**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: Kirubalini S I Department: IT



**Introduction and Overview**

Cloud CLI tools are command-line interfaces that let you manage cloud services (like AWS, Azure, or GCP) directly from your terminal. They're essential for automating tasks, quickly managing resources, and integrating cloud management into scripts and DevOps workflows. By installing the appropriate CLI for your cloud provider and learning its commands, you can efficiently interact with and control your cloud environment without needing a GUI.

**Objective**

The objective of using Cloud CLI tools is to efficiently manage and interact with cloud resources directly from the command line. This involves installing the appropriate CLI for your cloud provider (AWS, Azure, GCP, etc.), and then using it to perform tasks like listing available resources (e.g., storage buckets, virtual machines), uploading and downloading files to cloud storage, and managing virtual machines (creating, starting, stopping, deleting, configuring). Ultimately, the goal is to automate cloud management, improve efficiency, and integrate cloud operations into scripts and DevOps workflows.

**Importance of Cloud CLI**

Using Cloud CLI tools like the AWS CLI is super important for anyone working with cloud services. Here's why:

**Efficiency and Automation:** Imagine you need to create 100 virtual machines. Doing that through a web console would be incredibly time-consuming. With the CLI, you can write a script to automate the entire process in minutes. This saves you tons of time and reduces the risk of human error.

**DevOps and Infrastructure as Code:** In modern cloud environments, managing infrastructure as code (IaC) is crucial. CLI tools are fundamental to IaC, allowing you to define your infrastructure in code and manage it through version control, just like software. This enables you to easily reproduce environments, track changes, and automate deployments.

**Flexibility and Control:** CLIs offer a level of control and flexibility that GUIs often can't match. You can fine-tune configurations, access less common features, and perform complex operations that might be cumbersome or impossible through a GUI.

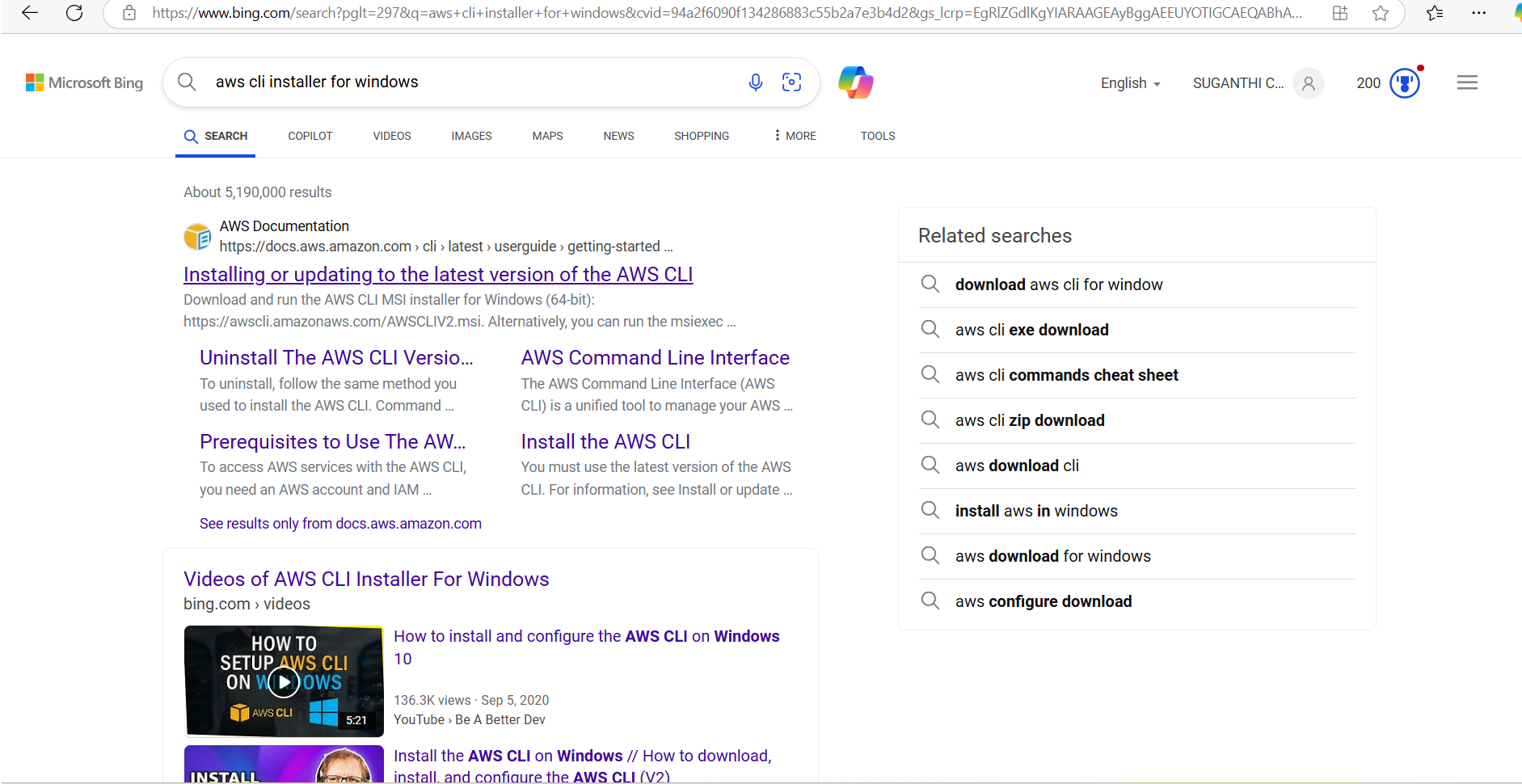
**Scripting and Integration:** CLIs let you integrate cloud management into your existing scripts and workflows. This is essential for automating tasks like backups, monitoring, and deployments.

