INDEX

S.NO	PROGRAMS	SIGN
1.	Write a program declaring a class Rectangle with data member's length and breadth and member functions Input, Output and CalcArea.	
2.	Write a program to demonstrate use of method overloading to calculate area of square, rectangle and triangle.	
3.	Write a program to demonstrate the use of static variable, static method and static block.	
4.	Write a program to demonstrate concept of ``this``.	
5.	Write a program to demonstrate multi-level and hierarchical inheritance.	
6.	Write a program to use super() to invoke base class constructor.	
7.	Write a program to demonstrate run-time polymorphism.	
8.	Write a program to demonstrate the concept of aggregation.	
9.	Write a program to demonstrate the concept of abstract class with constructor and ``final`` method.	
10.	Write a program to demonstrate unchecked exception.	
11.	Write a swing application that uses atleast 5 swing controls.	
12.	Write a program to implement border layout using Swing	

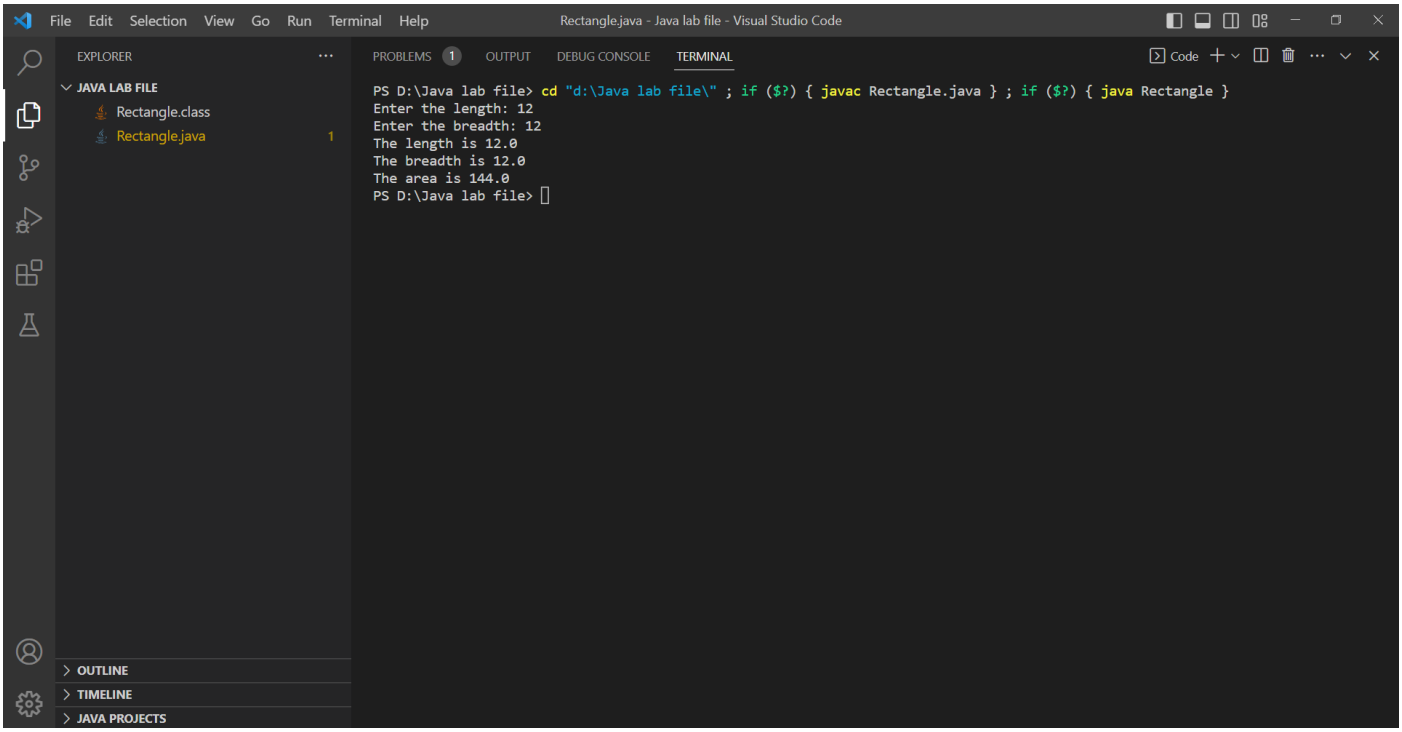
PROGRAMS

1. Write a program declaring a class Rectangle with data member's length and breadth and member functions Input, Output and Calc Area.

CODE: -

```
import java.util.Scanner;

public class Rectangle {
    private double length;
    private double breadth;

    public void input() {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter the length: ");
        length = sc.nextDouble();
        System.out.print("Enter the breadth: ");
        breadth = sc.nextDouble();
    }

    public void output() {
        System.out.println("The length is " + length);
        System.out.println("The breadth is " + breadth);
    }

    public double calcArea() {
        return length * breadth;
    }

    public static void main(String[] args) {
        Rectangle rect = new Rectangle();
        rect.input();
        rect.output();
        System.out.println("The area is " + rect.calcArea());
    }
}
```

