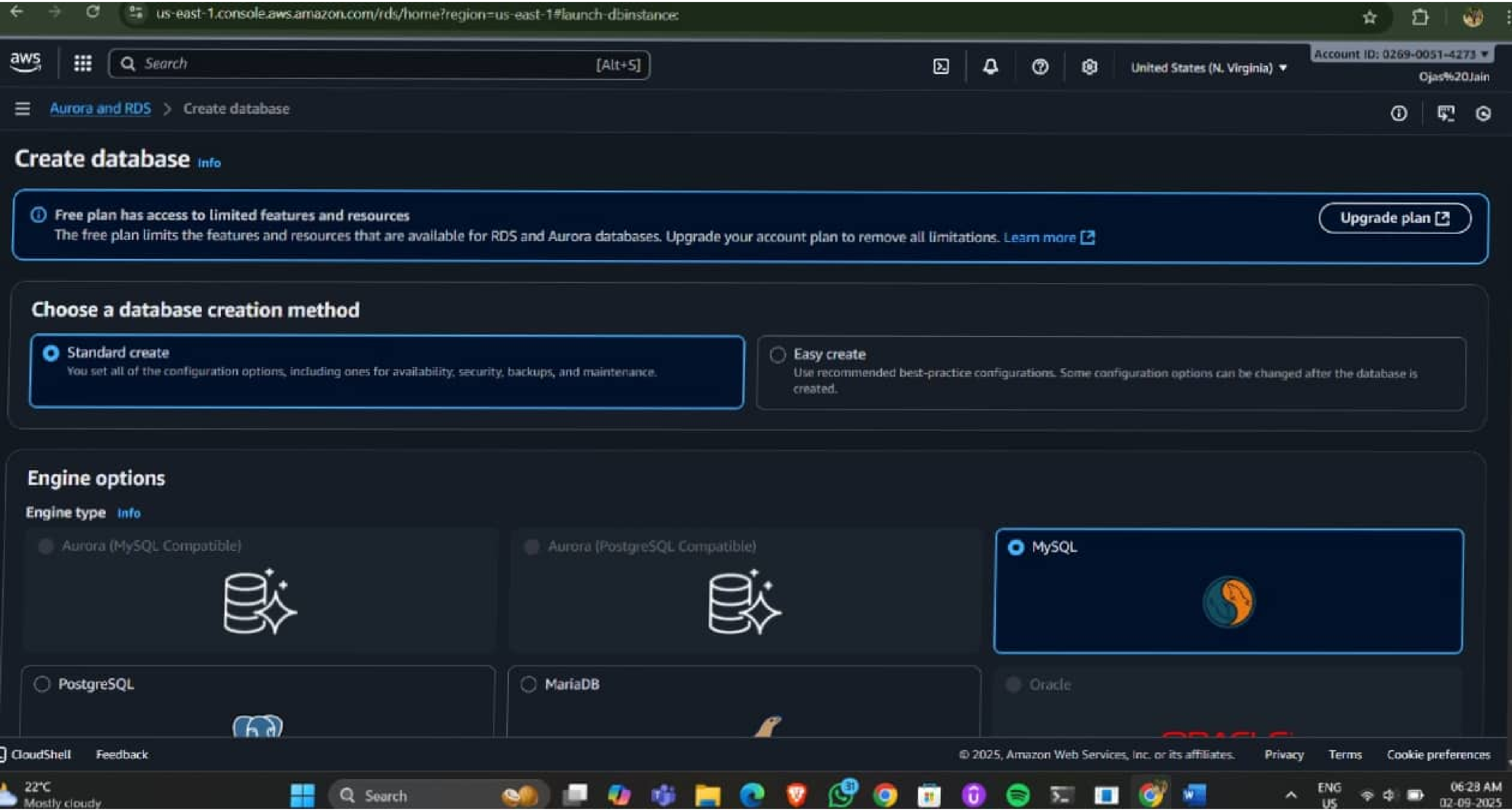


us-east-1.console.aws.amazon.com/rds/home?region=us-east-1#launch-dbinstance:
aws
Search
[Alt+S]
United States (N. Virginia) ▾
Account ID: 0269-0051-4273 ▾
Ojas%20Jain
Aurora and RDS > Create database

Create database

Info

Free plan has access to limited features and resources

The free plan limits the features and resources that are available for RDS and Aurora databases. Upgrade your account plan to remove all limitations. Learn more

Upgrade plan

Choose a database creation method

● Standard create
You set all of the configuration options, including ones for availability, security, backups, and maintenance.

○ Easy create
Use recommended best-practice configurations. Some configuration options can be changed after the database is created.

Engine options

Engine type Info

○ Aurora (MySQL Compatible)

○ Aurora (PostgreSQL Compatible)

