# Placement Empowerment Program

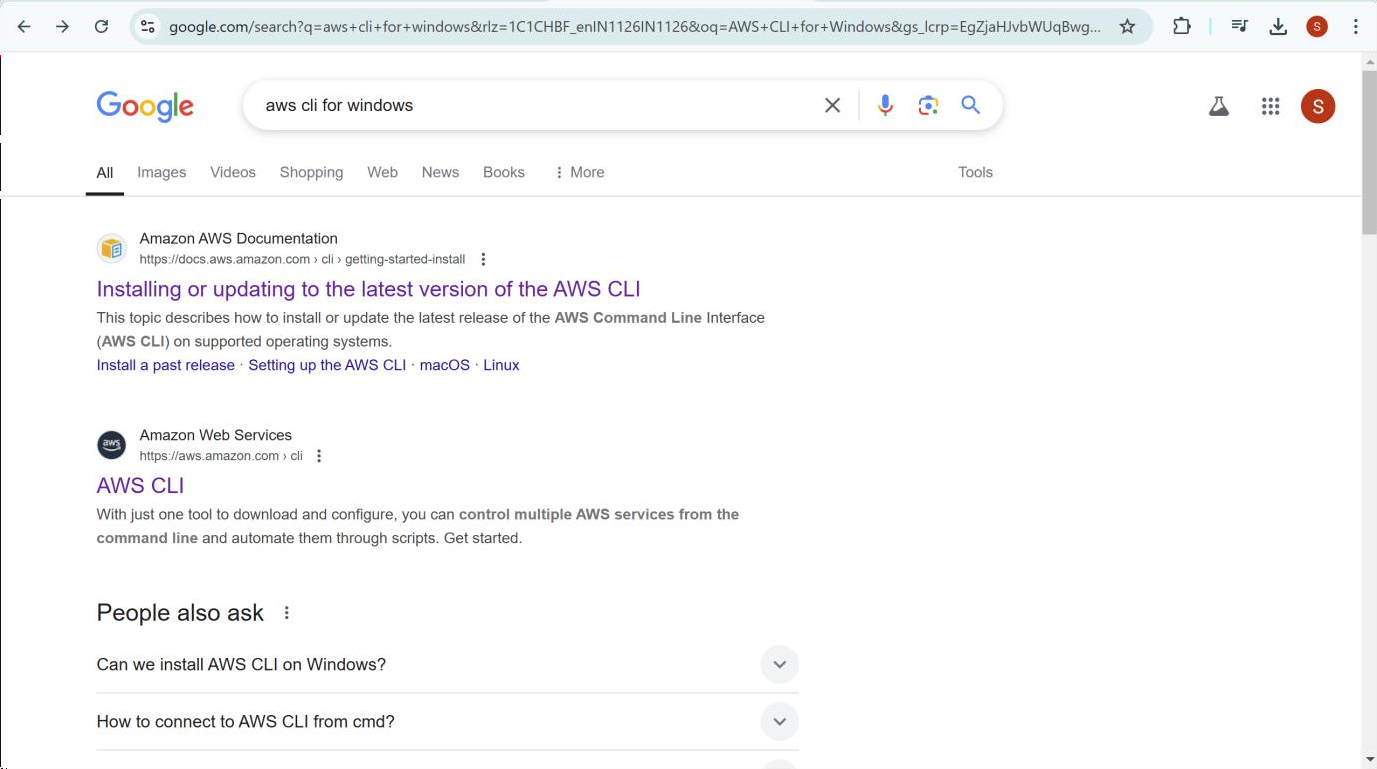
Poc 15: Use Cloud CLI Tools: Install the CLI for your cloud provider (e.g., AWS CLI). Use it to list resources, upload files to storage, and manage VMs.