OUTPUT: -

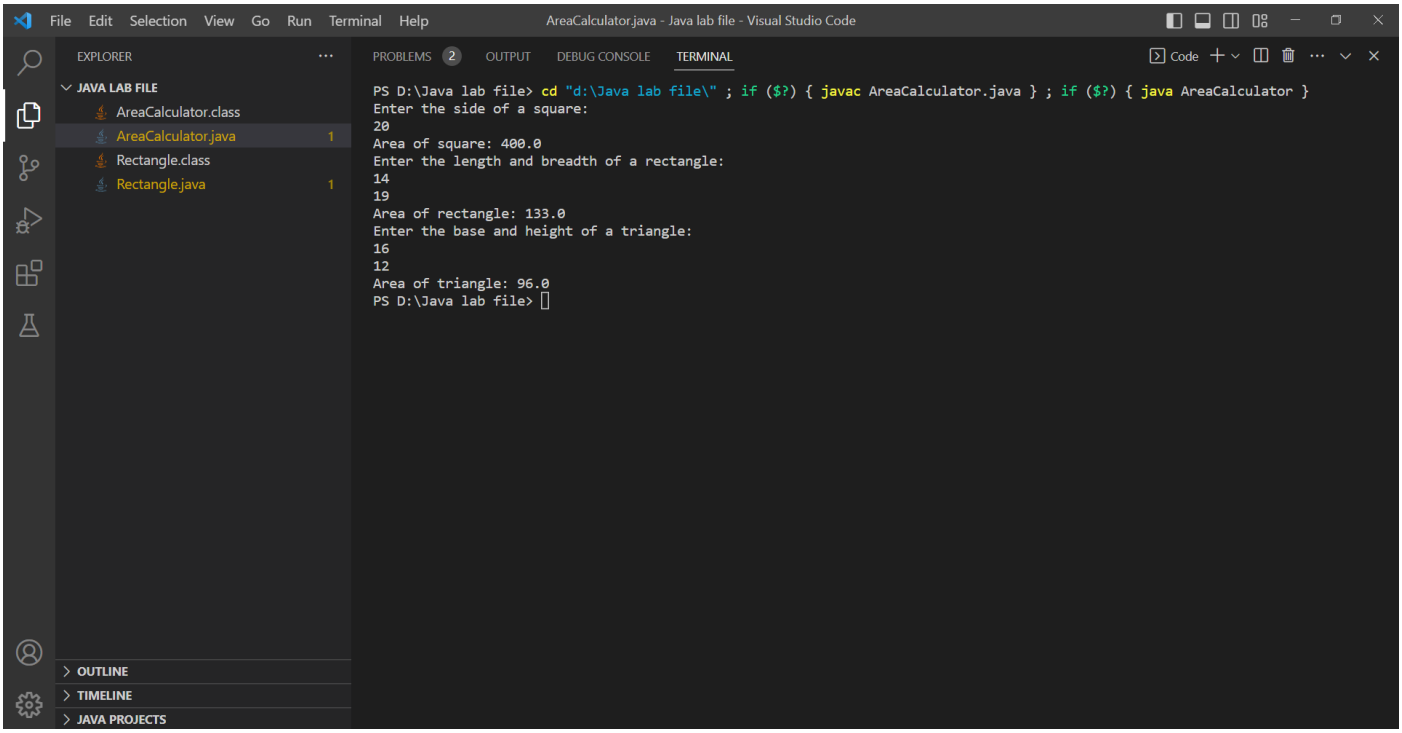

File Edit Selection View Go Run Terminal Help
Rectangle.java - Java lab file - Visual Studio Code
EXPLORER
JAVA LAB FILE
Rectangle.class
Rectangle.java
PROBLEMS 1
OUTPUT
DEBUG CONSOLE
TERMINAL
Code
PS D:\Java lab file> cd "d:\Java lab file\" ; if ($?) { javac Rectangle.java } ; if ($?) { java Rectangle }
Enter the length: 12
Enter the breadth: 12
The length is 12.0
The breadth is 12.0
The area is 144.0
PS D:\Java lab file>
OUTLINE
TIMELINE
JAVA PROJECTS

2. Write a program to demonstrate use of method overloading to calculate area of square, rectangle and triangle.

CODE: -

```
import java.util.Scanner;

public class AreaCalculator {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);

        System.out.println("Enter the side of a square: ");
        double side = sc.nextDouble();
        System.out.println("Area of square: " + area(side));

        System.out.println("Enter the length and breadth of a rectangle: ");
        double length = sc.nextDouble();
        double breadth = sc.nextDouble();
        System.out.println("Area of rectangle: " + area(length, breadth));

        System.out.println("Enter the base and height of a triangle: ");
        double base = sc.nextDouble();
        double height = sc.nextDouble();
        System.out.println("Area of triangle: " + area(base, height));
    }

    public static double area(double side) {
        return side * side;
    }

    public static double area(int length, int breadth) {
        return length * breadth;
    }

    public static double area(double base, double height) {
        return 0.5 * base * height;
    }

}
```

OUTPUT: -

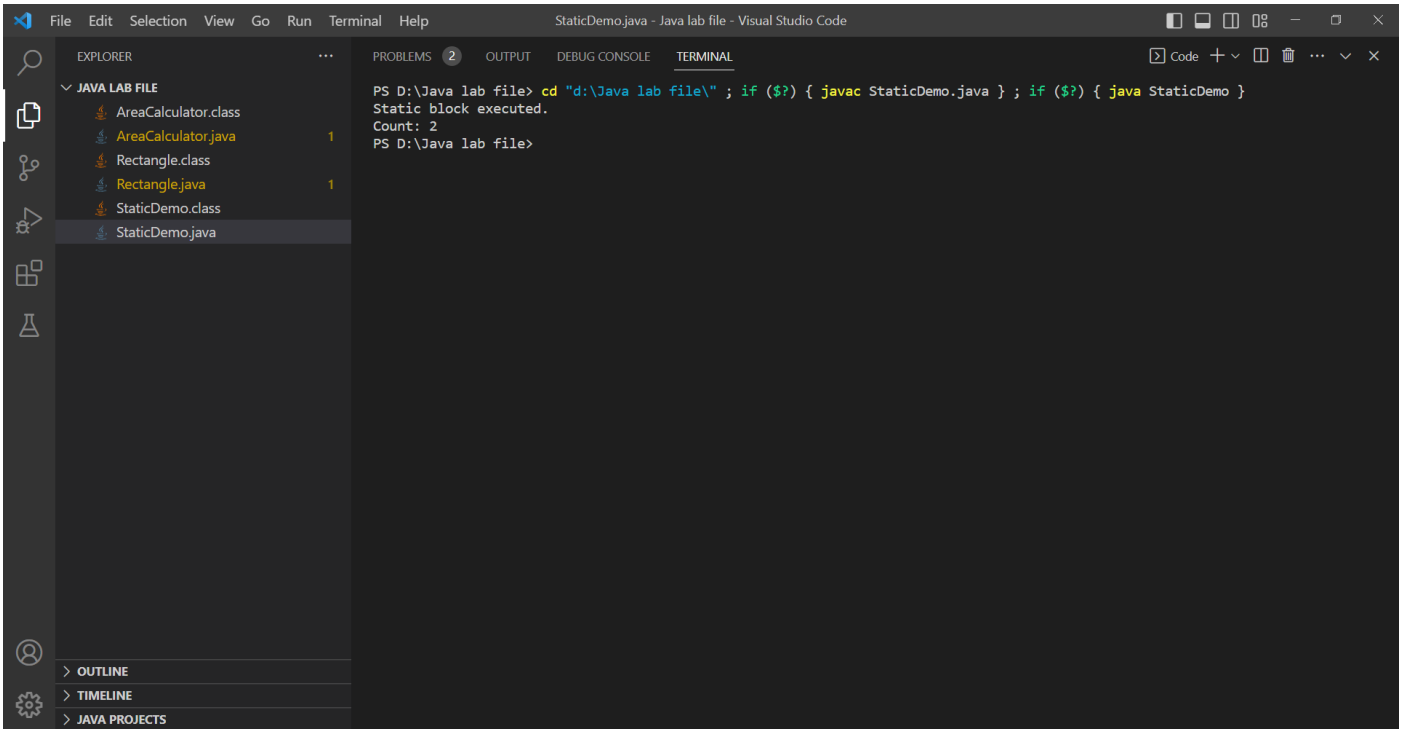

File Edit Selection View Go Run Terminal Help
AreaCalculator.java - Java lab file - Visual Studio Code
EXPLORER
JAVA LAB FILE
AreaCalculator.class
AreaCalculator.java 1
Rectangle.class
Rectangle.java 1
OUTLINE
TIMELINE
JAVA PROJECTS
PROBLEMS 2
OUTPUT
DEBUG CONSOLE
TERMINAL
Code
PS D:\Java lab file> cd "d:\Java lab file\" ; if ($?) { javac AreaCalculator.java } ; if ($?) { java AreaCalculator }
Enter the side of a square:
20
Area of square: 400.0
Enter the length and breadth of a rectangle:
14
19
Area of rectangle: 133.0
Enter the base and height of a triangle:
16
12
Area of triangle: 96.0
PS D:\Java lab file>

3. Write a program to demonstrate the use of static variable, static method and static block.

CODE: -

```
public class StaticDemo {
    static int count;

    static {
        System.out.println("Static block executed.");
        count = 0;
    }

    public static void incrementCount() {
        count++;
    }

    public static void main(String[] args) {
        StaticDemo.incrementCount();
        StaticDemo.incrementCount();
        System.out.println("Count: " + StaticDemo.count);
    }
}
```