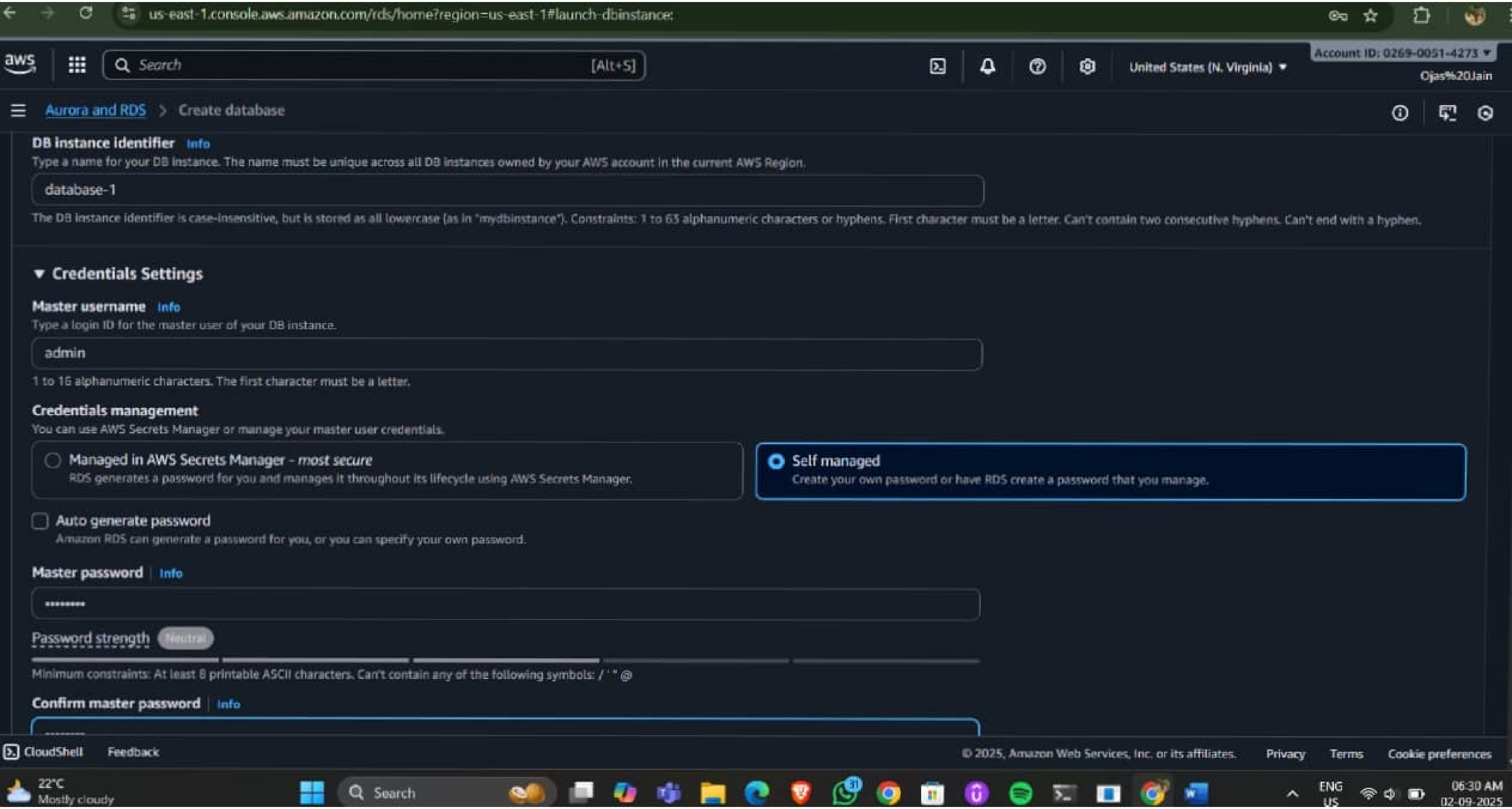
● MySQL

○ PostgreSQL

○ MariaDB

○ Oracle

CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences
22°C
Mostly cloudy
Search
ENG
US
06:28 AM
02-09-2025


us-east-1.console.aws.amazon.com/rds/home?region=us-east-1#launch-dbinstance:
aws
Search
[Alt+S]
United States (N. Virginia) ▾
Account ID: 0269-0051-4273 ▾
Ojas%20Jain
Aurora and RDS > Create database
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences
22°C
Mostly cloudy
Search
ENG
US
06:30 AM
02-09-2025

DB instance identifier Info

Type a name for your DB instance. The name must be unique across all DB instances owned by your AWS account in the current AWS Region.

database-1

The DB instance identifier is case-insensitive, but is stored as all lowercase (as in "mydbinstance"). Constraints: 1 to 63 alphanumeric characters or hyphens. First character must be a letter. Can't contain two consecutive hyphens. Can't end with a hyphen.

▼ **Credentials Settings**

Master username Info

Type a login ID for the master user of your DB instance.

admin

1 to 16 alphanumeric characters. The first character must be a letter.

Credentials management

You can use AWS Secrets Manager or manage your master user credentials.

○ Managed in AWS Secrets Manager - *most secure*
RDS generates a password for you and manages it throughout its lifecycle using AWS Secrets Manager.

● Self managed
Create your own password or have RDS create a password that you manage.

☐ Auto generate password
Amazon RDS can generate a password for you, or you can specify your own password.

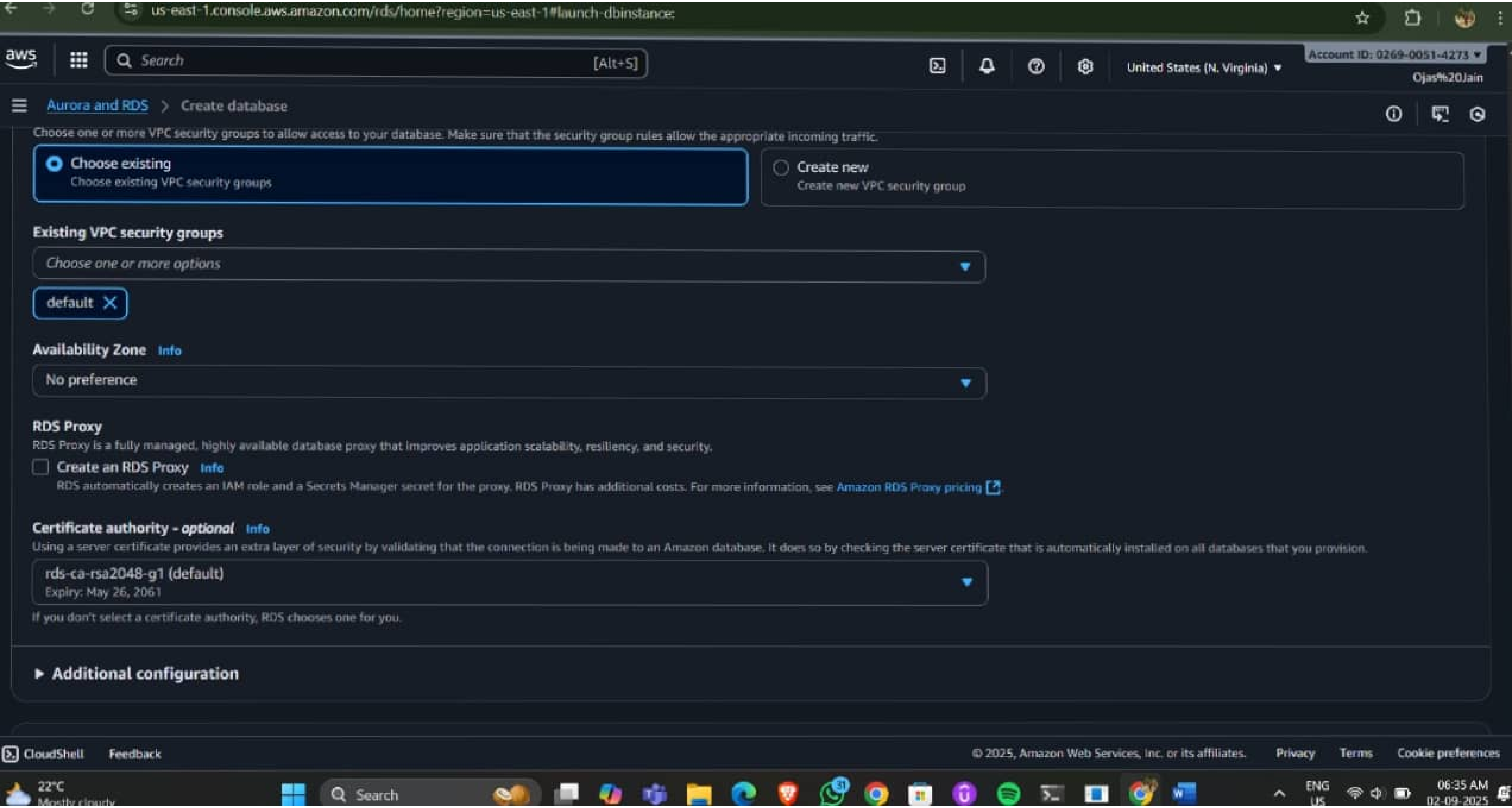
Master password | Info

•••••••••

Password strength Neutral

Minimum constraints: At least 8 printable ASCII characters. Can't contain any of the following symbols: / ' " @

Confirm master password | Info


us-east-1.console.aws.amazon.com/rds/home?region=us-east-1#launch-dbinstance:
aws
Search
[Alt+S]
United States (N. Virginia) ▾
Account ID: 0269-0051-4273 ▾
Ojas%20Jain
Aurora and RDS > Create database
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences
22°C
Mostly cloudy
Search
ENG
US
06:35 AM
02-09-2025

Choose one or more VPC security groups to allow access to your database. Make sure that the security group rules allow the appropriate incoming traffic.

