**CENTRAL HINDU GIRLS SCHOOL**

**KAMACHHA VARANASI**

****

**Session 2023-2024**

**Computer science**

**Topic – Movies ticket booking system**

**Submitted for: AISSCE**

**SUBMITTED TO:**

**VIVEK KUMAR SRIVASTAV**

**(COMPUTE TEACHER)**

**SUBMITTED BY:**

**SHREYA KUMARI**

**ROLL NO.:**

**CLASS XII(D**

**TABLE OF CONTENT**

**Topic page no:**

1. Certificate
2. Declaration
3. Acknowledgment
4. Hardware And Software Requirement
5. Code
6. output

**CERTIFICATE**

This is to certify that Ms. Shreya Kumari of class: **XII (D)** of **CENTRAL HINDU GIRLS SCHOOL** has done his project on the **MOVIE TICKET BOOKING SYSTEM** under my supervision. She has taken interest and has shown utmost sincerity in the completion of this project.

I certify this Project is up to my expectations & as per guidelines issued by **CBSE, NEW DELHI**.

Internal ExaminersignatureExternal Examiner signature

Principal Signature

**DECLARATION**

We hereby declare that the project entitled 'MOVIE TICKET BOOKING SYSTEM,' submitted as a mini-project in computer science to CENTRAL HINDU GIRLS SCHOOL, VARANASI (UP) is our genuine work done under the guidance of Vivek Kumar Srivastav.

DATE- NAME-Shreya Kumari

PLACE-

**ACKNOWLEGMENT**

I am grateful to Vivek Kumar Srivastav, my computer science teacher, who consistently provided valuable suggestions and guidance throughout the completion of my project. He has been a source of inspiration and has helped me to understand and remember important details of the project. The success of my project is attributed to his guidance.

Name of the student: Shreya Kumari.

Roll no:

Hardware and software Required

HARDWARE

* PC
* RAM
* Processor

SOFTWARE

* PYTHON 3.12.0
* sqlite3
* Python connector
* Operating system

**CODE**

import sqlite3

import random

class MovieTicketBookingSystem:

    def \_\_init\_\_(self):

        self.conn = sqlite3.connect('movie\_ticket\_system.db')

        self.create\_tables()

    def create\_tables(self):

        cursor = self.conn.cursor()

        cursor.execute('''

            CREATE TABLE IF NOT EXISTS users (

                username TEXT PRIMARY KEY,

                password TEXT

            )

        ''')

        cursor.execute('''

            CREATE TABLE IF NOT EXISTS movies (

                movie\_id INTEGER PRIMARY KEY,

                movie\_name TEXT

            )

        ''')

        cursor.execute('''

            CREATE TABLE IF NOT EXISTS bookings (

                username TEXT,

                movie\_id INTEGER,

                num\_tickets INTEGER,

                total\_cost INTEGER,

                FOREIGN KEY (username) REFERENCES users (username),

                FOREIGN KEY (movie\_id) REFERENCES movies (movie\_id)

            )

        ''')

        self.conn.commit()

    def register\_user(self, username, password):

        cursor = self.conn.cursor()

        cursor.execute('INSERT INTO users VALUES (?, ?)', (username, password))

        self.conn.commit()

        print("Registration successful.")

    def login\_user(self, username, password):

        cursor = self.conn.cursor()

        cursor.execute('SELECT \* FROM users WHERE username=? AND password=?', (username, password))

        result = cursor.fetchone()

        if result:

            print("Login successful.")

            return True

        else:

            print("Invalid username or password. Please try again.")

            return False

    def display\_movies(self):

        cursor = self.conn.cursor()

        cursor.execute('SELECT \* FROM movies')

        movies = cursor.fetchall()

        print("\nAvailable Movies:")

        for movie\_id, movie\_name in movies:

            print(f"{movie\_id}. {movie\_name}")

    def add\_movie(self, movie\_name):

        cursor = self.conn.cursor()

        cursor.execute('INSERT INTO movies (movie\_name) VALUES (?)', (movie\_name,))

        self.conn.commit()

        print(f"\nMovie '{movie\_name}' added successfully.")