OUTPUT: -

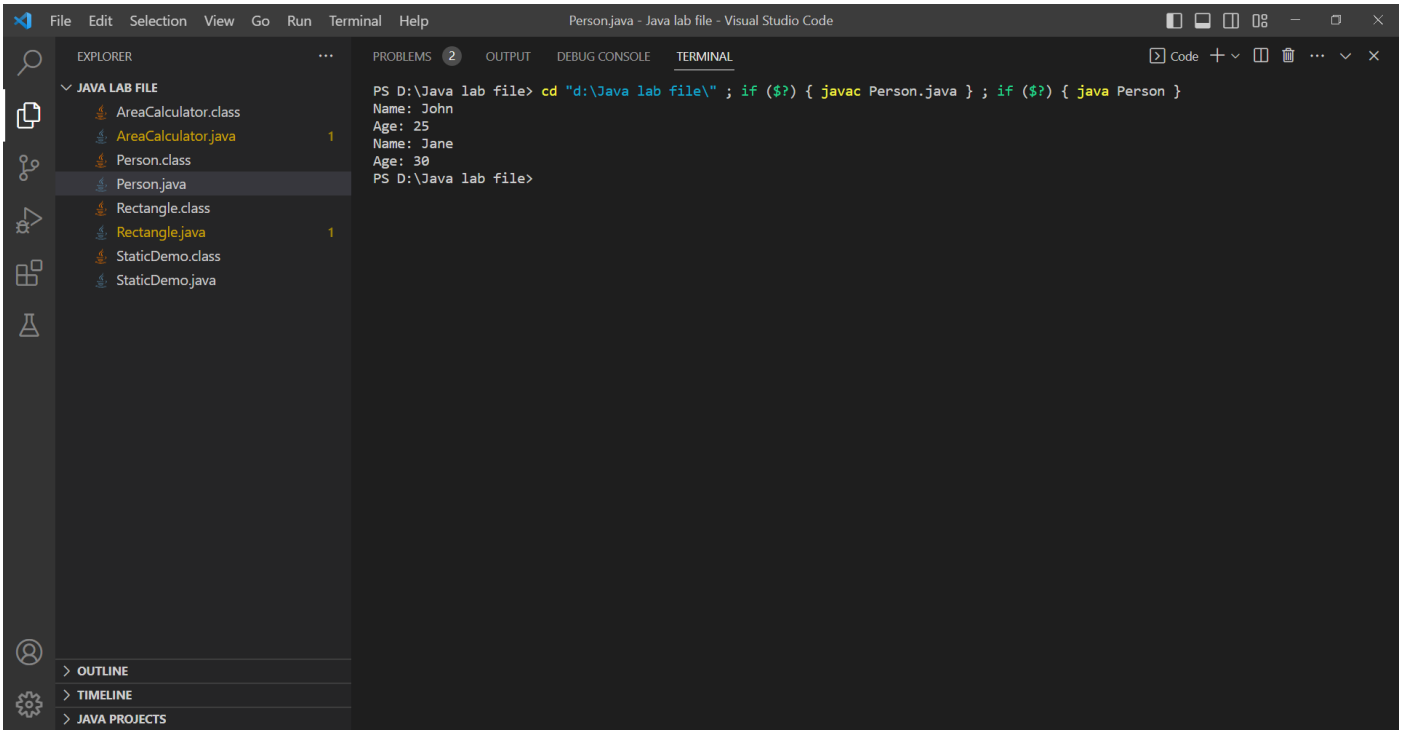

File Edit Selection View Go Run Terminal Help
StaticDemo.java - Java lab file - Visual Studio Code
EXPLORER
JAVA LAB FILE
AreaCalculator.class
AreaCalculator.java 1
Rectangle.class
Rectangle.java 1
StaticDemo.class
StaticDemo.java
OUTLINE
TIMELINE
JAVA PROJECTS
PROBLEMS 2
OUTPUT
DEBUG CONSOLE
TERMINAL
Code
PS D:\Java lab file> cd "d:\Java lab file\" ; if ($?) { javac StaticDemo.java } ; if ($?) { java StaticDemo }
Static block executed.
Count: 2
PS D:\Java lab file>

4. Write a program to demonstrate concept of ``this``.

CODE: -

```
public class Person {
    private String name;
    private int age;

    public Person(String name, int age) {
        this.name = name;
        this.age = age;
    }

    public void setName(String name) {
        this.name = name;
    }

    public void setAge(int age) {
        this.age = age;
    }

    public void printDetails() {
        System.out.println("Name: " + this.name);
        System.out.println("Age: " + this.age);
    }

    public static void main(String[] args) {
        Person person = new Person("John", 25);
        person.printDetails();

        person.setName("Jane");
        person.setAge(30);
        person.printDetails();
    }
}
```

OUTPUT: -

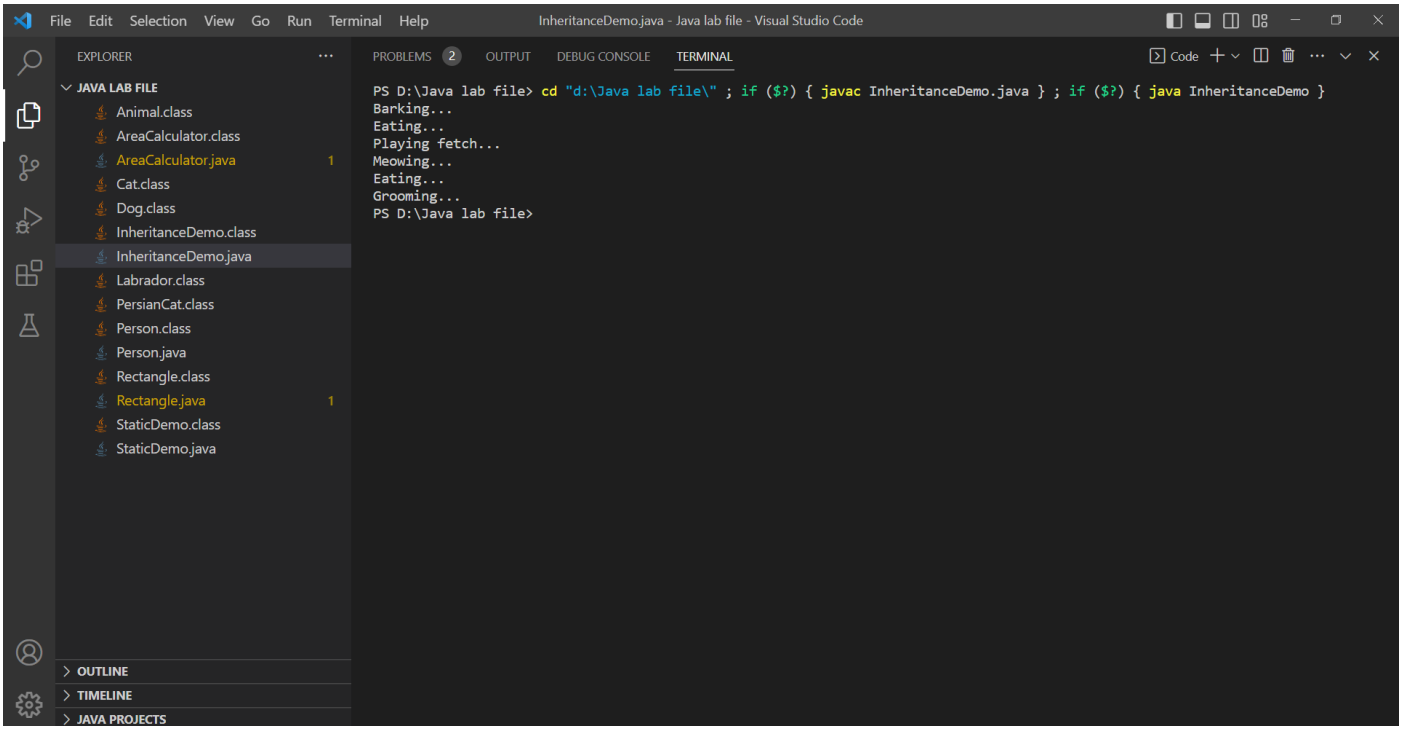

File Edit Selection View Go Run Terminal Help
Person.java - Java lab file - Visual Studio Code
EXPLORER
JAVA LAB FILE
AreaCalculator.class
AreaCalculator.java 1
Person.class
Person.java
Rectangle.class
Rectangle.java 1
StaticDemo.class
StaticDemo.java
OUTLINE
TIMELINE
JAVA PROJECTS
PROBLEMS 2
OUTPUT
DEBUG CONSOLE
TERMINAL
Code
PS D:\Java lab file> cd "d:\Java lab file\" ; if ($?) { javac Person.java } ; if ($?) { java Person }
Name: John
Age: 25
Name: Jane
Age: 30
PS D:\Java lab file>

