

Person-Student Management System



Submitted by:

Kazi Rifat Morshed (Student ID: 230220)

Md Rimon Islam (Student ID: 230236)

1st Year 2nd Term

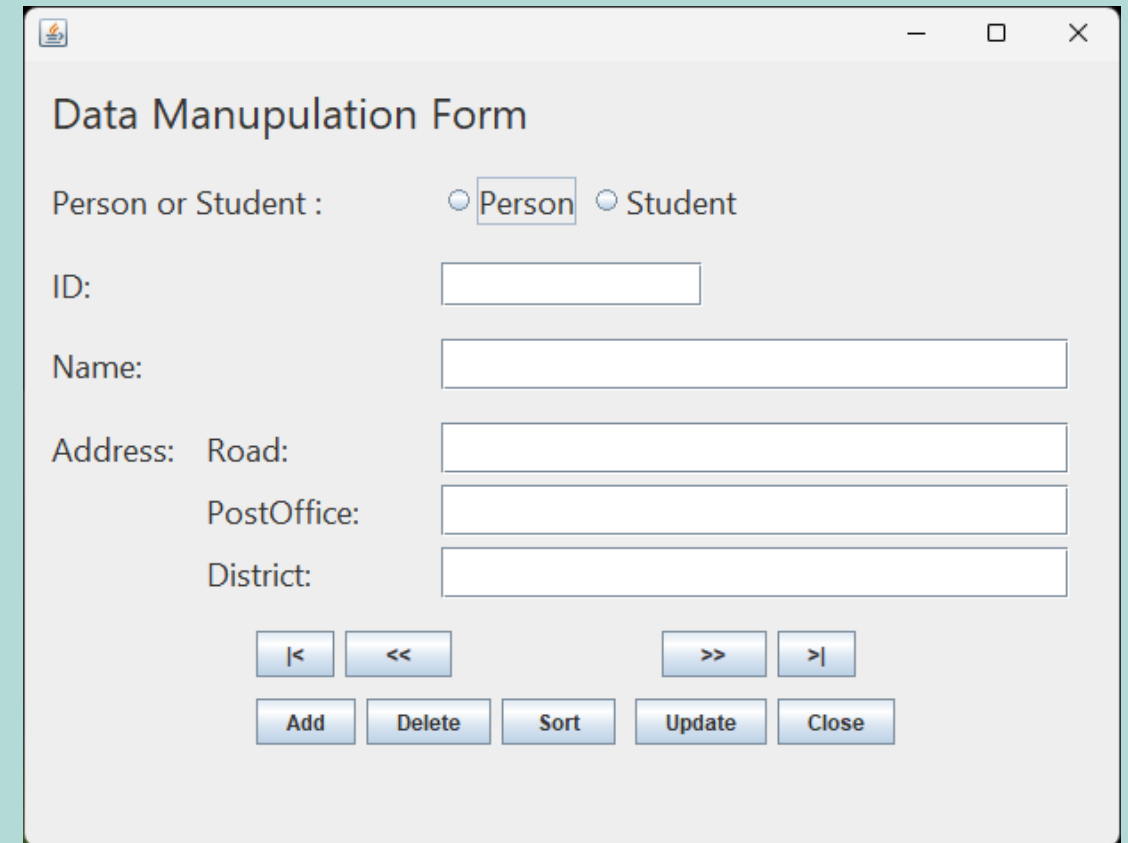
Computer Science and Engineering Discipline

Khulna University, Khulna

Project: Person-Student Management System in Java

Features:

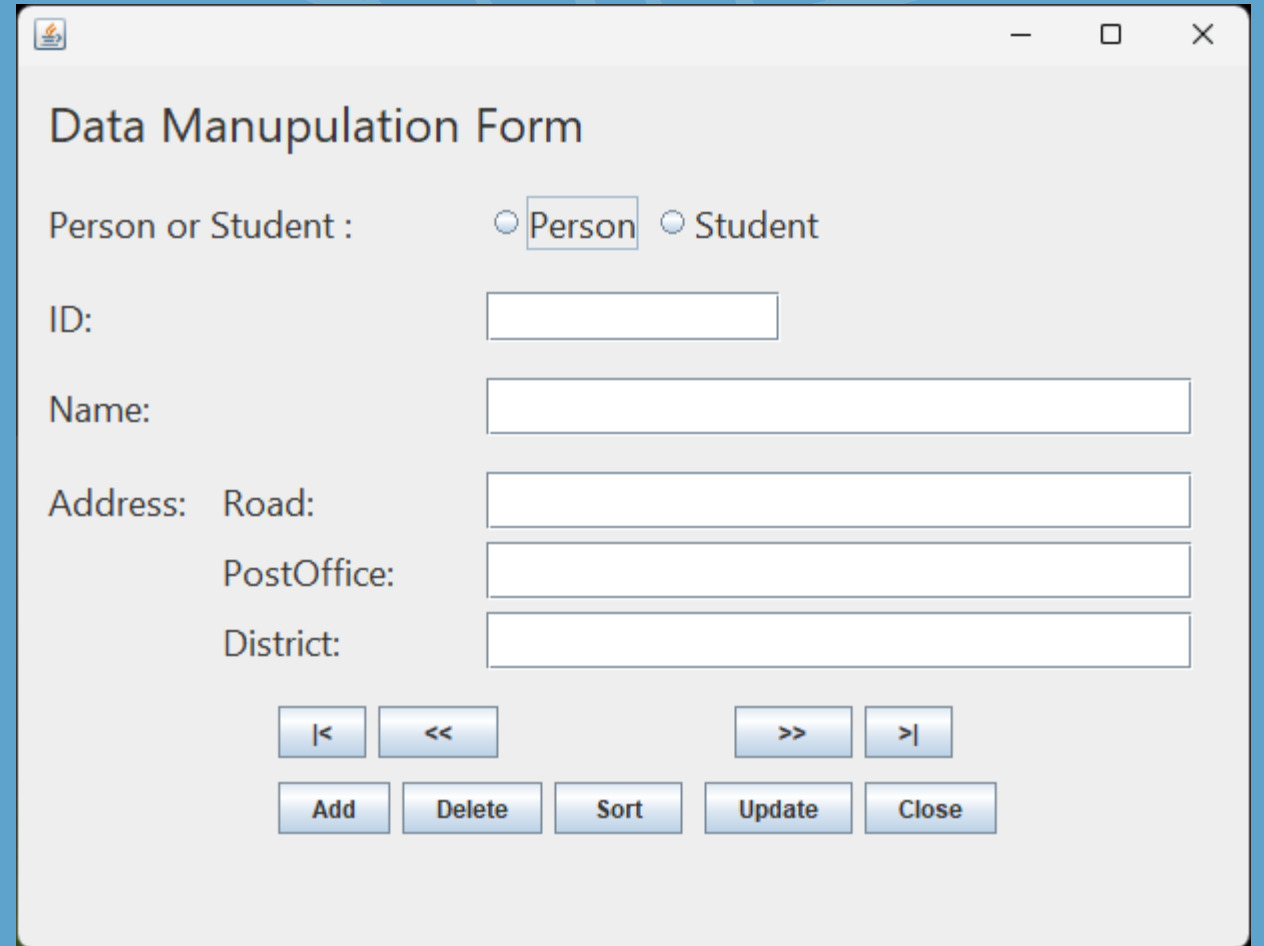
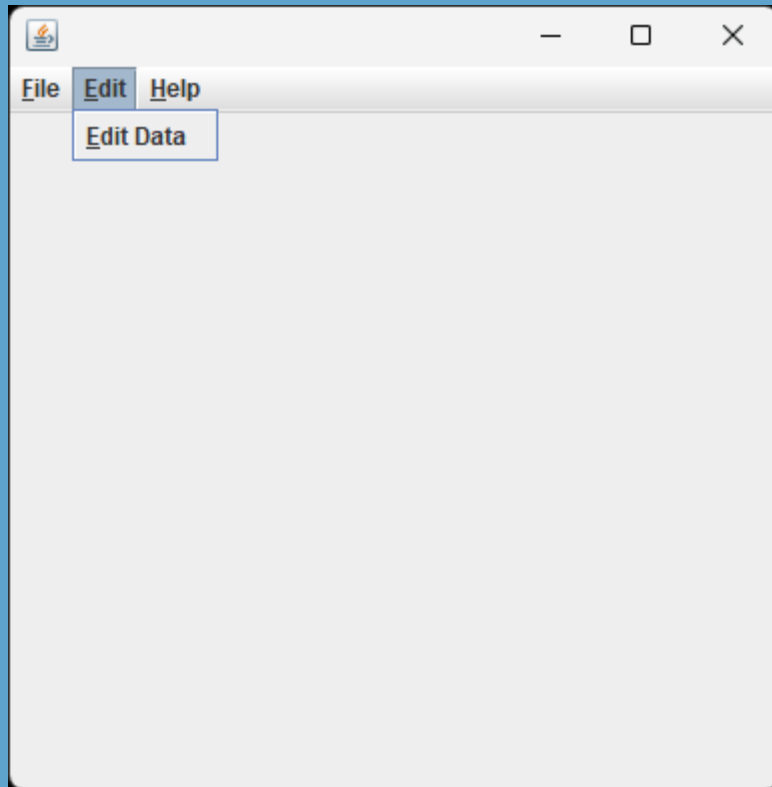
- Simple and User-friendly UI
- Person or Student selection radio button
- Add, Delete or Update Person or Student Information
- Sort alphabetically in ascending order
- Save to local memory
- Automatic load from local memory