    def add\_movies\_bulk(self, movies):

        cursor = self.conn.cursor()

        for movie\_name in movies:

            cursor.execute('INSERT INTO movies (movie\_name) VALUES (?)', (movie\_name,))

        self.conn.commit()

        print(f"\n{len(movies)} movies added successfully.")

    def book\_ticket(self, username, movie\_id, num\_tickets):

        cursor = self.conn.cursor()

        cursor.execute('SELECT \* FROM movies WHERE movie\_id=?', (movie\_id,))

        movie = cursor.fetchone()

        if movie:

            cost\_per\_ticket = random.randint(8, 15)

            total\_cost = cost\_per\_ticket \* num\_tickets

            cursor.execute('INSERT INTO bookings VALUES (?, ?, ?, ?)', (username, movie\_id, num\_tickets, total\_cost))

            self.conn.commit()

            print(f"\nBooking successful for {num\_tickets} tickets for {movie[1]}.")

            print(f"Total Cost: ${total\_cost}")

        else:

            print("Invalid movie selection.")

    def cancel\_booking(self, username):

        cursor = self.conn.cursor()

        cursor.execute('SELECT \* FROM bookings WHERE username=?', (username,))

        booking = cursor.fetchone()

        if booking:

            cursor.execute('DELETE FROM bookings WHERE username=?', (username,))

            self.conn.commit()

            print(f"\nBooking canceled for {booking[2]} tickets for movie ID {booking[1]}.")

        else:

            print("No active booking found.")

def main():

    movie\_system = MovieTicketBookingSystem()

    while True:

        print("\nMovie Ticket Booking System:")

        print("1. Register")

        print("2. Login")

        print("3. Display Movies")

        print("4. Add Movie")

        print("5. Add Movies (Bulk)")

        print("6. Book Ticket")

        print("7. Cancel Booking")

        print("8. Exit")

        choice = input("Enter your choice (1-8): ")

        if choice == '1':

            username = input("Enter username: ")

            password = input("Enter password: ")

            movie\_system.register\_user(username, password)

        elif choice == '2':

            username = input("Enter username: ")

            password = input("Enter password: ")

            if movie\_system.login\_user(username, password):

                while True:

                    inner\_choice = input("\n1. Display Movies\n2. Add Movie\n3. Add Movies (Bulk)\n4. Book Ticket\n5. Cancel Booking\n6. Logout\nEnter your choice (1-6): ")

                    if inner\_choice == '1':

                        movie\_system.display\_movies()

                    elif inner\_choice == '2':

                        movie\_name = input("Enter the name of the movie: ")

                        movie\_system.add\_movie(movie\_name)

                    elif inner\_choice == '3':

                        movies\_input = input("Enter multiple movie names separated by commas: ")

                        movies = [movie.strip() for movie in movies\_input.split(',')]

                        movie\_system.add\_movies\_bulk(movies)

                    elif inner\_choice == '4':

                        movie\_system.display\_movies()

                        movie\_id = int(input("Enter the movie ID: "))

                        num\_tickets = int(input("Enter the number of tickets: "))

                        movie\_system.book\_ticket(username, movie\_id, num\_tickets)

                    elif inner\_choice == '5':

                        movie\_system.cancel\_booking(username)

                    elif inner\_choice == '6':

                        print("Logging out.")

                        break

                    else:

                        print("Invalid choice. Please enter a number between 1 and 6.")

        elif choice == '3':

            movie\_system.display\_movies()

        elif choice == '4':

            movie\_name = input("Enter the name of the movie: ")

            movie\_system.add\_movie(movie\_name)

        elif choice == '5':

            movies\_input = input("Enter multiple movie names separated by commas: ")

            movies = [movie.strip() for movie in movies\_input.split(',')]

            movie\_system.add\_movies\_bulk(movies)

        elif choice == '6':

            print("Login required to book a ticket.")

        elif choice == '7':

            print("Login required to cancel a booking.")

