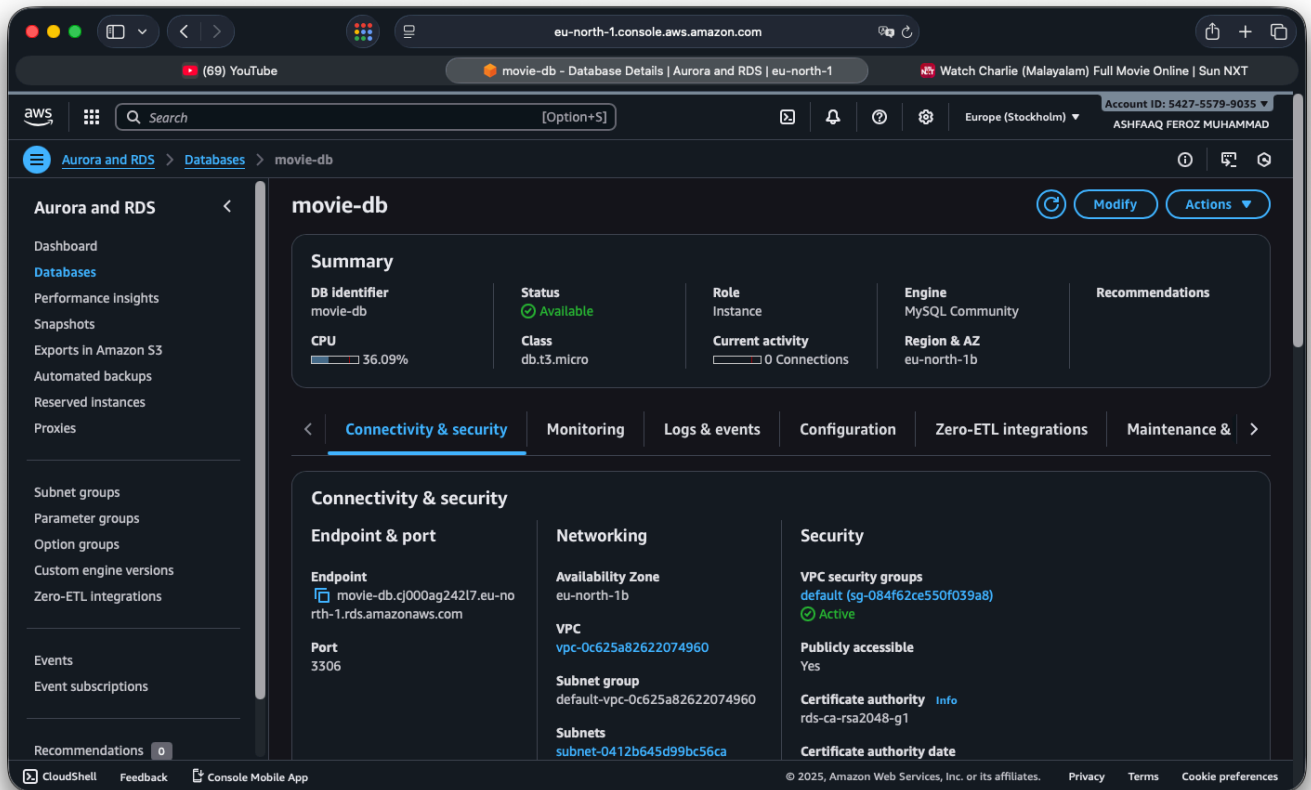# Final Report
Staff-Graded Assignment: AWS RDS – MySQL

Name: Ashfaaq Feroz Muhammad
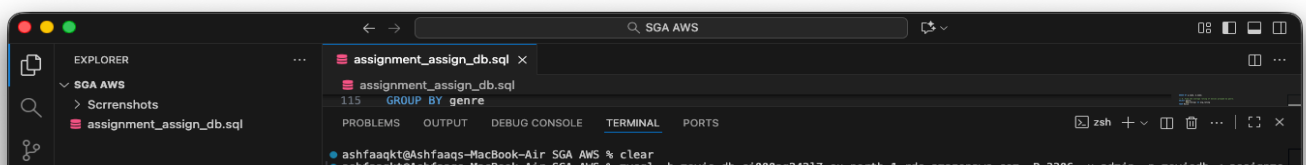ID: 2023EBCS005

AWS My SQL Database Dashboard:

- Created AWS account and created free tier database.
- Used ==Endpoint== to connect File to database.

First General Test if all are connected and working well

◊ Entered all the Queries in SQL file and ran a general Test after connecting to the database through terminal.