The screenshot shows a Java Swing window titled "Data Manipulation Form". The window contains the following elements:

- Title Bar:** Standard Windows-style title bar with minimize, maximize, and close buttons.
- Form Title:** "Data Manipulation Form" centered at the top.
- Person or Student Selection:** A label "Person or Student :" followed by two radio buttons: "Person" (selected) and "Student".
- Input Fields:** A series of text input fields for "ID:", "Name:", "Address: Road:", "PostOffice:", and "District:". The "Address: Road:" label is positioned to the left of the first address field.
- Navigation Buttons:** A row of four buttons: "<|", "<<", ">>", and ">|".
- Action Buttons:** A row of five buttons: "Add", "Delete", "Sort", "Update", and "Close".

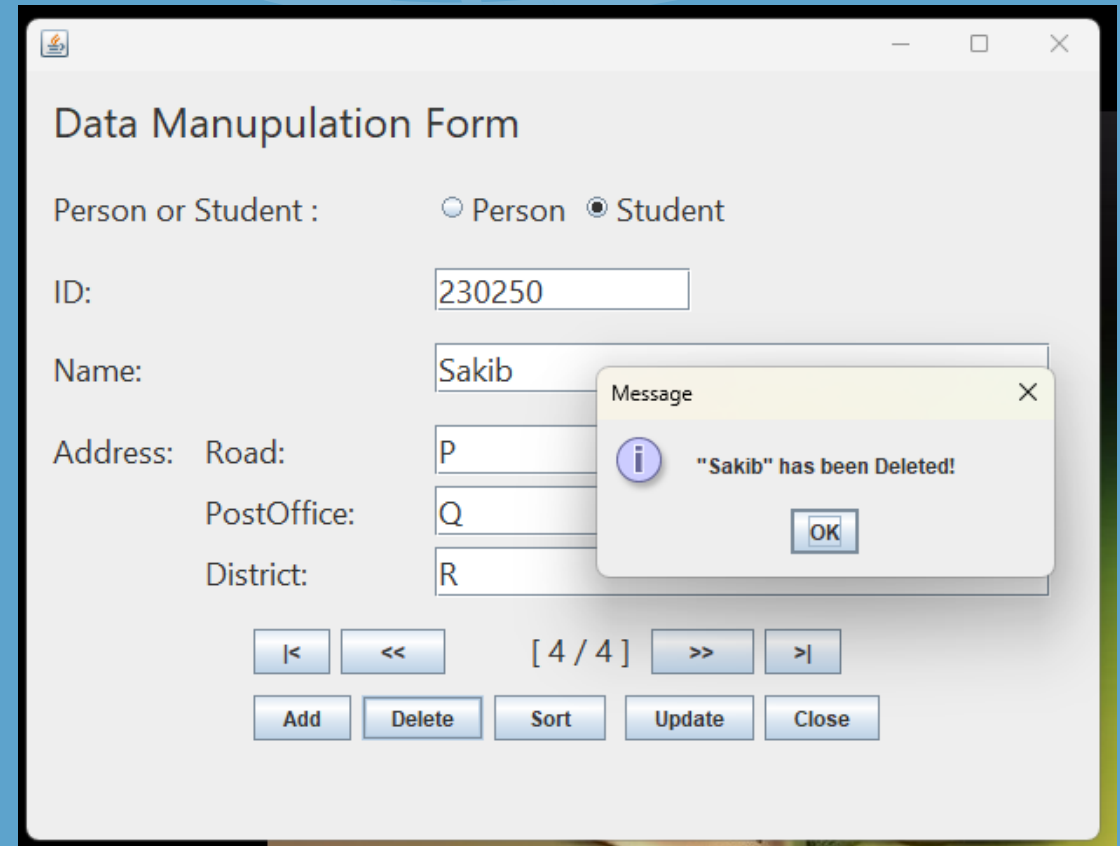
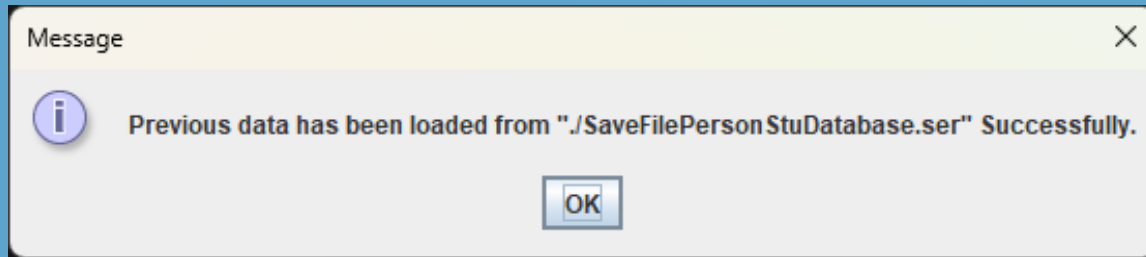
Project Demonstration



A screenshot of a 'Data Manipulation Form' window. The form contains the following elements:

- Title:** Data Manipulation Form
- Person or Student :** Radio buttons for 'Person' (selected) and 'Student'.
- ID:** A text input field.
- Name:** A text input field.
- Address: Road:** A text input field.
- PostOffice:** A text input field.
- District:** A text input field.
- Navigation Buttons:** '<', '<<', '>>', and '>|' buttons.
- Action Buttons:** 'Add', 'Delete', 'Sort', 'Update', and 'Close' buttons.

Project Demonstration



The "Data Manipulation Form" window displays a form for managing data. It includes a title bar with standard window controls. The form contains the following fields and controls:

- Person or Student :** Radio buttons for Person and Student.
- ID:** Text input field containing "230250".
- Name:** Text input field containing "Sakib".
- Address:** A group of text input fields for "Road:" (containing "P"), "PostOffice:" (containing "Q"), and "District:" (containing "R").
- Navigation:** A set of buttons including left and right arrow buttons, a page indicator "[4 / 4]", and double arrow buttons.
- Actions:** A row of buttons labeled "Add", "Delete", "Sort", "Update", and "Close".

A smaller "Message" dialog box is overlaid on the form, displaying an information icon (i) and the text: "'Sakib' has been Deleted!". It has an "OK" button.

Project Demonstration

Data Manipulation Form

Person or Student : Person Student

ID: _____

Name:

Address: Road:
PostOffice:
District:

Message

i "Rakib" has been Added!

