OBJECT ORIENTED CONCEPTS

SYNOPSIS

NAME	PRN	Roll No.
Ajinkya Kankirad	1032231916	11
Tanishka Kumar	1032231986	28
Pranav Jadhao	1032231941	18
Diya Patil	1032231899	06

TRAIN RESERVATION SYSTEM

PROBLEM STATEMENT:

The objective is to develop a comprehensive **Train Reservation System** using C++ that efficiently manages the processes associated with train ticket bookings and train schedule management. The system will have distinct roles for **users** and **admins**, providing role-based access and functionality tailored to each type of user.

The system includes a **secure login mechanism**, ensuring that users can log in using valid credentials before performing any operations. If a user account does not exist, the system should facilitate a user-friendly registration process.

FUNCTIONALITIES:

1. User login and registration.
2. Passenger ticket booking and cancellation.
3. Admin train schedule modification.
4. Admin train cancellation

C++ Features we will be using:

1. Inheritance:

In our problem statement , we will be having a '**Person**' class, and '**User'** and the '**Admin**' classes inherit from it. This means they automatically inherit common properties and methods from Person, such as a name or login information, but can also have their own unique attributes and behaviors such as source , destination etc.

2. Constructors:

In this case, the object '**Train**' is initialized with specific values for its Train Number, Source, and Destination attributes. This ensures that each train object has its own unique set of data.

3. Exception Handling:

By using try-catch blocks, the code can catch potential exceptions like invalid inputs, booking failures, or cancellations that don't exist. When an exception occurs, the program can handle it gracefully and provide informative messages to the user.

4. Virtual Functions:

Our problem statement allows us to use the concept of polymorphism. In our case, both ' **User** ' and ' **Admin** ' objects have a ' **viewInformation** ' method, but the implementation of this method could be different for each class, depending on the specific information they need to access.

Diya Patil :

Ajinkya Kankirad :

Tanishka Kumar :

Pranav Jadhao

SIGNATURE

flowchart fun

Book Ticket

Operations

User Login

Login Check — Exists

Start

Cancel Ticket

Register Success

Register User — Does Not Exist

Register Fail

Modify Schedule

Admin Operations

Admin Login

Admin Check

Cancel Train

Train Reservation System

source, destination — Unique Attributes

User

viewInformation() — method

Person — Inherits

Classes Used

Inheritance

admin level

Admin — Inherits

viewInformation() — method — Unique Attributes

Initialize Train — Constructors

C++ Features

Handle Exceptions — Exception Handling

Polymorphism in viewInformation() — Virtual Functions