Query 1:

```
SELECT *
FROM Movie
WHERE year > 2000;
```

```
ashfaaqkt@Ashfaaqs-MacBook-Air SGA AWS % mysql -h movie-db.cj000ag242l7.eu-north-1.rds.amazonaws.com -P 3306 -u admin -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 29
Server version: 8.0.42 Source distribution

Copyright (c) 2000, 2025, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> SELECT *
    -> FROM Movie
    -> WHERE year > 2000;
ERROR 1046 (3D000): No database selected
mysql> USE moviedb;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> SELECT *
    -> FROM Movie
    -> WHERE year > 2000;
+----+-----------------+-------------------------------+------+--------+-----------------+
| id | name            | description                   | year | rating | genre           |
+----+-----------------+-------------------------------+------+--------+-----------------+
|  1 | Inception       | A mind-bending sci-fi thriller| 2010 |    8.8 | Science Fiction |
|  2 | The Dark Knight | Batman faces the Joker        | 2008 |      9 | Action          |
|  3 | Interstellar    | Space exploration drama       | 2014 |    8.6 | Science Fiction |
+----+-----------------+-------------------------------+------+--------+-----------------+
3 rows in set (0.281 sec)

mysql>
```

Ln 59, Col 1 (38 selected)   Spaces: 4   UTF-8   LF   {} SQL   🐝   📡 Go Live   🔔

Used the database

The query 1 output.

Query 2:

```sql
SELECT name
FROM Person
WHERE role = 'Director';
```

```
mysql> SELECT name
    -> FROM Person
    -> WHERE role = 'Director';
+-----------------+
| name            |
+-----------------+
| Christopher Nolan |
| James Cameron   |
+-----------------+
2 rows in set (0.686 sec)

mysql>
```
Ln 66, Col 25 (48 selected)   Spaces: 4   UTF-8   LF   {} SQL   🐛   📡 Go Live   🔔

Query 3:

```sql
SELECT *
FROM Movie
WHERE rating > 8.0;
```

```
mysql> SELECT *
    -> FROM Movie
    -> WHERE rating > 8.0;
+----+------------------------+----------------------------+------+--------+-----------------+
| id | name                   | description                | year | rating | genre           |
+----+------------------------+----------------------------+------+--------+-----------------+
|  1 | Inception              | A mind-bending sci-fi thriller | 2010 |    8.8 | Science Fiction |
|  2 | The Dark Knight        | Batman faces the Joker     | 2008 |      9 | Action          |
|  3 | Interstellar           | Space exploration drama    | 2014 |    8.6 | Science Fiction |
|  5 | The Shawshank Redemption | A tale of hope in a prison | 1994 |    9.3 | Drama           |
+----+------------------------+----------------------------+------+--------+-----------------+
4 rows in set (0.291 sec)

mysql>
```
Ln 69, Col 1 (39 selected)   Spaces: 4   UTF-8   LF   {} SQL   🐛   📡 Go Live   🔔

Query 4:

```sql
SELECT COUNT(*) AS total_movies
FROM Movie;
```

```
mysql> SELECT *
    -> FROM Movie
    -> WHERE rating > 8.0;
+----+------------------------+----------------------------+------+--------+-----------------+
| id | name                   | description                | year | rating | genre           |
+----+------------------------+----------------------------+------+--------+-----------------+
|  1 | Inception              | A mind-bending sci-fi thriller | 2010 |    8.8 | Science Fiction |
|  2 | The Dark Knight        | Batman faces the Joker     | 2008 |      9 | Action          |
|  3 | Interstellar           | Space exploration drama    | 2014 |    8.6 | Science Fiction |
|  5 | The Shawshank Redemption | A tale of hope in a prison | 1994 |    9.3 | Drama           |
+----+------------------------+----------------------------+------+--------+-----------------+
4 rows in set (0.291 sec)

mysql>
```
Ln 69, Col 1 (39 selected)   Spaces: 4   UTF-8   LF   {} SQL   🐛   📡 Go Live   🔔

Query 5:

```sql
SELECT m.name AS movie_name,
       p.name AS director_name
FROM Movie m
JOIN MoviePerson mp ON m.id = mp.movie_id
JOIN Person p ON mp.person_id = p.id
WHERE p.role = 'Director';
```

```
mysql> SELECT m.name AS movie_name,
    ->        p.name AS director_name
    -> FROM Movie m
    -> JOIN MoviePerson mp ON m.id = mp.movie_id
    -> JOIN Person p ON mp.person_id = p.id
    -> WHERE p.role = 'Director';
+-----------------+-------------------+
| movie_name      | director_name     |
+-----------------+-------------------+
| Inception       | Christopher Nolan |
| The Dark Knight | Christopher Nolan |
| Interstellar    | Christopher Nolan |
| Titanic         | James Cameron     |
+-----------------+-------------------+
4 rows in set (0.336 sec)

mysql>
```