OK

< << [1 / 2] >> >|

Add Delete Sort Update Close

Data Manipulation Form

Person or Student : Person Student

ID:

Name:

Address: Road:
PostOffice:
District:

Message


i file has been saved to: ./SaveFilePersonStuDatabase.ser

OK

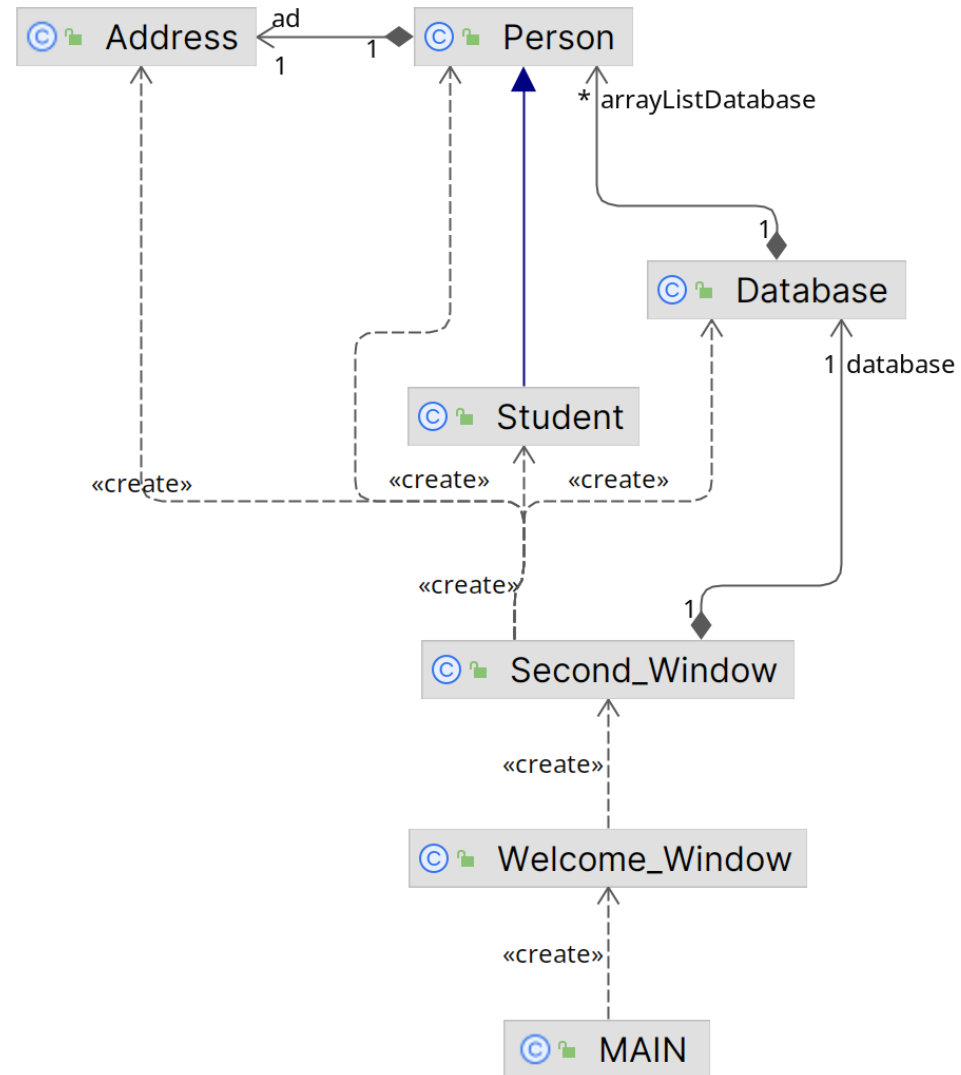
< << [10 / 10] >> >|