5. Write a program to demonstrate multi-level and hierarchical inheritance.

CODE: -

```
class Animal {
   public void eat() {
      System.out.println("Eating...");
   }
}

class Dog extends Animal {
   public void bark() {
      System.out.println("Barking...");
   }
}

class Cat extends Animal {
   public void meow() {
      System.out.println("Meowing...");
   }
}

class Labrador extends Dog {
   public void playFetch() {
      System.out.println("Playing fetch...");
   }
}

class PersianCat extends Cat {
   public void groom() {
      System.out.println("Grooming...");
   }
}

public class InheritanceDemo {
   public static void main(String[] args) {
      Labrador labrador = new Labrador();
      labrador.bark();
      labrador.eat();
      labrador.playFetch();

      PersianCat persianCat = new PersianCat();
      persianCat.meow();
      persianCat.eat();
      persianCat.groom();
   }
}
```

OUTPUT: -

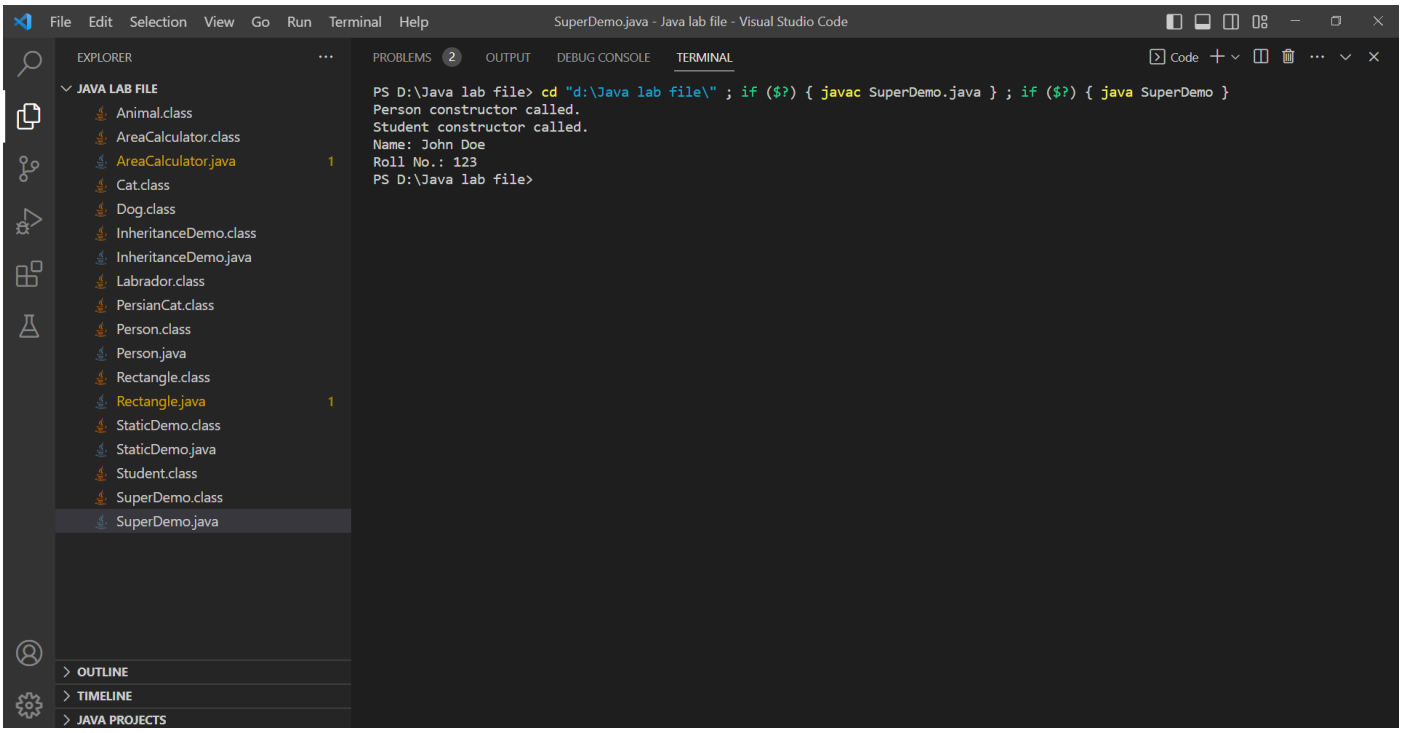

File Edit Selection View Go Run Terminal Help
InheritanceDemo.java - Java lab file - Visual Studio Code
EXPLORER
JAVA LAB FILE
Animal.class
AreaCalculator.class
AreaCalculator.java 1
Cat.class
Dog.class
InheritanceDemo.class
InheritanceDemo.java
Labrador.class
PersianCat.class
Person.class
Person.java
Rectangle.class
Rectangle.java 1
StaticDemo.class
StaticDemo.java
OUTLINE
TIMELINE
JAVA PROJECTS
PROBLEMS 2
OUTPUT
DEBUG CONSOLE
TERMINAL
Code
PS D:\Java lab file> cd "d:\Java lab file\" ; if ($?) { javac InheritanceDemo.java } ; if ($?) { java InheritanceDemo }
Barking...
Eating...
Playing fetch...
Meowing...
Eating...
Grooming...
PS D:\Java lab file>

6. Write a program to use super() to invoke base class constructor.

CODE: -

```
class Person {
   private String name;

   public Person(String name) {
      this.name = name;
      System.out.println("Person constructor called.");
   }

   public void printName() {
      System.out.println("Name: " + name);
   }
}

class Student extends Person {
   private int rollNo;

   public Student(String name, int rollNo) {
      super(name); // invoking base class constructor
      this.rollNo = rollNo;
      System.out.println("Student constructor called.");
   }

   public void printRollNo() {
      System.out.println("Roll No.: " + rollNo);
   }
}

public class SuperDemo {
   public static void main(String[] args) {
      Student student = new Student("John Doe", 123);
      student.printName();
      student.printRollNo();
   }
}
```