Name: Subiksha J Department: Ads

# Steps

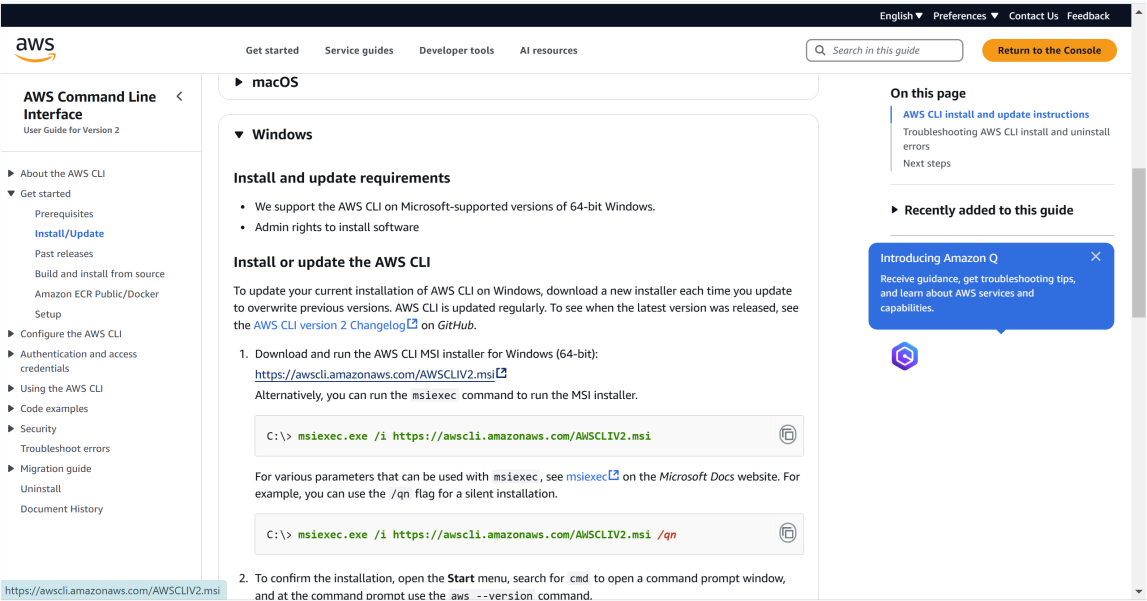
## Step 1:

Search for **aws cli for windows** and click the first link to proceed.



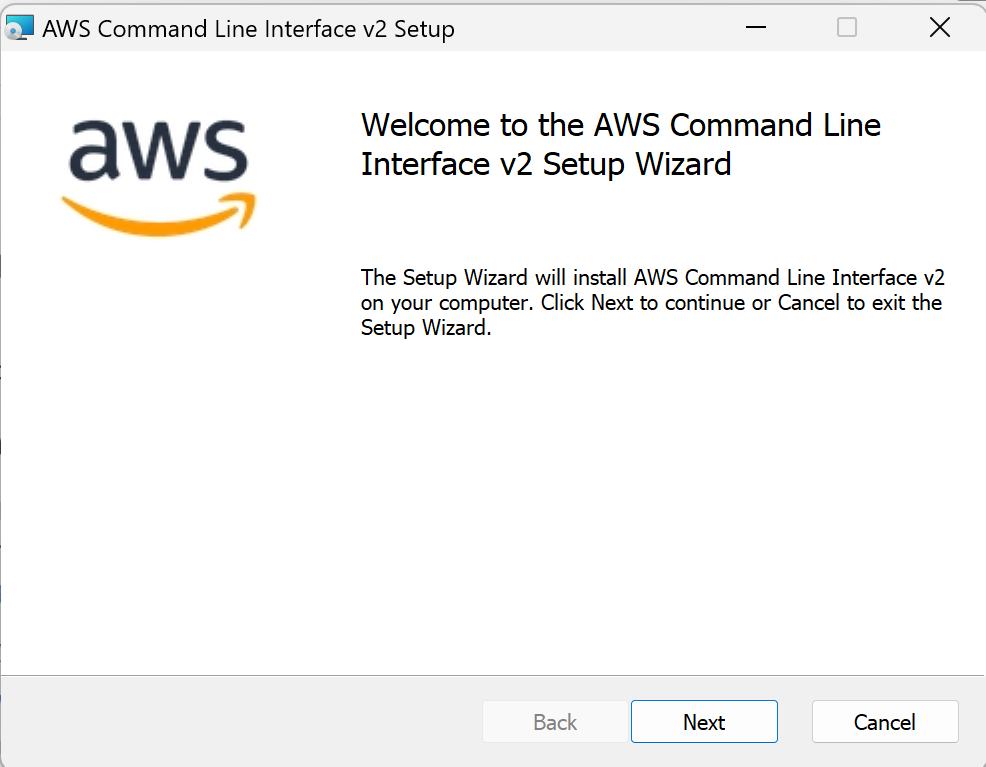
## Step 2:

In that page for windows click the downlad link it will start downloading.



## Step 3:

Follow up the installation wizard.

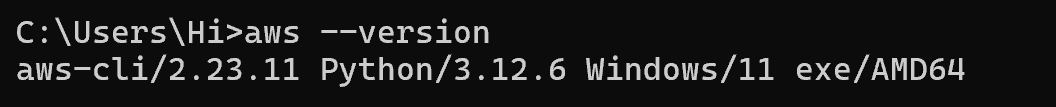


## Step 4:

Open **Command Prompt** and Type the following command and press Enter:

**aws --version**

If this appears, the AWS CLI is installed successfully! ◻



## Step 4:

**Configure the AWS CLI**

Now that the AWS CLI is installed, we need to configure it with your AWS account credentials so that you can interact with AWS services like S3 and EC2.

1. **Open Command Prompt**
2. **Run the Configuration Command**:

In the command prompt, type the following command and press Enter: aws configure

1. **Enter Your AWS Credentials**: This will prompt you for four pieces of information:

**AWS Access Key ID**:

You need to get this from your AWS account.

**How to Get It**:

* 1. Go to the [AWS IAM Console](https://console.aws.amazon.com/iam/).
  2. Click on **Users** on the left-hand menu.
  3. Click on your username (or create a new user if needed).

-> Under the **Security Credentials** tab, find the **Access Keys** section.

-> Click **Create New Access Key**. It will give you an **Access Key ID** and **Secret Access Key**. Keep this information safe!

-> Copy the **Access Key ID** and paste it when prompted.

* 1. **AWS Secret Access Key**:

This is the **Secret Access Key** that is shown when you create a new access key.

Paste this when prompted in the CLI.

* 1. **Default Region Name**:

You can enter a region code like us-east-1

* 1. **Default Output Format**: Choose json

## Step 5:

**List AWS Resources Using the AWS CLI**

S3 (Simple Storage Service) is where you store files in AWS. To list your S3 buckets, follow these steps:

1. List S3 Buckets**:**

Run the following command in Command Prompt

**aws s3 ls**

1. Expected Result**:**

If you have any S3 buckets, they will appear as a list.

If you don't have any, you might see an empty output like this:

A new bucket will be listed here when created.

This command helps you verify the S3 storage you currently have.

## Step 6:

Launch an Ec2 instance in the console and Run the following command:

**aws ec2 describe-instances --query "Reservations[].Instances[].{ID:InstanceId,State:State.Name,Typ e:InstanceType,Name:Tags[?Key=='Name'].Value | [0]}"**

Expected Result**:**

This will display details about your EC2 instances. You’ll see something like:

## Step 7:

**Upload Files to S3 Using AWS CLI**

If you don’t already have an S3 bucket, create a new one.

Run this command to upload a file (replace file.txt with the path to the file you want to upload, and my-unique-bucket-name with your bucket name). Note : The whatever the file u want to store it must be in separate folder.

