

## **Placement Empowerment Program**

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

**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: N.V ANANTHA NARAYANAN      Department: CSE**

# Introduction and Overview

Cloud CLI tools, such as AWS CLI, allow users to interact with cloud services directly from the terminal, enabling automation and efficient resource management. This task involves installing the AWS CLI, configuring it with AWS credentials, and using it to perform basic operations like listing resources, uploading files to S3, and managing EC2 instances. CLI tools offer a faster and scriptable alternative to the AWS Management Console, improving productivity. By completing this task, you 'll gain hands-on experience in cloud automation and resource control using command-line commands.

## Objective

The goal of this project is to:

1. Learn Cloud CLI Basics – Install and configure AWS CLI to interact with cloud resources using command-line commands.
2. Manage Cloud Resources – Use AWS CLI to list cloud resources, upload files to S3, and manage EC2 instances efficiently.
3. Enhance Automation Skills – Gain hands-on experience in automating cloud tasks, improving efficiency over manual AWS Management Console operations.

## Importance of Cloud CLI

**Hands-on Learning & Efficiency** – Cloud CLI provides direct interaction with cloud services, enabling faster and more efficient management compared to the web console.

**Automation & Scripting** – It allows users to automate repetitive tasks, such as resource provisioning and deployments, improving productivity.

**Remote Cloud Management** – With CLI tools, users can manage cloud resources from any terminal, making it ideal for DevOps, remote administration, and large-scale cloud operations.

# Step-by-Step Overview

## Step 1:

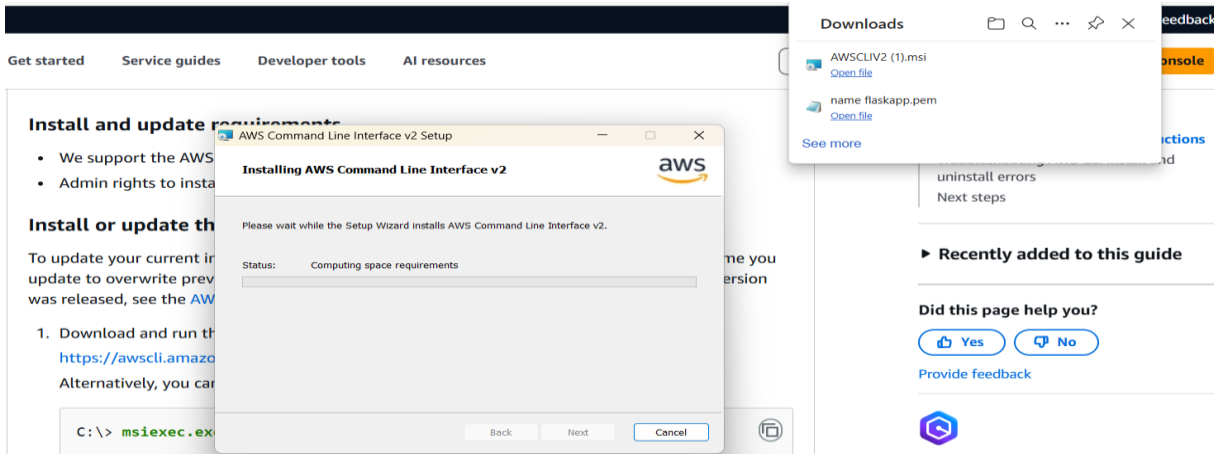
Search for "AWS CLI Installer for Windows" on Google and click the first link to access the official website.

The screenshot shows a Bing search results page for the query "aws cli installer for windows". The top result is from AWS Documentation, titled "Installing or updating to the latest version of the AWS CLI". Below the title, it says "Download and run the AWS CLI MSI installer for Windows (64-bit): https://awscli.amazonaws.com/AWSCLIV2.msi. Alternatively, you can run the msixec ...". There are two sub-links: "Uninstall The AWS CLI Versio..." and "AWS Command Line Interface". To the right, there is a "Related searches" box with links like "download aws cli for window", "aws cli exe download", "aws cli commands cheat sheet", "aws cli zip download", "aws download cli", "install aws in windows", "aws download for windows", and "aws configure download". Below the main result, there is a "Videos of AWS CLI Installer For Windows" section with a video thumbnail titled "How to install and configure the AWS CLI on Windows 10".