OUTPUT: -

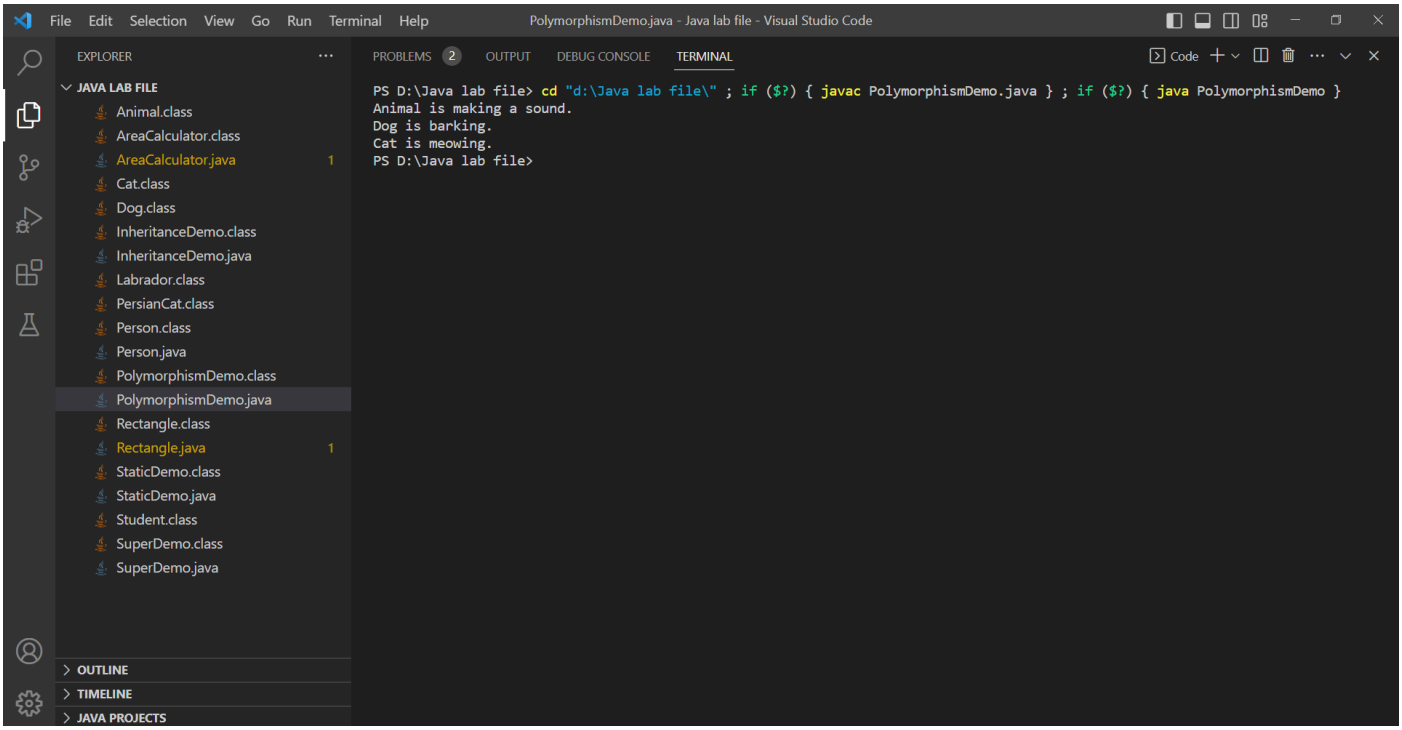

File Edit Selection View Go Run Terminal Help
SuperDemo.java - Java lab file - Visual Studio Code
EXPLORER
JAVA LAB FILE
Animal.class
AreaCalculator.class
AreaCalculator.java
Cat.class
Dog.class
InheritanceDemo.class
InheritanceDemo.java
Labrador.class
PersianCat.class
Person.class
Person.java
Rectangle.class
Rectangle.java
StaticDemo.class
StaticDemo.java
Student.class
SuperDemo.class
SuperDemo.java
OUTLINE
TIMELINE
JAVA PROJECTS
PROBLEMS 2
OUTPUT
DEBUG CONSOLE
TERMINAL
Code
PS D:\Java lab file> cd "d:\Java lab file\" ; if ($?) { javac SuperDemo.java } ; if ($?) { java SuperDemo }
Person constructor called.
Student constructor called.
Name: John Doe
Roll No.: 123
PS D:\Java lab file>

7. . Write a program to demonstrate run-time polymorphism.

CODE: -

```
class Animal {
   public void makeSound() {
      System.out.println("Animal is making a sound.");
   }
}

class Dog extends Animal {
   public void makeSound() {
      System.out.println("Dog is barking.");
   }
}

class Cat extends Animal {
   public void makeSound() {
      System.out.println("Cat is meowing.");
   }
}

public class PolymorphismDemo {
   public static void main(String[] args) {
      Animal animal = new Animal();
      animal.makeSound();

      Animal dog = new Dog();
      dog.makeSound();

      Animal cat = new Cat();
      cat.makeSound();
   }
}
```

OUTPUT: -


File Edit Selection View Go Run Terminal Help
PolymorphismDemo.java - Java lab file - Visual Studio Code
EXPLORER
JAVA LAB FILE
Animal.class
AreaCalculator.class
AreaCalculator.java
Cat.class
Dog.class
InheritanceDemo.class
InheritanceDemo.java
Labrador.class
PersianCat.class
Person.class
Person.java
PolymorphismDemo.class
PolymorphismDemo.java
Rectangle.class
Rectangle.java
StaticDemo.class
StaticDemo.java
Student.class
SuperDemo.class
SuperDemo.java
OUTLINE
TIMELINE
JAVA PROJECTS
PROBLEMS 2
OUTPUT
DEBUG CONSOLE
TERMINAL
Code
PS D:\Java lab file> cd "d:\Java lab file\" ; if ($?) { javac PolymorphismDemo.java } ; if ($?) { java PolymorphismDemo }
Animal is making a sound.
Dog is barking.
Cat is meowing.
PS D:\Java lab file>

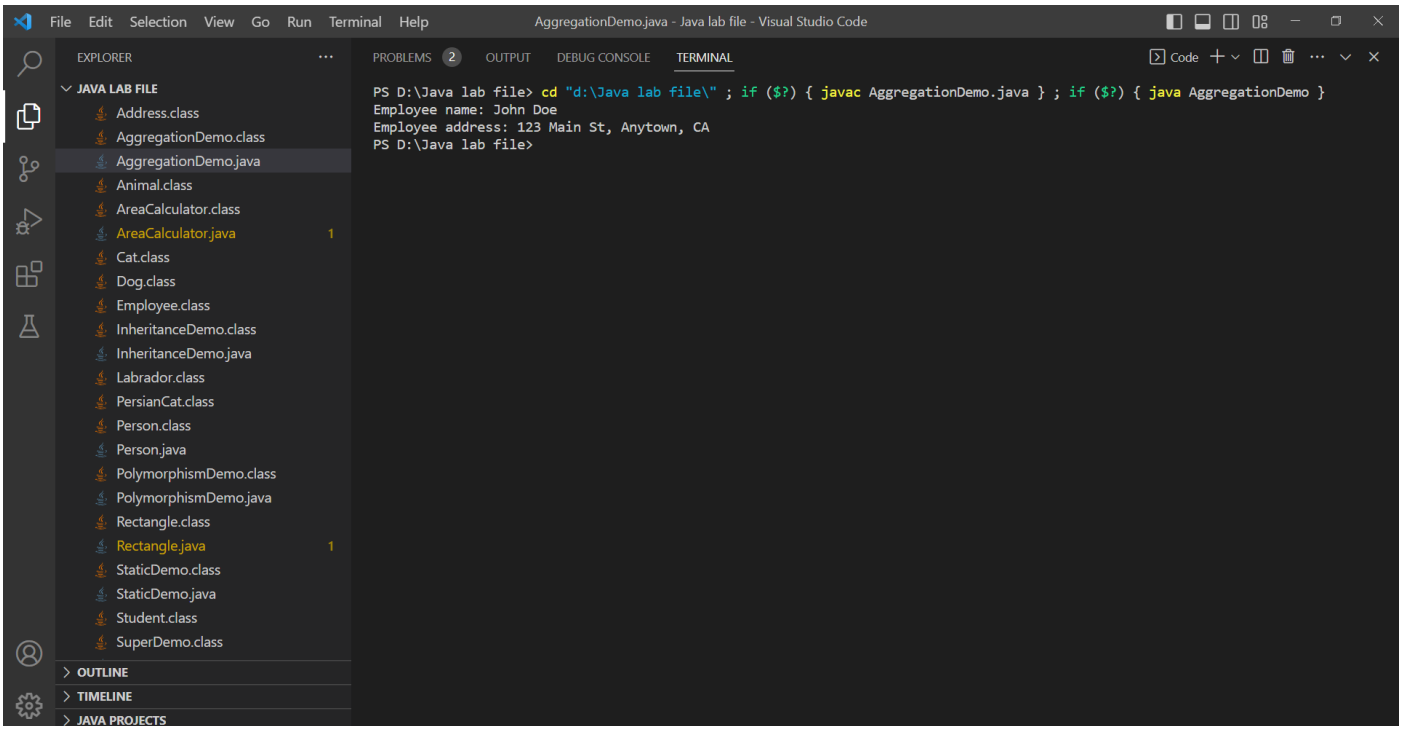
8. Write a program to demonstrate the concept of aggregation.

