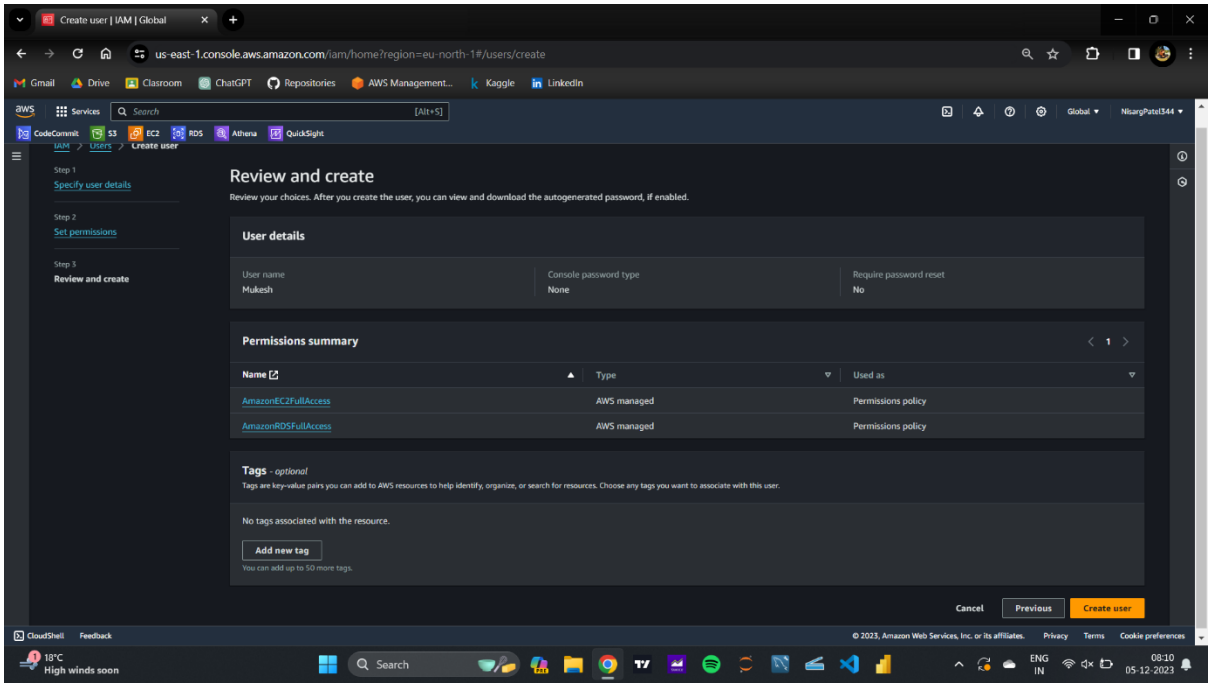
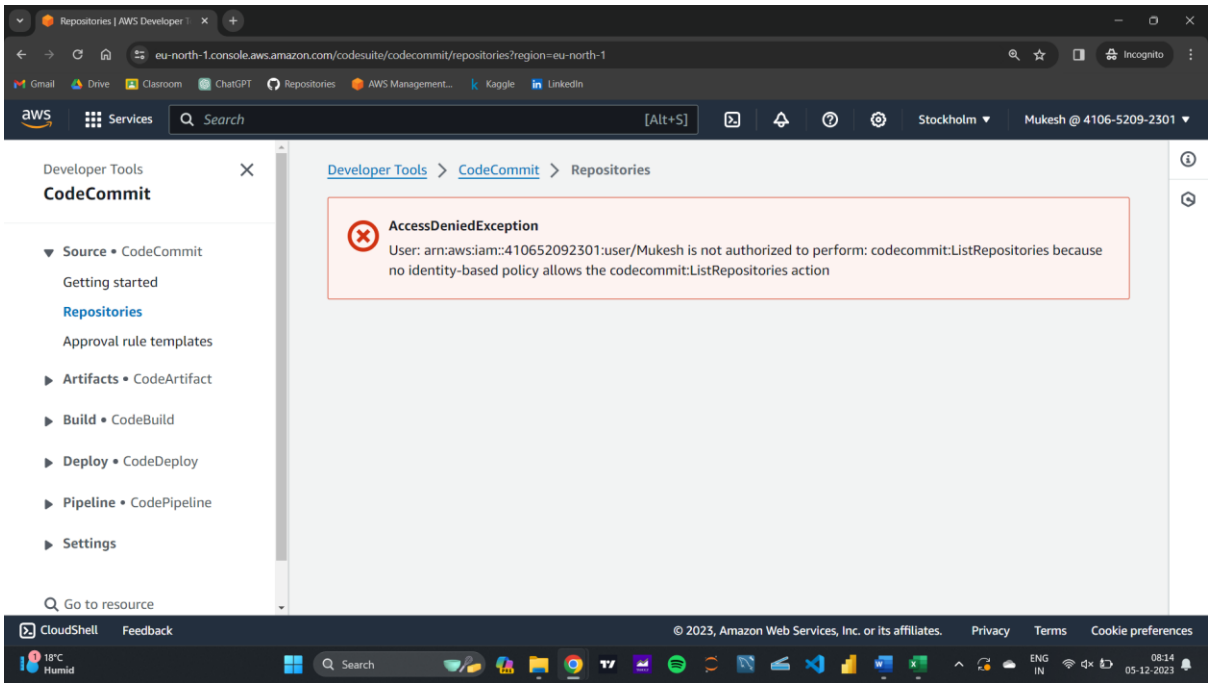
