

Maurey Shane M. Reyes

BSCS-C204

Finals Lab Task 5. CRUD CLI using Python and MySQL

Finals Lab Task 5. CLI using Mysql and Python

1. Make sure you have installed the following prerequisites before proceeding:

a. Mysql-connector

b. Mysql-connector-python

c. Xampp is running along with Apache and Mysql in the background

2. Create the following database in Mysql;

a. Database name: moviesDB with the ff: fields:

movie_id int(10) Primary Key

title varchar(50) NOT NULL

main_actor varchar(50) NOT NULL

director varchar(50) NOT NULL

genre varchar(25) NOT NULL

gross_sales float

ratings (G, PG, R13, R16,X) varchar(5)

b. Insert at_least 5 records

c. Create a user named test_user and assign a password and give it an admin

access by checking necessary SQL functions

3. Guided by the Demo code attached in this task.
test_DemoDB.py

4. Kindly continue working on the code that will allow the user to navigate through the

Database and perform simple CRUD operations.
Follow the following CLI Menu

Options:

5. The user should be able perform the ff: in your program.

MOVIE DATABASE CRUD APP

1- Add New Record

2- View all records,

3- Update a Record and show the updates,

4- Delete a record

5- Search A Record

6- Display Total Numbers of Movies stored in the database

7- Exit

6. For additional challenge, Task – You are to add a SEARCH option in the MENU that will

allow the user to search by Name or emp_id, then display the information about the

record being search. You may use Like statement and fetchOne method in my SQL to

do this,

7. You are also going to add a method that will display the the total number of records in

your database – You may use SQL count statement for this.

8. What to submit:

a. UI Menu

b. Sample Output

c. Source Code

d. Exported sql file

```
import mysql.connector

conn = mysql.connector.connect(
    host="localhost",
    user="test_user",
    password="yourpassword",
    database="moviesDB"
)
cursor = conn.cursor()

def get_new_movie_id():
    cursor.execute("SELECT movie_id FROM movies ORDER BY movie_id DESC LIMIT 1")
    last_record = cursor.fetchone()

    if last_record:
        last_id = last_record[0]      # e.g., 'M005'
        num = int(last_id[1:]) + 1    # extract number → 5 → +1 = 6
    else:
        num = 1 # first record

    new_id = f"M{num:03d}"
    return new_id
```

```
def add_movie():
    print("\n--- Add New Movie ---")

    movie_id = get_new_movie_id()
    print(f"Generated Movie ID: {movie_id}")

    title = input("Enter Title: ")
    main_actor = input("Enter Main Actor: ")
    director = input("Enter Director: ")
    genre = input("Enter Genre: ")
    gross_sales = float(input("Enter Gross Sales: "))
    ratings = input("Enter Rating (G, PG, R13, R16, X): ")

    cursor.execute( operation: """
        INSERT INTO movies1 (movie_id, title, main_actor, director, genre, gross_sales, ratings)
        VALUES (%s, %s, %s, %s, %s, %s, %s)
    """, params: (movie_id, title, main_actor, director, genre, gross_sales, ratings))

    conn.commit()
    print("Movie added successfully!\n")
```

```
def view_movies():
    print("\n--- Movie List ---")
    cursor.execute("SELECT * FROM movies1")
    rows = cursor.fetchall()

    if rows:
        for row in rows:
            movie_id = row[0]
            title = row[1]
            main_actor = row[2]
            director = row[3]
            genre = row[4]
            gross_sales = float(row[5])
            ratings = row[6]

            print(f"ID: {movie_id} | Title: {title} | Actor: {main_actor} | Director: {director} | "
                  f"Genre: {genre} | Gross Sales: {gross_sales} | Rating: {ratings}")
    else:
        print("No movies found.")
    print()
```

```
def update_movie():
    print("\n--- Update Movie ---")
    movie_id = input("Enter Movie ID to update: ")

    title = input("Enter new Title: ")
    main_actor = input("Enter new Main Actor: ")
    director = input("Enter new Director: ")
    genre = input("Enter new Genre: ")
    gross_sales = float(input("Enter new Gross Sales: "))
    ratings = input("Enter new Rating: ")

    cursor.execute(operation: """
        UPDATE movies1
        SET title=%s, main_actor=%s, director=%s, genre=%s, gross_sales=%s, ratings=%s
        WHERE movie_id=%s
    """, params: (title, main_actor, director, genre, gross_sales, ratings, movie_id))

    conn.commit()
    print("Movie updated successfully!\n")
```

```
def delete_movie():
    print("\n--- Delete Movie ---")
    movie_id = input("Enter Movie ID to delete: ")

    cursor.execute(operation: "DELETE FROM movies1 WHERE movie_id=%s", params: (movie_id,))
    conn.commit()
    print("Movie deleted successfully!\n")
```

```

def search_movie():
    print("\n--- Search Movie ---")
    keyword = input("Enter Movie ID or Title to search: ")

    cursor.execute(operation: """
        SELECT * FROM movies1
        WHERE movie_id LIKE %s OR title LIKE %
        """, params: ('%' + keyword + '%', '%' + keyword + '%'))

    rows = cursor.fetchall()

    if rows:
        for row in rows:
            print(row)
    else:
        print("No matching movie found.")
    print()