**Remote Management:** You can manage your cloud resources from anywhere with an internet connection, even if you don't have access to a GUI. This is particularly useful for server administration and troubleshooting.

**Step-by-Step Overview**

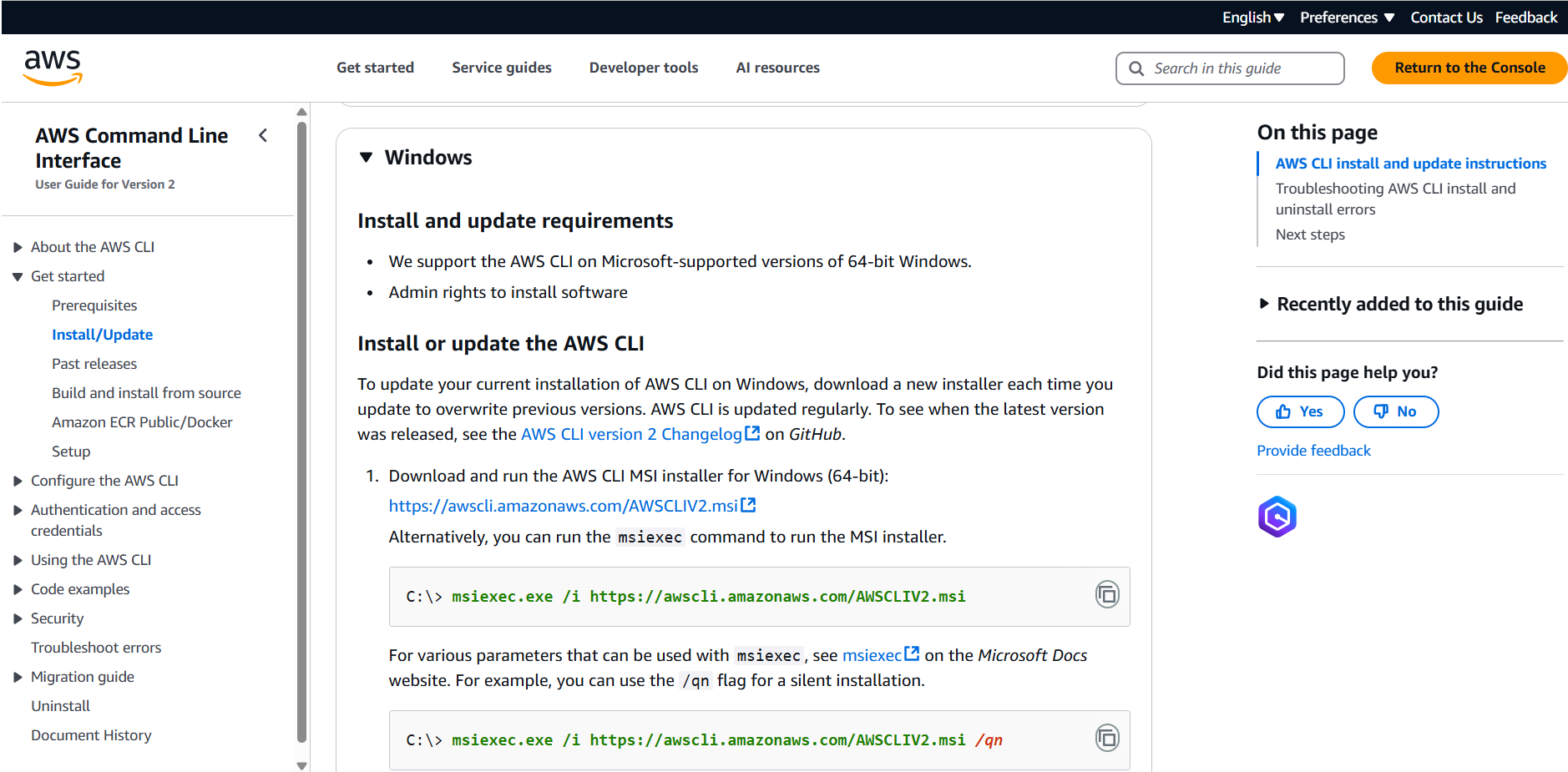
**Step1:**

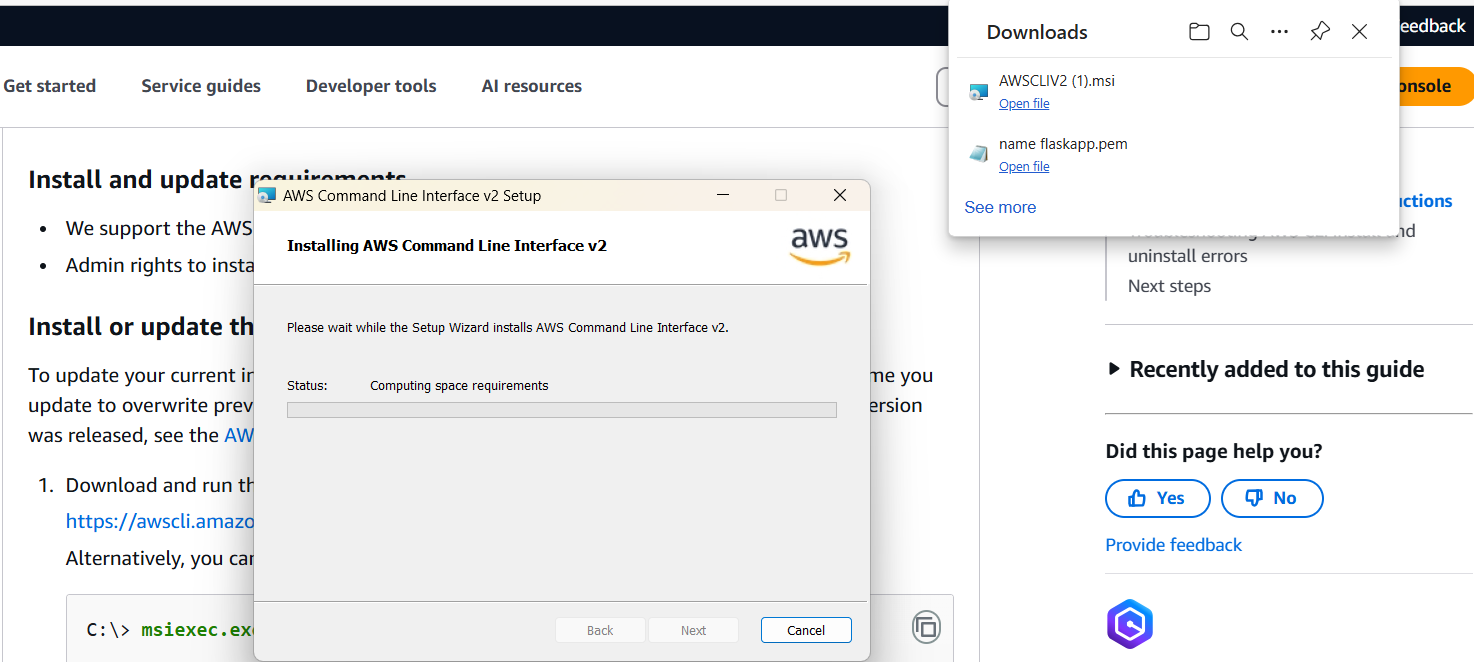
Search for "AWS CLI Installer for Windows" on Google and click the first link to access the official website.



