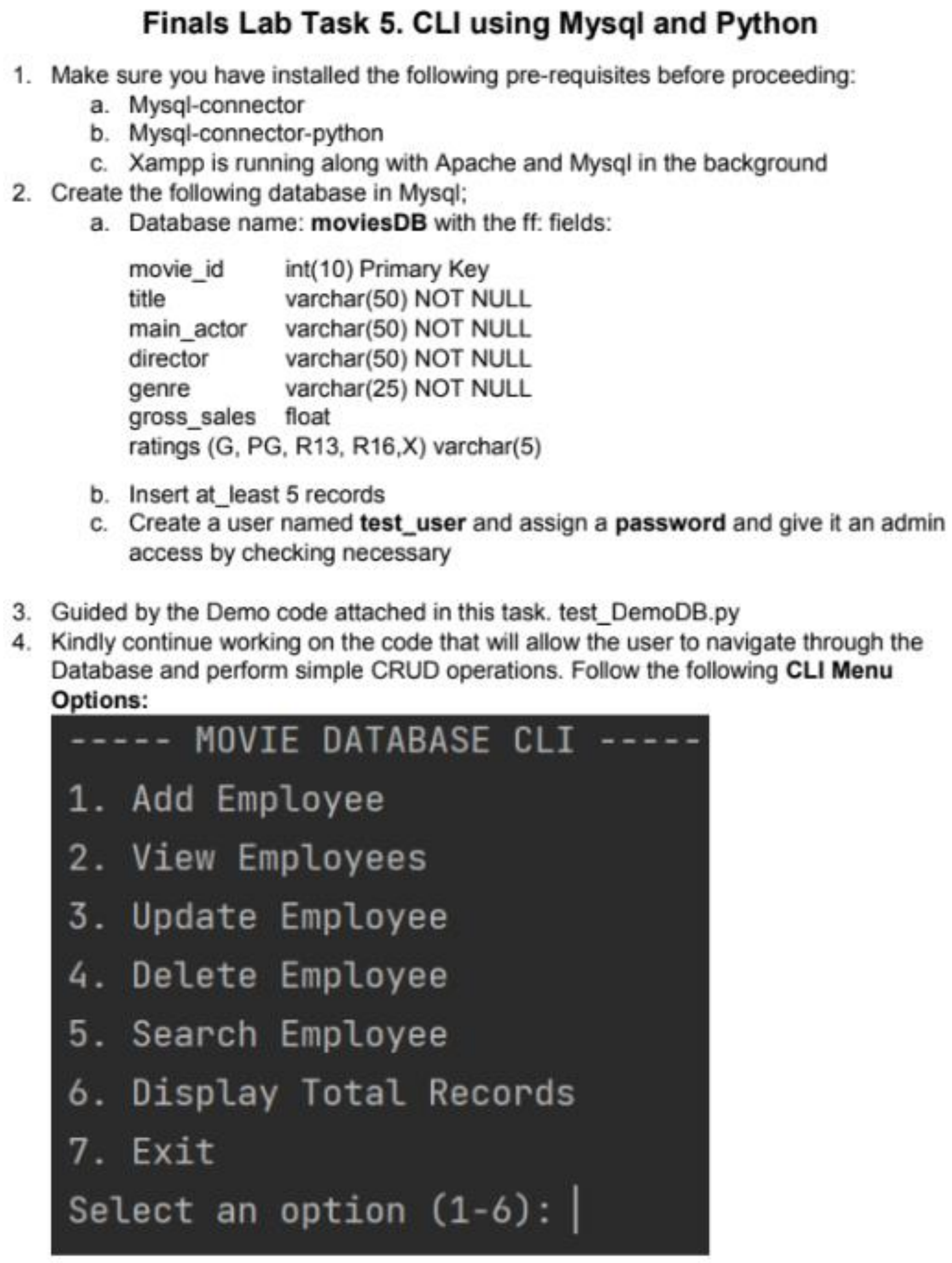
Mechure, Liann S.   November 29, 2025

BSCS – C204   7OOP

Finals Lab Task #1

Problem:



Finals Lab Task 5. CLI using Mysql and Python

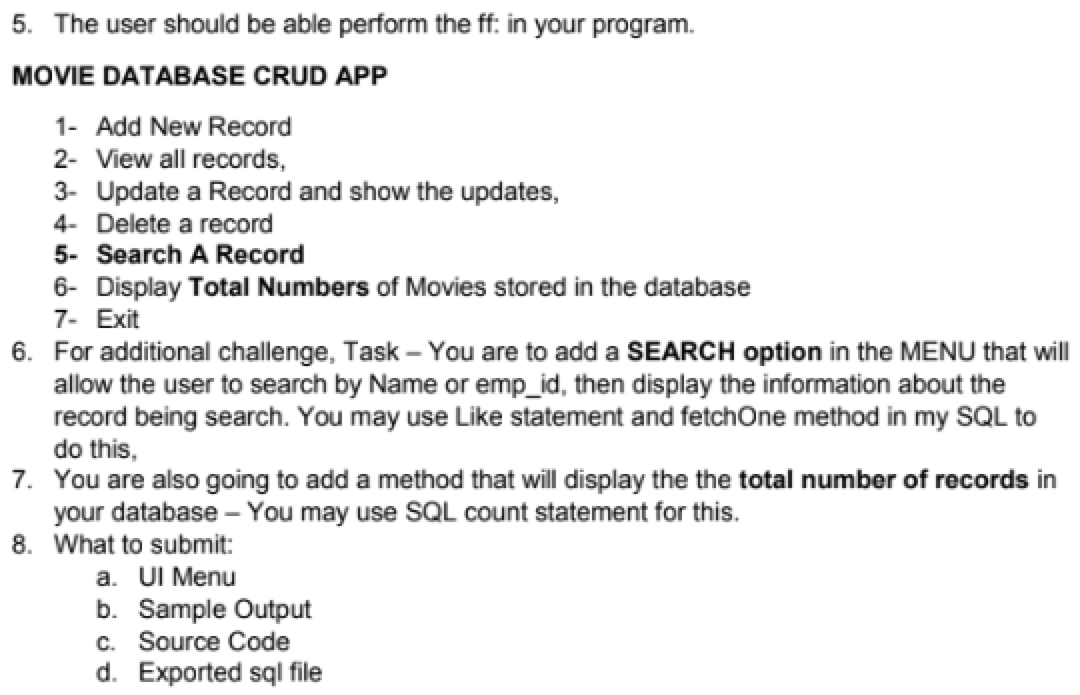
1. Make sure you have installed the following pre-requisites before proceeding:
    a. Mysql-connector
    b. Mysql-connector-python
    c. Xampp is running along with Apache and Mysql in the background
2. Create the following database in Mysql;
    a. Database name: **moviesDB** with the ff: fields:

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

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:**

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



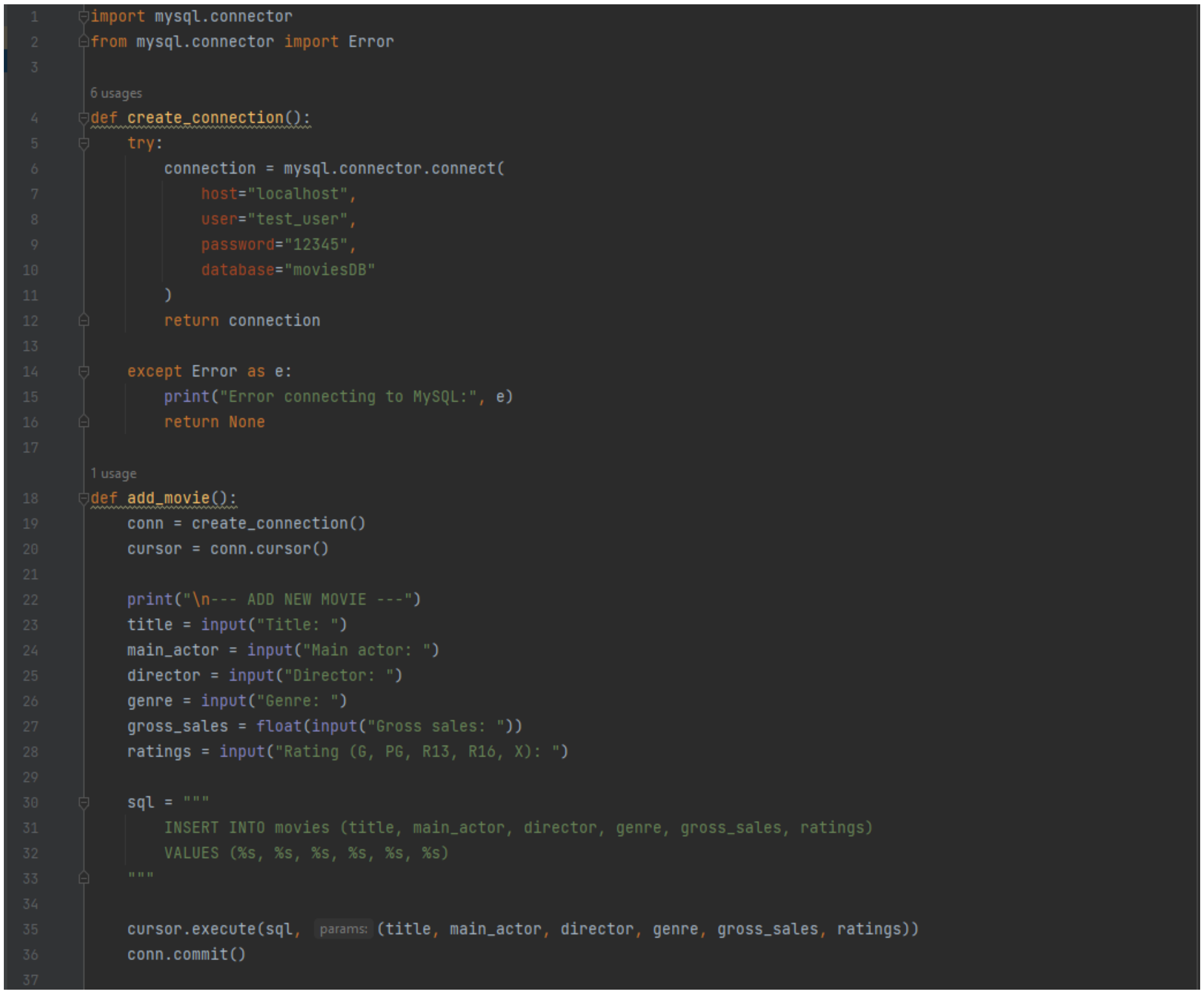
5. The user should be able perform the ff: in your program.