```

```

def display_total():
    cursor.execute("SELECT COUNT(*) FROM movies1")
    total = cursor.fetchone()[0]
    print(f"\nTotal number of movies: {total}\n")

```

```

def menu():
    while True:
        print("----- MOVIE DATABASE CLI -----")
        print("1. Add Movie")
        print("2. View Movies")
        print("3. Update Movie")
        print("4. Delete a Movie")
        print("5. Search a Movie")
        print("6. Display Total Records")
        print("7. Exit")

        choice = input("Select an option (1-7): ")

```

```

choice = input("Select an option (1-7): ")

if choice == '1':
    add_movie()
elif choice == '2':
    view_movies()
elif choice == '3':
    update_movie()
elif choice == '4':
    delete_movie()
elif choice == '5':
    search_movie()
elif choice == '6':
    display_total()
elif choice == '7':
    print("Exiting program...")
    break
else:
    print("Invalid option. Try again.\n")

menu()

```

```
----- MOVIE DATABASE CLI -----  
1. Add Movie  
2. View Movies  
3. Update Movie  
4. Delete a Movie  
5. Search a Movie  
6. Display Total Records  
7. Exit  
Select an option (1-7): 1
```

```
--- Add New Movie ---  
Generated Movie ID: M001  
Enter Title: 3 Idiots  
Enter Main Actor: Aaron Aquino  
Enter Director: Maurey Reyes  
Enter Genre: Liann  
Enter Gross Sales: 70  
Enter Rating (G, PG, R13, R16, X): R16  
Movie added successfully!
```

```
--- Add New Movie ---|  
Generated Movie ID: M001  
Enter Title: 3 Idiots  
Enter Main Actor: Aaron Aquino  
Enter Director: Maurey Reyes  
Enter Genre: Liann  
Enter Gross Sales: 70  
Enter Rating (G, PG, R13, R16, X): R16  
Movie added successfully!
```

```
Select an option (1-7): 2
```

```
--- Movie List ---  
ID: M001 | Title: 3 Idiots | Actor: Aaron Aquino | Director: Maurey Reyes | Genre: Liann | Gross Sales: 70.0 | Rating: R16  
ID: M002 | Title: Fengshui | Actor: Kris Aquino | Director: Jose Marie Chan | Genre: Horror | Gross Sales: 79.0 | Rating: R16  
ID: M003 | Title: Girl Boy Bakla Tomboy | Actor: Vice Ganda | Director: Ping Lacson | Genre: Comedy | Gross Sales: 80.0 | Rating: R13  
ID: M004 | Title: 7 Sundays | Actor: Ruru Madrid | Director: Dingdong Dantes | Genre: Drama | Gross Sales: 81.0 | Rating: PG  
ID: M005 | Title: Barcelona | Actor: Kitchie Nadal | Director: Coco Martin | Genre: Drama | Gross Sales: 82.0 | Rating: R13  
ID: M006 | Title: 18 Again | Actor: Zac Mandiq | Director: Aaron Aquino | Genre: Romance | Gross Sales: 83.0 | Rating: R13
```

```
Select an option (1-7): 3
```

```
--- Update Movie ---  
Enter Movie ID to update: M002  
Enter new Title: 47 Meters Down  
Enter new Main Actor: Mark Fho  
Enter new Director: Ji Lin  
Enter new Genre: Fantasy  
Enter new Gross Sales: 84  
Enter new Rating: PG  
Movie updated successfully!
```

```
Select an option (1-7): 4
```

```
--- Delete Movie ---
```

```
Enter Movie ID to delete: M003
```

```
Movie deleted successfully!
```

```
Select an option (1-7): 5
```

```
--- Search Movie ---
```

```
Enter Movie ID or Title to search: M001
```

```
('M001', '3 Idiots', 'Aaron Aquino', 'Maurey Reyes', 'Liann', Decimal('70.00'), 'R16')
```

```
Select an option (1-7): 6
```

```
Total number of movies: 5
```

```
----- MOVIE DATABASE CLI -----
```

1. Add Movie
2. View Movies
3. Update Movie
4. Delete a Movie
5. Search a Movie
6. Display Total Records
7. Exit

```
Select an option (1-7): 7
```

```
Exiting program...
```

```
Process finished with exit code 0
```