VIT - Vellore

Name: RONIT MEXSON .
Email: ronit.mexson2024@vitstudent.ac.in
Roll no: 24BAI0036
Phone: 9999999999
Branch: ARUMUGA ARUN R_OOPS
Department: admin
Batch: VL2024250502365
Degree: admin

Scan to verify results

BCSE102P_Structured and Object Oriented Programming Lab_VL2024250502365

VIT V_Structured and OOP_Lab 6_COD_Easy_Multiple inheritance

Attempt : 1
Total Mark : 20
Marks Obtained : 20

Section 1 : Coding

1. Problem Statement

Design a ticket booking system that handles both concert tickets and movie tickets. Implement the system using the concept of multiple inheritance. The system should allow the user to input the details of a concert ticket and a movie ticket, and display the booking details.

Define a base class Ticket and two derived classes ConcertTicket and MovieTicket. The Ticket class stores the details of an event and its price, while the derived classes ConcertTicket and MovieTicket inherit from Ticket and provide specific implementations for their respective events.

Define a class Booking, which inherits from both ConcertTicket and MovieTicket. The Booking class represents a booking that includes both a

concert ticket and a movie ticket. It stores the details of both events and provides a function displayBookingDetails to display the booking details.

Answer

```
#include <iostream>
#include <iomanip>
#include <string>
using namespace std;

class Ticket {
protected:
    string event;
    int price;
public:
    Ticket(string e, int p) : event(e), price(p) {}
};

class ConcertTicket : public Ticket {
public:
    ConcertTicket(string e, int p) : Ticket(e, p) {}
    void displayConcertDetails() {
        cout << "Concert Details: " << endl;
        cout << "Event: " << event << endl;
        cout << "Price: $" << price << "\n" << endl;
    }
};

class MovieTicket : public Ticket {
public:
    MovieTicket(string e, int p) : Ticket(e, p) {}
    void displayMovieDetails() {
        cout << "Movie Details: " << endl;
        cout << "Event: " << event << endl;
        cout << "Price: $" << price << endl;
    }
};

class Booking : public ConcertTicket, public MovieTicket {
public:
    Booking(string concertEvent, int concertPrice, string movieEvent, int
moviePrice)
```

```
    : ConcertTicket(concertEvent, concertPrice), MovieTicket(movieEvent,
moviePrice) {}

    void displayBookingDetails() {
        displayConcertDetails();
        displayMovieDetails();
    }
};

int main() {
    string concertEvent, movieEvent;
    int concertPrice, moviePrice;

    getline(cin, concertEvent);
    cin >> concertPrice;
    cin.ignore();
    getline(cin, movieEvent);
    cin >> moviePrice;

    Booking booking(concertEvent, concertPrice, movieEvent, moviePrice);
    booking.displayBookingDetails();

    return 0;
}
```

Status : Correct **Marks : 10/10**

2. Problem Statement

Harry, a recent graduate, is excited about buying his first car and considering a loan. To calculate the total interest paid, he wants to design a class structure using multiple inheritance.

Create a class named Loan inheriting from the classes: Principal and InterestRate. This program calculates the total interest paid over the loan period, aiding Harry in understanding the financial implications.

Principal class - stores the price as a protected attributeInterestRate class - stores interest rate as a protected attributeLoan class - calculates total interest

Note: Total Interest = price * interest rate * years.

Answer

```
// You are using GCC
#include <iostream>
#include <iomanip>
using namespace std;

class Principal {
protected:
    double price;
public:
    Principal(double p) : price(p) {}
};

class InterestRate {
protected:
    double rate;
public:
    InterestRate(double r) : rate(r) {}
};

class Loan : public Principal, public InterestRate {
public:
    Loan(double p, double r) : Principal(p), InterestRate(r) {}

    double calculateTotalInterest(int years) {
        return price * rate * years;
    }
};

int main() {
    double price, rate;
    int years;

    cin >> price;
    cin >> rate;
    cin >> years;

    Loan loan(price, rate);
    double totalInterest = loan.calculateTotalInterest(years);
```

24BAI0036

```
    cout << fixed << setprecision(2);
    cout << "Total interest paid: Rs." << totalInterest << endl;

    return 0;
}
```

Status : Correct **Marks : 10/10**

24BAI0036