**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





**Step 3:**

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



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**

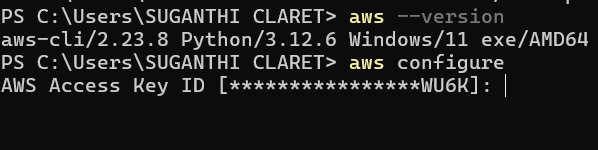
It will ask for:

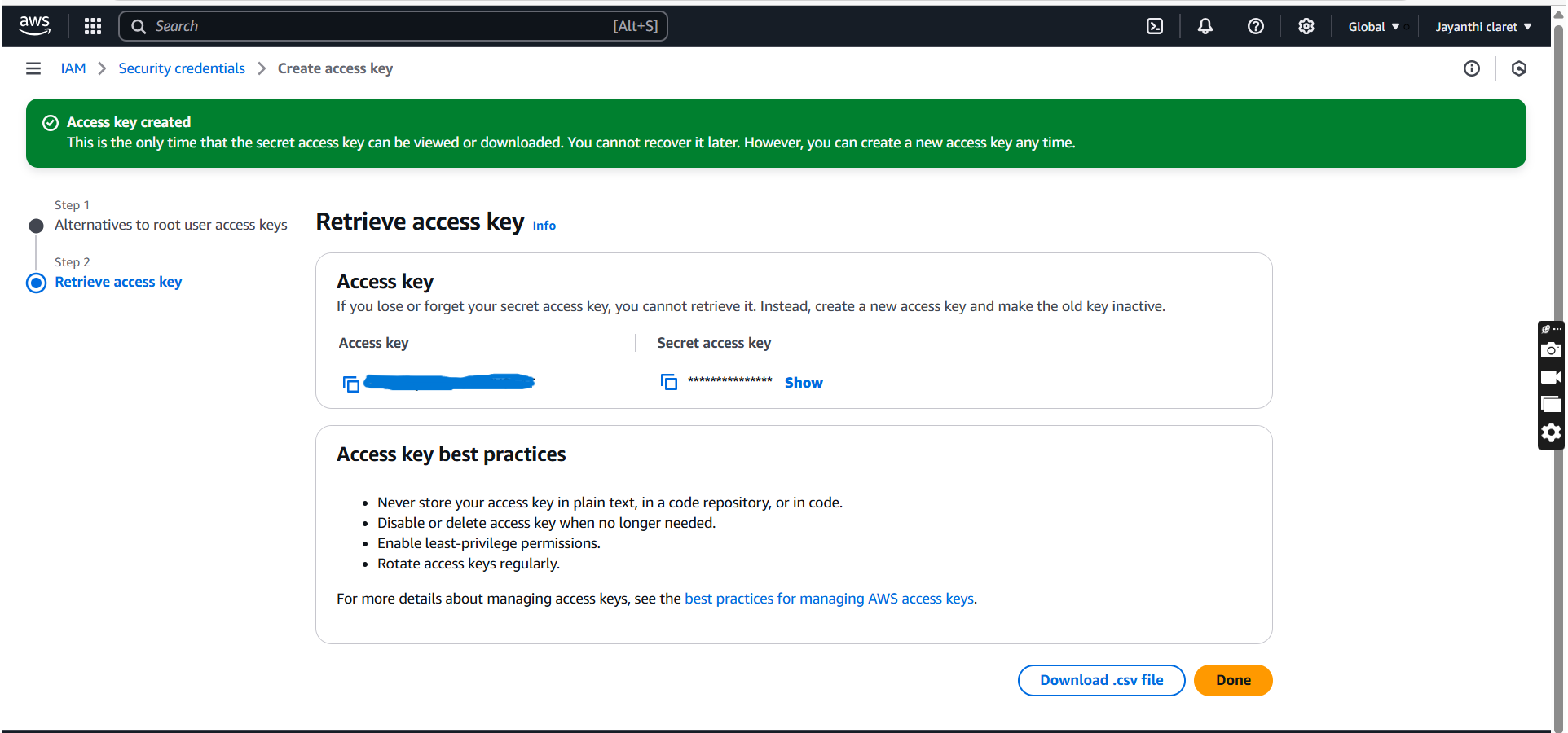
AWS Access Key ID → Get it from AWS IAM > Security Credentials