● Choose existing
Choose existing VPC security groups

○ Create new
Create new VPC security group

Existing VPC security groups

Choose one or more options ▾

default ✕

Availability Zone Info

No preference ▾

RDS Proxy

RDS Proxy is a fully managed, highly available database proxy that improves application scalability, resiliency, and security.

☐ Create an RDS Proxy Info

RDS automatically creates an IAM role and a Secrets Manager secret for the proxy. RDS Proxy has additional costs. For more information, see Amazon RDS Proxy pricing.

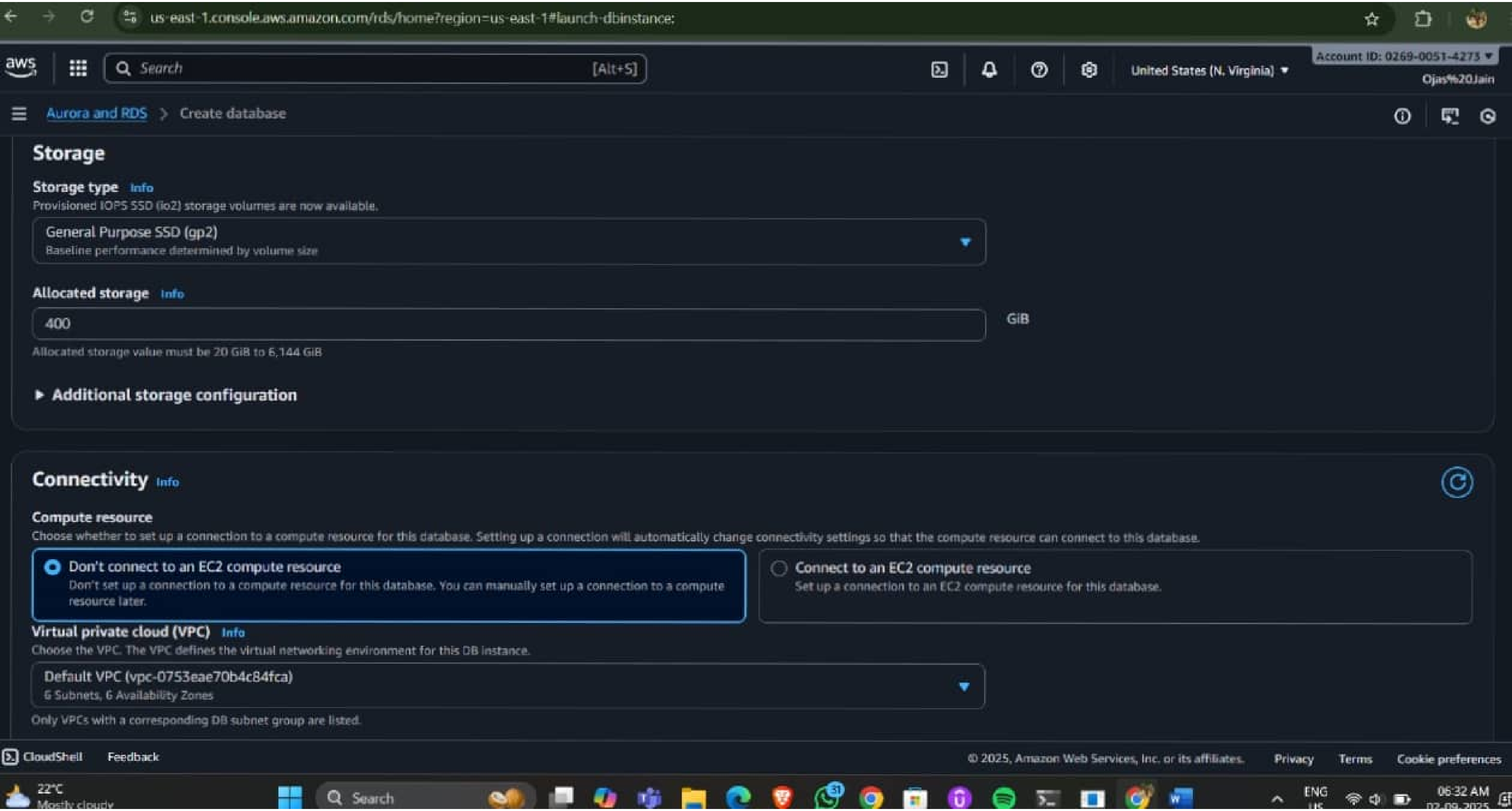
Certificate authority - *optional* Info

Using a server certificate provides an extra layer of security by validating that the connection is being made to an Amazon database. It does so by checking the server certificate that is automatically installed on all databases that you provision.

rds-ca-rsa2048-g1 (default) ▾
Expiry: May 26, 2061

If you don't select a certificate authority, RDS chooses one for you.

▶ **Additional configuration**


us-east-1.console.aws.amazon.com/rds/home?region=us-east-1#launch-dbinstance:
aws
Search [Alt+S]
United States (N. Virginia)
Account ID: 0269-0051-4273
Ojas%20Jain
Aurora and RDS > Create database
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences
22°C
Mostly cloudy
Search
ENG
06:32 AM

Storage

Storage type Info

Provisioned IOPS SSD (io2) storage volumes are now available.

General Purpose SSD (gp2)
Baseline performance determined by volume size

Allocated storage Info

400 GiB

Allocated storage value must be 20 GiB to 6,144 GiB

▸ **Additional storage configuration**

Connectivity Info

Compute resource

Choose whether to set up a connection to a compute resource for this database. Setting up a connection will automatically change connectivity settings so that the compute resource can connect to this database.

● Don't connect to an EC2 compute resource
Don't set up a connection to a compute resource for this database. You can manually set up a connection to a compute resource later.