CODE: -

```
class Address {
    private String street;
    private String city;
    private String state;

    public Address(String street, String city, String state) {
        this.street = street;
        this.city = city;
        this.state = state;
    }

    public String getStreet() {
        return street;
    }

    public String getCity() {
        return city;
    }

    public String getState() {
        return state;
    }
}

class Employee {
    private String name;
    private Address address;

    public Employee(String name, Address address) {
        this.name = name;
        this.address = address;
    }

    public String getName() {
        return name;
    }

    public Address getAddress() {
        return address;
    }
}

public class AggregationDemo {
    public static void main(String[] args) {
        Address address = new Address("123 Main St", "Anytown", "CA");
        Employee employee = new Employee("John Doe", address);

        System.out.println("Employee name: " + employee.getName());
        System.out.println("Employee address: " + employee.getAddress().getStreet() + ", " +
```

```
                    employee.getAddress().getCity() + ", " + employee.getAddress().getState());
        }
    }
```

OUTPUT: -

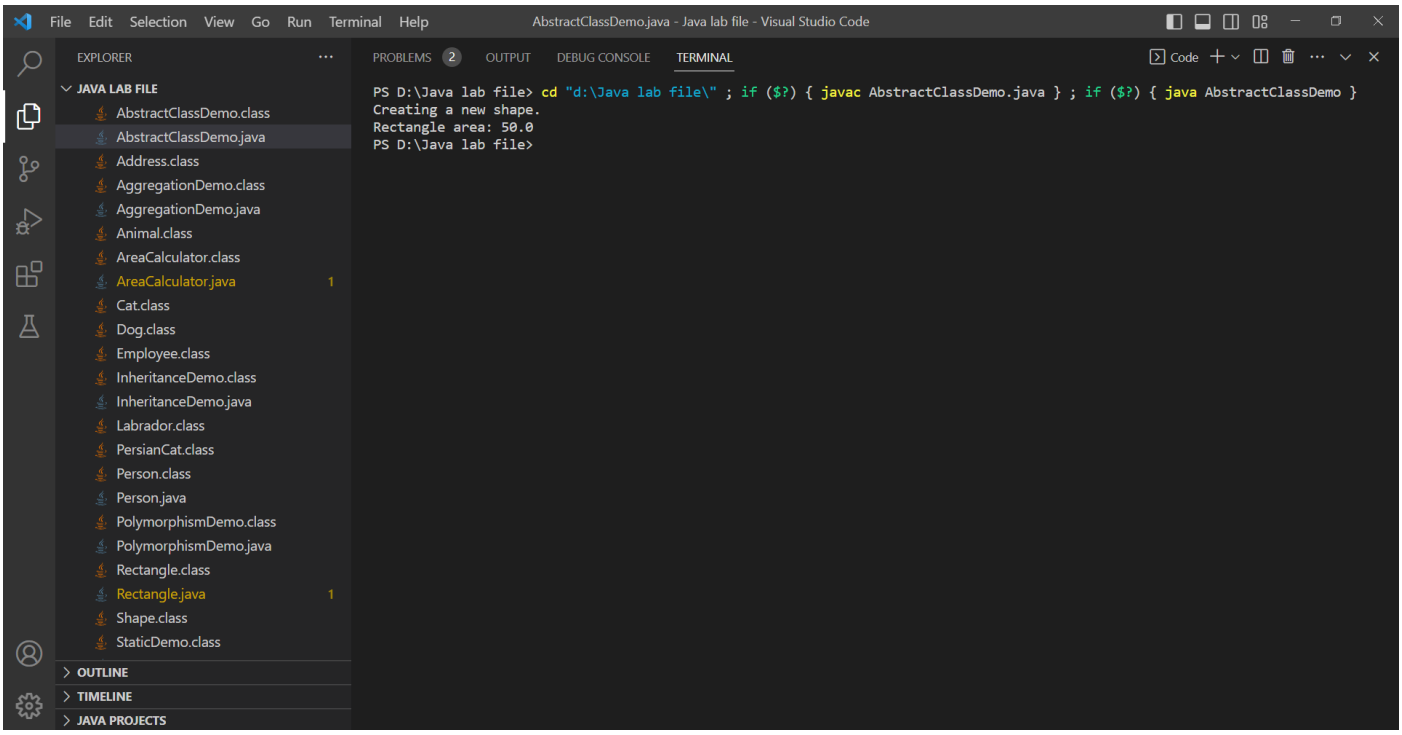

File Edit Selection View Go Run Terminal Help
AggregationDemo.java - Java lab file - Visual Studio Code
EXPLORER
JAVA LAB FILE
Address.class
AggregationDemo.class
AggregationDemo.java
Animal.class
AreaCalculator.class
AreaCalculator.java 1
Cat.class
Dog.class
Employee.class
InheritanceDemo.class
InheritanceDemo.java
Labrador.class
PersianCat.class
Person.class
Person.java
PolymorphismDemo.class
PolymorphismDemo.java
Rectangle.class
Rectangle.java 1
StaticDemo.class
StaticDemo.java
Student.class
SuperDemo.class
OUTLINE
TIMELINE
JAVA PROJECTS
PROBLEMS 2
OUTPUT
DEBUG CONSOLE
TERMINAL
Code
PS D:\Java lab file> cd "d:\Java lab file\" ; if ($?) { javac AggregationDemo.java } ; if ($?) { java AggregationDemo }
Employee name: John Doe
Employee address: 123 Main St, Anytown, CA
PS D:\Java lab file>

9. Write a program to demonstrate the concept of abstract class with constructor and ``final`` method.

CODE: -

```
abstract class Shape {
   protected double area;

   public Shape() {
      System.out.println("Creating a new shape.");
   }

   public final double getArea() {
      return area;
   }

   public abstract void calculateArea();
}

class Rectangle extends Shape {
   private double length;
   private double width;

   public Rectangle(double length, double width) {
      this.length = length;
      this.width = width;
   }

   public void calculateArea() {
      area = length * width;
   }
}

public class AbstractClassDemo {
   public static void main(String[] args) {
      Rectangle rectangle = new Rectangle(5, 10);
      rectangle.calculateArea();
      System.out.println("Rectangle area: " + rectangle.getArea());
   }
}
```

OUTPUT: -


File Edit Selection View Go Run Terminal Help
AbstractClassDemo.java - Java lab file - Visual Studio Code
EXPLORER
JAVA LAB FILE
AbstractClassDemo.class
AbstractClassDemo.java
Address.class
AggregationDemo.class
AggregationDemo.java
Animal.class
AreaCalculator.class
AreaCalculator.java
Cat.class
Dog.class
Employee.class
InheritanceDemo.class
InheritanceDemo.java
Labrador.class
PersianCat.class
Person.class
Person.java
PolymorphismDemo.class
PolymorphismDemo.java
Rectangle.class
Rectangle.java
Shape.class
StaticDemo.class
OUTLINE
TIMELINE
JAVA PROJECTS
PROBLEMS 2
OUTPUT
DEBUG CONSOLE
TERMINAL
Code
PS D:\Java lab file> cd "d:\Java lab file\" ; if ($?) { javac AbstractClassDemo.java } ; if ($?) { java AbstractClassDemo }
Creating a new shape.
Rectangle area: 50.0
PS D:\Java lab file>