MOVIE DATABASE CRUD APP

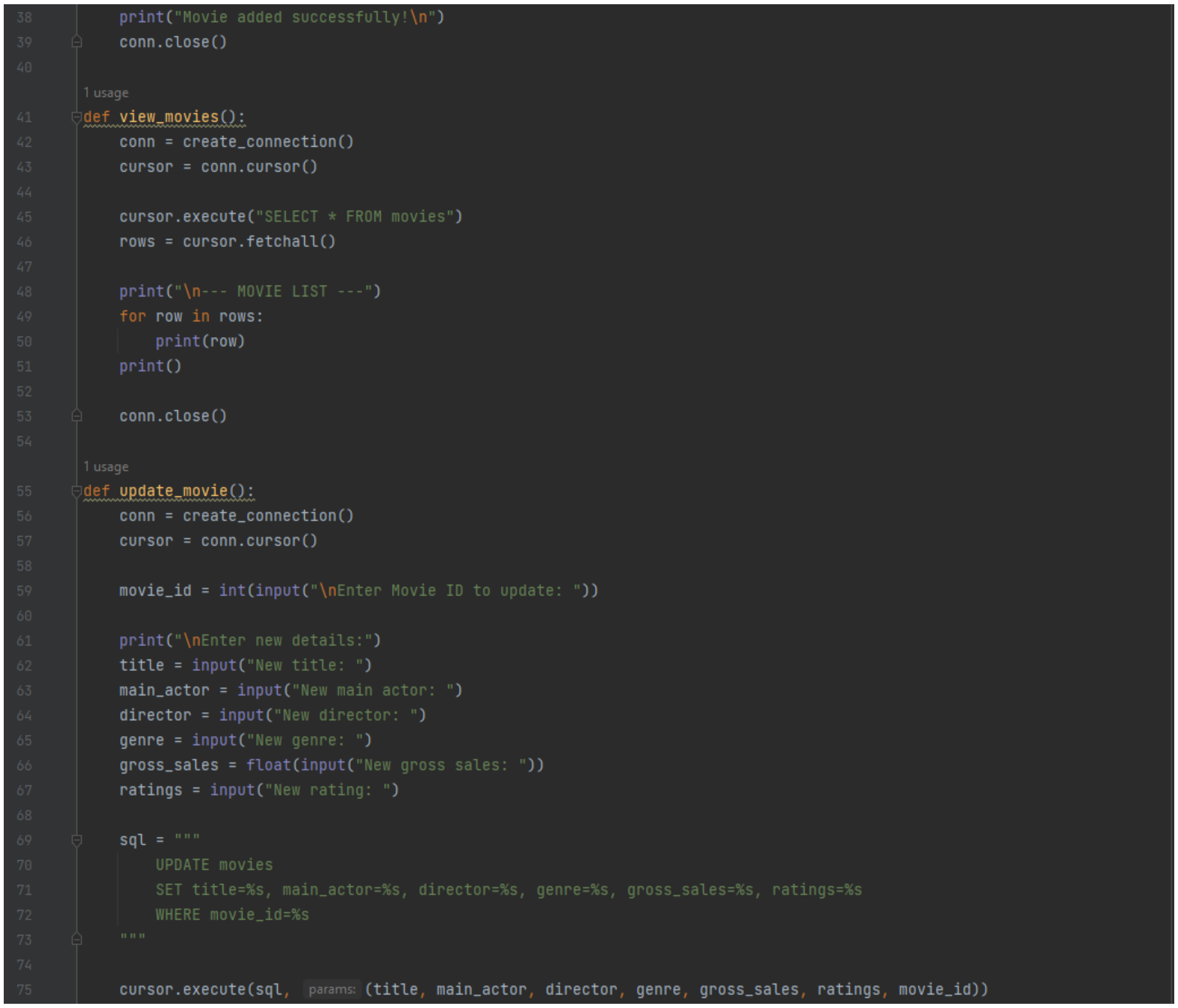
1- Add New Record
2- View all records,
3- Update a Record and show the updates,
4- Delete a record
5- Search A Record
6- Display **Total Numbers** of Movies stored in the database
7- Exit

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

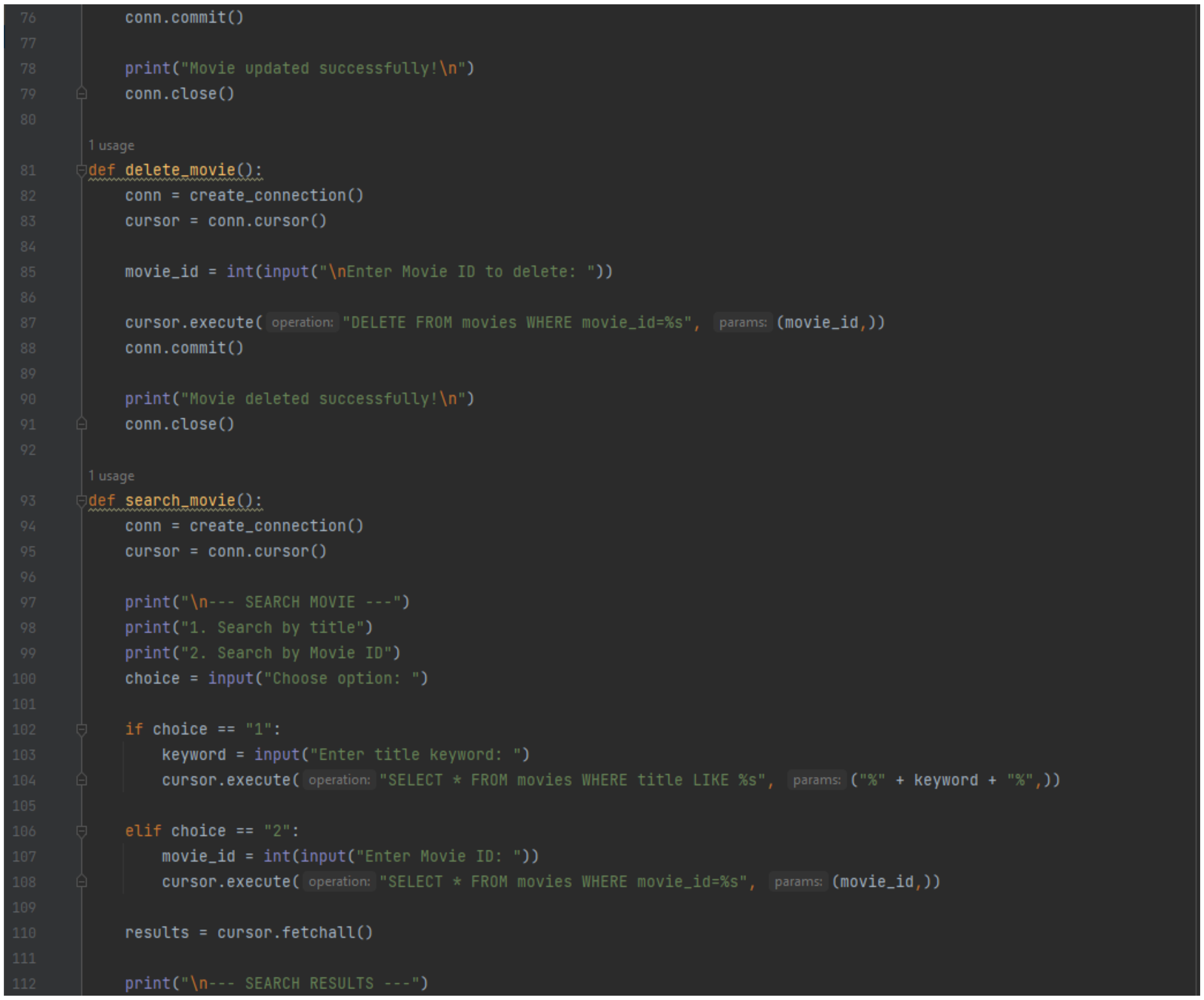
Source Code:



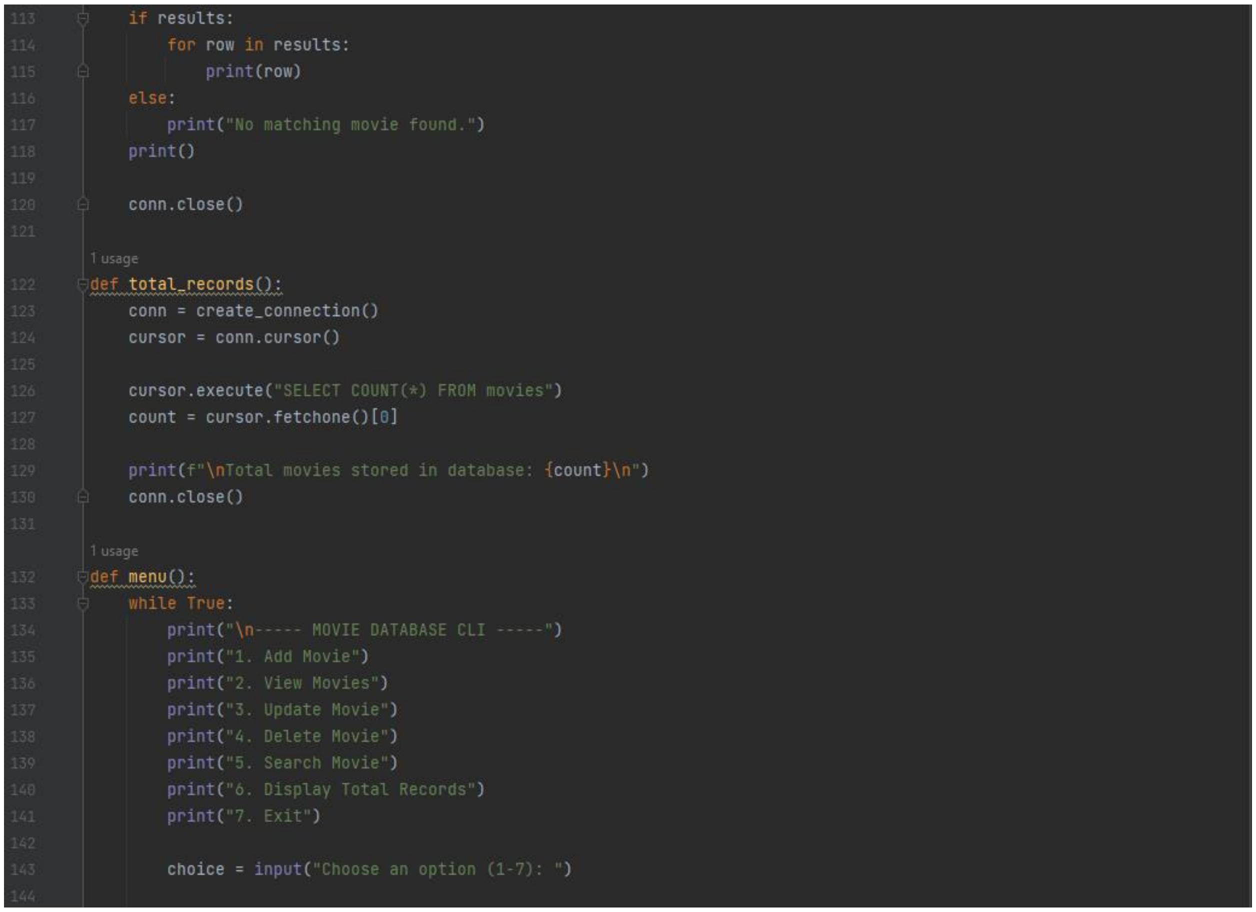
```python
import mysql.connector
from mysql.connector import Error

6 usages
def create_connection():
    try:
        connection = mysql.connector.connect(
            host="localhost",
            user="test_user",
            password="12345",
            database="moviesDB"
        )
        return connection

    except Error as e:
        print("Error connecting to MySQL:", e)
        return None

1 usage
def add_movie():
    conn = create_connection()
    cursor = conn.cursor()

    print("\n--- ADD NEW MOVIE ---")
    title = input("Title: ")
    main_actor = input("Main actor: ")
    director = input("Director: ")
    genre = input("Genre: ")
    gross_sales = float(input("Gross sales: "))
    ratings = input("Rating (G, PG, R13, R16, X): ")

    sql = """
        INSERT INTO movies (title, main_actor, director, genre, gross_sales, ratings)
        VALUES (%s, %s, %s, %s, %s, %s)
    """

    cursor.execute(sql,  params: (title, main_actor, director, genre, gross_sales, ratings))
    conn.commit()

```



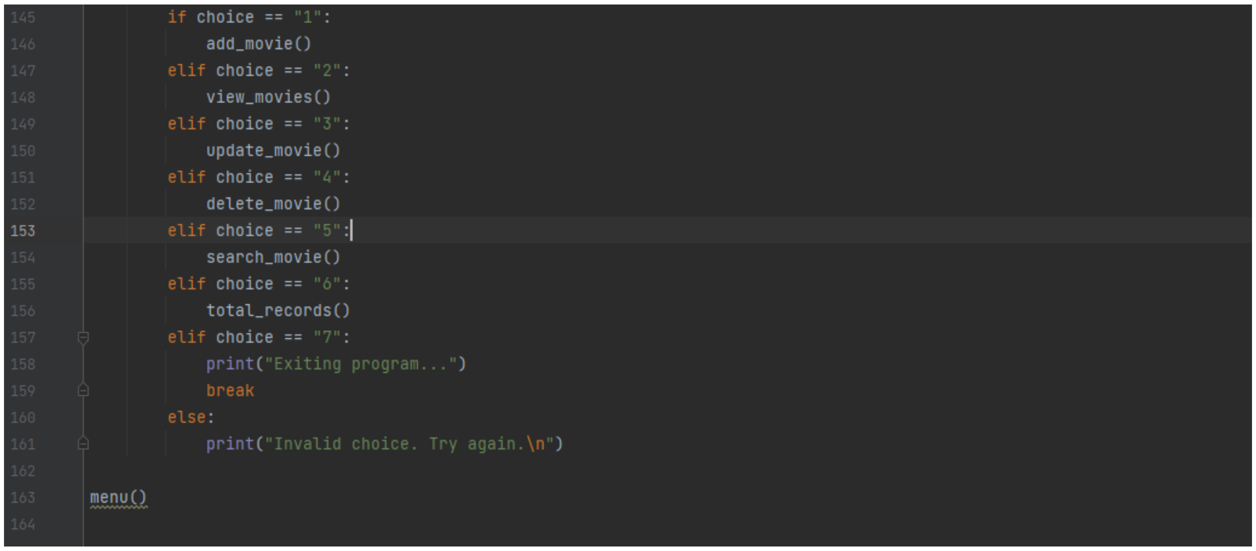
```python
    print("Movie added successfully!\n")
    conn.close()

1 usage
def view_movies():
    conn = create_connection()
    cursor = conn.cursor()

    cursor.execute("SELECT * FROM movies")
    rows = cursor.fetchall()

    print("\n--- MOVIE LIST ---")
    for row in rows:
        print(row)
    print()

    conn.close()

1 usage
def update_movie():
    conn = create_connection()
    cursor = conn.cursor()

    movie_id = int(input("\nEnter Movie ID to update: "))

    print("\nEnter new details:")
    title = input("New title: ")
    main_actor = input("New main actor: ")
    director = input("New director: ")
    genre = input("New genre: ")
    gross_sales = float(input("New gross sales: "))
    ratings = input("New rating: ")

    sql = """
        UPDATE movies
        SET title=%s, main_actor=%s, director=%s, genre=%s, gross_sales=%s, ratings=%s
        WHERE movie_id=%s
    """

    cursor.execute(sql,  params: (title, main_actor, director, genre, gross_sales, ratings, movie_id))
```



```
    conn.commit()

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

1 usage
def delete_movie():
    conn = create_connection()
    cursor = conn.cursor()

    movie_id = int(input("\nEnter Movie ID to delete: "))

    cursor.execute( operation: "DELETE FROM movies WHERE movie_id=%s",  params: (movie_id,))
    conn.commit()

    print("Movie deleted successfully!\n")
    conn.close()

1 usage
def search_movie():
    conn = create_connection()
    cursor = conn.cursor()

    print("\n--- SEARCH MOVIE ---")
    print("1. Search by title")
    print("2. Search by Movie ID")
    choice = input("Choose option: ")

    if choice == "1":
        keyword = input("Enter title keyword: ")
        cursor.execute( operation: "SELECT * FROM movies WHERE title LIKE %s",  params: ("%" + keyword + "%",))

    elif choice == "2":
        movie_id = int(input("Enter Movie ID: "))
        cursor.execute( operation: "SELECT * FROM movies WHERE movie_id=%s",  params: (movie_id,))

    results = cursor.fetchall()

    print("\n--- SEARCH RESULTS ---")
```