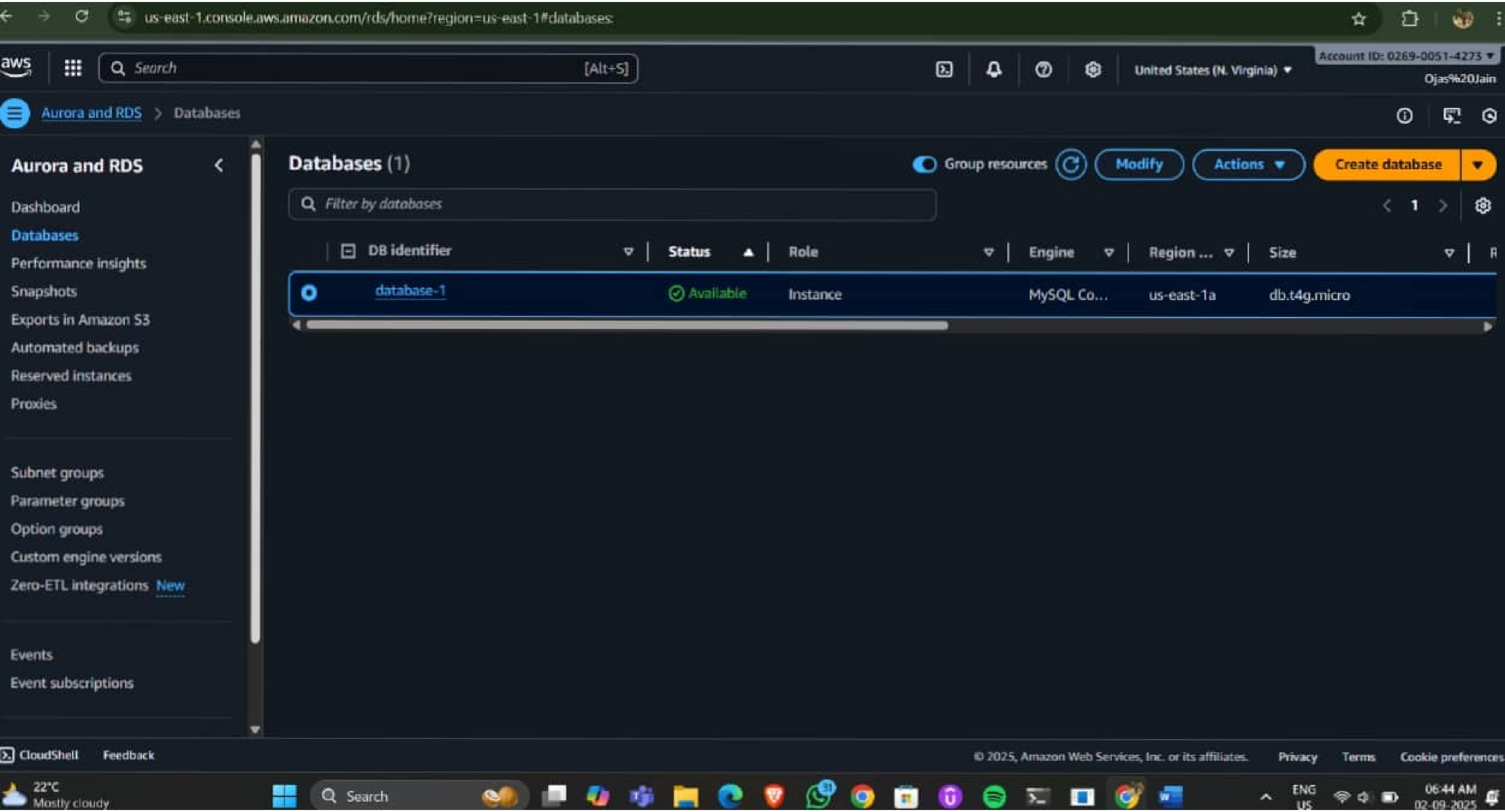
○ Connect to an EC2 compute resource
Set up a connection to an EC2 compute resource for this database.

Virtual private cloud (VPC) Info

Choose the VPC. The VPC defines the virtual networking environment for this DB instance.

Default VPC (vpc-0753eae70b4c84fca)
6 Subnets, 6 Availability Zones

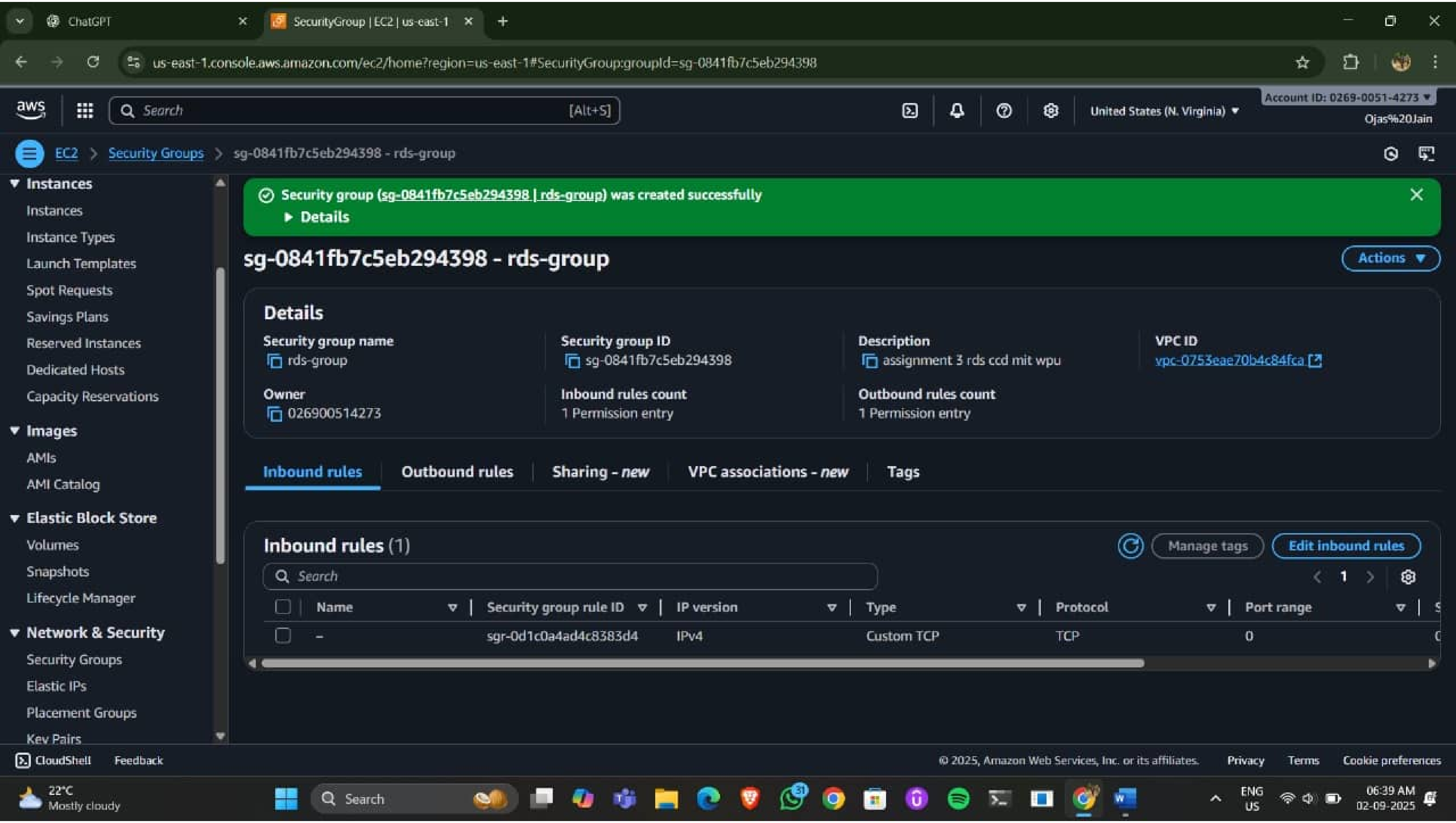
Only VPCs with a corresponding DB subnet group are listed.



