Name: CHINMAYA GARNAIK

Course: FYMCA

Division: B

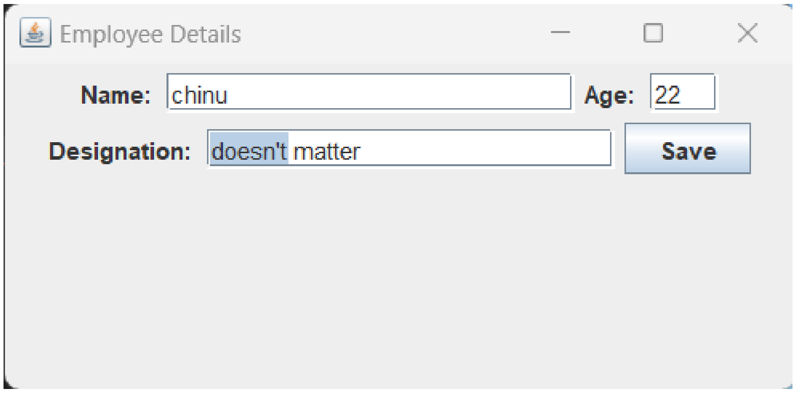
PRN: 1132220942

ADVANCE JAVA ASSIGNMENT 1

1. Create an interface for Employee Details

```
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;

public class EmployeeDetails extends JFrame {
    private JLabel lblName;
    private JTextField txtName;
    private JLabel lblAge;
    private JTextField txtAge;
    private JLabel lblDesignation;
    private JTextField txtDesignation;
    private JButton btnSave;

    public EmployeeDetails() {
        setTitle("Employee Details");
        setSize(300, 200);
        setLayout(new FlowLayout());
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

        lblName = new JLabel("Name: ");
        txtName = new JTextField(20);
        lblAge = new JLabel("Age: ");
        txtAge = new JTextField(3);
        lblDesignation = new JLabel("Designation: ");
        txtDesignation = new JTextField(20);
        btnSave = new JButton("Save");
        add(lblName);
        add(txtName);
        add(lblAge);
        add(txtAge);
        add(lblDesignation);
        add(txtDesignation);
        add(btnSave);

        setVisible(true);
    }

    public static void main(String[] args) {
        new EmployeeDetails();
    }
}
```

Output:



Employee Details

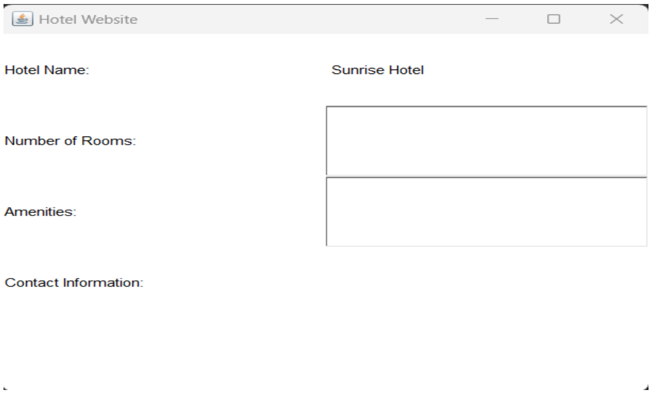
Name: chinu Age: 22

Designation: doesn't matter

2. Create an interface for Hotel website.

```
import java.awt.*;
import java.awt.event.*;

public class question2 extends Frame implements ActionListener {
    private Label hotelNameLabel, roomsLabel, amenitiesLabel,
contactLabel,
            hotelNameText;
    private TextField roomsText, amenitiesText, contactText;
    private Button submitButton, exitButton;

    public question2() {
        // Set the frame properties
        setTitle("Hotel Website");
        setSize(500, 400);
        setLayout(new GridLayout(5, 2));
        setVisible(true);
        // Add the labels and text fields for hotel name, number
of rooms,
        // amenities, and contact information
        hotelNameLabel = new Label("Hotel Name: ");
        add(hotelNameLabel);
        hotelNameText = new Label(" Sunrise Hotel ");
        add(hotelNameText);
        roomsLabel = new Label("Number of Rooms:");
        add(roomsLabel);
        roomsText = new TextField();
        add(roomsText);
        amenitiesLabel = new Label("Amenities:");
        add(amenitiesLabel);
        amenitiesText = new TextField();
        add(amenitiesText);
        contactLabel = new Label("Contact Information:");
        add(contactLabel);
        contactText = new TextField();
        add(contactText);
        // Add the submit and exit buttons
        submitButton = new Button("Submit");
        add(submitButton);
        submitButton.addActionListener(this);
        exitButton = new Button("Exit");
        add(exitButton);
        exitButton.addActionListener(this);
        // Set the default close operation
        addWindowListener(new WindowAdapter() {
            public void windowClosing(WindowEvent windowEvent) {
                System.exit(0);
            }
        });
    }

    public void actionPerformed(ActionEvent e) {
        if (e.getSource() == submitButton) {
            String hotelName = hotelNameText.getText();
            int numberOfRooms =
Integer.parseInt(roomsText.getText());
            String amenities = amenitiesText.getText();
            String contactInfo = contactText.getText();
            System.out.println("Hotel Name: " + hotelName);
            System.out.println("Number of Rooms: " +
numberOfRooms);
            System.out.println("Amenities: " + amenities);
            System.out.println("Contact Information: " +
contactInfo);
        } else if (e.getSource() == exitButton) {
            System.exit(0);
        }
    }

    public static void main(String[] args) {
        new question2();
    }
}
```

Output:



Hotel Name: Sunrise Hotel

Number of Rooms:

Amenities:

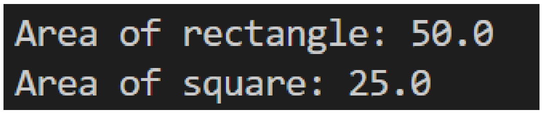
Contact Information:

3. Implement Interface and abstract class.

```
interface Shape {
    double area();
}

abstract class Polygon implements Shape {
    protected int numSides;

    Polygon(int numSides) {
        this.numSides = numSides;
    }
}

class Rectangle extends Polygon {
    private double width;
    private double height;

    Rectangle(double width, double height) {
        super(4);
        this.width = width;
        this.height = height;
    }

    @Override
    public double area() {
        return width * height;
    }
}

class Square extends Polygon {
    private double side;

    Square(double side) {
        super(4);
        this.side = side;
    }

    @Override
    public double area() {
        return side * side;
    }
}
```

```
class question3 {
    public static void main(String[] args) {
        Rectangle rectangle = new Rectangle(5, 10);
        Square square = new Square(5);
        System.out.println("Area of rectangle: " +
rectangle.area());
        System.out.println("Area of square: " + square.area());
    }
}
```

Output:

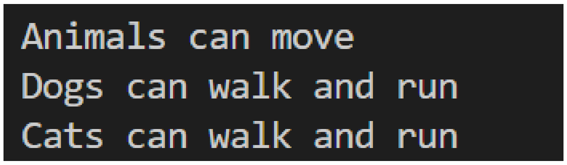


```
Area of rectangle: 50.0
Area of square: 25.0
```

4. Implement inheritance and polymorphism.

```java
class Animal {
    public void move() {
        System.out.println("Animals can move");
    }
}

class Dog extends Animal {
    public void move() {
        System.out.println("Dogs can walk and run");
    }

    public void bark() {
        System.out.println("Dogs can bark");
    }
}

class Cat extends Animal {
    public void move() {
        System.out.println("Cats can walk and run");
    }

    public void meow() {
        System.out.println("Cats can meow");
    }
}

public class question4 {
    public static void main(String[] args) {
        Animal a = new Animal();
        Animal b = new Dog();
        Animal c = new Cat();
        a.move();
        b.move();
        c.move();
    }
}
```

Output:

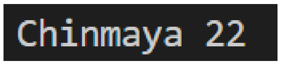


```
Animals can move
Dogs can walk and run
Cats can walk and run
```

5. Implement parameterized constructor.

```java
public class question5 {
    String studentName;
    int studentAge;

    // constructor
    question5(String name, int age) {
        studentName = name;
        studentAge = age;
    }

    void display() {
        System.out.println(studentName + " " + studentAge);
    }

    public static void main(String args[]) {
        question5 myObj = new question5("Chinmaya", 22);
        myObj.display();
    }
}
```

Output:



```
Chinmaya 22
```