

VIT - Vellore

Name: DIVYANSHU SINGH .

Email: divyanshu.singh2024a@vitstudent.ac.in

Roll no: 24BCT0101

Phone: 9999999999

Branch: PRIYADHARSINI M_OOPS

Department: admin

Batch: VL2024250502354

Degree: admin

Scan to verify results



BCSE102P_Structured and Object Oriented Programming Lab_VL2024250502354

VIT V_Structured and OOP_Lab 5_COD_Hard_Friend Functions and Friend Classes

Attempt : 1

Total Mark : 10

Marks Obtained : 10

Section 1 : Coding

1. Problem Statement

Seema is building a ticket reservation system and she wants to reserve tickets and cancel for a particular id. She created two classes: Ticket and Reservation. Each Ticket has a private attribute ticketId, and each Reservation has a private attribute reservedTickets (an array of tickets). The Reservation class contains a friend function `cancelReservation(Reservation&, const Ticket&)` that allows a ticket to be canceled from a reservation.

Implement the classes and the `cancelReservation` function and write a program to manage ticket reservations.

Answer

```
#include <iostream>
#include <vector>
using namespace std;
class Ticket {
private:
    int ticketId;
public:
    Ticket(int id) : ticketId(id) {}
    int getId() const { return ticketId; }
    friend class Reservation;
};
class Reservation {
private:
    vector<Ticket> reservedTickets;
public:
    void addTicket(int id) {
        reservedTickets.push_back(Ticket(id));
    }
    void displayTickets() const {
        cout << "Reserved Tickets: ";
        for (const auto &ticket : reservedTickets) {
            cout << ticket.getId() << " ";
        }
        cout << endl;
    }
    friend void cancelReservation(Reservation &res, int ticketId);
};
void cancelReservation(Reservation &res, int ticketId) {
    bool found = false;
    for (auto it = res.reservedTickets.begin(); it != res.reservedTickets.end(); ++it) {
        if (it->getId() == ticketId) {
            res.reservedTickets.erase(it);
            found = true;
            break;
        }
    }
    if (found) {
        cout << "Ticket with ID " << ticketId << " has been canceled." << endl;
    } else {
        cout << "Ticket with ID " << ticketId << " not found in the reservation." << endl;
    }
}
```

```
}  
}  
int main() {  
    int N, ticketId, cancelId;  
    cin >> N;  
    Reservation res;  
  
    for (int i = 0; i < N; i++) {  
        cin >> ticketId;  
        res.addTicket(ticketId);  
    }  
  
    cin >> cancelId;  
    res.displayTickets();  
    cancelReservation(res, cancelId);  
    res.displayTickets();  
  
    return 0;  
}
```

Status : Correct

Marks : 10/10