## Step 2:

Click on the "Install/Update" option located on the left-hand side of the Apache Lounge website. Select the link regarding your OS, Install by using the link provided else use the *msiexec* command

The screenshot shows the AWS Command Line Interface User Guide for Version 2. The left sidebar has a navigation menu with options like "About the AWS CLI", "Get started", "Prerequisites", "Install/Update", "Past releases", "Build and install from source", "Amazon ECR Public/Docker", "Setup", "Configure the AWS CLI", "Authentication and access credentials", "Using the AWS CLI", "Code examples", "Security", "Troubleshoot errors", "Migration guide", "Uninstall", and "Document History". The main content area is titled "Windows" and "Install and update requirements". It lists two requirements: "We support the AWS CLI on Microsoft-supported versions of 64-bit Windows." and "Admin rights to install software". Below this, it says "Install or update the AWS CLI" and provides instructions: "To update your current installation of AWS CLI on Windows, download a new installer each time you update to overwrite previous versions. AWS CLI is updated regularly. To see when the latest version was released, see the AWS CLI version 2 Changelog on GitHub." It then lists step 1: "Download and run the AWS CLI MSI installer for Windows (64-bit): https://awscli.amazonaws.com/AWSCLIV2.msi". It also mentions that alternatively, you can run the `msiexec` command to run the MSI installer. A code block shows the command: `C:\> msiexec.exe /i https://awscli.amazonaws.com/AWSCLIV2.msi`. Below this, it says "For various parameters that can be used with msiexec, see msiexec on the Microsoft Docs website. For example, you can use the /qn flag for a silent installation." Another code block shows the command: `C:\> msiexec.exe /i https://awscli.amazonaws.com/AWSCLIV2.msi /qn`. On the right side, there is a "On this page" section with a link to "AWS CLI install and update instructions" and a "Recently added to this guide" section. At the bottom, there is a "Did this page help you?" section with "Yes" and "No" buttons and a "Provide feedback" link.



## Step 3:

Once installed, verify the installation by opening Command Prompt (cmd) or PowerShell and running **aws --version**

```
PS C:\Users\SUGANTHI CLARET> aws --version
aws-cli/2.23.8 Python/3.12.6 Windows/11 exe/AMD64
```

It should return something like

aws-cli/2.x.x Python/3.x.x Windows/x86\_64

## Step 4:

Before using AWS CLI, you need to configure it with your AWS credentials.

Open Command Prompt and type **aws configure**

It will ask for:

AWS Access Key ID → Get it from AWS IAM > Security Credentials

AWS Secret Access Key → Get it from AWS IAM > Security Credentials

Default region name → Example: us-east-1 (Find yours in AWS Console)

Default output format → Keep it as json or press Enter for default

```
PS C:\Users\SUGANTHI CLARET> aws --version
aws-cli/2.23.8 Python/3.12.6 Windows/11 exe/AMD64
PS C:\Users\SUGANTHI CLARET> aws configure
AWS Access Key ID [*****WU6K]: |
```

aws [Search] [Alt+S] Global Jayanthi claret



IAM > Security credentials > Create access key

Access key created  
This is the only time that the secret access key can be viewed or downloaded. You cannot recover it later. However, you can create a new access key any time.

Step 1 Alternatives to root user access keys  
Step 2 Retrieve access key

### Retrieve access key [Info](#)

**Access key**  
If you lose or forget your secret access key, you cannot retrieve it. Instead, create a new access key and make the old key inactive.

Access key	Secret access key
 [Redacted]	 ***** <a href="#">Show</a>

**Access key best practices**

- Never store your access key in plain text, in a code repository, or in code.
- Disable or delete access key when no longer needed.
- Enable least-privilege permissions.
- Rotate access keys regularly.

For more details about managing access keys, see the [best practices for managing AWS access keys](#).

[Download .csv file](#) [Done](#)

## Step 5:

To see all storage buckets, Type **aws s3 ls** in cmd

To check running EC2 instances **aws ec2 describe-instances** in cmd

```
PS C:\Users\SUGANTHI CLARET> aws s3 ls
2025-01-28 22:49:31 my-storage-bucket-abc
PS C:\Users\SUGANTHI CLARET> |
```

```

PS C:\Users\SUGANTHI CLARET> aws ec2 describe-instances
{
  "Reservations": [
    {
      "ReservationId": "r-0e5c8403c37eb7d9e",
      "OwnerId": "58508058359",
      "Groups": [],
      "Instances": [
        {
          "Architecture": "x86_64",
          "BlockDeviceMappings": [],
          "ClientToken": "289d059a-3abf-4fde-b650-ad71f77665be",
          "EbsOptimized": false,
          "EnaSupport": true,
          "Hypervisor": "xen",
          "NetworkInterfaces": [],
          "RootDeviceName": "/dev/sda1",
          "RootDeviceType": "ebs",
          "SecurityGroups": [],
          "StateReason": {
            "Code": "Client.UserInitiatedShutdown",
            "Message": "Client.UserInitiatedShutdown: User initiated shutdown"
          },
          "Tags": [
            {
              "Key": "Name",
              "Value": "task14"
            }
          ],
          "VirtualizationType": "hvm",

