Lab 7 CSE- 322 Cloud Computing

Name: Gaurav Malave

Roll No: 2022BCD0017

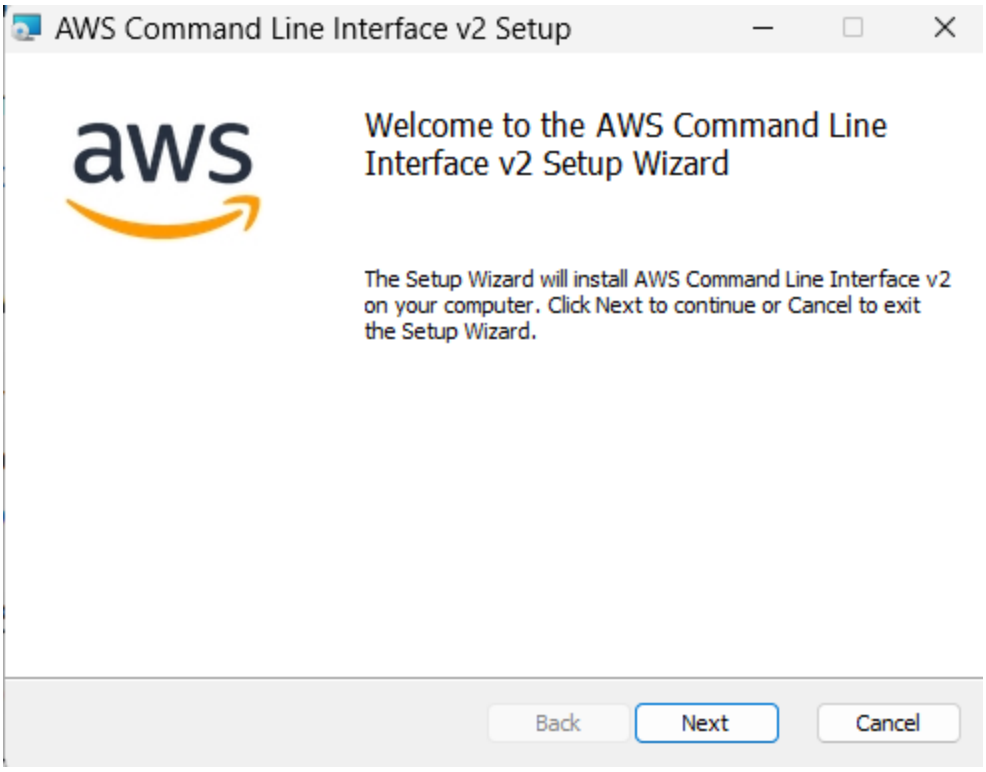
Batch: 2

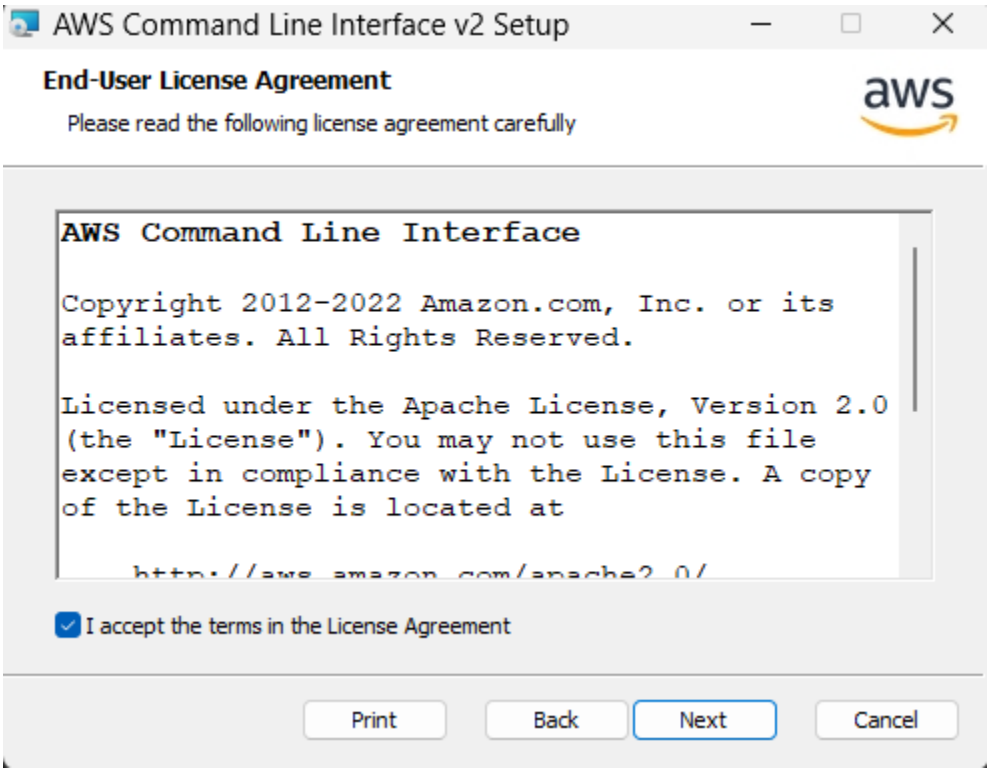
Task 1: Develop a cloud database using AWS DynamoDB on your name.