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

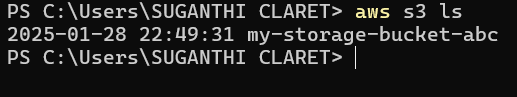


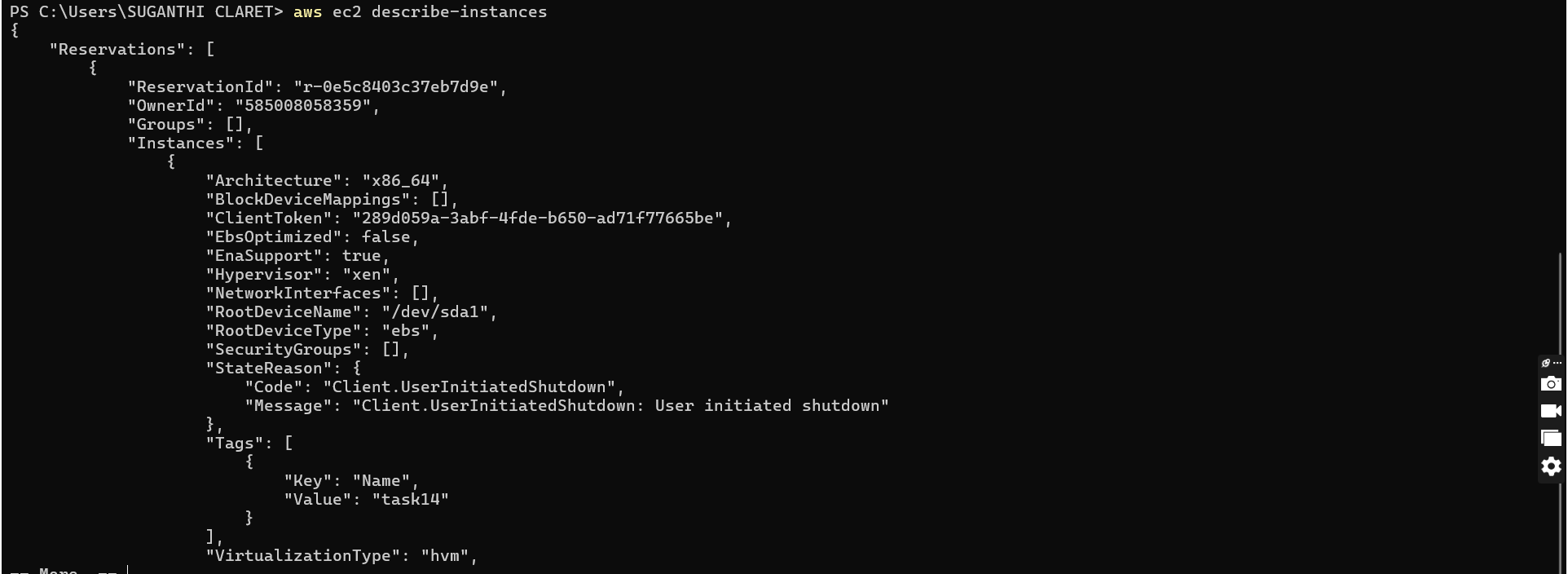


**Step 5:**

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

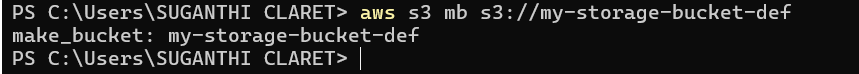
To check running EC2 instances **aws ec2 describe-instances** in cmd