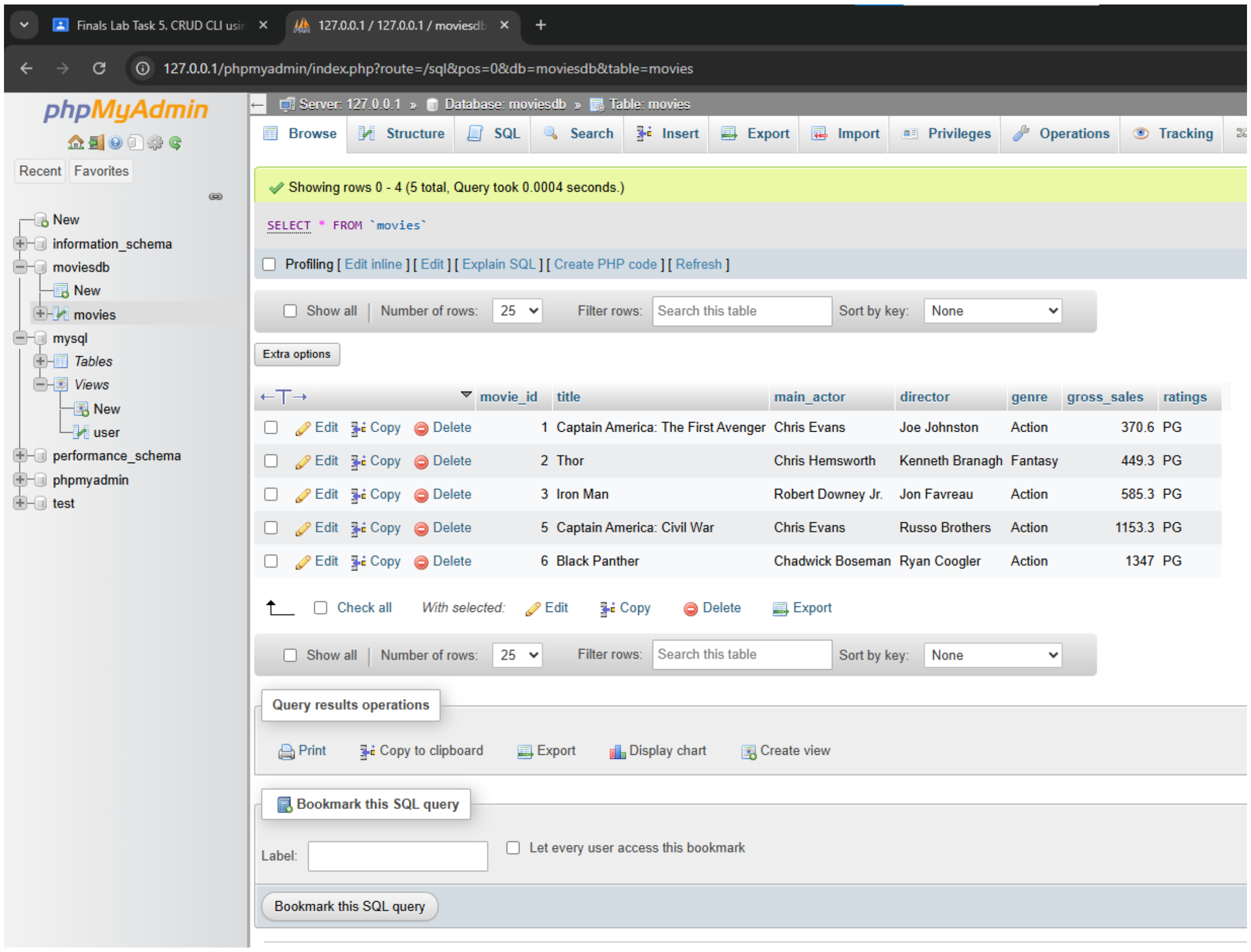


```python
    if results:
        for row in results:
            print(row)
    else:
        print("No matching movie found.")
    print()

    conn.close()

def total_records():
    conn = create_connection()
    cursor = conn.cursor()

    cursor.execute("SELECT COUNT(*) FROM movies")
    count = cursor.fetchone()[0]

    print(f"\nTotal movies stored in database: {count}\n")
    conn.close()

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

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

```



```python
        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":
            total_records()
        elif choice == "7":
            print("Exiting program...")
            break
        else:
            print("Invalid choice. Try again.\n")

menu()
```

Sample Output:



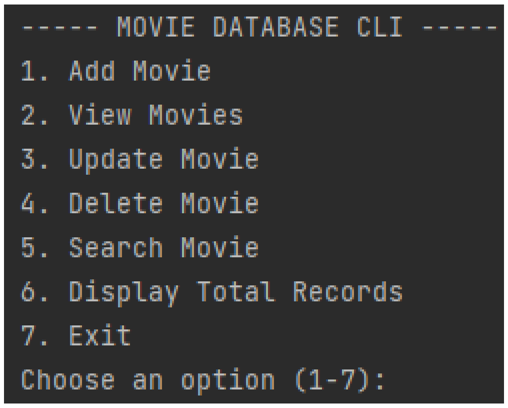
127.0.0.1/phpmyadmin/index.php?route=/sql&pos=0&db=moviesdb&table=movies

Server: 127.0.0.1 » Database: moviesdb » Table: movies

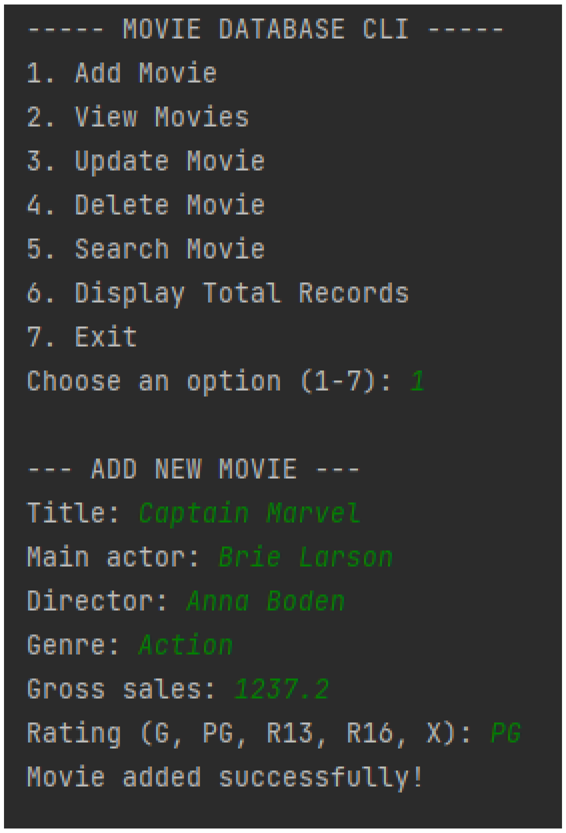
Showing rows 0 - 4 (5 total, Query took 0.0004 seconds.)

```
SELECT * FROM `movies`
```