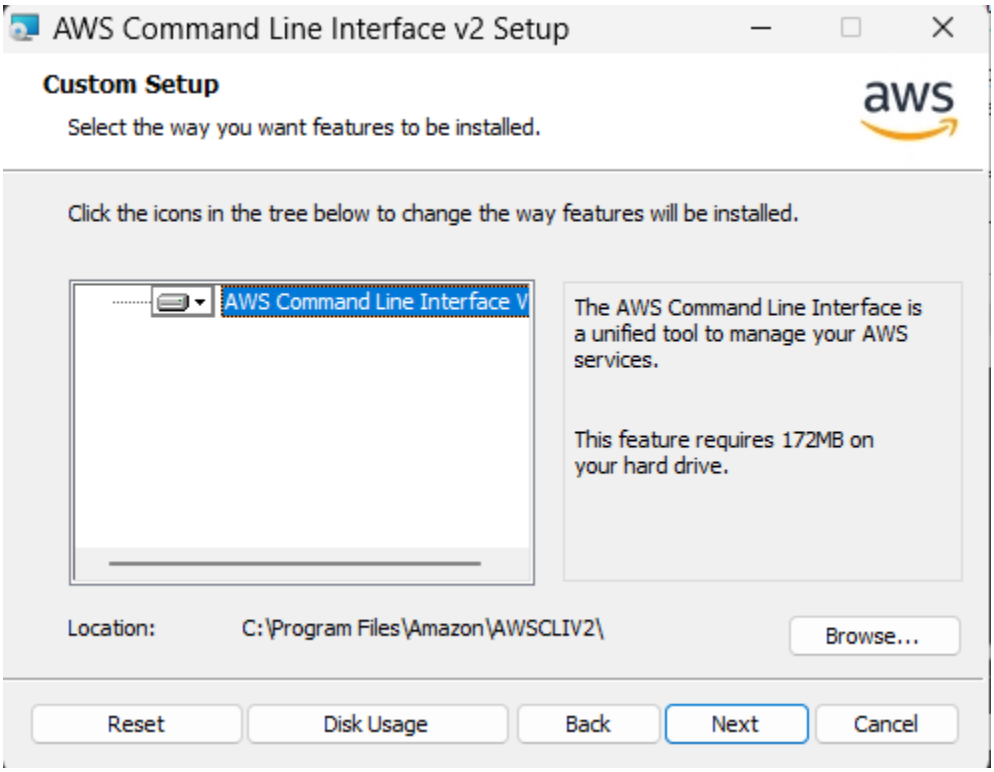
We will create a **DynamoDB table** on AWS with a primary key and populate it with some data.