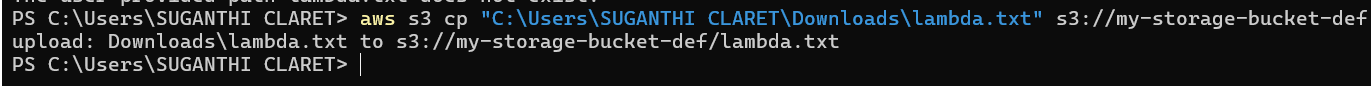


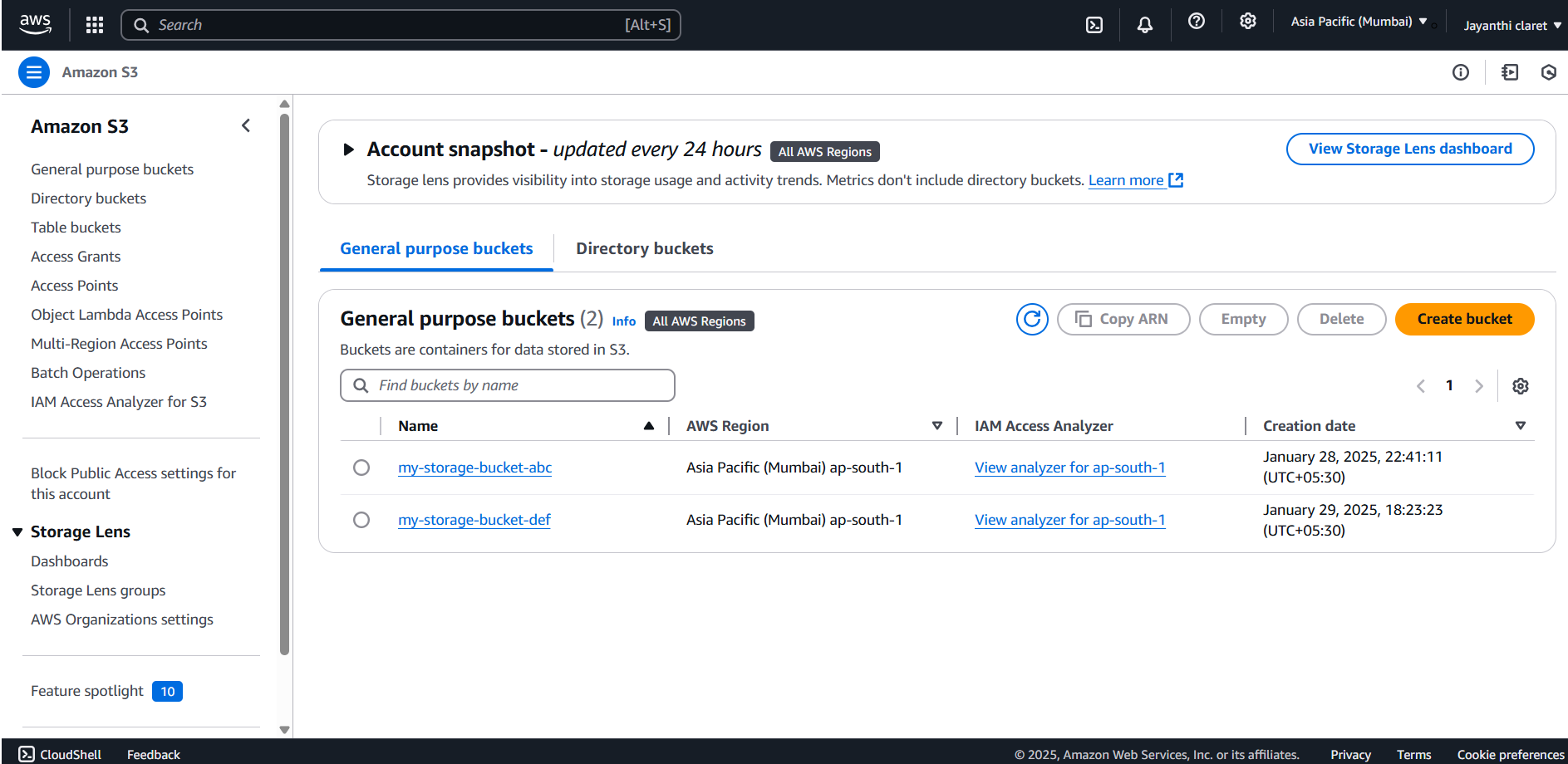
**Step 6:**

