- Once connected, run:
- `username -a` # Check Linux version
- `df -h` # Check disk usage

`uptime` # Check system uptime

- To **stop** the instance:

`aws ec2 stop-instances --instance-ids i-xxxxxxxxxxxxxx`

- To **terminate** (delete) the instance:

Go to stop instance and then finally delete it.

```
5-ec2-3-208-15-132.compute-1.amazonaws.com
The authenticity of host 'ec2-3-208-15-132.compute-1.amazonaws.com (3.208.15.132)' can't be established.
ED25519 key fingerprint is SHA256:TaDbdW+uAOY5xLYHJny+Kc04T260gjtF4zjJHOR0siw.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'ec2-3-208-15-132.compute-1.amazonaws.com' (ED25519) to the list of known hosts.
Welcome to Ubuntu 24.04.1 LTS (GNU/Linux 6.8.0-1021-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/pro

System information as of Wed Jan 29 03:54:43 UTC 2025

System load:  0.22          Processes:           105
Usage of /:   24.9% of 6.71GB Users logged in:        0
Memory usage: 20%          IPv4 address for enX0: 172.31.29.146
Swap usage:   0%

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

The list of available updates is more than a week old.
To check for new updates run: sudo apt update

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.
```

# Conclusion

- Created an AWS Free Tier account
- Launched an EC2 instance
- Connected to the instance using SSH