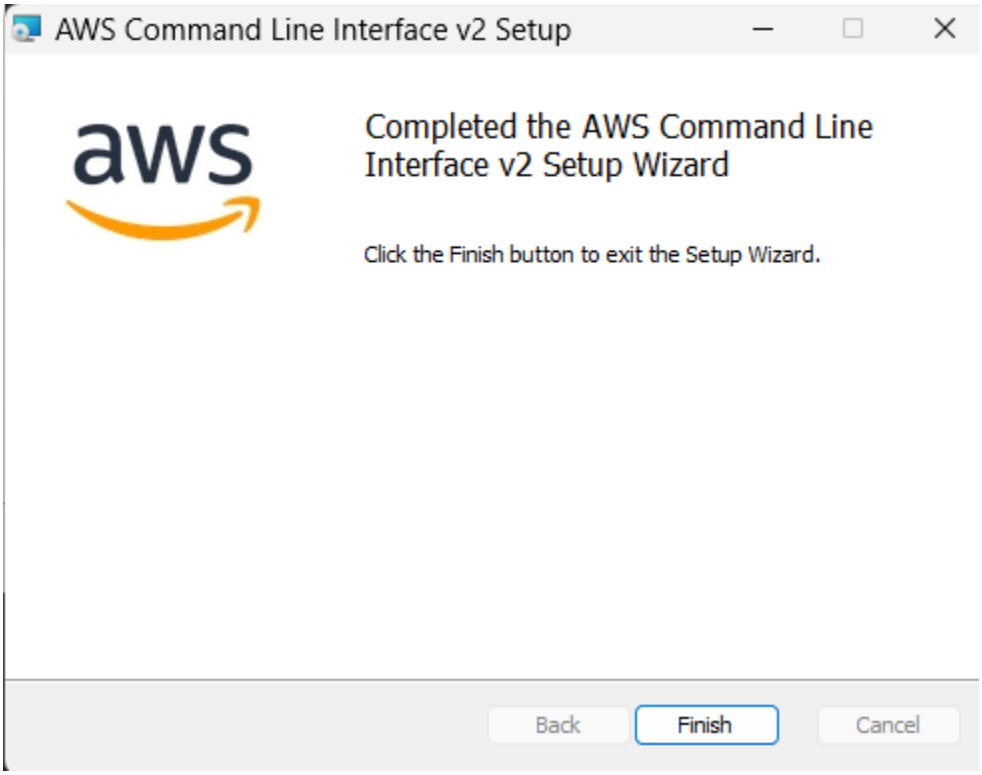
Step 1: Set Up AWS Account and IAM User

1. **Sign in to AWS Console**:
    - Go to AWS Management Console.
    - If you don't have an account, create one.


AWS Command Line Interface v2 Setup
aws
Welcome to the AWS Command Line Interface v2 Setup Wizard
The Setup Wizard will install AWS Command Line Interface v2 on your computer. Click Next to continue or Cancel to exit the Setup Wizard.
Back
Next
Cancel