Add Delete Sort Update Close

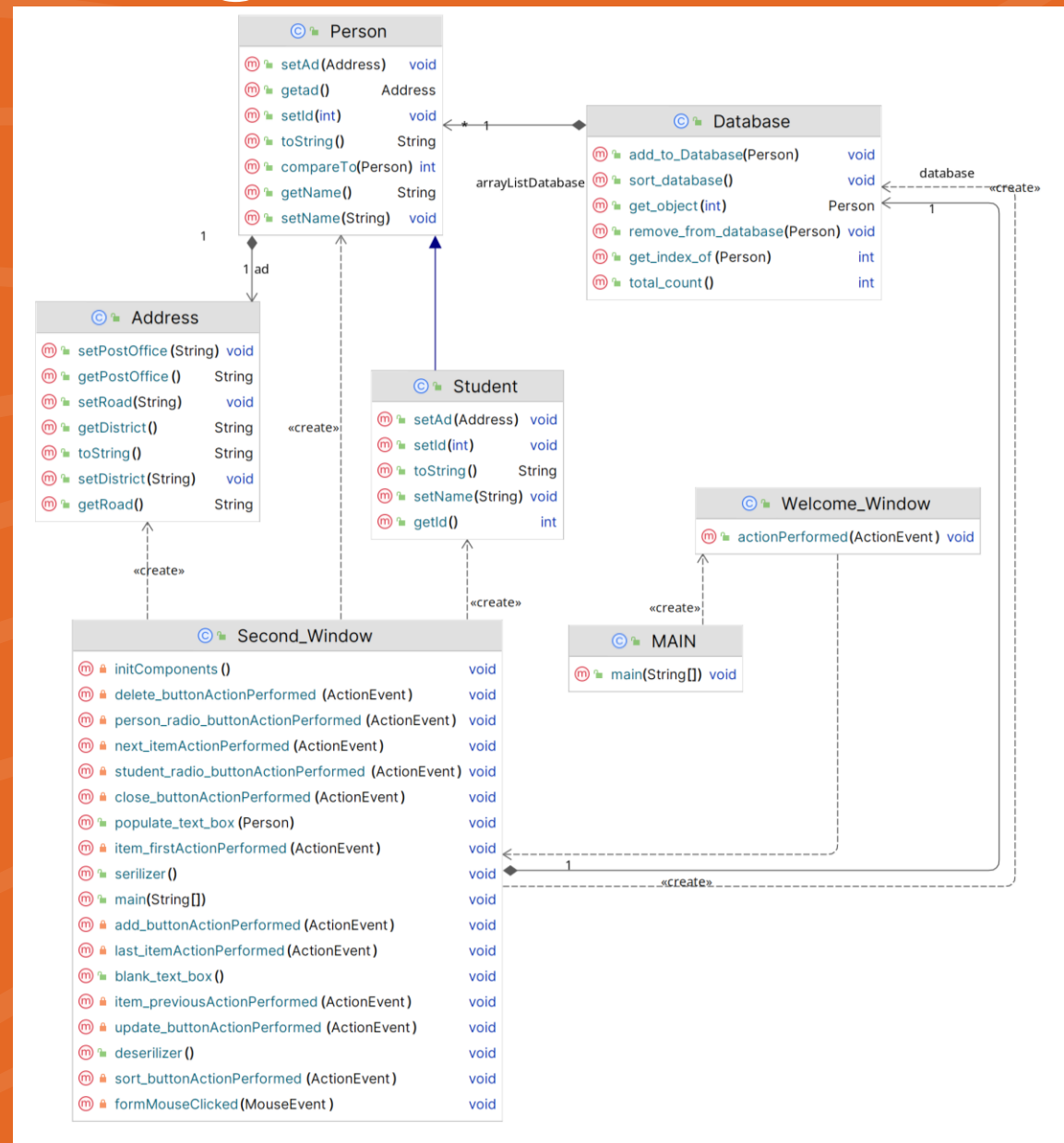
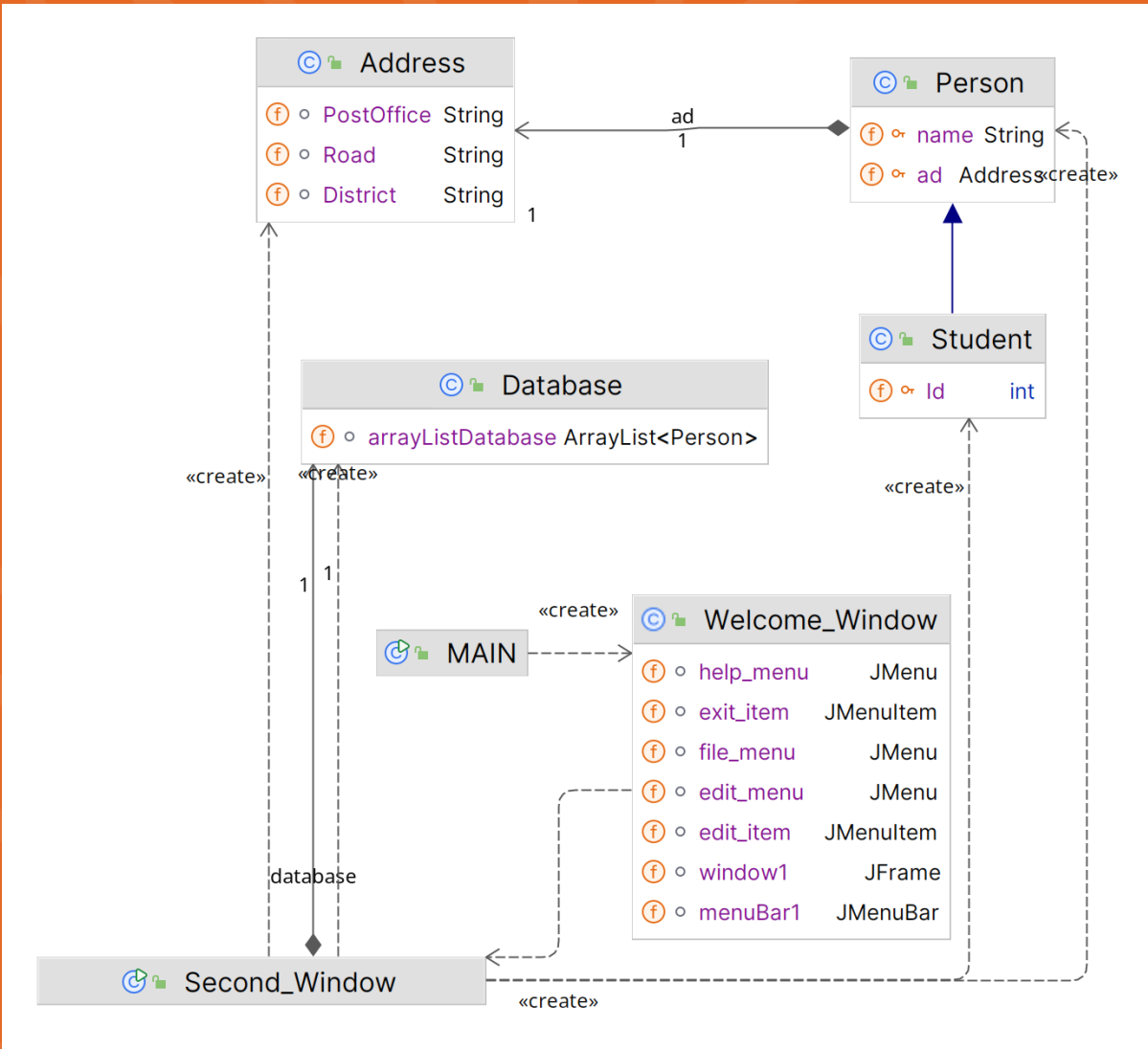
Step by Step Procedure

