 

**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: Nanthini R Department: ADS



**Introduction**

Cloud computing allows users to deploy and manage virtual machines (VMs) without investing in expensive hardware. Amazon Web Services (AWS) provides a Free Tier that enables users to create and run virtual machines at no cost for the first 12 months.

This guide will walk you through creating an AWS Free Tier account, launching a virtual machine (EC2 instance), and connecting to it via SSH from a Windows system. By the end, you will have a working VM in the cloud that you can use for hosting applications, testing environments, or learning cloud computing.

### ****Overview****

### The process of setting up an AWS virtual machine involves several steps:

### Creating an AWS Free Tier account – Signing up for AWS and accessing the Free Tier benefits.

### Launching an EC2 instance – Configuring and deploying a virtual machine.

### Generating and using an SSH key – Creating a secure authentication method for accessing the VM.

### Connecting to the VM via SSH – Using tools like PuTTY or Windows Terminal to establish a secure connection.

**Step-by-Step Overview**

**Step 1: Create an AWS Free Tier Account**

Go to the AWS Free Tier page

Click "Create an AWS Account" and enter your email, name, and password.

Choose Personal or Business account type and enter your details.

Enter billing information

Complete phone verification by entering a code sent via SMS.

Select the Free Support Plan and proceed to the AWS Management Console

**Step 2: Launch an EC2 Virtual Machine**

1. Log in to the AWS Management Console
2. In the search bar, type EC2 and click EC2 Dashboard.
3. Click Launch Instance to create a new virtual machine.
4. Enter an Instance Name
5. Under Amazon Machine Image (AMI), select Ubuntu or Amazon Linux 2023 (Free Tier Eligible).
6. Under Instance Type, choose t2.micro
7. In the Key Pair section, click Create a new key pair, name it and download the .pem file. Store this file securely!
8. Under Network Settings, allow SSH traffic (default settings are fine).
9. Click Launch Instance and wait for the instance to be created**.**

**Step 3: Connect to Your Virtual Machine via SSH**

Using PuTTY

Download and install PuTTY

Open PuTTYgen, click Load, and select your my-key.pem file.

1. Click Save private key (it will be saved as a .ppk file).
2. Open PuTTY, enter the public IP address of your instance, and go to Connection → SSH → Auth to load your .ppk file.
3. Click Open and log in with the username ec2-user.

**Outcome**

Deploying a static website using GitHub Pages on Windows provides a simple, efficient, and cost-effective way to host web projects without the need for complex server configurations. By following this process, you have successfully set up a GitHub repository, pushed your static website files, and enabled GitHub Pages for public hosting.

With this setup, any updates you make to your local website can be easily pushed to GitHub, ensuring that your site remains up to date with minimal effort. Additionally, GitHub Pages supports further enhancements such as custom domains, Jekyll integration, and automation with GitHub Actions, allowing you to expand your site's capabilities.