10. Write a program to demonstrate checked exception during file handling.

CODE: -

```
import java.io.File;
import java.io.FileNotFoundException;
import java.util.Scanner;

public class FileHandlingDemo {
    public static void main(String[] args) {
        try {
            File file = new File("input.txt");
            Scanner scanner = new Scanner(file);
            while (scanner.hasNextLine()) {
                String line = scanner.nextLine();
                System.out.println(line);
            }
            scanner.close();
        } catch (FileNotFoundException e) {
            System.out.println("File not found!");
            e.printStackTrace();
        }
    }
}
```

OUTPUT: -

File Edit Selection View Go Run Terminal Help — input.txt - Java lab file - Visual Studio Code

EXPLORER

JAVA LAB FILE

- AbstractClassDemo.class
- AbstractClassDemo.java
- Address.class
- AggregationDemo.class
- AggregationDemo.java
- Animal.class
- AreaCalculator.class
- AreaCalculator.java 1
- Cat.class
- Dog.class
- Employee.class
- FileHandlingDemo.class
- FileHandlingDemo.java
- InheritanceDemo.class
- InheritanceDemo.java
- input.txt
- Labrador.class
- PersianCat.class
- Person.class
- Person.java
- PolymorphismDemo.class
- PolymorphismDemo.java
- Rectangle.class

OUTLINE
TIMELINE
JAVA PROJECTS

PolymorphismDemo.java | AggregationDemo.java | AbstractClassDemo.java | FileHandlingDemo.java | input.txt

input.txt

```
1  VISHAL
2
```

PROBLEMS 2 | OUTPUT | DEBUG CONSOLE | TERMINAL

```
PS D:\Java lab file> cd "d:\Java lab file\" ; if ($?) { javac FileHandlingDemo.java } ; if ($?) { java FileHandlingDemo }
VISHAL
PS D:\Java lab file>
```

File Edit Selection View Go Run Terminal Help — input.txt - Java lab file - Visual Studio Code

EXPLORER

JAVA LAB FILE

- AbstractClassDemo.class
- AbstractClassDemo.java
- Address.class
- AggregationDemo.class
- AggregationDemo.java
- Animal.class
- AreaCalculator.class
- AreaCalculator.java 1
- Cat.class
- Dog.class
- Employee.class
- FileHandlingDemo.class
- FileHandlingDemo.java
- InheritanceDemo.class
- InheritanceDemo.java
- input.txt
- Labrador.class
- PersianCat.class
- Person.class
- Person.java
- PolymorphismDemo.class
- PolymorphismDemo.java
- Rectangle.class

OUTLINE
TIMELINE
JAVA PROJECTS

PROBLEMS 2 | OUTPUT | DEBUG CONSOLE | TERMINAL

```
PS D:\Java lab file> cd "d:\Java lab file\" ; if ($?) { javac FileHandlingDemo.java } ; if ($?) { java FileHandlingDemo }
VISHAL
PS D:\Java lab file>
```

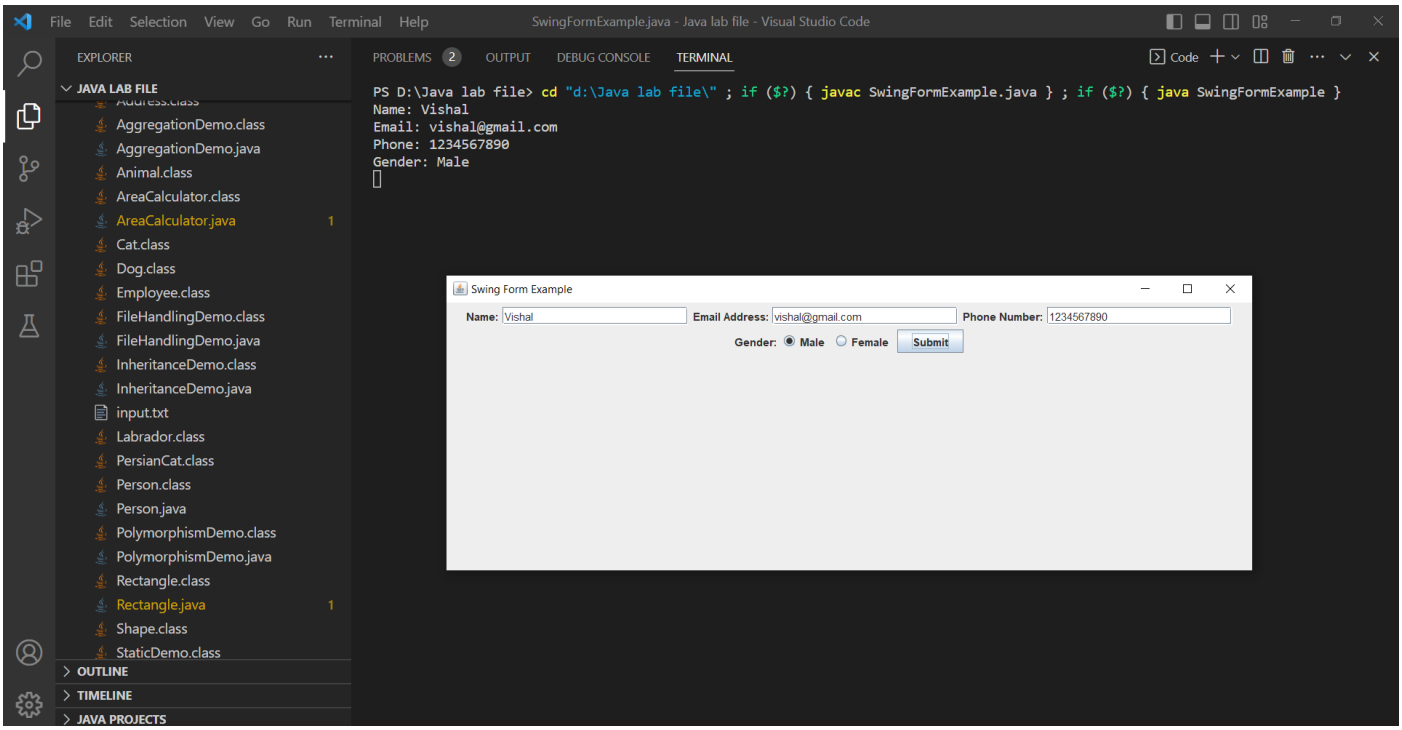
11. Write a swing application that uses at least 5 swing control.

CODE: -

```
import java.awt.FlowLayout;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;

import javax.swing.ButtonGroup;
import javax.swing.JButton;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JRadioButton;
import javax.swing.JTextField;

public class SwingFormExample {
    public static void main(String[] args) {
        JFrame frame = new JFrame("Swing Form Example");

        JLabel nameLabel = new JLabel("Name:");
        JLabel emailLabel = new JLabel("Email Address:");
        JLabel phoneLabel = new JLabel("Phone Number:");
        JLabel genderLabel = new JLabel("Gender:");

        JTextField nameField = new JTextField(20);
        JTextField emailField = new JTextField(20);
        JTextField phoneField = new JTextField(20);

        JRadioButton maleRadioButton = new JRadioButton("Male");
        JRadioButton femaleRadioButton = new JRadioButton("Female");
        ButtonGroup genderButtonGroup = new ButtonGroup();
        genderButtonGroup.add(maleRadioButton);
        genderButtonGroup.add(femaleRadioButton);

        JButton submitButton = new JButton("Submit");

        submitButton.addActionListener(new ActionListener() {
            @Override
            public void actionPerformed(ActionEvent e) {
                String name = nameField.getText();
                String email = emailField.getText();
                String phone = phoneField.getText();
                String gender = maleRadioButton.isSelected() ? "Male" : "Female";

                System.out.println("Name: " + name);
                System.out.println("Email: " + email);
                System.out.println("Phone: " + phone);
                System.out.println("Gender: " + gender);
            }
        });

        frame.setLayout(new FlowLayout());
        frame.add(nameLabel);
        frame.add(nameField);
```

```
        frame.add(emailLabel);
        frame.add(emailField);
        frame.add(phoneLabel);
        frame.add(phoneField);
        frame.add(genderLabel);
        frame.add(maleRadioButton);
        frame.add(femaleRadioButton);
        frame.add(submitButton);
        frame.setSize(300, 200);
        frame.setVisible(true);
    }
}
```