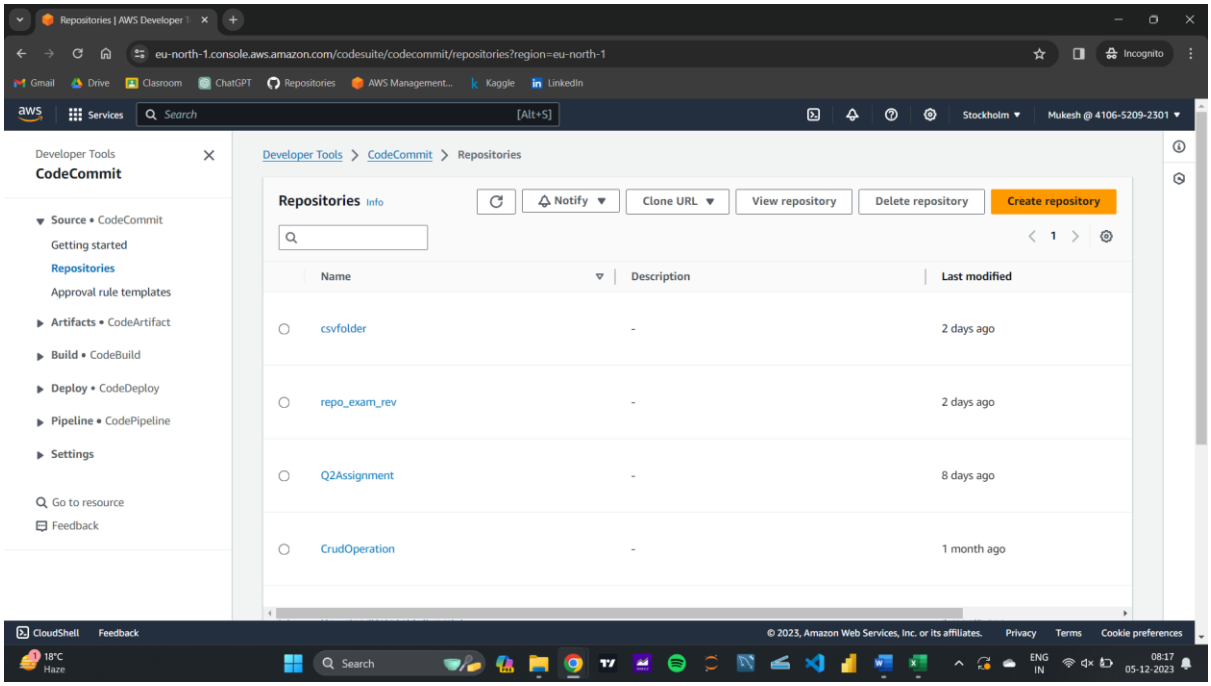
This is the copyright of Patel Nisarg

