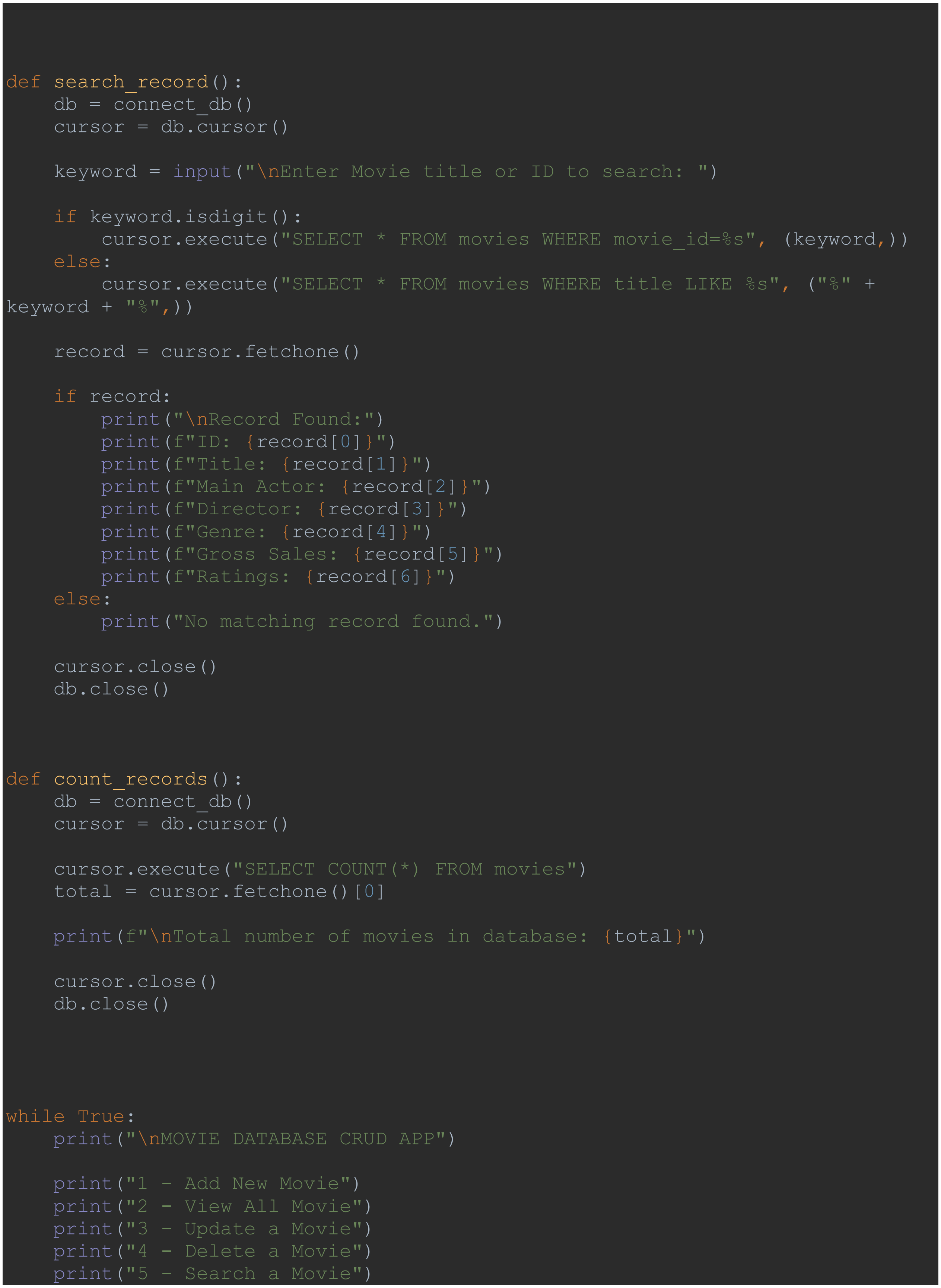
Bondoc, Moses

7OOP

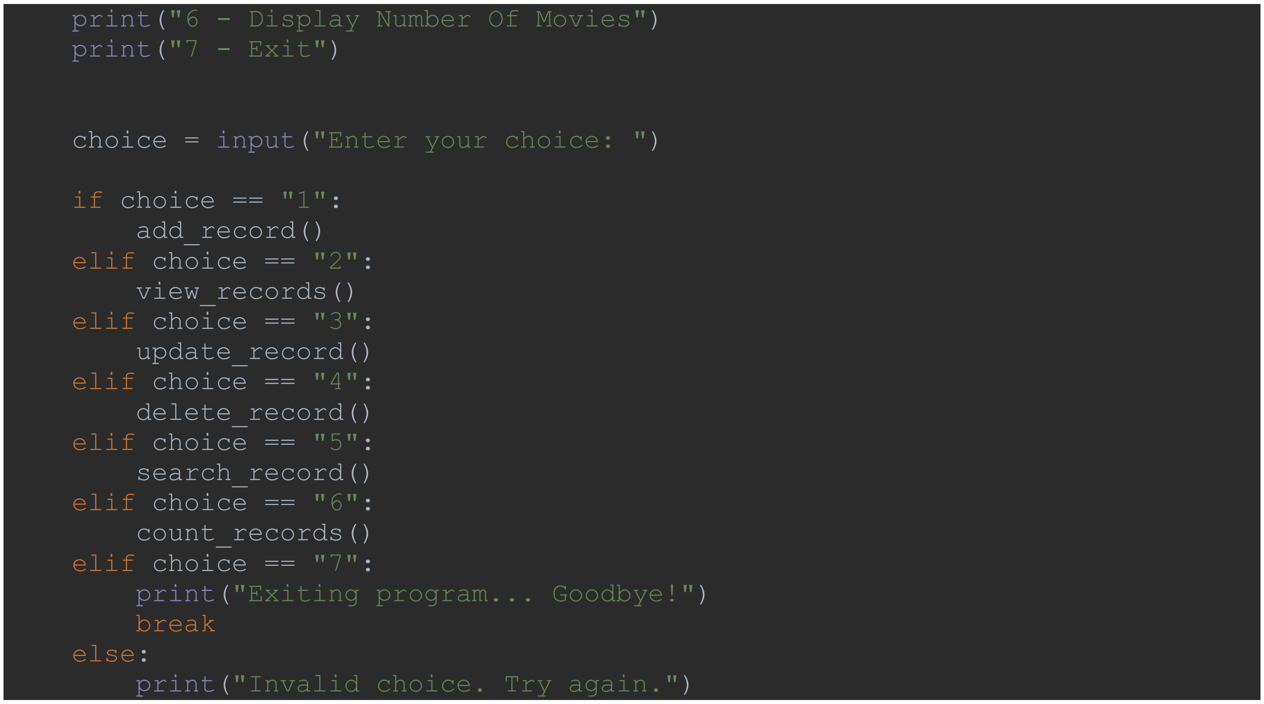
Program:

```
import mysql.connector


def connect_db():
    return mysql.connector.connect(
        host="localhost",
        user="test_user",
        password="yourpassword",
        database="moviesDB"
    )


def add_record():
    db = connect_db()
    cursor = db.cursor()

    print("\nADD NEW MOVIE RECORD")
    title = input("Title: ")
    actor = input("Main Actor: ")
    director = input("Director: ")
    genre = input("Genre: ")
    gross = float(input("Gross Sales: "))
    rating = 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)"
    values = (title, actor, director, genre, gross, rating)
    cursor.execute(sql, values)

    db.commit()
    print("\nMovie added successfully!")