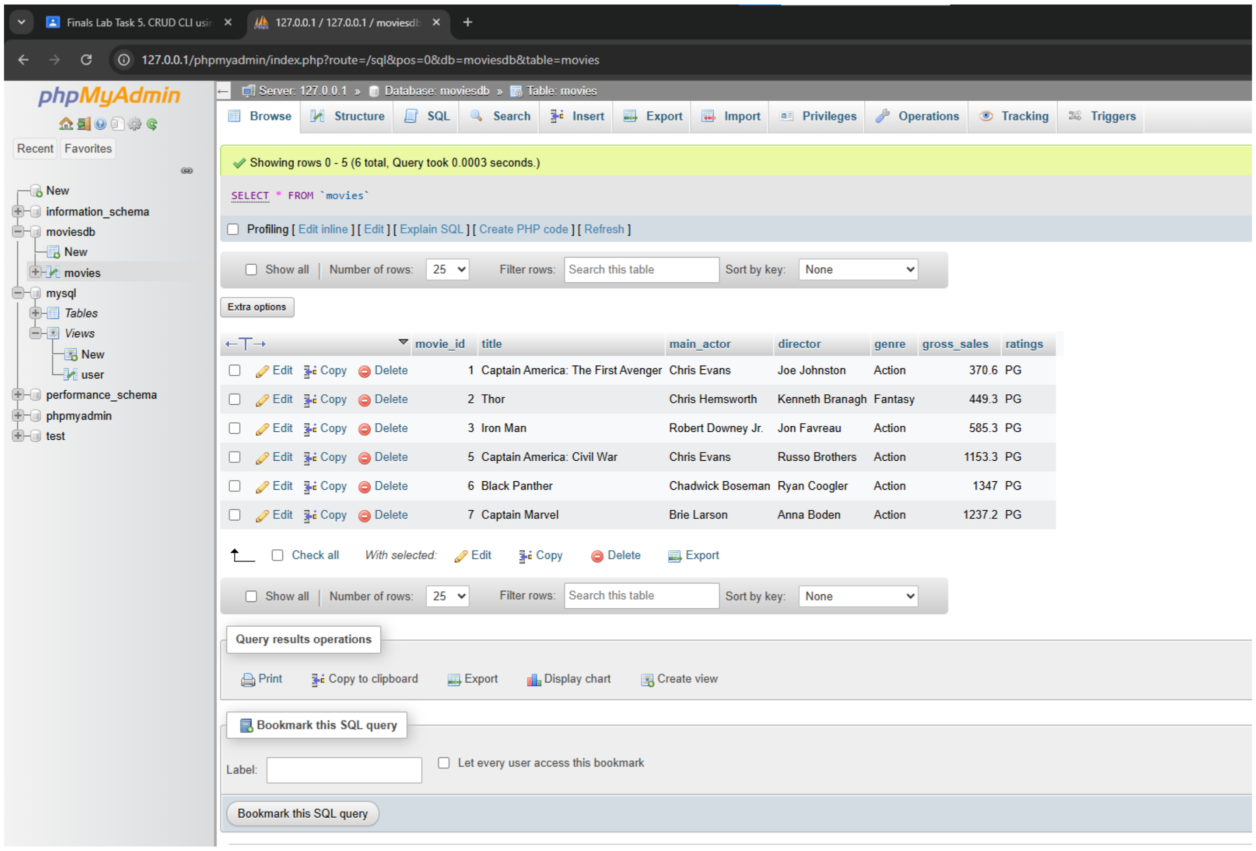
movie_id	title	main_actor	director	genre	gross_sales	ratings
1	Captain America: The First Avenger	Chris Evans	Joe Johnston	Action	370.6	PG
2	Thor	Chris Hemsworth	Kenneth Branagh	Fantasy	449.3	PG
3	Iron Man	Robert Downey Jr.	Jon Favreau	Action	585.3	PG
5	Captain America: Civil War	Chris Evans	Russo Brothers	Action	1153.3	PG
6	Black Panther	Chadwick Boseman	Ryan Coogler	Action	1347	PG



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



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

--- ADD NEW MOVIE ---
Title: Captain Marvel
Main actor: Brie Larson
Director: Anna Boden
Genre: Action
Gross sales: 1237.2
Rating (G, PG, R13, R16, X): PG
Movie added successfully!
```



Finals Lab Task 5. CRUD CLI usi | 127.0.0.1 / 127.0.0.1 / moviesdb

127.0.0.1/phpmyadmin/index.php?route=/sql&pos=0&db=moviesdb&table=movies

Server: 127.0.0.1 » Database: moviesdb » Table: movies

✓ Showing rows 0 - 5 (6 total, Query took 0.0003 seconds.)