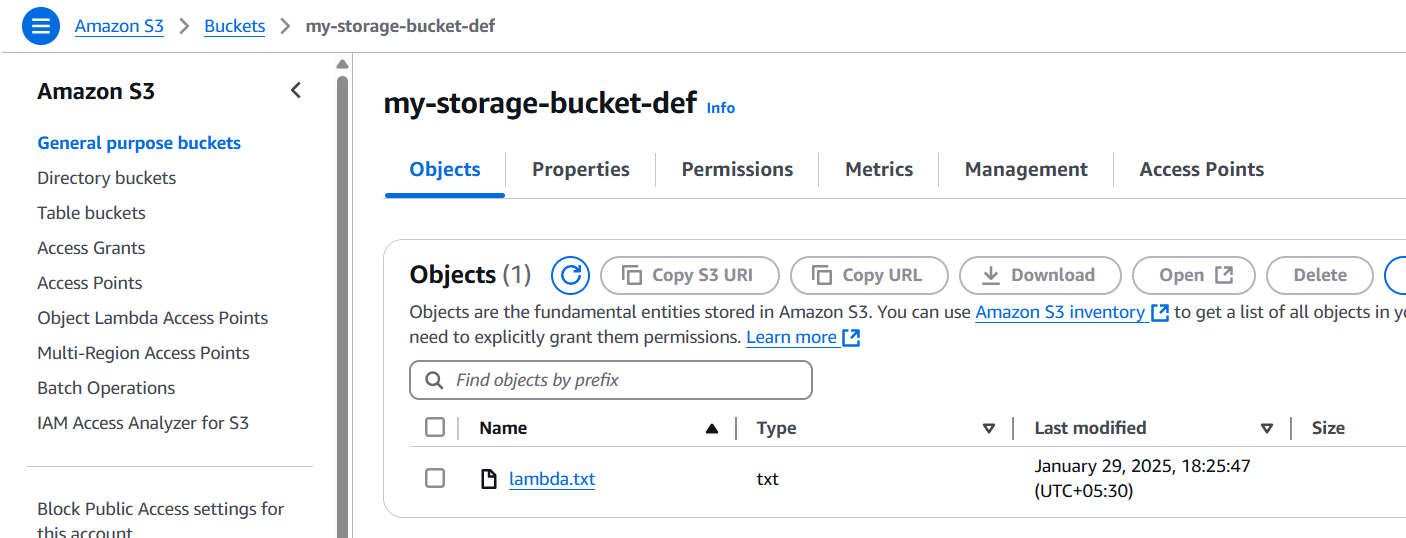
Create an S3 Bucket by typing **aws s3 mb s3://your-unique-bucket-name** in cmd



