**LAB SESSION 11**

**Question 1: Composition**

Create a Book class that has title, author, and a Date object for the publication date. Implement a method to display the book details.

**Code:**

#include <iostream>

using namespace std;

int startlab11()

{

    cout << "Name: Saad Ali Khan(SE-23083)" << endl;

    cout << "Start of Lab 11" << endl;

    return 0;

}

class Date

{

public:

    int day, month, year;

    Date(int d, int m, int y) : day(d), month(m), year(y) {}

};

class Book

{

private:

    string title;

    string author;

    Date publicationDate;

public:

    Book(const string &t, const string &a, const Date &d) : title(t), author(a), publicationDate(d) {}

    void display() const

    {

        cout << "Title: " << title << "\nAuthor: " << author

             << "\nPublication Date: " << publicationDate.day << "/"

             << publicationDate.month << "/" << publicationDate.year << endl;

    }

};

int l11q1()

{

    Date date(15, 7, 2021);

    Book book("The Great Gatsby", "F. Scott Fitzgerald", date);

    book.display();

    return 0;

    return 0;

}

int main()

{

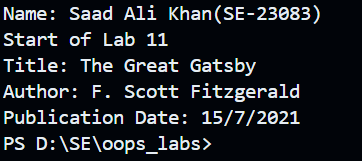
    startlab11();

    l11q1();

    return 0;

}

**Output:**

****

**Question 2: Aggregation**

Create a Library class that can hold multiple Book objects. Implement methods to add books and display all books in the library.

**Code:**

#include <iostream>

#include <vector>

#include <string>

using namespace std;

int startlab11()

{

    cout << "Name: Saad Ali Khan(SE-23083)" << endl;

    cout << "Lab 11" << endl;

    return 0;

}

class Date

{

public:

    int day, month, year;

    Date(int d, int m, int y) : day(d), month(m), year(y) {}

};

class Book

{

private:

    string title;

    string author;

    Date publicationDate;

public:

    Book(const string &t, const string &a, const Date &d) : title(t), author(a), publicationDate(d) {}

    void display() const

    {

        cout << "Title: " << title << "\nAuthor: " << author

             << "\nPublication Date: " << publicationDate.day << "/"

             << publicationDate.month << "/" << publicationDate.year << endl;

    }

};

class Library

{

private:

    vector<Book> books;

public:

    void addBook(const Book &book)

    {

        books.push\_back(book);

    }

    void displayBooks() const

    {

        for (const auto &book : books)

        {

            book.display();

            cout << endl;

        }

    }

};

int l11q2()

{

    Library library;

    Date date1(15, 7, 2021);

    Book book1("The Great Gatsby", "F. Scott Fitzgerald", date1);

    Date date2(1, 1, 2000);

    Book book2("Harry Potter and the Philosopher's Stone", "J.K. Rowling", date2);

    library.addBook(book1);

    library.addBook(book2);

    library.displayBooks();

    return 0;

    return 0;

}

int main()

{

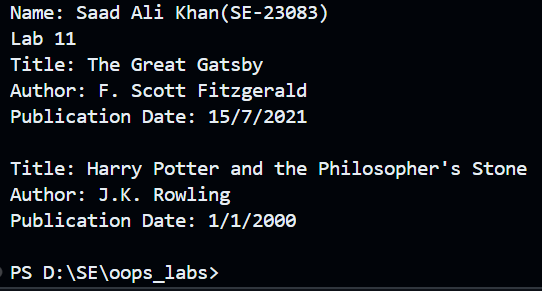
    startlab11();

    l11q2();

    return 0;

}

**Output:**

****

**Question 3: Composition with push\_back**

Create a Playlist class that contains a collection of Song objects. Each Song object should have a title, artist, and duration. Implement methods to add songs and display the playlist.

**Code:**

#include <iostream>

#include <vector>

#include <string>

using namespace std;

int startlab11()

{

    cout << "Name: Saad Ali Khan(SE-23083)" << endl;

    cout << "Lab 11" << endl;

    return 0;

}

class Song

{

private:

    string title;

    string artist;

    int duration; // duration in seconds

public:

    Song(const string &t, const string &a, int d) : title(t), artist(a), duration(d) {}

    void display() const

    {

        cout << "Title: " << title << "\nArtist: " << artist << "\nDuration: " << duration << " seconds" << endl;

    }

};

class Playlist

{

private:

    vector<Song> songs;

public:

    void addSong(const Song &song)

    {

        songs.push\_back(song);

    }

    void displayPlaylist() const

    {

        for (const auto &song : songs)

        {

            song.display();

            cout << endl;

        }

    }

};

int l11q3()

{

    Playlist playlist;

    Song song1("Shape of You", "Ed Sheeran", 263);

    Song song2("Blinding Lights", "The Weeknd", 200);

    playlist.addSong(song1);

    playlist.addSong(song2);

    playlist.displayPlaylist();

    return 0;

}

int main()

{

    startlab11();

    l11q3();

    return 0;

}

**Output:**