```
SELECT * FROM `movies`
```

Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

Extra options

	movie_id	title	main_actor	director	genre	gross_sales	ratings
Edit Copy Delete	1	Captain America: The First Avenger	Chris Evans	Joe Johnston	Action	370.6	PG
Edit Copy Delete	2	Thor	Chris Hemsworth	Kenneth Branagh	Fantasy	449.3	PG
Edit Copy Delete	3	Iron Man	Robert Downey Jr.	Jon Favreau	Action	585.3	PG
Edit Copy Delete	5	Captain America: Civil War	Chris Evans	Russo Brothers	Action	1153.3	PG
Edit Copy Delete	6	Black Panther	Chadwick Boseman	Ryan Coogler	Action	1347	PG
Edit Copy Delete	7	Captain Marvel	Brie Larson	Anna Boden	Action	1237.2	PG

Check all | With selected: Edit Copy Delete Export

Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

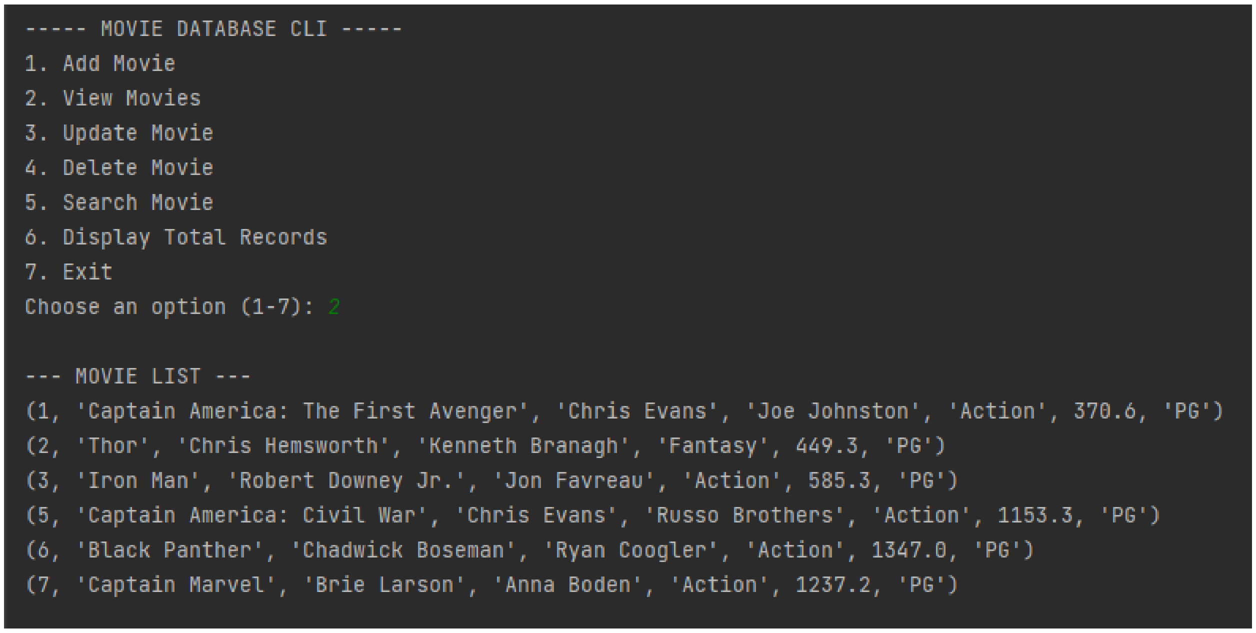
Query results operations

Print | Copy to clipboard | Export | Display chart | Create view

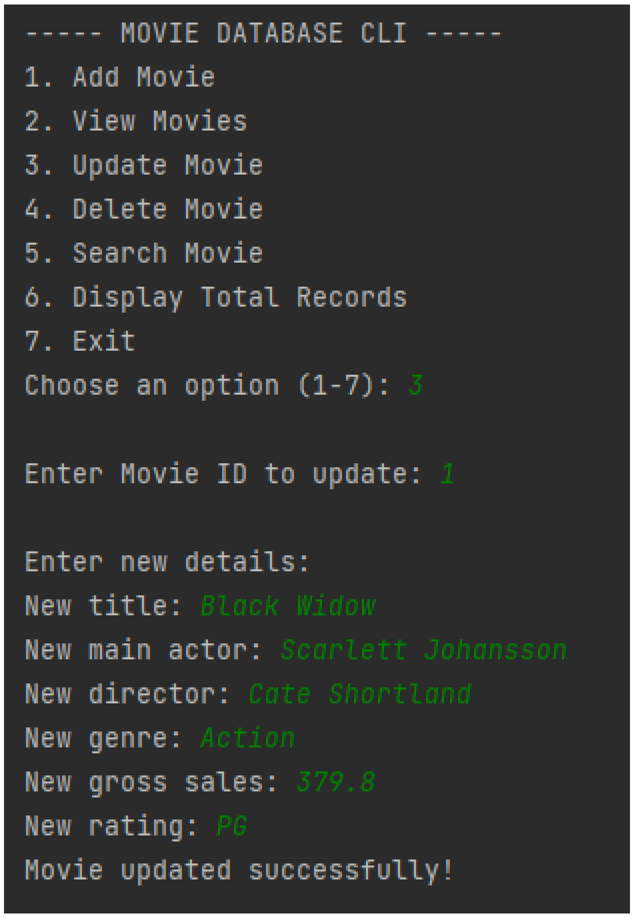
Bookmark this SQL query

Label: | Let every user access this bookmark

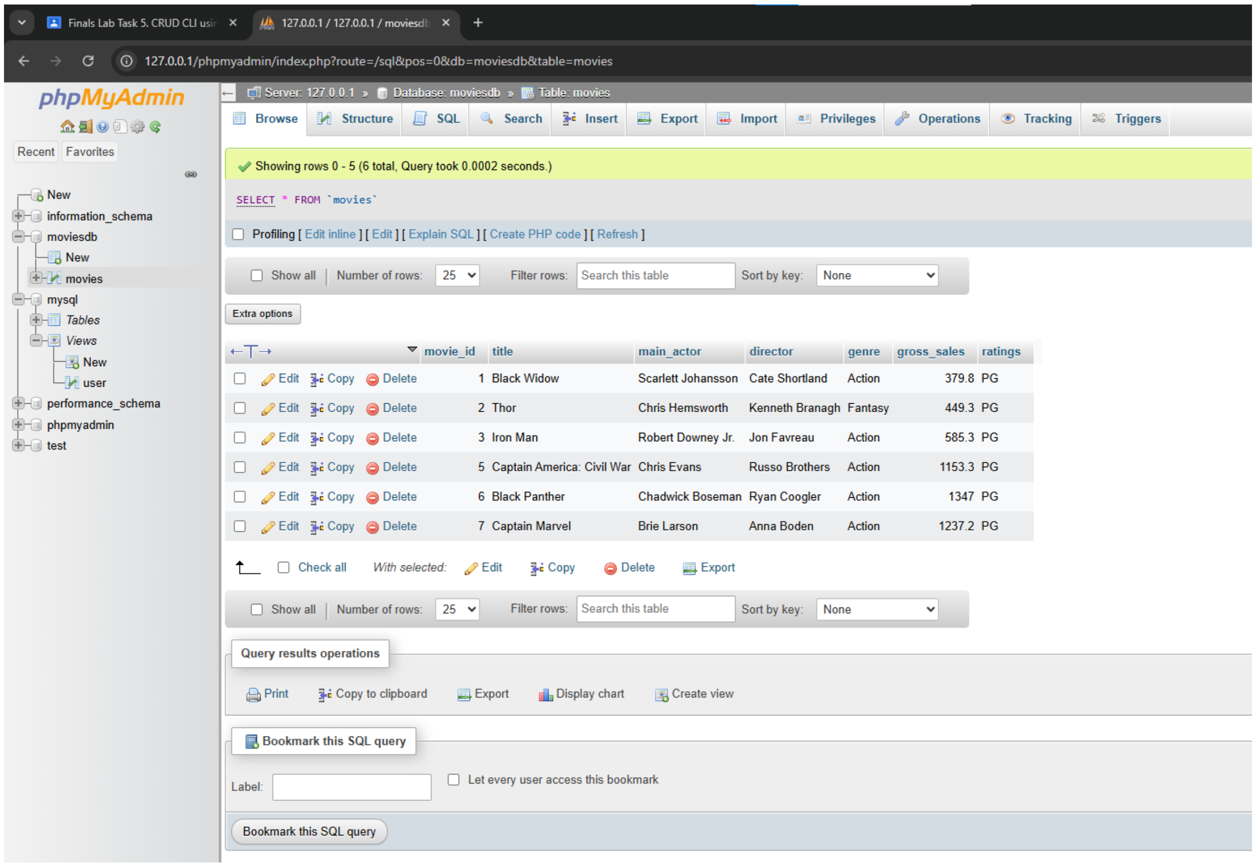
Bookmark this SQL query



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

--- MOVIE LIST ---
(1, 'Captain America: The First Avenger', 'Chris Evans', 'Joe Johnston', 'Action', 370.6, 'PG')
(2, 'Thor', 'Chris Hemsworth', 'Kenneth Branagh', 'Fantasy', 449.3, 'PG')
(3, 'Iron Man', 'Robert Downey Jr.', 'Jon Favreau', 'Action', 585.3, 'PG')
(5, 'Captain America: Civil War', 'Chris Evans', 'Russo Brothers', 'Action', 1153.3, 'PG')
(6, 'Black Panther', 'Chadwick Boseman', 'Ryan Coogler', 'Action', 1347.0, 'PG')
(7, 'Captain Marvel', 'Brie Larson', 'Anna Boden', 'Action', 1237.2, 'PG')
```



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

Enter Movie ID to update: 1

Enter new details:
New title: Black Widow
New main actor: Scarlett Johansson
New director: Cate Shortland
New genre: Action
New gross sales: 379.8
New rating: PG
Movie updated successfully!
```

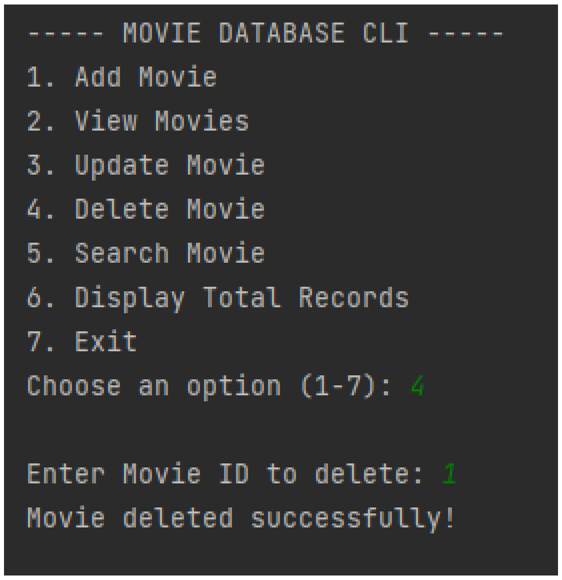


127.0.0.1/phpmyadmin/index.php?route=/sql&pos=0&db=moviesdb&table=movies

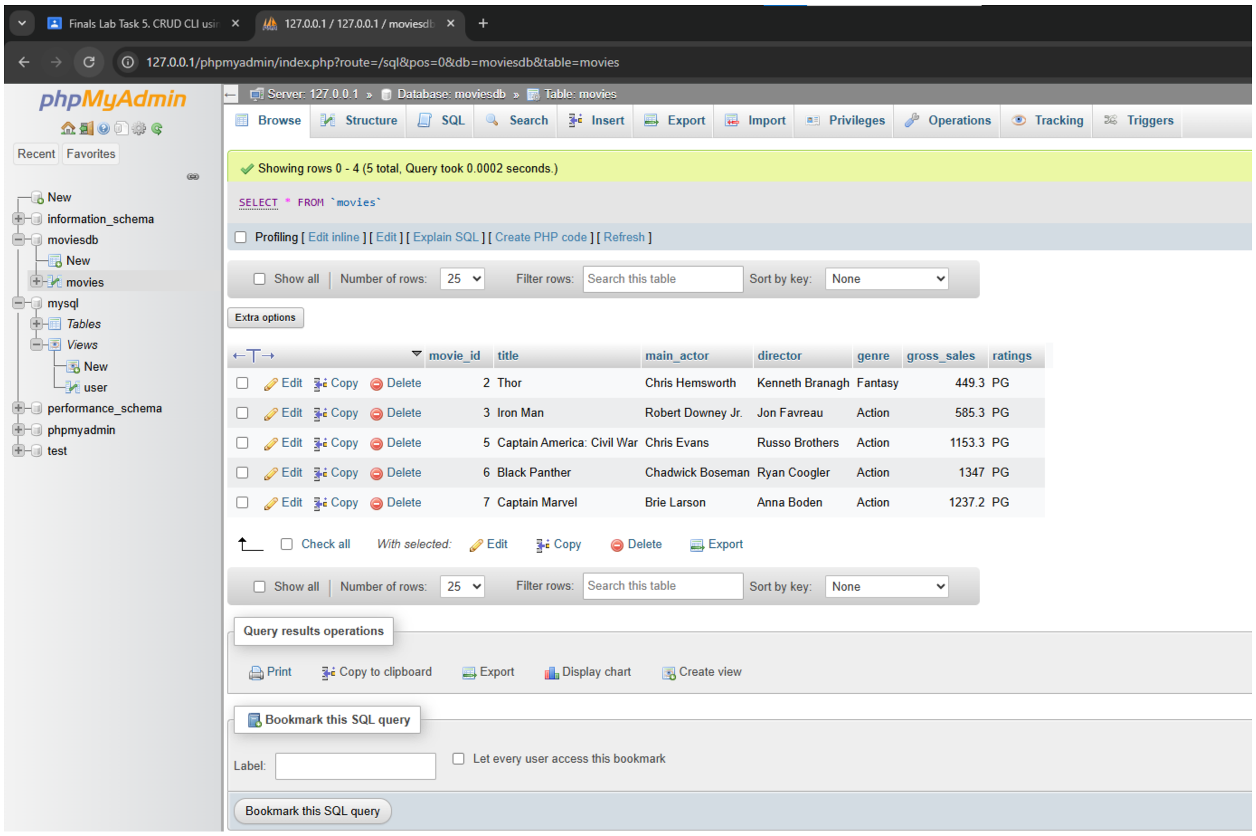
✓ Showing rows 0 - 5 (6 total, Query took 0.0002 seconds.)