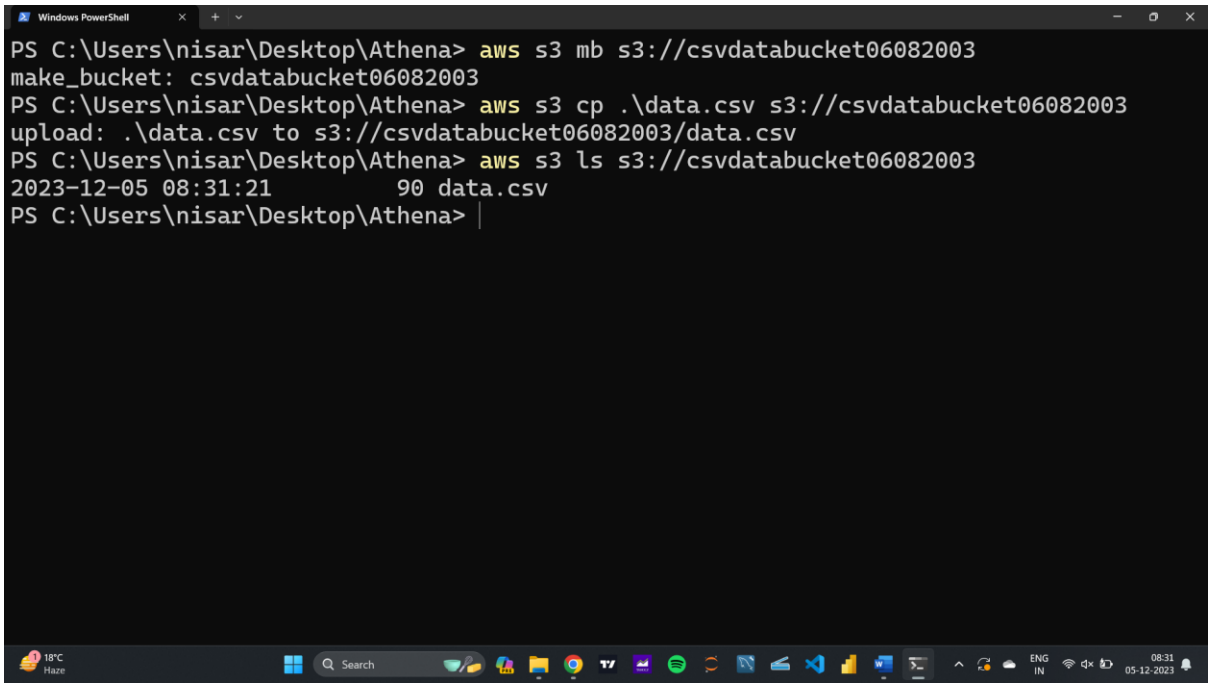
Question 1.A

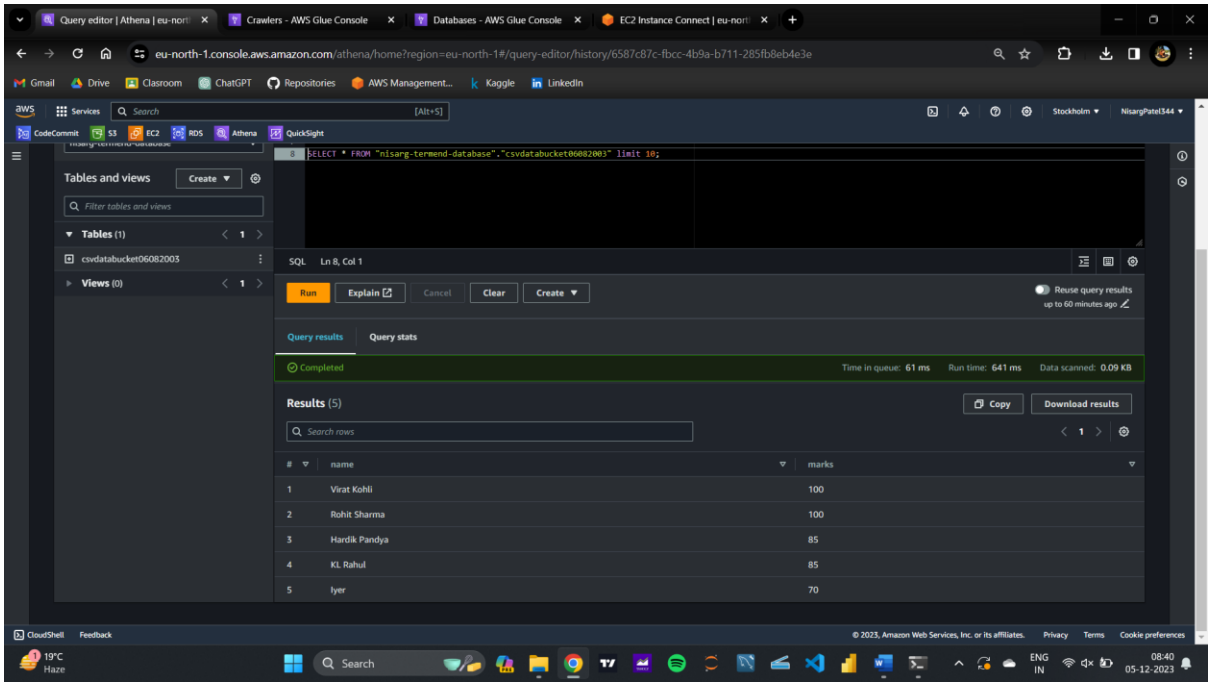




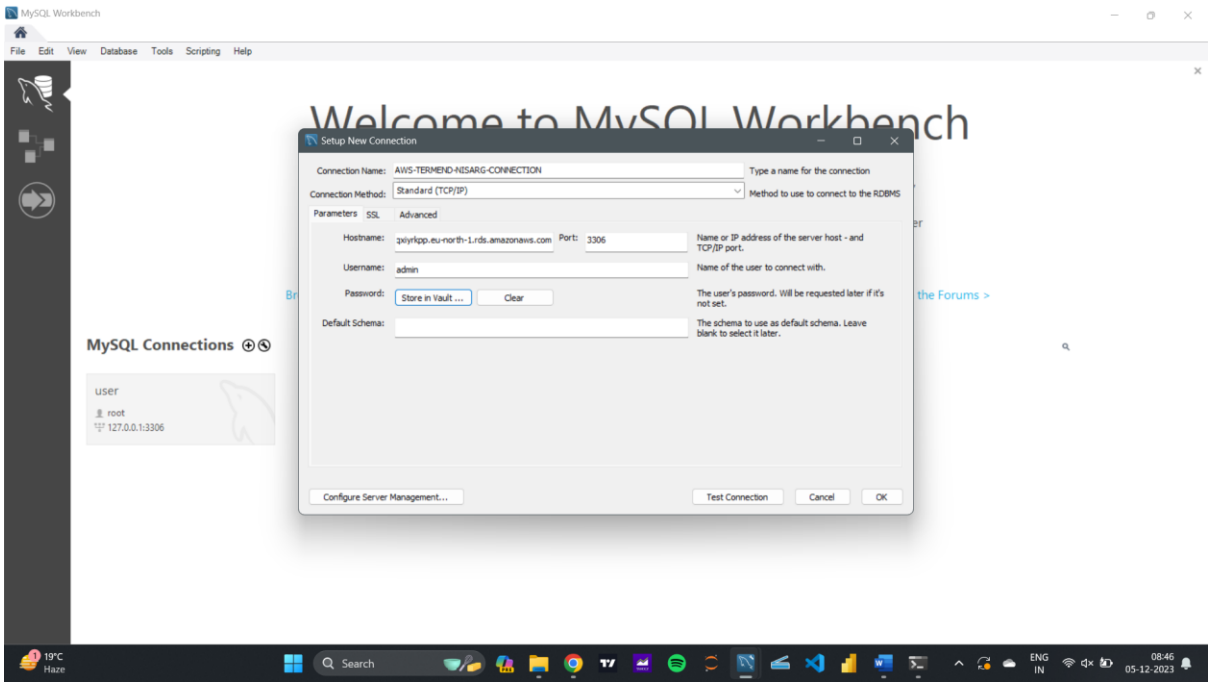

Repositories | AWS Developer
eu-north-1.console.aws.amazon.com/codesuite/codecommit/repositories?region=eu-north-1
Incognito
Gmail
Drive
Classroom
ChatGPT
Repositories
AWS Management...
Kaggle
LinkedIn
aws
Services
Search
[Alt+S]
Stockholm
Mukesh @ 4106-5209-2301
Developer Tools
CodeCommit
Source • CodeCommit
Getting started
Repositories
Approval rule templates
Artifacts • CodeArtifact
Build • CodeBuild
Deploy • CodeDeploy
Pipeline • CodePipeline
Settings
Go to resource
Feedback
Developer Tools > CodeCommit > Repositories
Repositories Info
Notify
Clone URL
View repository
Delete repository
Create repository
Name
Description
Last modified
csvfolder
-
2 days ago
repo_exam_rev
-
2 days ago
Q2Assignment
-
8 days ago
CrudOperation
-
1 month ago
CloudShell
Feedback
© 2023, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences
18°C
Haze
Search
ENG
IN
08:17
05-12-2023