us-east-1.console.aws.amazon.com/rds/home?region=us-east-1#databases:
aws
Search [Alt+S]
United States (N. Virginia)
Account ID: 0269-0051-4273
Ojas%20Jain
Aurora and RDS > Databases
Aurora and RDS
Dashboard
Databases
Performance insights
Snapshots
Exports in Amazon S3
Automated backups
Reserved instances
Proxies
Subnet groups
Parameter groups
Option groups
Custom engine versions
Zero-ETL integrations New
Events
Event subscriptions
Group resources
Modify
Actions
Create database
Filter by databases
1
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences
22°C
Mostly cloudy
Search
ENG
US
06:44 AM
02-09-2025

Databases (1)

	DB identifier	Status	Role	Engine	Region ...	Size
●	database-1	Available	Instance	MySQL Co...	us-east-1a	db.t4g.micro


ChatGPT
SecurityGroup | EC2 | us-east-1
us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#SecurityGroup:groupId=sg-0841fb7c5eb294398
aws
Search
[Alt+S]
United States (N. Virginia)
Account ID: 0269-0051-4273
Ojas%20Jain
EC2 > Security Groups > sg-0841fb7c5eb294398 - rds-group
Instances
Instances
Instance Types
Launch Templates
Spot Requests
Savings Plans
Reserved Instances
Dedicated Hosts
Capacity Reservations
Images
AMIs
AMI Catalog
Elastic Block Store
Volumes
Snapshots
Lifecycle Manager
Network & Security
Security Groups
Elastic IPs
Placement Groups
Key Pairs
Details
Actions
Inbound rules
Outbound rules
Sharing - new
VPC associations - new
Tags
Manage tags
Edit inbound rules
Search
1
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences
22°C
Mostly cloudy
Search
ENG
US
06:39 AM
02-09-2025