```
SELECT * FROM `movies`
```

	movie_id	title	main_actor	director	genre	gross_sales	ratings
Edit Copy Delete	1	Black Widow	Scarlett Johansson	Cate Shortland	Action	379.8	PG
Edit Copy Delete	2	Thor	Chris Hemsworth	Kenneth Branagh	Fantasy	449.3	PG
Edit Copy Delete	3	Iron Man	Robert Downey Jr.	Jon Favreau	Action	585.3	PG
Edit Copy Delete	5	Captain America: Civil War	Chris Evans	Russo Brothers	Action	1153.3	PG
Edit Copy Delete	6	Black Panther	Chadwick Boseman	Ryan Coogler	Action	1347	PG
Edit Copy Delete	7	Captain Marvel	Brie Larson	Anna Boden	Action	1237.2	PG



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

Enter Movie ID to delete: 1
Movie deleted successfully!
```



127.0.0.1/phpmyadmin/index.php?route=/sql&pos=0&db=moviesdb&table=movies

✓ Showing rows 0 - 4 (5 total, Query took 0.0002 seconds.)

```
SELECT * FROM `movies`
```

Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

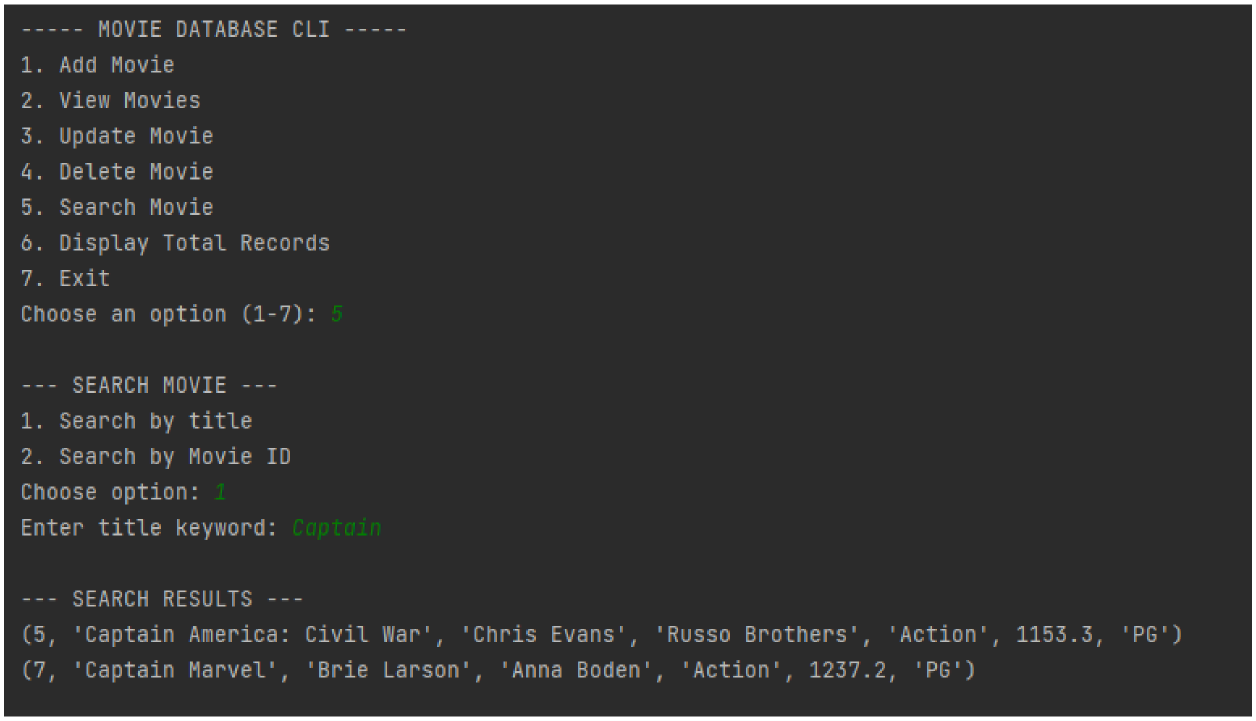
Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

	movie_id	title	main_actor	director	genre	gross_sales	ratings
Edit Copy Delete	2	Thor	Chris Hemsworth	Kenneth Branagh	Fantasy	449.3	PG
Edit Copy Delete	3	Iron Man	Robert Downey Jr.	Jon Favreau	Action	585.3	PG
Edit Copy Delete	5	Captain America: Civil War	Chris Evans	Russo Brothers	Action	1153.3	PG
Edit Copy Delete	6	Black Panther	Chadwick Boseman	Ryan Coogler	Action	1347	PG
Edit Copy Delete	7	Captain Marvel	Brie Larson	Anna Boden	Action	1237.2	PG

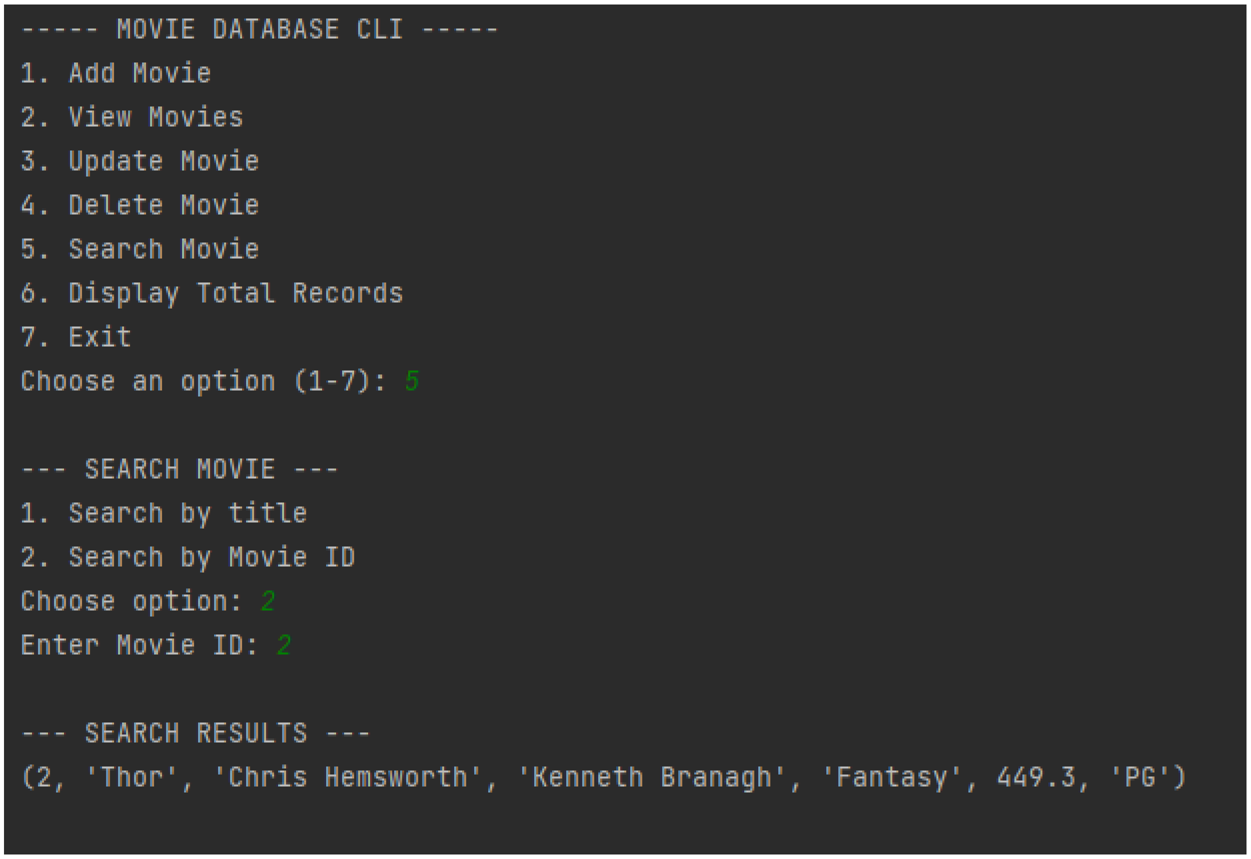
Check all | With selected: Edit | Copy | Delete | Export

Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

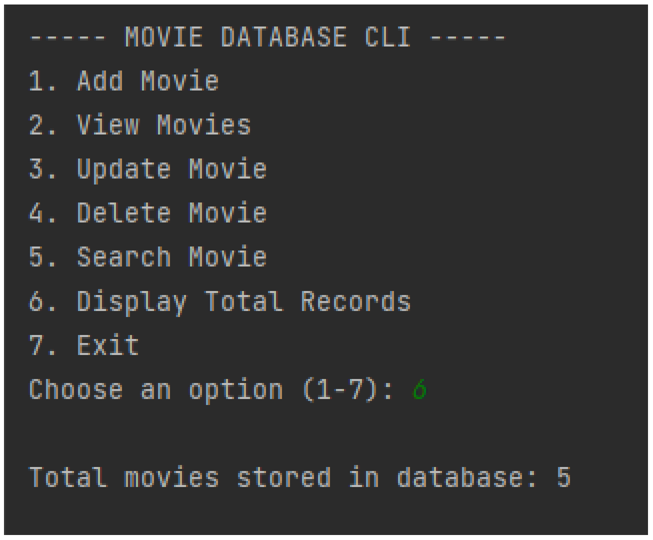
Label: | Let every user access this bookmark



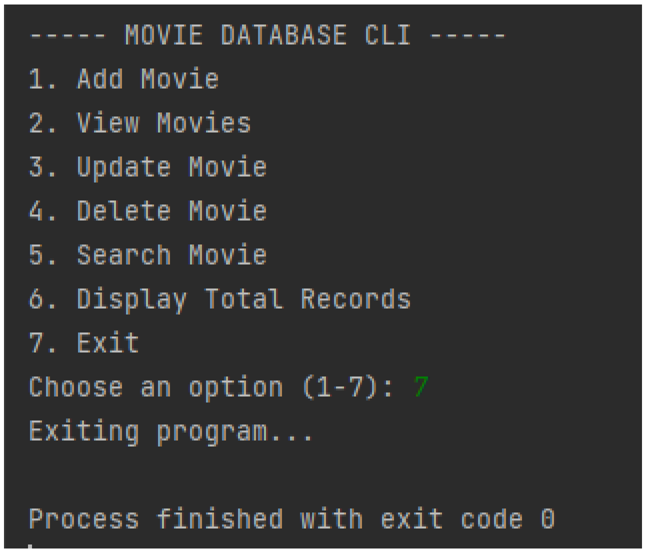
```
----- MOVIE DATABASE CLI -----
1. Add Movie
2. View Movies
3. Update Movie
4. Delete Movie
5. Search Movie
6. Display Total Records
7. Exit
Choose an option (1-7): 5

--- SEARCH MOVIE ---
1. Search by title
2. Search by Movie ID
Choose option: 1
Enter title keyword: Captain

--- SEARCH RESULTS ---
(5, 'Captain America: Civil War', 'Chris Evans', 'Russo Brothers', 'Action', 1153.3, 'PG')
(7, 'Captain Marvel', 'Brie Larson', 'Anna Boden', 'Action', 1237.2, 'PG')
```



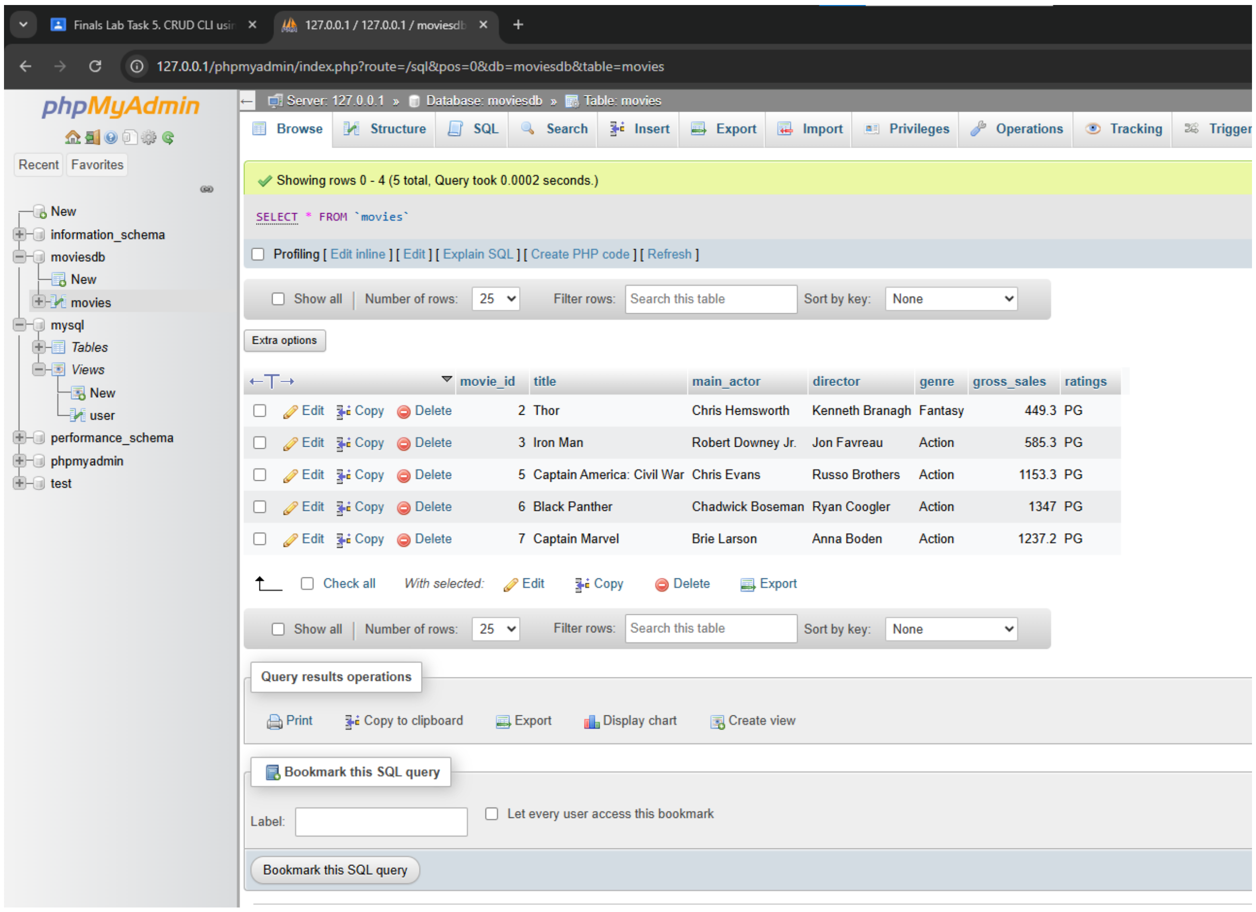
```
----- MOVIE DATABASE CLI -----
1. Add Movie
2. View Movies
3. Update Movie
4. Delete Movie
5. Search Movie
6. Display Total Records
7. Exit
Choose an option (1-7): 5

--- SEARCH MOVIE ---
1. Search by title
2. Search by Movie ID
Choose option: 2
Enter Movie ID: 2

--- SEARCH RESULTS ---
(2, 'Thor', 'Chris Hemsworth', 'Kenneth Branagh', 'Fantasy', 449.3, 'PG')
```



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

Total movies stored in database: 5
```



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

Process finished with exit code 0
```



127.0.0.1/phpmyadmin/index.php?route=/sql&pos=0&db=moviesdb&table=movies

Showing rows 0 - 4 (5 total, Query took 0.0002 seconds.)

```
SELECT * FROM `movies`
```

	movie_id	title	main_actor	director	genre	gross_sales	ratings
Edit Copy Delete	2	Thor	Chris Hemsworth	Kenneth Branagh	Fantasy	449.3	PG
Edit Copy Delete	3	Iron Man	Robert Downey Jr.	Jon Favreau	Action	585.3	PG
Edit Copy Delete	5	Captain America: Civil War	Chris Evans	Russo Brothers	Action	1153.3	PG
Edit Copy Delete	6	Black Panther	Chadwick Boseman	Ryan Coogler	Action	1347	PG
Edit Copy Delete	7	Captain Marvel	Brie Larson	Anna Boden	Action	1237.2	PG