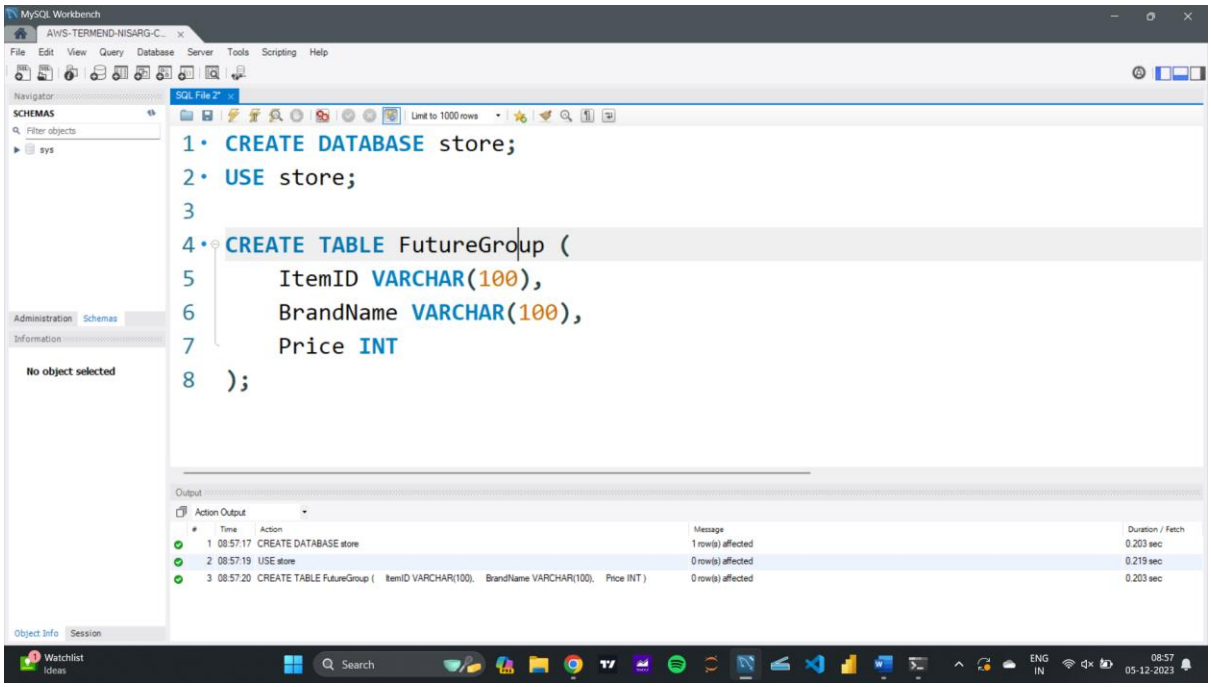
Question 1.B


Windows PowerShell
PS C:\Users\nisar\Desktop\Athena> aws s3 mb s3://csvdatabucket06082003
make_bucket: csvdatabucket06082003
PS C:\Users\nisar\Desktop\Athena> aws s3 cp .\data.csv s3://csvdatabucket06082003
upload: .\data.csv to s3://csvdatabucket06082003/data.csv
PS C:\Users\nisar\Desktop\Athena> aws s3 ls s3://csvdatabucket06082003
2023-12-05 08:31:21 90 data.csv
PS C:\Users\nisar\Desktop\Athena>


eu-north-1.console.aws.amazon.com/athena/home?region=eu-north-1#/query-editor/history/6587c87c-fbcc-4b9a-b711-285fb8eb4e3e
SELECT * FROM "nisarg-termend-database","csvdatabucket06082003" limit 10;
Tables and views
Create
Tables (1)
csvdatabucket06082003
Views (0)
Run
Explain
Cancel
Clear
Create
Query results
Query stats
Completed
Time in queue: 61 ms
Run time: 641 ms
Data scanned: 0.09 KB
Results (5)
Copy
Download results
name
marks
1 Virat Kohli 100
2 Rohit Sharma 100
3 Hardik Pandya 85
4 KL Rahul 85
5 Iyer 70

