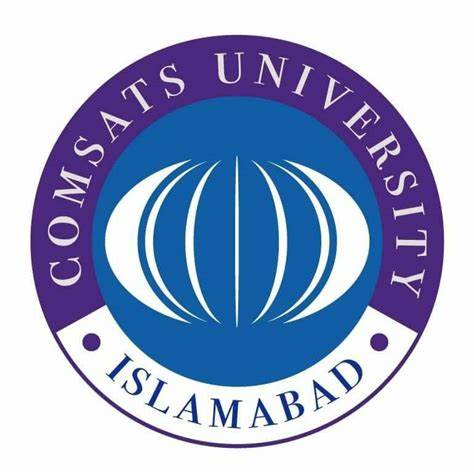
****

**OBJECT ORIENTED PROGRAMMING**

**“FINAL PROJECT”**

**GROUP:**

**Members:**

Syed Ali Shoaib Hasan (SP22-BCS-097)

Amna Jamal (SP22-BCS-010)

**Class**: BCS-3A

**INTRODUCTION:**

The project is about “NADRA Online Booking System” and is dedicated towards offering a platform that connects users and NADRA employees. This system is designed to facilitate the management of bookings and information related to individuals using various classes. The classes included in this system are Address, Person, Client, Employee, Passport, and ID Card. With these classes, the NADRA Booking System enables efficient handling of address information, managing client and employee details, tracking passport and ID card information, and facilitating the booking process.

**Classes included:**

* Address
* Person
* Client
* Employee
* Passport
* ID Card
* ClientEmployee
* ClientGuiStep1
* Menu
* midWindow
* employeeDetails
* changePassword

**Features used:**

* GUI
* Composition
* Array List
* Inheritance
* Interface (ActionListener)
* Classes
* File Handling (Serialization)

**Explanation of classes:**

* **Address:**

The Address class is used to store and manage physical address information in a system. It organizes details like house number, street name, area, city, and province in a structured way. This makes it easy to access and change address information when needed. The class has functions to set the address attributes, retrieve specific parts of the address, and convert the address into a readable format. By implementing the Serializable interface, the class allows address objects to be saved, sent, and stored in the system.

* **Person:**

The Person class is used to represent individuals in a system. It stores information such as the person's name, age, and address. The class implements Serializable to allow objects to be saved, transmitted, and stored. It provides constructors to initialize the attributes and getter/setter methods to access and modify the values. The toString() method formats the person's details into a readable string representation. Overall, the Person class serves the purpose of accurately capturing and managing personal information, making it a fundamental component in applications that deal with individuals' data.

* **Client:**

The Client class extends the Person class and is used for managing client information in a system. It includes attributes such as emailID, password, and purpose. The class also has instances of Passport and IDCard classes to store additional identification details. The class provides methods to set and retrieve the passport, ID card, purpose, email, and password. It also has functions for setting up a client account, logging in, and searching for client information. The class handles file operations to store and retrieve client data. Overall, the Client class serves the purpose of managing client accounts, authentication, and data storage for a booking system.

* **Employee:**

The purpose of the given class "Employee" is to represent an employee in a system. It extends the "Person" class and implements the "Serializable" interface, allowing objects of this class to be serialized and deserialized.

The class defines various attributes of an employee such as position, ID, working hours, CNIC, email ID, password, and branch. It also maintains a list of clients associated with the employee. The class provides methods to add clients to the client list, retrieve the client list, and perform various operations such as inserting/updating employee data, checking work ID validity, searching for employee records, setting passwords, booking employees, updating client lists, and login authentication.

Additionally, the class interacts with a file named "EmployeeList.txt" to store and retrieve employee data using serialization.

* **Passport:**

The purpose of the "Passport" class is to represent a passport object. It extends the "Person" class and implements the "Serializable" interface. The class stores information such as the validity of the passport, passport number, and the type of request (urgent or normal). It also provides methods for generating a passport number, setting the request type, and checking the existence of passport details. Additionally, it allows for the creation of passport records, retrieval of passport details based on the passport number, and displaying the passport information.

* **ID Card:**

The purpose of the "IDCard" class is to represent an ID card object. It extends the "Person" class and implements the "Serializable" interface. The class stores information such as the validity of the ID card, CNIC number, and the type of request (urgent or normal). It also provides methods for generating a CNIC number, setting the request type, and checking the existence of ID card details. Additionally, it allows for the creation of ID card records, retrieval of ID card details based on the CNIC number, and displaying the ID card information.

* **ClientEmployee:**

The purpose of the "ClientEmployee" class is to create a graphical user interface (GUI) for the NADRA Online Reservation System. It extends the "JFrame" class and implements the "ActionListener" interface. It provides various UI components such as labels, buttons, text fields, and password fields for employee and client interactions. It also handles user actions, such as logging in, signing up, and performing different tasks based on the selected options. User can select whether he or she is a user or a NADRA employee and then accordingly sign up or log in to their account in the actionPerformed function.

* **clientGUIStep1:**

The purpose of the class clientGUIstep1 is to create a graphical user interface (GUI) for the NADRA Online Reservation System. It extends the JFrame class and implements the ActionListener interface. The class defines various GUI components such as labels, text fields, buttons, radio buttons, and combo boxes to gather information from the user. It also handles user actions, such as button clicks, through the actionPerformed method. The class is responsible for processing the user input and performing actions based on the selected options, such as ID card renewal, ID card application, passport renewal, passport application, or logging out.

* **employeeDetails:**

The "employeeDetails" class is a GUI window that displays detailed information about an employee in the NADRA Online Reservation System. It extends the "JFrame" class and implements the "ActionListener" interface. The constructor takes an "Employee" object as a parameter and populates the window with labels and buttons to display and interact with the employee details. The class also handles button click events, allowing the user to go back to the main menu or log out.

* **changePassword:**

The code defines a GUI window for changing an employee's password. It includes labels and text fields for entering the old and new passwords, as well as a submit button. When the submit button is clicked, the new password is retrieved, the employee's password is updated, the current window is hidden, and a new menu window is displayed.

* **midWindow:**

The code defines a GUI window that serves as a menu for an employee. It includes buttons for accessing employee details, client details, and changing the password. When a button is clicked, a corresponding action is triggered: showing employee details, opening the password change window, or displaying a list of client details. The window is created with a specific employee object and provides various options for interaction based on the user's selection.

* **Menu:**

The code creates a GUI menu window with "Client" and "NADRA Employee" buttons. When a button is clicked, it hides the current window and opens a new window of the "ClientEmployee" class, passing an argument of 1 for the "Client" button or 2 for the "NADRA Employee" button.