Query 6:

```sql
SELECT p.name AS actor_name,
       COUNT(mp.movie_id) AS movie_count
FROM Person p
JOIN MoviePerson mp ON p.id = mp.person_id
WHERE p.role IN ('Actor', 'Actress')
GROUP BY p.id, p.name
ORDER BY movie_count DESC;
```

```
mysql> SELECT p.name AS actor_name,
    ->        COUNT(mp.movie_id) AS movie_count
    -> FROM Person p
    -> JOIN MoviePerson mp ON p.id = mp.person_id
    -> WHERE p.role IN ('Actor', 'Actress')
    -> GROUP BY p.id, p.name
    -> ORDER BY movie_count DESC;
+--------------------+-------------+
| actor_name         | movie_count |
+--------------------+-------------+
| Anne Hathaway      |           2 |
| Leonardo DiCaprio  |           1 |
| Christian Bale     |           1 |
| Morgan Freeman     |           1 |
| Kate Winslet       |           1 |
| Tim Robbins        |           1 |
| Matthew McConaughey|           1 |
+--------------------+-------------+
7 rows in set (0.250 sec)

mysql>
```

Query 7:

```sql
SELECT name, rating
FROM Movie
ORDER BY rating DESC
LIMIT 2;
```

```
mysql> SELECT name, rating
    -> FROM Movie
    -> ORDER BY rating DESC
    -> LIMIT 2;
+------------------------+--------+
| name                   | rating |
+------------------------+--------+
| The Shawshank Redemption |   9.3 |
| The Dark Knight        |      9 |
+------------------------+--------+
2 rows in set (0.257 sec)

mysql>
```

Query 8:

```sql
SELECT p.name AS actor_name,
       p.birth_year,
       m.name AS movie_name
FROM Person p
JOIN MoviePerson mp ON p.id = mp.person_id
JOIN Movie m ON mp.movie_id = m.id
WHERE p.role IN ('Actor', 'Actress')
  AND p.birth_year < 1975
ORDER BY p.name, m.name;
```

```
mysql> SELECT p.name AS actor_name,
    ->        p.birth_year,
    ->        m.name AS movie_name
    -> FROM Person p
    -> JOIN MoviePerson mp ON p.id = mp.person_id
    -> JOIN Movie m ON mp.movie_id = m.id
    -> WHERE p.role IN ('Actor', 'Actress')
    ->   AND p.birth_year < 1975
    -> ORDER BY p.name, m.name;
+---------------------+------------+-------------------------+
| actor_name          | birth_year | movie_name              |
+---------------------+------------+-------------------------+
| Christian Bale      |       1974 | The Dark Knight         |
| Leonardo DiCaprio   |       1974 | Inception               |
| Matthew McConaughey |       1969 | Interstellar            |
| Morgan Freeman      |       1937 | The Shawshank Redemption |
| Tim Robbins         |       1958 | The Shawshank Redemption |
+---------------------+------------+-------------------------+
5 rows in set (0.319 sec)

mysql>
```

Ln 110, Col 1 (258 selected)   Spaces: 4   UTF-8   LF   {} SQL   Go Live

Query 9:

```sql
SELECT genre,
       AVG(rating) AS avg_rating
FROM Movie
GROUP BY genre
ORDER BY avg_rating DESC;
```

```
mysql> SELECT genre,
    ->        AVG(rating) AS avg_rating
    -> FROM Movie
    -> GROUP BY genre
    -> ORDER BY avg_rating DESC;
+-----------------+-------------------+
| genre           | avg_rating        |
+-----------------+-------------------+
| Drama           | 9.300000190734863 |
| Action          |                 9 |
| Science Fiction | 8.700000286102295 |
| Romance         | 7.900000095367432 |
+-----------------+-------------------+
4 rows in set (0.218 sec)

mysql>
```

Ln 116, Col 26 (98 selected)   Spaces: 4   UTF-8   LF   {} SQL   Go Live

Query 10:

```sql
SELECT DISTINCT p2.name AS actor_name
FROM Person p1
JOIN MoviePerson mp1 ON p1.id = mp1.person_id
JOIN Movie m ON mp1.movie_id = m.id
JOIN MoviePerson mp2 ON m.id = mp2.movie_id
JOIN Person p2 ON mp2.person_id = p2.id
WHERE p1.role = 'Director'
  AND p1.name = 'Christopher Nolan'
  AND p2.role IN ('Actor', 'Actress');
```

```
mysql> SELECT DISTINCT p2.name AS actor_name
    -> FROM Person p1
    -> JOIN MoviePerson mp1 ON p1.id = mp1.person_id
    -> JOIN Movie m ON mp1.movie_id = m.id
    -> JOIN MoviePerson mp2 ON m.id = mp2.movie_id
    -> JOIN Person p2 ON mp2.person_id = p2.id
    -> WHERE p1.role = 'Director'
    ->   AND p1.name = 'Christopher Nolan'
    ->   AND p2.role IN ('Actor', 'Actress');
+--------------------+
| actor_name         |
+--------------------+
| Leonardo DiCaprio  |
| Anne Hathaway      |
| Christian Bale     |
| Matthew McConaughey|
+--------------------+
4 rows in set (0.291 sec)

mysql>
```

THE END