    cursor.close()
    db.close()



def view_records():
    db = connect_db()
    cursor = db.cursor()

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

    print("\nMOVIE LIST:")

    for row in results:
        print(f"ID: {row[0]} | Title: {row[1]} | Actor: {row[2]} | Director:
{row[3]}")
```

```
    cursor.close()
    db.close()



def update_record():
    db = connect_db()
    cursor = db.cursor()

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

    print("\nEnter new values (leave blank to keep old value):")
    cursor.execute("SELECT * FROM movies WHERE movie_id = %s", (movie_id,))
    record = cursor.fetchone()

    if not record:
        print("Record not found.")
        return

    new_title = input(f"Title ({record[1]}): ") or record[1]
    new_actor = input(f"Main Actor ({record[2]}): ") or record[2]
    new_director = input(f"Director ({record[3]}): ") or record[3]
    new_genre = input(f"Genre ({record[4]}): ") or record[4]
    new_gross = input(f"Gross Sales ({record[5]}): ") or record[5]
    new_rating = input(f"Rating ({record[6]}): ") or record[6]

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

    values = (new_title, new_actor, new_director, new_genre, new_gross,
new_rating, movie_id)
    cursor.execute(sql, values)
    db.commit()

    print("\nRecord updated successfully!")

    cursor.close()
    db.close()



def delete_record():
    db = connect_db()
    cursor = db.cursor()

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

    sql = "DELETE FROM movies WHERE movie_id=%s"
    cursor.execute(sql, (movie_id,))
    db.commit()

    print("Record deleted successfully!")

    cursor.close()
    db.close()
```

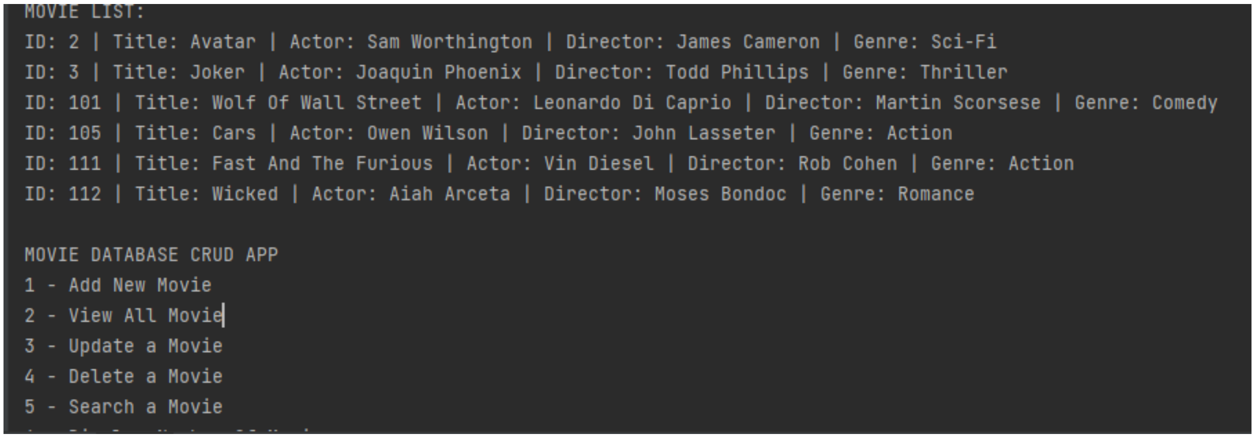


```
def search_record():
    db = connect_db()
    cursor = db.cursor()

    keyword = input("\nEnter Movie title or ID to search: ")

    if keyword.isdigit():
        cursor.execute("SELECT * FROM movies WHERE movie_id=%s", (keyword,))
    else:
        cursor.execute("SELECT * FROM movies WHERE title LIKE %s", ("%" +
keyword + "%",))

    record = cursor.fetchone()

    if record:
        print("\nRecord Found:")
        print(f"ID: {record[0]}")
        print(f"Title: {record[1]}")
        print(f"Main Actor: {record[2]}")
        print(f"Director: {record[3]}")
        print(f"Genre: {record[4]}")
        print(f"Gross Sales: {record[5]}")
        print(f"Ratings: {record[6]}")
    else:
        print("No matching record found.")