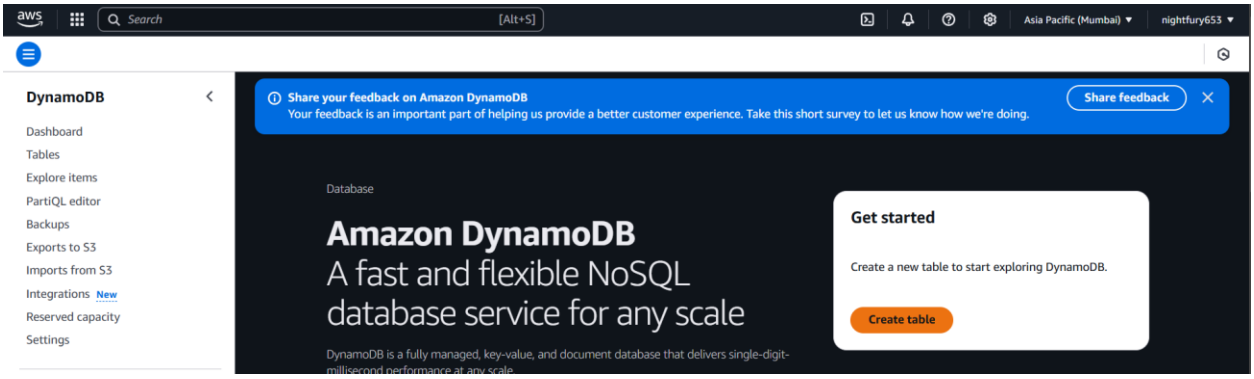

AWS Command Line Interface v2 Setup
End-User License Agreement
Please read the following license agreement carefully
aws
AWS Command Line Interface
Copyright 2012-2022 Amazon.com, Inc. or its affiliates. All Rights Reserved.
Licensed under the Apache License, Version 2.0 (the "License"). You may not use this file except in compliance with the License. A copy of the License is located at
I accept the terms in the License Agreement
Print
Back
Next
Cancel


AWS Command Line Interface v2 Setup
Custom Setup
Select the way you want features to be installed.
aws
Click the icons in the tree below to change the way features will be installed.
AWS Command Line Interface V
The AWS Command Line Interface is a unified tool to manage your AWS services.
This feature requires 172MB on your hard drive.
Location:
C:\Program Files\Amazon\AWSCLIV2\
Browse...
Reset
Disk Usage
Back
Next
Cancel


AWS Command Line Interface v2 Setup
aws
Completed the AWS Command Line Interface v2 Setup Wizard
Click the Finish button to exit the Setup Wizard.
Back
Finish
Cancel

Create an IAM User with DynamoDB Permissions:

- Open the **IAM console**: AWS IAM Console.
- Go to **Users** → **Add users**.
- Set the user name as **yourname-dynamodb-user**.
- Enable **"Access key – Programmatic access"**.
- Attach an **AWS managed policy**:
  - Choose **AmazonDynamoDBFullAccess**.