**aws s3 cp C:\path\to\your\file.txt s3://my-unique-bucket-name/ -- recursive**

## Step 8:

To check if the file is uploaded successfully, you can list the contents of your S3 bucket:

**aws s3 ls s3://my-unique-bucket-name/**

## Step 9:

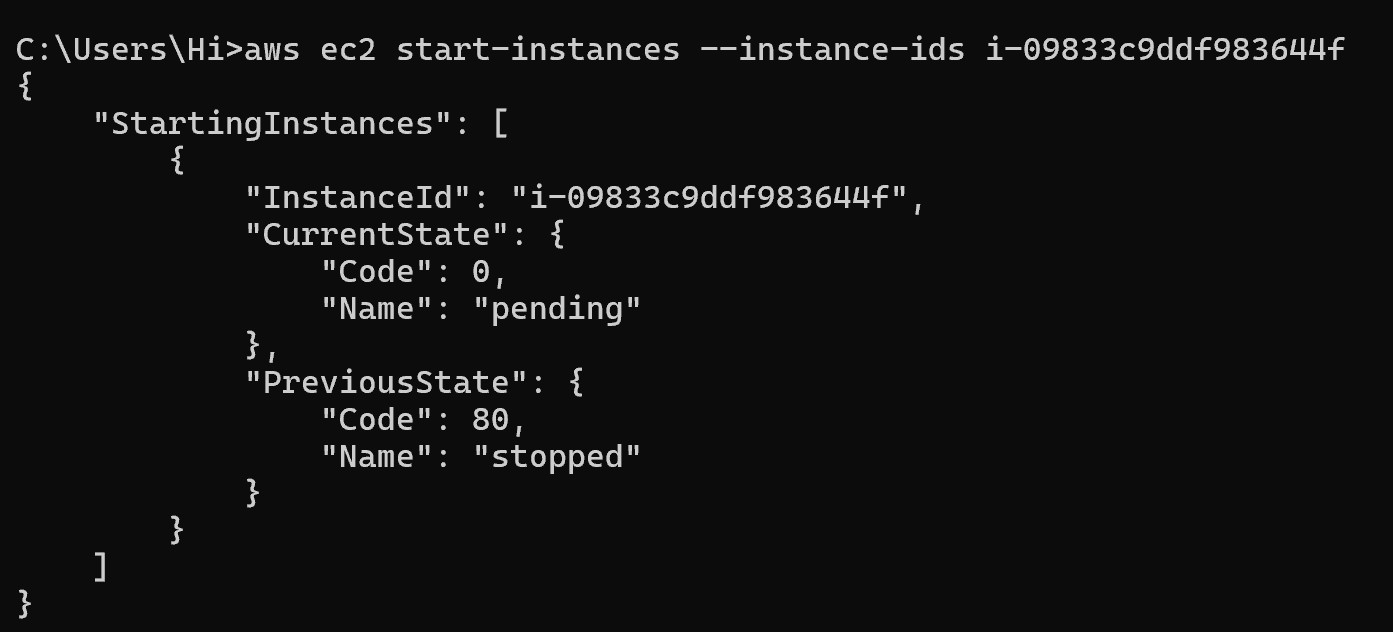
Run the following command to list your EC2 instances, including their ID, state, type, and name (if tagged with "Name"):

**aws ec2 describe-instances --query "Reservations[].Instances[].{ID:InstanceId,State:State.Name,Typ e:InstanceType,Name:Tags[?Key=='Name'].Value | [0]}"**

## Step 10:

Now Stop the Ec2 instance you have created . If you have any stopped EC2 instances and want to start them, use this command. Replace <instance-id> with the ID of the instance you want to start (you can get it from the previous command's output):

**aws ec2 start-instances --instance-ids <instance-id>**



## Step 11:

You can verify the state of your EC2 instance (whether it’s running or stopped) by using the following command:

**aws ec2 describe-instances --instance-ids <instance-id> --query "Reservations[].Instances[].State.Name"**

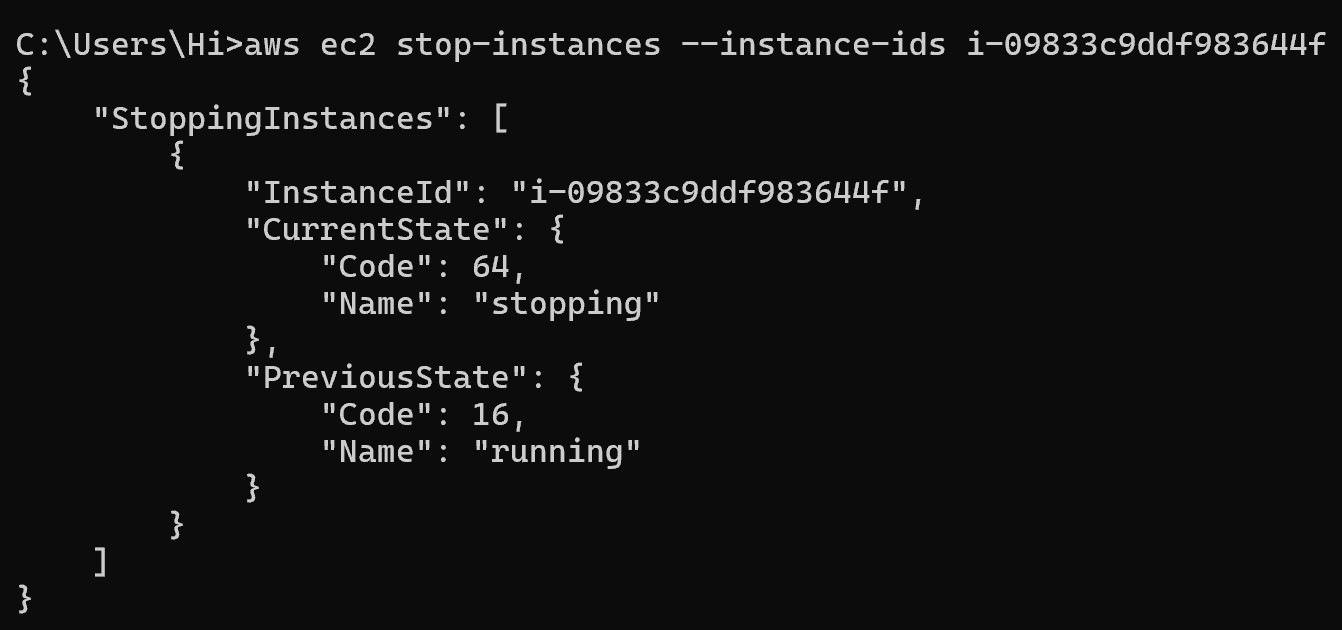


## Step 12:

To stop an EC2 instance, use this command. Again, replace

<instance-id> with the actual instance ID:

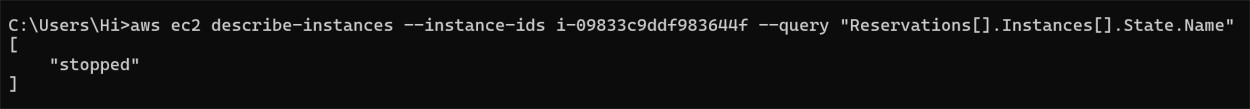
**aws ec2 stop-instances --instance-ids <instance-id>**



## Step 13:

Again verify the state of your EC2 instance (whether it’s running or stopped) by using the following command:

**aws ec2 describe-instances --instance-ids <instance-id> --query "Reservations[].Instances[].State.Name"**



## Step 14:

**Clean Up Resources**

* **Delete a Single File**:

**aws s3 rm s3://my-unique-bucket-name/file.pdf**

Replace file.pdf with the actual name of the file you want to delete.

* **Delete All Files in a Bucket** (if you want to delete everything):

**aws s3 rm s3://my-unique-bucket-name/--recursive**

## Step 15:

Once the files are deleted, you can delete the bucket itself.

* **Delete an S3 Bucket**:

**aws s3 rb s3://my-unique-bucket-name/ --force**

Then verify it by :

**aws s3 ls**

It returns nothing so it means there is no bucket (deleted)

## Step 16:

If you no longer need the EC2 instances, you can terminate them to avoid ongoing charges.

**aws ec2 terminate-instances --instance-ids <instance-id>**

Then verify it by :

**aws ec2 describe-instances --query "Reservations[].Instances[].{ID:InstanceId,State:State.Name}"**