Question 2.B

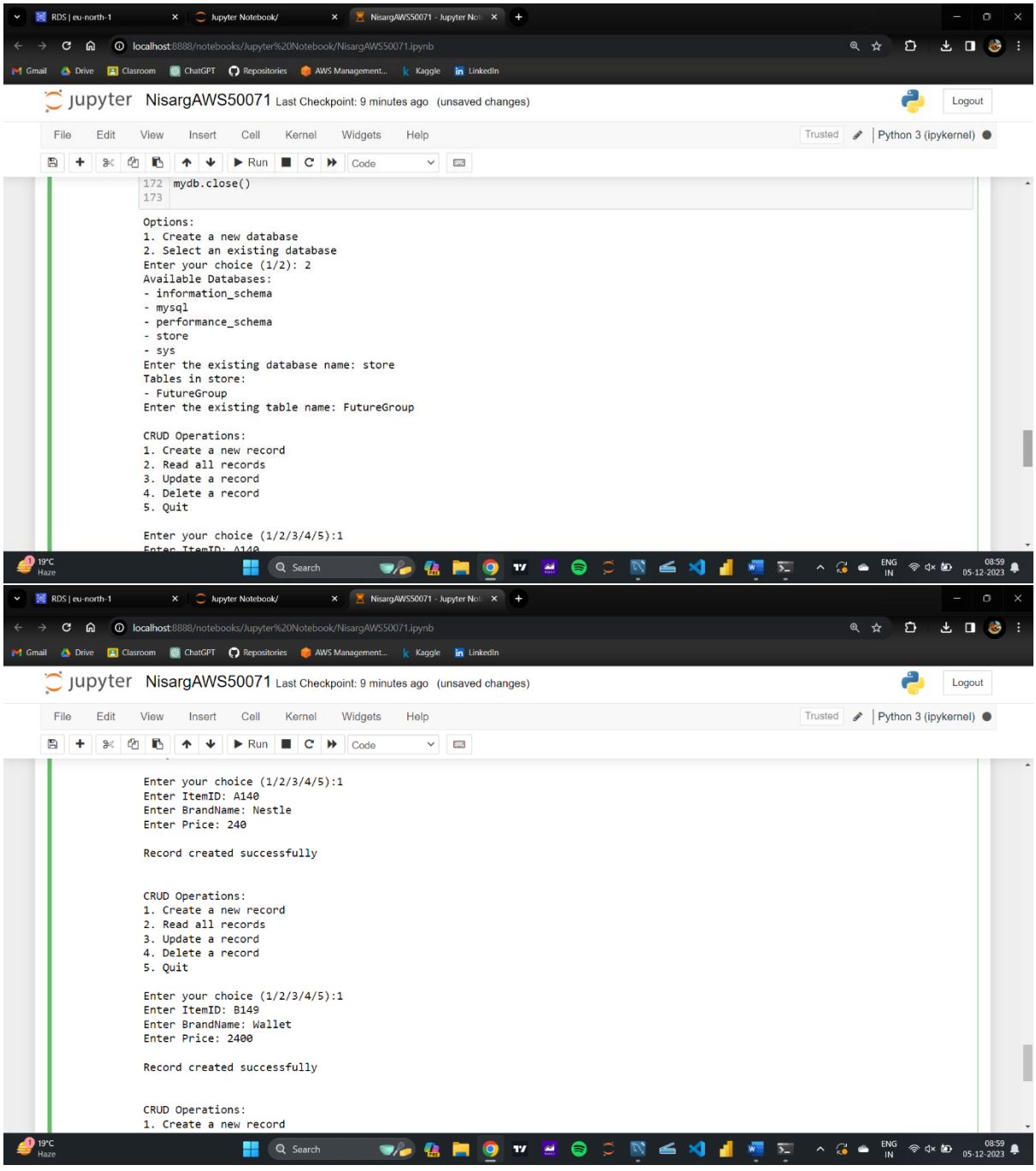

MySQL Workbench
File Edit View Database Tools Scripting Help
Welcome to MySQL Workbench
Setup New Connection
Connection Name: AWS-TERMEND-NISARG-CONNECTION
Type a name for the connection
Connection Method: Standard (TCP/IP)
Method to use to connect to the RDBMS
Parameters SSL Advanced
Hostname: :xiyrkpp.eu-north-1.rds.amazonaws.com Port: 3306
Name or IP address of the server host - and TCP/IP port.
Username: admin
Name of the user to connect with.
Password: Store in Vault ... Clear
The user's password. Will be requested later if it's not set.
Default Schema:
The schema to use as default schema. Leave blank to select it later.
Configure Server Management...
Test Connection
Cancel
OK
MySQL Connections
user
root
127.0.0.1:3306
the Forums >
19°C Haze
Search
ENG IN
08:46 05-12-2023