OUTPUT: -

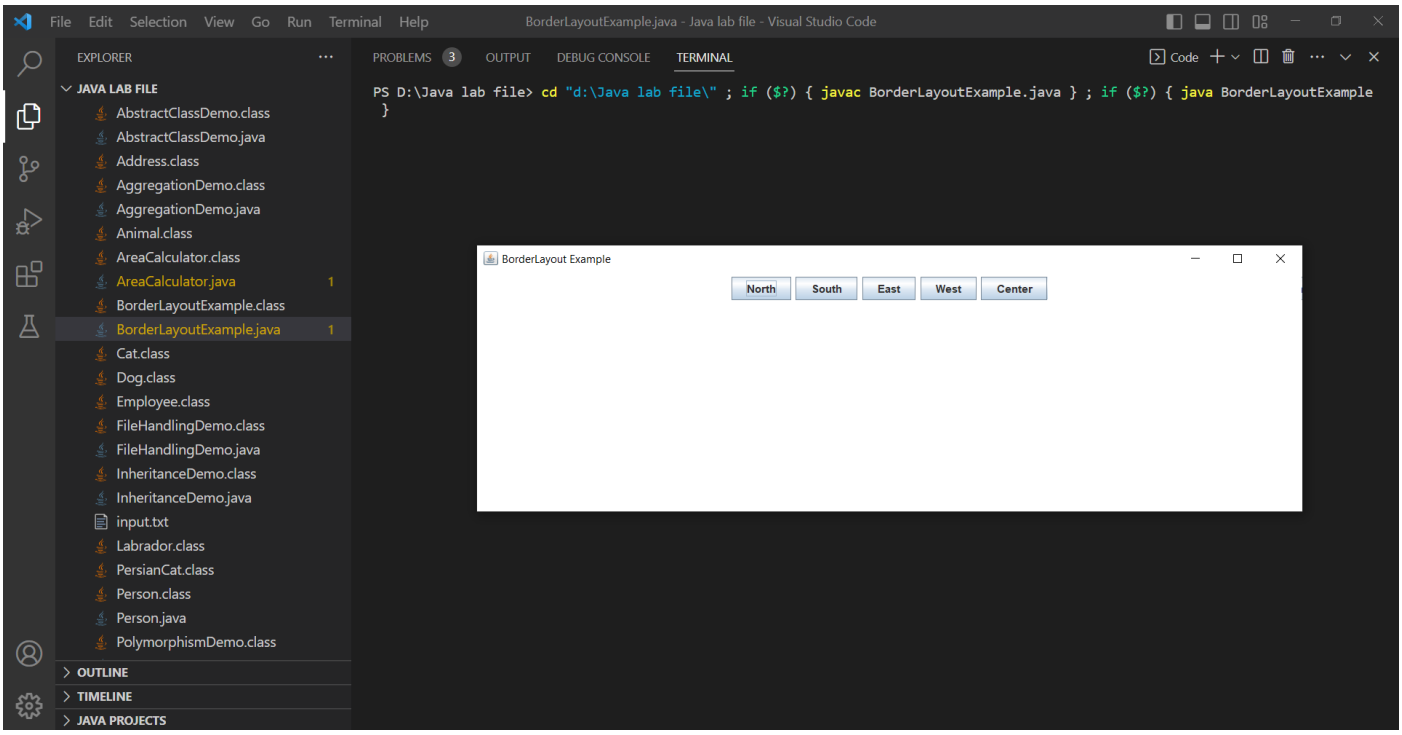

File Edit Selection View Go Run Terminal Help
SwingFormExample.java - Java lab file - Visual Studio Code
EXPLORER
JAVA LAB FILE
AggregationDemo.class
AggregationDemo.java
Animal.class
AreaCalculator.class
AreaCalculator.java
Cat.class
Dog.class
Employee.class
FileHandlingDemo.class
FileHandlingDemo.java
InheritanceDemo.class
InheritanceDemo.java
input.txt
Labrador.class
PersianCat.class
Person.class
Person.java
PolymorphismDemo.class
PolymorphismDemo.java
Rectangle.class
Rectangle.java
Shape.class
StaticDemo.class
OUTLINE
TIMELINE
JAVA PROJECTS
PROBLEMS 2
OUTPUT
DEBUG CONSOLE
TERMINAL
Code
PS D:\Java lab file> cd "d:\Java lab file\" ; if ($?) { javac SwingFormExample.java } ; if ($?) { java SwingFormExample }
Name: Vishal
Email: vishal@gmail.com
Phone: 1234567890
Gender: Male
Swing Form Example
Name: Vishal
Email Address: vishal@gmail.com
Phone Number: 1234567890
Gender: Male Female
Submit

12. Write a program to implement border layout using Swing.

CODE: -

```
import java.awt.BorderLayout;
import java.awt.Color;
import java.awt.Dimension;
import javax.swing.JButton;
import javax.swing.JFrame;
import javax.swing.JPanel;
import javax.swing.SwingUtilities;

public class BorderLayoutExample {
   public static void main(String[] args) {
      JFrame frame = new JFrame("BorderLayout Example");
      frame.setPreferredSize(new Dimension(400, 300));

      JPanel buttonPanel = new JPanel();
      buttonPanel.setBackground(Color.WHITE);

      JButton northButton = new JButton("North");
      JButton southButton = new JButton("South");
      JButton eastButton = new JButton("East");
      JButton westButton = new JButton("West");
      JButton centerButton = new JButton("Center");
      buttonPanel.add(northButton, BorderLayout.NORTH);
      buttonPanel.add(southButton, BorderLayout.SOUTH);
      buttonPanel.add(eastButton, BorderLayout.EAST);
      buttonPanel.add(westButton, BorderLayout.WEST);
      buttonPanel.add(centerButton, BorderLayout.CENTER);

      frame.getContentPane().add(buttonPanel);

      frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
      frame.pack();
      frame.setVisible(true);
   }
}
```

OUTPUT: -


File Edit Selection View Go Run Terminal Help
BorderLayoutExample.java - Java lab file - Visual Studio Code
EXPLORER
JAVA LAB FILE
AbstractClassDemo.class
AbstractClassDemo.java
Address.class
AggregationDemo.class
AggregationDemo.java
Animal.class
AreaCalculator.class
AreaCalculator.java
BorderLayoutExample.class
BorderLayoutExample.java
Cat.class
Dog.class
Employee.class
FileHandlingDemo.class
FileHandlingDemo.java
InheritanceDemo.class
InheritanceDemo.java
input.txt
Labrador.class
PersianCat.class
Person.class
Person.java
PolymorphismDemo.class
OUTLINE
TIMELINE
JAVA PROJECTS
PROBLEMS 3
OUTPUT
DEBUG CONSOLE
TERMINAL
Code
PS D:\Java lab file> cd "d:\Java lab file\" ; if ($?) { javac BorderLayoutExample.java } ; if ($?) { java BorderLayoutExample
}
BorderLayout Example
North
South
East
West
Center