        elif choice == '8':

            print("Exiting Movie Ticket Booking System. Goodbye!")

            break

        else:

            print("Invalid choice. Please enter a number between 1 and 8.")

if \_\_name\_\_ == "\_\_main\_\_":

    main()

**OUTPUT**

Movie Ticket Booking System:

1. Register

2. Login

3. Display Movies

4. Add Movie

5. Add Movies (Bulk)

6. Book Ticket

7. Cancel Booking

8. Exit

Enter your choice (1-8): 3

Available Movies:

1. animal

2. Harry potter

3. Fantastic beast

4. Bridge of spies

5. Avatar

6. Iron man3

7. spiderman far from home

Movie Ticket Booking System:

1. Register

2. Login

3. Display Movies

4. Add Movie

5. Add Movies (Bulk)

6. Book Ticket

7. Cancel Booking

8. Exit

Enter your choice (1-8): 1

Enter username: Shreya Kumari

Enter password: 12345678

Registration successful.

Movie Ticket Booking System:

1. Register

2. Login

3. Display Movies

4. Add Movie

5. Add Movies (Bulk)

6. Book Ticket

7. Cancel Booking

8. Exit

Enter your choice (1-8): 2

Enter username: Shreya Kumari

Enter password: 12345678

Login successful.

1. Display Movies

2. Add Movie

3. Add Movies (Bulk)

4. Book Ticket

5. Cancel Booking

6. Logout

Enter your choice (1-6): 1

Available Movies:

1. animal

2. Harry potter

3. Fantastic beast

4. Bridge of spies

5. Avatar

6. Iron man3

7. spiderman far from home

1. Display Movies

2. Add Movie

3. Add Movies (Bulk)

4. Book Ticket

5. Cancel Booking

6. Logout

Enter your choice (1-6): 2

Enter the name of the movie: Oppenheimer

Movie 'Oppenheimer' added successfully.

1. Display Movies

2. Add Movie

3. Add Movies (Bulk)

4. Book Ticket

5. Cancel Booking

6. Logout

Enter your choice (1-6): 3

Enter multiple movie names separated by commas: The marvel, Barbie, Aquaman,Lift,The return, For all mankind,Reacher

7 movies added successfully.

1. Display Movies

2. Add Movie

3. Add Movies (Bulk)

4. Book Ticket

5. Cancel Booking

6. Logout

Enter your choice (1-6): 4

Available Movies:

1. animal

2. Harry potter

3. Fantastic beast

4. Bridge of spies

5. Avatar

6. Iron man3

7. spiderman far from home

8. Oppenheimer

9. The marvel

10. Barbie

11. Aquaman

12. Lift

13. The return

14. For all mankind

15. Reacher

Enter the movie ID: 8

Enter the number of tickets: 2

Booking successful for 2 tickets for Oppenheimer.

Total Cost: $22

1. Display Movies

2. Add Movie

3. Add Movies (Bulk)

4. Book Ticket

5. Cancel Booking

6. Logout

Enter your choice (1-6): 5

Booking canceled for 2 tickets for movie ID 8.

1. Display Movies

2. Add Movie

3. Add Movies (Bulk)

4. Book Ticket

5. Cancel Booking

6. Logout

Enter your choice (1-6): 6

Logging out.

Movie Ticket Booking System:

1. Register

2. Login

3. Display Movies

4. Add Movie

5. Add Movies (Bulk)

6. Book Ticket

7. Cancel Booking

8. Exit

Enter your choice (1-8): 6

Login required to book a ticket.

Movie Ticket Booking System:

1. Register

2. Login

3. Display Movies

4. Add Movie

5. Add Movies (Bulk)

6. Book Ticket

7. Cancel Booking

8. Exit

Enter your choice (1-8): 8

Exiting Movie Ticket Booking System. Goodbye!

**BIBLIOGRAPHY**

1. Class 11 Book(Sumitra Arora Python)
2. Class 12 Book(Sumitra Arora Python)
3. www.python.com
4. www.wikipedia.com
5. www.google.com