Security group (sg-0841fb7c5eb294398 | rds-group) was created successfully

sg-0841fb7c5eb294398 - rds-group

Details

Security group name
rds-group

Security group ID
sg-0841fb7c5eb294398

Description
assignment 3 rds ccd mit wpu

VPC ID
vpc-0753eae70b4c84fca

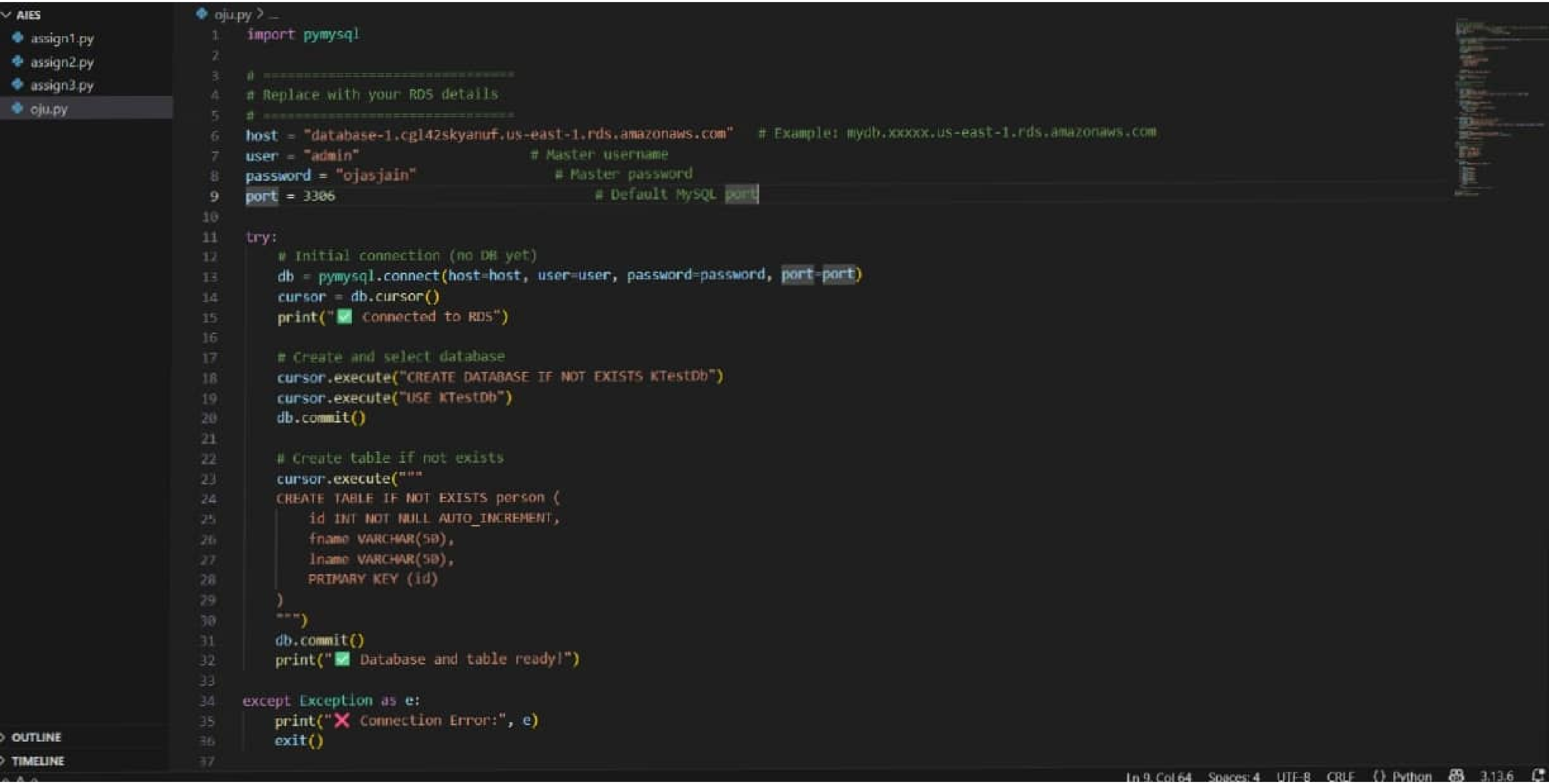
Owner
026900514273

Inbound rules count
1 Permission entry

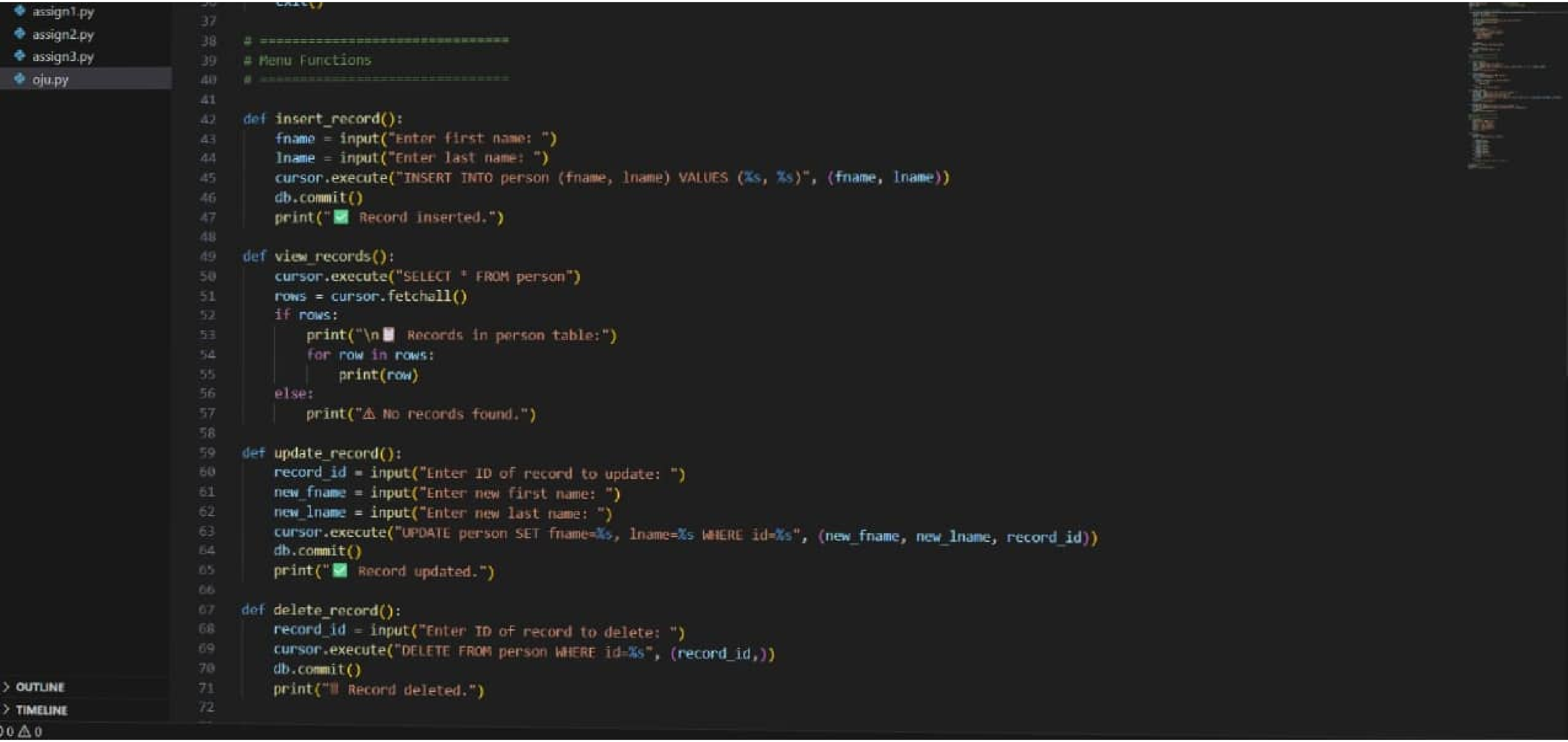
Outbound rules count
1 Permission entry

Inbound rules (1)

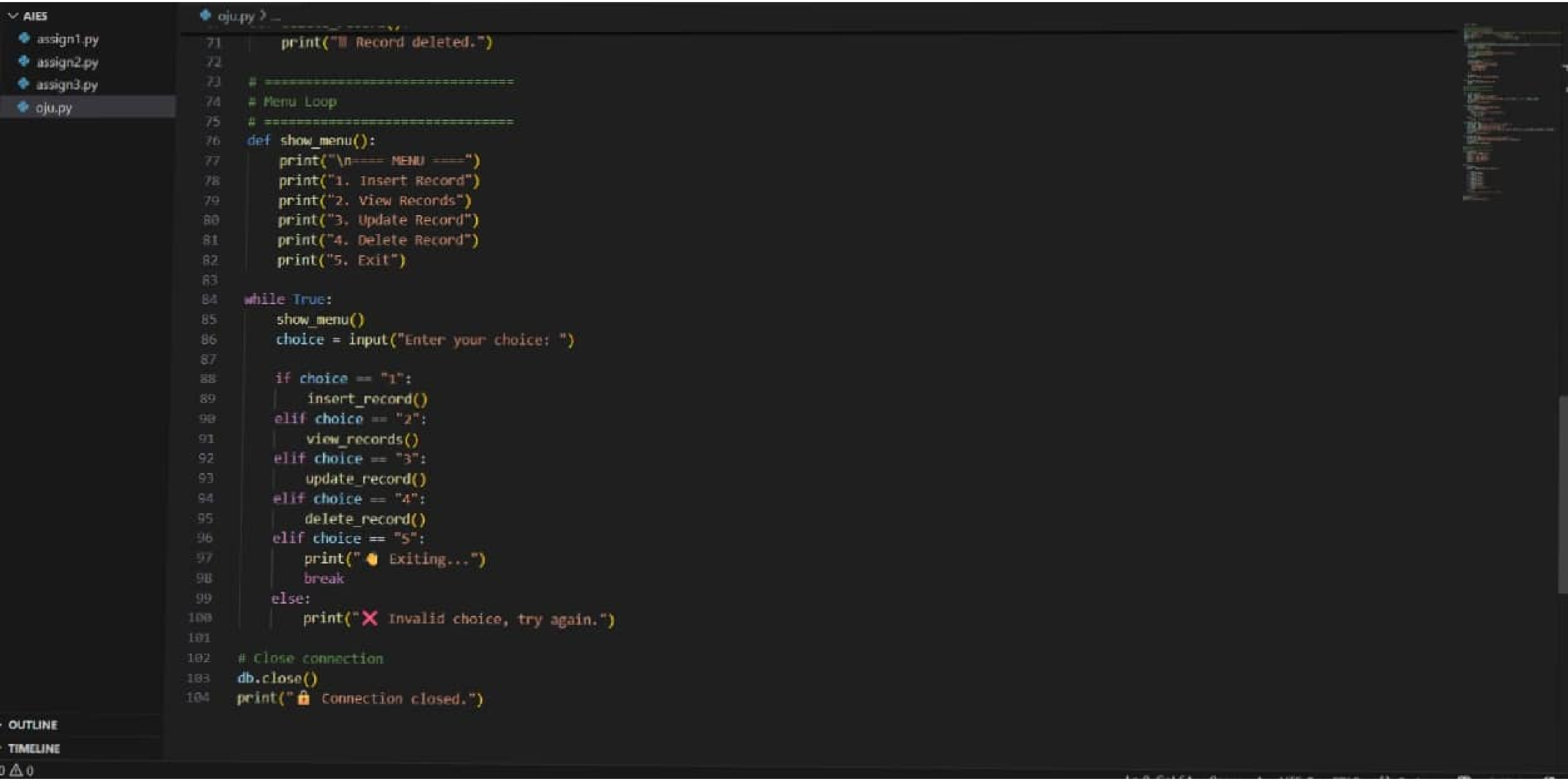
☐	Name	Security group rule ID	IP version	Type	Protocol	Port range
☐	–	sgr-0d1c0a4ad4c8383d4	IPv4	Custom TCP	TCP	0