- Rough Drawing of UI Interface to determine necessary Swing components
 - Identifying necessary classes and Drawing UML Class Diagram according to requirement
 - Writing Basic Classes (Address, Person, Student, Database, ...) as base of back-end side
 - Drag and Drop GUI Components to design front-end User Interface (UI)
 - Defining instructions(code) for GUI components in respective method blocks
 - Implement ActionListener Interface
 - Implement Serialization and Deserialization
 - Continuous Testing and Bug Fixing
 - Export to executable JAR file
- 
- A decorative graphic in the bottom-left corner of the slide, consisting of a stylized leaf or fan shape with multiple curved lines radiating from the center, rendered in a light orange color.

UML Class Diagram (Class name and relations only)



UML Class Diagram

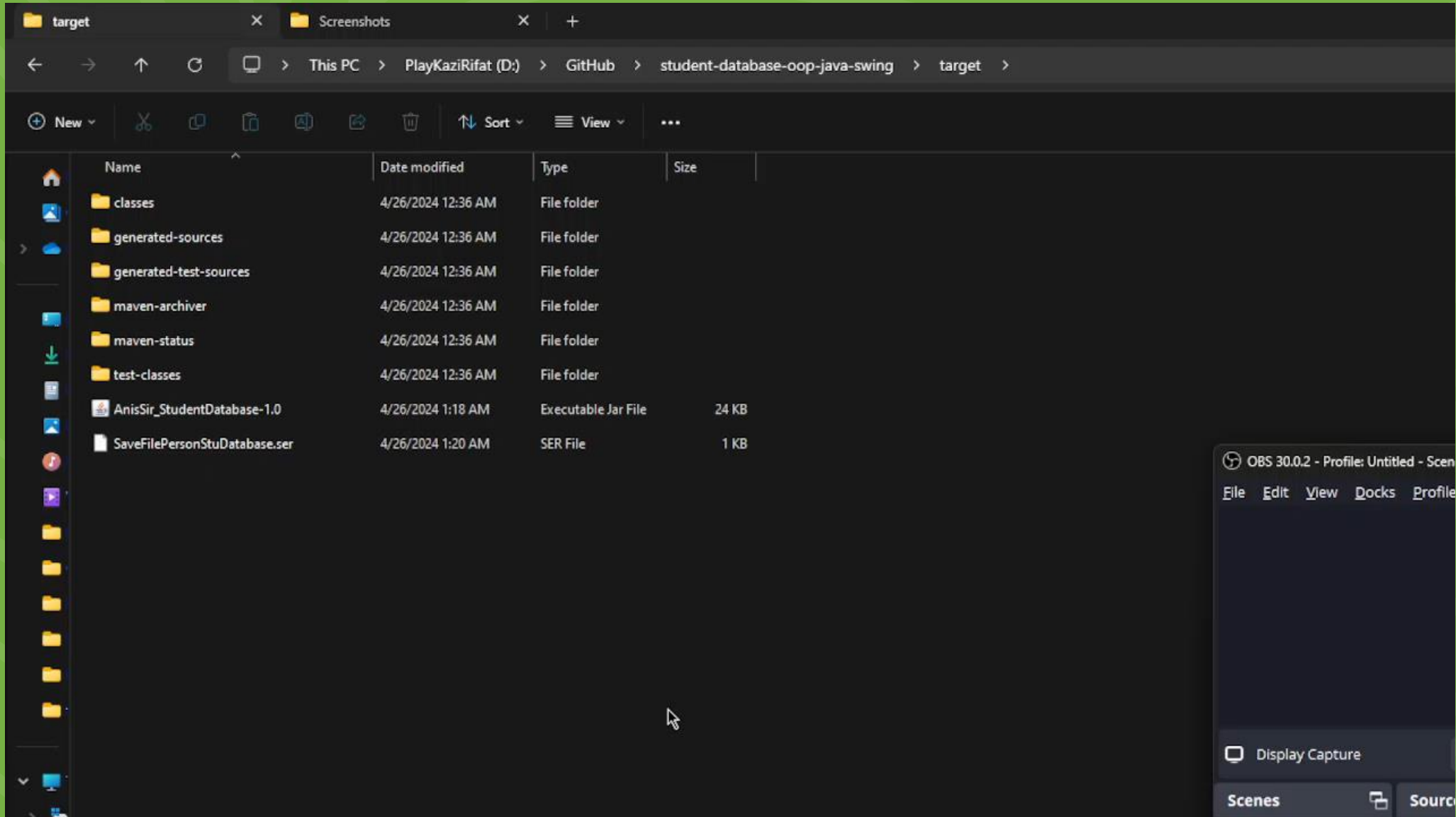




Essential Swing Components

- **JFrame:** JFrame represents a window on the screen and holds all the other GUI components.
- **JLabel:** JLabel displays a text or image on the screen.
- **JButton:** JButton is a clickable button that triggers an action when pressed.
- **JMenuBar:** JMenuBar is a horizontal bar that holds multiple JMenu objects and appears at the top of the JFrame by default.
- **JMenu:** An individual menu within a menu bar. It holds a list of menu items.
- **JRadioButton:** A button that represents selection of one option from a set of choices.
- **TextField:** JTextField is a component used in Java Swing to allow users to input or edit a single line of text.

Project Demonstration (Video)



Questions & answers



Thank you very much