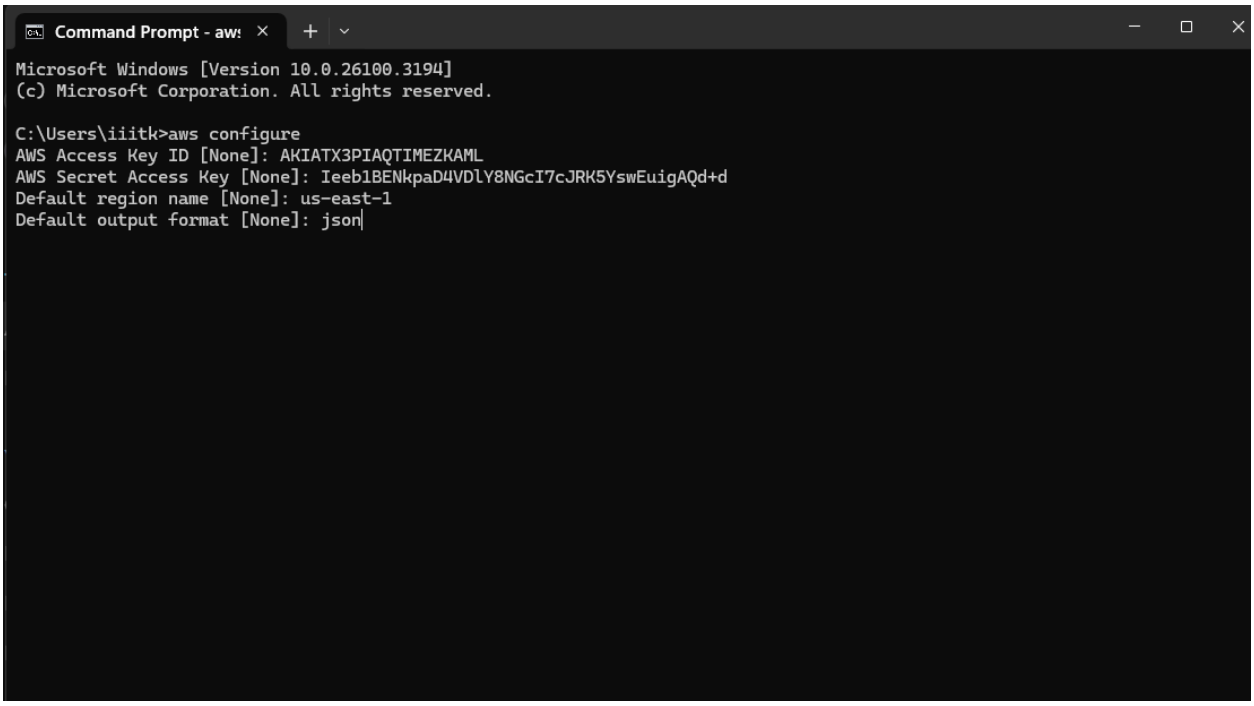


> Enter your **AWS Access Key ID**.

> Enter your **AWS Secret Access Key**.

> Choose a default region (e.g., **us-east-1**).

> Choose output format (default: **json**).



Generate and Download Credentials:

- Copy the **Access Key ID** and **Secret Access Key** (or download the credentials file).
- Store them safely.

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
AKIATX3PIAQTIMEZKAML	********** Show

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

Access keys (1)

Actions | Create access key

Use access keys to send programmatic calls to AWS from the AWS CLI, AWS Tools for PowerShell, AWS SDKs, or direct AWS API calls. You can have a maximum of two access keys (active or inactive) at a time. Learn more

Access key ID	Created on	Access key last used	Region last used	Service last used	Status
AKIATX3PIAQTIMEZKAML		None	N/A	N/A	Active

Table details Info

DynamoDB is a schemaless database that requires only a table name and a primary key when you create the table.

Table name

This will be used to identify your table.

2022BCD0017

Between 3 and 255 characters, containing only letters, numbers, underscores (_), hyphens (-), and periods (.).

Partition key

The partition key is part of the table's primary key. It is a hash value that is used to retrieve items from your table and allocate data across hosts for scalability and availability.

Gaurav Malave | String

1 to 255 characters and case sensitive.

Sort key - *optional*

You can use a sort key as the second part of a table's primary key. The sort key allows you to sort or search among all items sharing the same partition key.

Enter the sort key name | String

1 to 255 characters and case sensitive.

Step 3: Create a DynamoDB Table

```
C:\Users\iiitk>aws dynamodb create-table ^
More?     --table-name Gaurav_MalaveDB ^
More?     --attribute-definitions AttributeName=ID,AttributeType=S ^
More?     --key-schema AttributeName=ID,KeyType=HASH ^
More?     --billing-mode PAY_PER_REQUEST ^
More?     --region us-east-1
{
    "TableDescription": {
        "AttributeDefinitions": [
            {
                "AttributeName": "ID",
                "AttributeType": "S"
            }
        ],
        "TableName": "Gaurav_MalaveDB",
        "KeySchema": [
            {
                "AttributeName": "ID",
                "KeyType": "HASH"
            }
        ],
        "TableStatus": "CREATING",
        "CreationDateTime": "2025-02-25T11:54:26.342000+05:30",
        "ProvisionedThroughput": {
            "NumberOfDecreasesToday": 0,
            "ReadCapacityUnits": 0,
            "WriteCapacityUnits": 0
        },
        "TableSizeBytes": 0,
        "ItemCount": 0,
        "TableArn": "arn:aws:dynamodb:us-east-1:257394476070:table/Gaurav_MalaveDB",
        "TableId": "80c5ffce-8bb0-4fa9-829f-bb2d59d73522",
        "BillingModeSummary": {
            "BillingMode": "PAY_PER_REQUEST"
        },
        "DeletionProtectionEnabled": false
    }
}
```

Verify Table Creation

```
C:\Users\iiitk>aws dynamodb put-item ^
More?     --table-name Gaurav_MalaveDB ^
More?     --item "{""ID"": {""S"": ""1""}, ""Name"": {""S"": ""Gaurav""}, ""Age"": {""N"": ""23""}, ""City"": {""S"": ""Bangalore""}}" ^
More?     --region us-east-1
```

Task 2: Query the database from your local machine using AWS CLI commands.