AIES
assign1.py
assign2.py
assign3.py
oju.py
oju.py > ...
OUTLINE
TIMELINE
Ln 9, Col 64
Spaces: 4
UTF-8
CRLF
{} Python
3.13.6

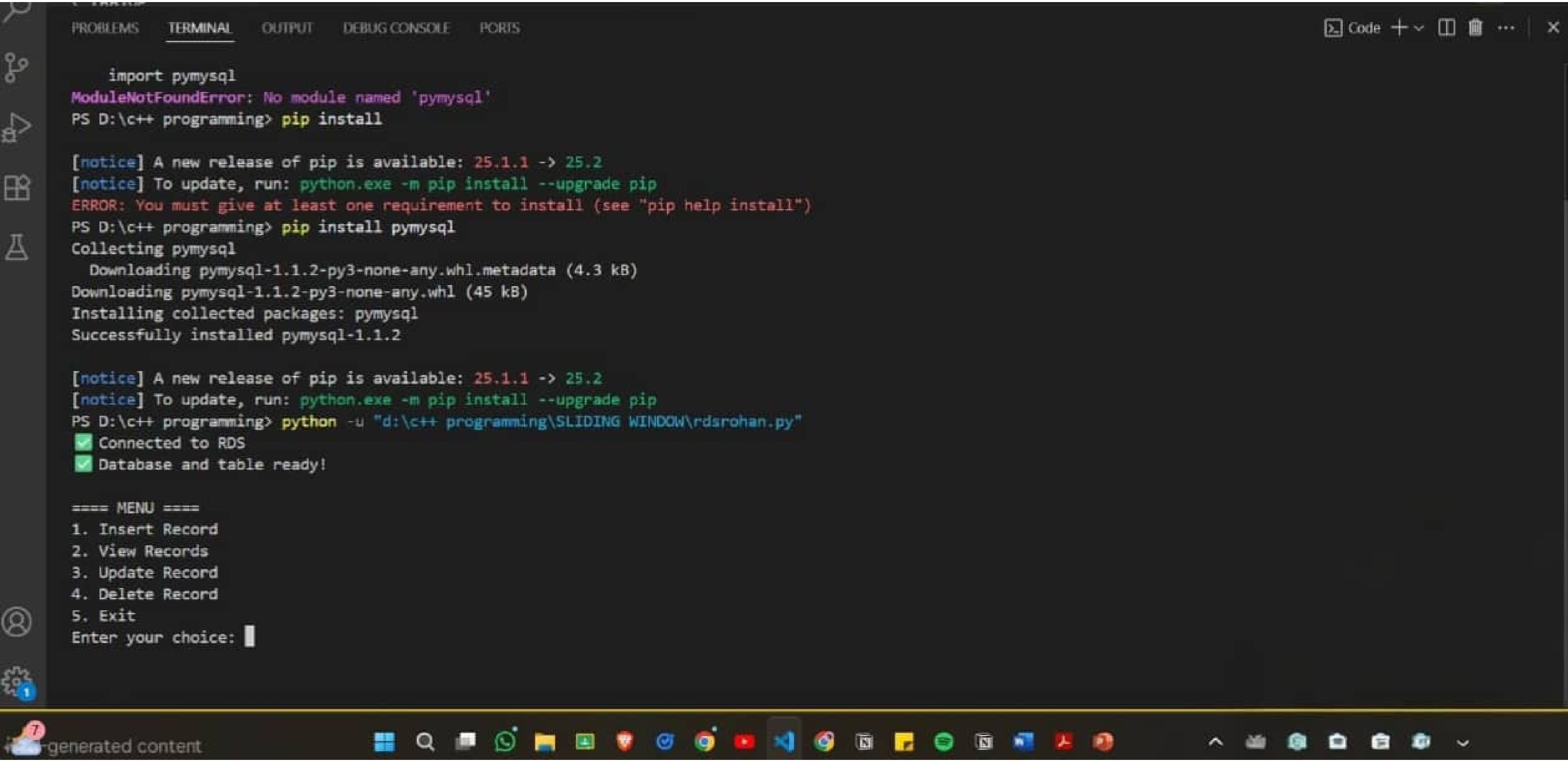
```python
import pymysql

# ===============================
# Replace with your RDS details
# ===============================
host = "database-1.cgl42skyanuf.us-east-1.rds.amazonaws.com"   # Example: mydb.xxxxx.us-east-1.rds.amazonaws.com
user = "admin"                     # Master username
password = "ojasjain"                 # Master password
port = 3306                                # Default MySQL port

try:
    # Initial connection (no DB yet)
    db = pymysql.connect(host=host, user=user, password=password, port=port)
    cursor = db.cursor()
    print("✅ Connected to RDS")

    # Create and select database
    cursor.execute("CREATE DATABASE IF NOT EXISTS KTestDb")
    cursor.execute("USE KTestDb")
    db.commit()

    # Create table if not exists
    cursor.execute("""
    CREATE TABLE IF NOT EXISTS person (
        id INT NOT NULL AUTO_INCREMENT,
        fname VARCHAR(50),
        lname VARCHAR(50),
        PRIMARY KEY (id)
    )
    """)
    db.commit()
    print("✅ Database and table ready!")

except Exception as e:
    print("❌ Connection Error:", e)
    exit()
```