    cursor.close()
    db.close()



def count_records():
    db = connect_db()
    cursor = db.cursor()

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

    print(f"\nTotal number of movies in database: {total}")

    cursor.close()
    db.close()




while True:
    print("\nMOVIE DATABASE CRUD APP")

    print("1 - Add New Movie")
    print("2 - View All Movie")
    print("3 - Update a Movie")
    print("4 - Delete a Movie")
    print("5 - Search a Movie")
```

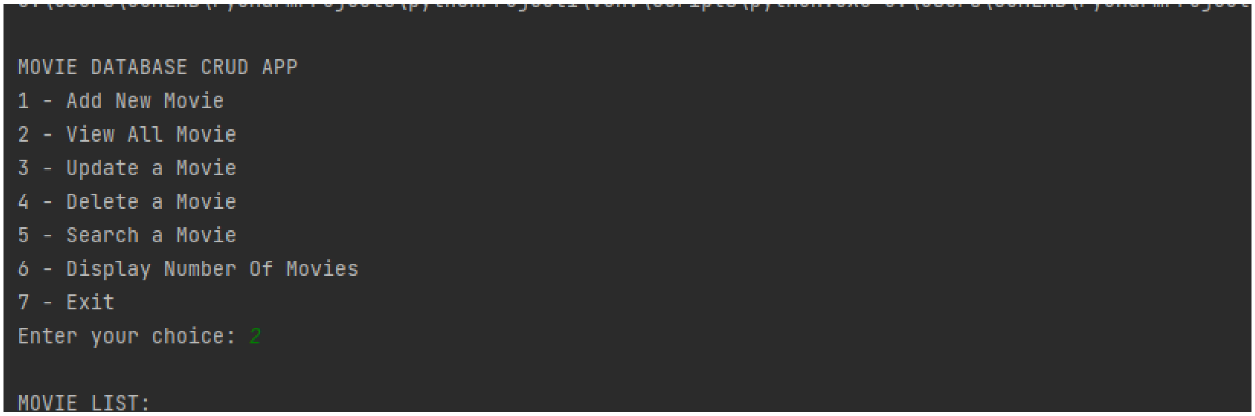


```
    print("6 - Display Number Of Movies")
    print("7 - Exit")


    choice = input("Enter your choice: ")

    if choice == "1":
        add_record()
    elif choice == "2":
        view_records()
    elif choice == "3":
        update_record()
    elif choice == "4":
        delete_record()
    elif choice == "5":
        search_record()
    elif choice == "6":
        count_records()
    elif choice == "7":
        print("Exiting program... Goodbye!")
        break
    else:
        print("Invalid choice. Try again.")
```

Output:



```
MOVIE LIST:
ID: 2 | Title: Avatar | Actor: Sam Worthington | Director: James Cameron | Genre: Sci-Fi
ID: 3 | Title: Joker | Actor: Joaquin Phoenix | Director: Todd Phillips | Genre: Thriller
ID: 101 | Title: Wolf Of Wall Street | Actor: Leonardo Di Caprio | Director: Martin Scorsese | Genre: Comedy
ID: 105 | Title: Cars | Actor: Owen Wilson | Director: John Lasseter | Genre: Action
ID: 111 | Title: Fast And The Furious | Actor: Vin Diesel | Director: Rob Cohen | Genre: Action
ID: 112 | Title: Wicked | Actor: Aiah Arceta | Director: Moses Bondoc | Genre: Romance

MOVIE DATABASE CRUD APP
1 - Add New Movie
2 - View All Movie
3 - Update a Movie
4 - Delete a Movie
5 - Search a Movie
```



```
MOVIE DATABASE CRUD APP
1 - Add New Movie
2 - View All Movie
3 - Update a Movie
4 - Delete a Movie
5 - Search a Movie
6 - Display Number Of Movies
7 - Exit
Enter your choice: 2

MOVIE LIST:
```