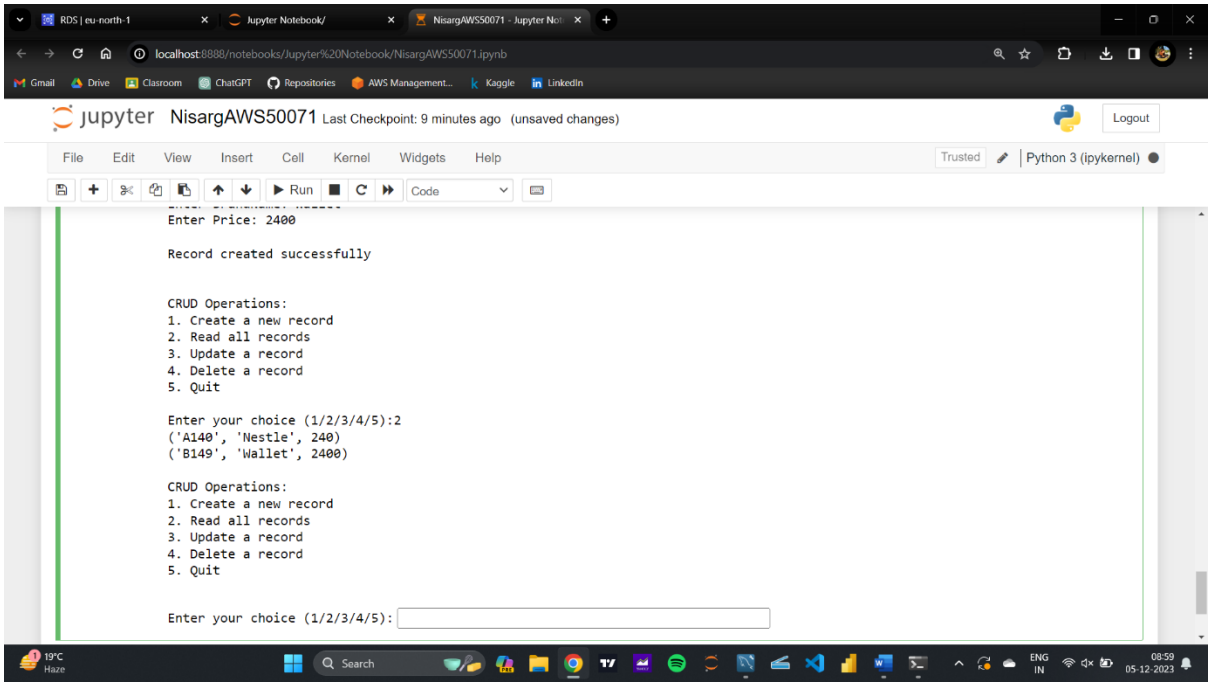

MySQL Workbench
AWS-TERMEND-NISARG-C...
File Edit View Query Database Server Tools Scripting Help
Navigator
SCHEMAS
Filter objects
sys
Administration Schemas
Information
No object selected
Object Info Session
SQL File 2*
Limit to 1000 rows
1 • CREATE DATABASE store;
2 • USE store;
3
4 • CREATE TABLE FutureGroup (
5 ItemID VARCHAR(100),
6 BrandName VARCHAR(100),
7 Price INT
8);
Output
Action Output
Time Action Message Duration / Fetch
1 08:57:17 CREATE DATABASE store 1 row(s) affected 0.203 sec
2 08:57:19 USE store 0 row(s) affected 0.219 sec
3 08:57:20 CREATE TABLE FutureGroup (ItemID VARCHAR(100), BrandName VARCHAR(100), Price INT) 0 row(s) affected 0.203 sec
Watchlist Ideas
Search
ENG IN
08:57 05-12-2023