Step 1: Query Data from Local Machine

1. **Retrieve all items from the table**:

```
C:\Users\iiitk>aws dynamodb scan --table-name Gaurav_MalaveDB --region us-east-1
{
    "Items": [
        {
            "City": {
                "S": "Bangalore"
            },
            "ID": {
                "S": "1"
            },
            "Age": {
                "N": "23"
            },
            "Name": {
                "S": "Gaurav"
            }
        }
    ],
    "Count": 1,
    "ScannedCount": 1,
    "ConsumedCapacity": null
}
```

Query an item by its primary key (ID = 1):

```
C:\Users\iiitk>aws dynamodb get-item ^
More?     --table-name Gaurav_MalaveDB ^
More?     --key "{""ID"": {""S"": ""1""}}" ^
More?     --region us-east-1
{
    "Item": {
        "Skill": {
            "S": "Machine Learning"
        },
        "City": {
            "S": "Hyderabad"
        },
        "ID": {
            "S": "1"
        },
        "Age": {
            "N": "24"
        },
        "Name": {
            "S": "Gaurav"
        }
    }
}
```

Step 2: Update an Item

If you want to **update the Age field**, run

```
C:\Users\iiitk>aws dynamodb update-item ^
More?     --table-name Gaurav_MalaveDB ^
More?     --key "{""ID"": {""S"": ""1""}}" ^
More?     --update-expression "SET Age = :newAge, City = :newCity, Skill = :newSkill" ^
More?     --expression-attribute-values "{"":newAge"": {""N"": ""24""}, "":newCity"": {""S"": ""Hyderabad""}, "":newSkill"": {""S"": ""Machine Learning""}}" ^
More?     --region us-east-1
```

Step 3: Delete an Item

To delete an item where ID=1

```
C:\Users\iiitk>aws dynamodb delete-item ^
More?     --table-name Gaurav_MalaveDB ^
More?     --key "{""ID"": {""S"": ""1""}}" ^
More?     --region us-east-1
```

```
C:\Users\iiitk>aws dynamodb scan --table-name Gaurav_MalaveDB --region us-east-1
{
    "Items": [],
    "Count": 0,
    "ScannedCount": 0,
    "ConsumedCapacity": null
}
```

Step 4: Delete the DynamoDB Table

If you no longer need the table

```
C:\Users\iiitk>aws dynamodb delete-table --table-name Gaurav_MalaveDB --region us-east-1
{
    "TableDescription": {
        "TableName": "Gaurav_MalaveDB",
        "TableStatus": "DELETING",
        "ProvisionedThroughput": {
            "NumberOfDecreasesToday": 0,
            "ReadCapacityUnits": 0,
            "WriteCapacityUnits": 0
        },
        "TableSizeBytes": 0,
        "ItemCount": 0,
        "TableArn": "arn:aws:dynamodb:us-east-1:257394476070:table/Gaurav_MalaveDB",
        "TableId": "80c5ffce-8bb0-4fa9-829f-bb2d59d73522",
        "BillingModeSummary": {
            "BillingMode": "PAY_PER_REQUEST",
            "LastUpdateToPayPerRequestDateTime": "2025-02-25T11:54:26.342000+05:30"
        },
        "DeletionProtectionEnabled": false
    }
}
```

```
C:\Users\iiitk>aws dynamodb describe-table --table-name Gaurav_MalaveDB

An error occurred (ResourceNotFoundException) when calling the DescribeTable operation: Requested resource not found: Table: Gaurav_MalaveDB not found
```

Summary

Step 1 Set up AWS account and IAM user

Step 2 Install & configure AWS CLI

Step 3 Create DynamoDB table (Gaurav_MalaveDB)

Step 4 Insert sample data

Step 5 Query data from local machine

Step 6 Update an item

Step 7 Delete an item

Step 8 Delete the table