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



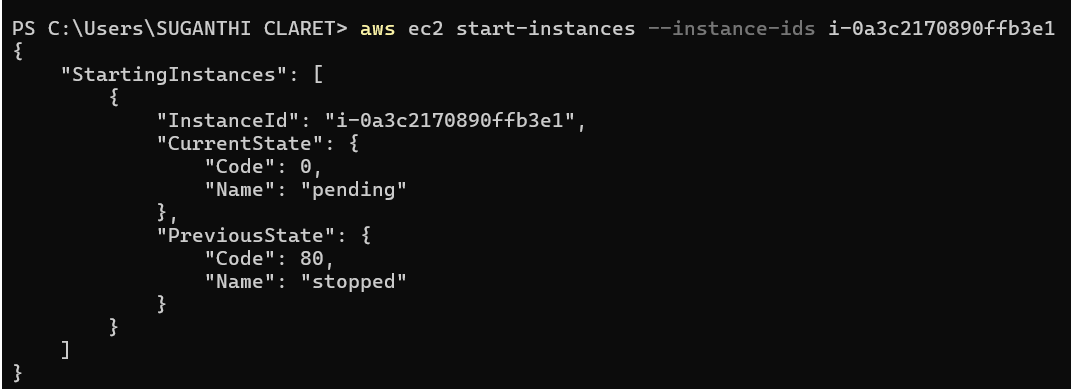


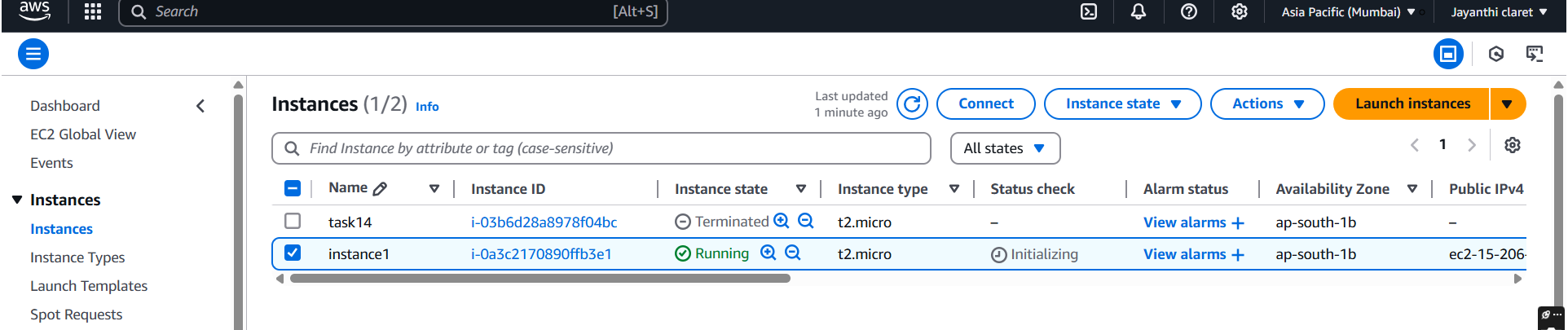


**Step 7:**

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

Replace <INSTANCE\_ID> with your actual instance ID





**Expected Outcome**

Using Cloud CLI tools like the AWS CLI leads to several positive outcomes:

**Increased Efficiency:** Tasks are completed faster, especially repetitive ones, through automation and direct command execution. You avoid navigating complex GUIs.

**Improved Automation:** Scripts and workflows can be created to automate complex cloud management processes, reducing manual effort and errors.

**Greater Flexibility:** Cloud resources can be managed from any environment with a terminal, enhancing accessibility and control.

**Enhanced DevOps Practices:** CLI tools are crucial for Infrastructure as Code (IaC), enabling version control and automated deployment of cloud infrastructure.

**Simplified Management:** Listing resources, uploading files, and managing VMs becomes more straightforward and consistent through standardized commands.

**Reduced Costs (potentially):** Automation and efficient resource management can lead to optimized cloud spending.

**Faster Development Cycles:** Developers can quickly provision and manage resources for testing and development, accelerating the development process.