```

## Step 6:

Create an S3 Bucket by typing **aws s3 mb s3://your-unique-bucket-name** in cmd

```

PS C:\Users\SUGANTHI CLARET> aws s3 mb s3://my-storage-bucket-def
make_bucket: my-storage-bucket-def
PS C:\Users\SUGANTHI CLARET> |

```

Upload a file to S3 Bucket by typing **aws s3 cp yourfile.txt s3://your-unique-bucket-name/** in cmd

```

PS C:\Users\SUGANTHI CLARET> aws s3 cp "C:\Users\SUGANTHI CLARET\Downloads\lambda.txt" s3://my-storage-bucket-def
upload: Downloads\lambda.txt to s3://my-storage-bucket-def/lambda.txt
PS C:\Users\SUGANTHI CLARET> |

```

**Amazon S3**

General purpose buckets  
Directory buckets  
Table buckets  
Access Grants  
Access Points  
Object Lambda Access Points  
Multi-Region Access Points  
Batch Operations  
IAM Access Analyzer for S3

Block Public Access settings for this account

**Storage Lens**  
Dashboards  
Storage Lens groups  
AWS Organizations settings

Feature spotlight 10

**Account snapshot - updated every 24 hours** All AWS Regions  
Storage lens provides visibility into storage usage and activity trends. Metrics don't include directory buckets. [Learn more](#)

[View Storage Lens dashboard](#)

**General purpose buckets** Directory buckets

**General purpose buckets (2)** Info All AWS Regions  
Buckets are containers for data stored in S3.

Find buckets by name

	Name	AWS Region	IAM Access Analyzer	Creation date
<input type="radio"/>	<a href="#">my-storage-bucket-abc</a>	Asia Pacific (Mumbai) ap-south-1	<a href="#">View analyzer for ap-south-1</a>	January 28, 2025, 22:41:11 (UTC+05:30)
<input type="radio"/>	<a href="#">my-storage-bucket-def</a>	Asia Pacific (Mumbai) ap-south-1	<a href="#">View analyzer for ap-south-1</a>	January 29, 2025, 18:23:23 (UTC+05:30)

CloudShell Feedback

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

**Amazon S3**

General purpose buckets  
Directory buckets  
Table buckets  
Access Grants  
Access Points  
Object Lambda Access Points  
Multi-Region Access Points  
Batch Operations  
IAM Access Analyzer for S3

Block Public Access settings for this account

**my-storage-bucket-def** Info

**Objects** Properties Permissions Metrics Management Access Points

**Objects (1)** Copy S3 URI Copy URL Download Open Delete

Objects are the fundamental entities stored in Amazon S3. You can use [Amazon S3 inventory](#) to get a list of all objects in your bucket. [Learn more](#)

Find objects by prefix

	Name	Type	Last modified	Size
<input type="checkbox"/>	<a href="#">lambda.txt</a>	txt	January 29, 2025, 18:25:47 (UTC+05:30)	

## Step 7:

To Start an EC2 Instance, Type **aws ec2 start-instances --instance-ids <INSTANCE\_ID>** in cmd

Replace <INSTANCE\_ID> with your actual instance ID

```
PS C:\Users\SUGANTHI CLARET> aws ec2 start-instances --instance-ids i-0a3c2170890ffb3e1
{
  "StartingInstances": [
    {
      "InstanceId": "i-0a3c2170890ffb3e1",
      "CurrentState": {
        "Code": 0,
        "Name": "pending"
      },
      "PreviousState": {
        "Code": 80,
        "Name": "stopped"
      }
    }
  ]
}
```

	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4
<input type="checkbox"/>	task14	i-03b6d28a8978f04bc	Terminated	t2.micro	-	View alarms +	ap-south-1b	-
<input checked="" type="checkbox"/>	instance1	i-0a3c2170890ffb3e1	Running	t2.micro	Initializing	View alarms +	ap-south-1b	ec2-15-206

## Expected Outcome

By completing this POC, you will:

1. **Successful Installation & Configuration** – AWS CLI will be installed and configured with the correct credentials, allowing seamless interaction with AWS services.
2. **Ability to List Cloud Resources** – You will be able to list AWS resources such as S3 buckets, EC2 instances, and IAM users using CLI commands.
3. **File Management in S3** – You will gain hands-on experience in uploading, downloading, and managing files in Amazon S3 using the CLI.
4. **EC2 Instance Control** – You will learn how to start, stop, and reboot EC2 instances from the command line, improving your cloud management skills.
5. **Improved Automation Skills** – By using CLI instead of the AWS Console, you will develop automation capabilities essential for DevOps and cloud computing.