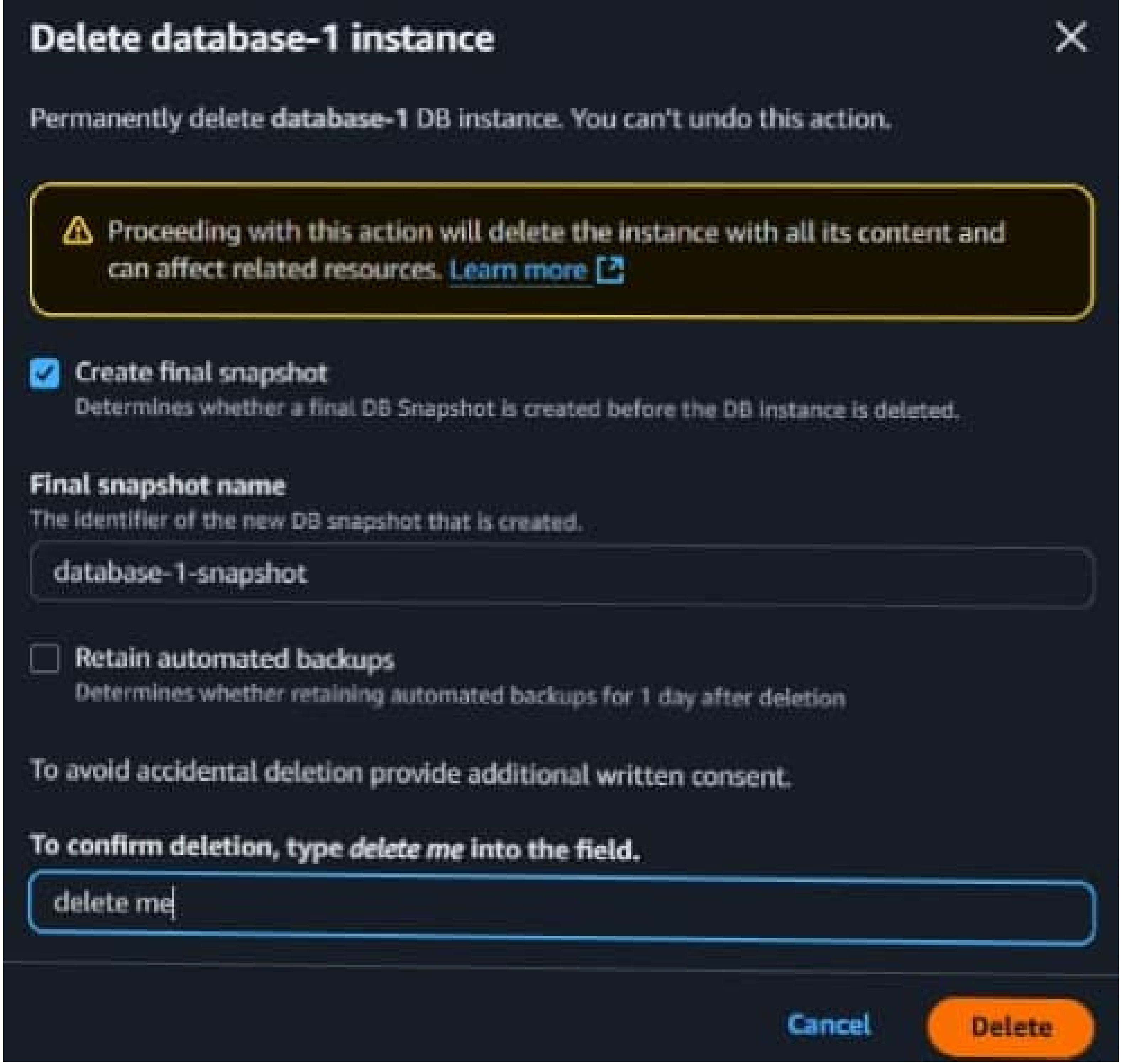
```
    exit()

# ================================
# Menu Functions
# ================================

def insert_record():
    fname = input("Enter first name: ")
    lname = input("Enter last name: ")
    cursor.execute("INSERT INTO person (fname, lname) VALUES (%s, %s)", (fname, lname))
    db.commit()
    print("✅ Record inserted.")

def view_records():
    cursor.execute("SELECT * FROM person")
    rows = cursor.fetchall()
    if rows:
        print("\n📋 Records in person table:")
        for row in rows:
            print(row)
    else:
        print("⚠ No records found.")

def update_record():
    record_id = input("Enter ID of record to update: ")
    new_fname = input("Enter new first name: ")
    new_lname = input("Enter new last name: ")
    cursor.execute("UPDATE person SET fname=%s, lname=%s WHERE id=%s", (new_fname, new_lname, record_id))
    db.commit()
    print("✅ Record updated.")

def delete_record():
    record_id = input("Enter ID of record to delete: ")
    cursor.execute("DELETE FROM person WHERE id=%s", (record_id,))
    db.commit()
    print("🗑 Record deleted.")
```


AIES
assign1.py
assign2.py
assign3.py
oju.py
oju.py › ...
OUTLINE
TIMELINE
0 ⚠ 0

```python
        print("🗑 Record deleted.")

# ==============================
# Menu Loop
# ==============================
def show_menu():
    print("\n==== MENU ====")
    print("1. Insert Record")
    print("2. View Records")
    print("3. Update Record")
    print("4. Delete Record")
    print("5. Exit")

while True:
    show_menu()
    choice = input("Enter your choice: ")

    if choice == "1":
        insert_record()
    elif choice == "2":
        view_records()
    elif choice == "3":
        update_record()
    elif choice == "4":
        delete_record()
    elif choice == "5":
        print("👋 Exiting...")
        break
    else:
        print("❌ Invalid choice, try again.")

# Close connection
db.close()
print("🔒 Connection closed.")
```



```
    import pymysql
ModuleNotFoundError: No module named 'pymysql'
PS D:\c++ programming> pip install

[notice] A new release of pip is available: 25.1.1 -> 25.2
[notice] To update, run: python.exe -m pip install --upgrade pip
ERROR: You must give at least one requirement to install (see "pip help install")
PS D:\c++ programming> pip install pymysql
Collecting pymysql
  Downloading pymysql-1.1.2-py3-none-any.whl.metadata (4.3 kB)
Downloading pymysql-1.1.2-py3-none-any.whl (45 kB)
Installing collected packages: pymysql
Successfully installed pymysql-1.1.2

[notice] A new release of pip is available: 25.1.1 -> 25.2
[notice] To update, run: python.exe -m pip install --upgrade pip
PS D:\c++ programming> python -u "d:\c++ programming\SLIDING WINDOW\rdsrohan.py"
✅ Connected to RDS
✅ Database and table ready!

==== MENU ====
1. Insert Record
2. View Records
3. Update Record
4. Delete Record
5. Exit
Enter your choice: 
```



Delete database-1 instance

Permanently delete **database-1** DB instance. You can't undo this action.

⚠ Proceeding with this action will delete the instance with all its content and can affect related resources. Learn more

☑ Create final snapshot
Determines whether a final DB Snapshot is created before the DB instance is deleted.

Final snapshot name
The identifier of the new DB snapshot that is created.
database-1-snapshot

☐ Retain automated backups
Determines whether retaining automated backups for 1 day after deletion

To avoid accidental deletion provide additional written consent.

To confirm deletion, type *delete me* into the field.
delete me