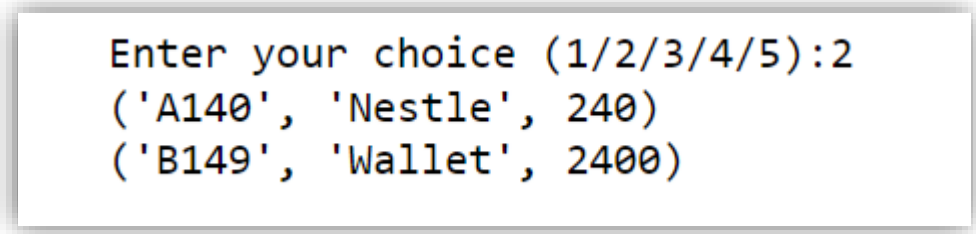
Python Code Made By Me For Operation RDS Database using Python

```
import mysql.connector

# Establish a connection to the MySQL database
mydb = mysql.connector.connect(
    host="nisarg-termend-50071-database.ctifqxiyrkpp.eu-north-1.rds.amazonaws.com",
    user="admin",
    password="11111111"
)

# Create a cursor to execute SQL queries
mycursor = mydb.cursor()

# Initialize empty attributes list and table_name
attributes = []
table_name = ""

# Execute a query to show available databases
mycursor.execute("SHOW DATABASES")
databases = [db[0] for db in mycursor.fetchall()]

# Function to create a new record
def create_record():
    if not attributes:
        print("No attributes defined. Please select or create a table first.")
        return
    values = []
    placeholders = []
    for attribute in attributes:
        if attribute[0] == 'id':
            continue
        value = input(f"Enter {attribute[0]}: ")
        values.append(value)
        placeholders.append("%s")
    insert_query = f"INSERT INTO {table_name} ("
    insert_query += ", ".join([attribute[0] for attribute in attributes if attribute[0] != 'id'])
    insert_query += ") VALUES ("
    insert_query += ", ".join(placeholders)
    insert_query += ")"
    try:
        mycursor.execute(insert_query, values)
        mydb.commit()
        print("\nRecord created successfully\n")
    except mysql.connector.Error as err:
        print(f"Error: {err}")

# Function to read all records
def read_records():
    mycursor.execute(f"SELECT * FROM {table_name}")
    records = mycursor.fetchall()
    if not records:
        print("No records found")
    else:
        for record in records:
            print(record)

# Function to update a record
def update_record():
    if not attributes:
        print("No attributes defined. Please select or create a table first.")
        return
    record_id = input("Enter the ID of the record you want to update: ")
    values = []
    for attribute in attributes:
        value = input(f"Enter new {attribute[0]}: ")
        values.append(value)
    update_query = f"UPDATE {table_name} SET "
    update_query += ", ".join([f"{attribute[0]} = %s" for attribute in attributes])
    update_query += " WHERE id = %s"  # Assuming the primary key is 'id'
    values.append(record_id)
    try:
        mycursor.execute(update_query, values)
        mydb.commit()
        print("\nRecord updated successfully\n")
    except mysql.connector.Error as err:
        print(f"Error: {err}")

# Function to delete a record
def delete_record():
    if not attributes:
        print("No attributes defined. Please select or create a table first.")
        return
    record_id = input("Enter the ID of the record you want to delete: ")
    delete_query = f"DELETE FROM {table_name} WHERE id = %s"  # Assuming the primary key is 'id'
```

```
    try:
        mycursor.execute(delete_query, (record_id,))
        mydb.commit()
        print("\nRecord deleted successfully\n")
    except mysql.connector.Error as err:
        print(f"Error: {err}")

print("Options:")
print("1. Create a new database")
print("2. Select an existing database")
choice = input("Enter your choice (1/2): ")

if choice == "1":
    # Create a new database
    database_name = input("Enter the database name: ")
    mycursor.execute(f"CREATE DATABASE IF NOT EXISTS {database_name}")
    mycursor.execute(f"USE {database_name}")
    table_name = input("Enter the table name (not a reserved keyword): ")
    attributes = []
    while True:
        attribute_name = input("Enter an attribute name (or 'done' to finish): ")
        if attribute_name.lower() == 'done':
            break
        attribute_datatype = input(f"Enter the datatype for {attribute_name}: ")
        attribute_length = input(f"Enter the length for {attribute_name} (or 'max' for maximum length): ")
        attributes.append((attribute_name, attribute_datatype, attribute_length))
    create_table_query = f"CREATE TABLE IF NOT EXISTS `{table_name}` ("
    create_table_query += f"id INT AUTO_INCREMENT PRIMARY KEY, "
    for attribute in attributes:
        attribute_name, attribute_datatype, attribute_length = attribute
        if attribute_length.lower() == 'max':
            create_table_query += f"`{attribute_name}` {attribute_datatype}, "
        else:
            create_table_query += f"`{attribute_name}` {attribute_datatype}({attribute_length}), "
    create_table_query = create_table_query[:-2] + ")"
    mycursor.execute(create_table_query)

elif choice == "2":
    # Select an existing database
    print("Available Databases:")
    for db in databases:
        print(f"- {db}")
    selected_database = input("Enter the existing database name: ")
    if selected_database in databases:
        mycursor.execute(f"USE {selected_database}")
        mycursor.execute(f"SHOW TABLES IN {selected_database}")
        tables = [table[0] for table in mycursor.fetchall()]
        if tables:
            print(f"Tables in {selected_database}:")
            for table in tables:
                print(f"- {table}")
            table_name = input("Enter the existing table name: ")
            mycursor.execute(f"DESCRIBE `{table_name}`")
            table_attributes = mycursor.fetchall()
            attributes = [(attr[0], attr[1]) for attr in table_attributes]

            # Check if the 'id' column is in the attributes list and remove it
            attributes = [attr for attr in attributes if attr[0] != 'id']
        else:
            print(f"No tables found in {selected_database}. You can create a new table.")
            table_name = input("Enter the table name: ")
    else:
        print("Database not found. You can create a new database.")
        database_name = input("Enter the database name: ")


while True:
    print("\nCRUD Operations:")
    print("1. Create a new record")
    print("2. Read all records")
    print("3. Update a record")
    print("4. Delete a record")
    print("5. Quit\n")
    choice = input("Enter your choice (1/2/3/4/5):")
    if choice == "1":
        create_record()
    elif choice == "2":
        read_records()
    elif choice == "3":
        update_record()
    elif choice == "4":
        delete_record()
    elif choice == "5":
        break
    else:
        print("Invalid choice. Please enter a valid option.")

# Close the database connection
mydb.close()
```


jupyter NisargAWS50071 Last Checkpoint: 9 minutes ago (unsaved changes)
Logout
File Edit View Insert Cell Kernel Widgets Help
Trusted Python 3 (ipykernel)
Run Code
172 mydb.close()
173
Options:
1. Create a new database
2. Select an existing database
Enter your choice (1/2): 2
Available Databases:
- information_schema
- mysql
- performance_schema
- store
- sys
Enter the existing database name: store
Tables in store:
- FutureGroup
Enter the existing table name: FutureGroup
CRUD Operations:
1. Create a new record
2. Read all records
3. Update a record
4. Delete a record
5. Quit
Enter your choice (1/2/3/4/5):1
Enter your choice (1/2/3/4/5):1
Enter ItemID: A140
Enter BrandName: Nestle
Enter Price: 240
Record created successfully
CRUD Operations:
1. Create a new record
2. Read all records
3. Update a record
4. Delete a record
5. Quit
Enter your choice (1/2/3/4/5):1
Enter ItemID: B149
Enter BrandName: Wallet
Enter Price: 2400
Record created successfully
CRUD Operations:
1. Create a new record





This is the screenshot from the upper photo to check precisely, You can verify from upper screenshot also, this is just to visualize easy