

## Pro 1 Interface

```
package javaInterface;
import java.util.Scanner;

class Rectangle {
    double length, width;

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

    double calculateArea() {
        return length * width;
    }

    double calculatePerimeter() {
        return 2 * (length + width);
    }
}

class Circle {
    double radius;

    Circle(double radius) {
        this.radius = radius;
    }

    double calculateArea() {
        return Math.PI * radius * radius;
    }

    double calculatePerimeter() {
        return 2 * Math.PI * radius;
    }
}

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

        System.out.print("Enter the length of the rectangle: ");
        double length = scanner.nextDouble();
```

```
System.out.print("Enter the width of the rectangle: ");
double width = scanner.nextDouble();

Rectangle rectangle = new Rectangle(length, width);

System.out.println("Rectangle Area: " + rectangle.calculateArea());
System.out.println("Rectangle Perimeter: " +
rectangle.calculatePerimeter());

System.out.print("Enter the radius of the circle: ");
double radius = scanner.nextDouble();

Circle circle = new Circle(radius);

System.out.println("Circle Area: " + circle.calculateArea());
System.out.println("Circle Perimeter: " + circle.calculatePerimeter());

scanner.close();
}
}
```

## Por 2 Exception Handling

**package** exam;

**import** java.util.Scanner;

**import** java.io.\*;

**public class** MS {

**public static void** main(String[] args) {

        Scanner sc =**new** Scanner(System.**in**);

**try** {

            System.**out**.println("Enter a Number to be divided :");

**int** num = sc.nextInt();

            System.**out**.println("Result"+10/(num));

        }

**catch**(ArithmeticException e) {

            System.**out**.println(e.getMessage());

        }

**try** {

            System.**out**.println("Enter a String :");

            String text =sc.next();

            text=**null**;

            System.**out**.println("String lenght is"+text.length());

        }

**catch**(NullPointerException e) {

            System.**out**.println(e.getMessage());

        }

**try** {

            System.**out**.println("Enter a index value ");

**int** index=sc.nextInt();

**int** [] num = {1,2,3};

            System.**out**.println("Index value is:"+num[index]);

        }

**catch**(ArrayIndexOutOfBoundsException e) {

            System.**out**.println(e.getMessage());

        }

```

try {
    System.out.println("Enter a number:");
    int num = Integer.parseInt(sc.next());
    System.out.println("Result:"+num);
}
catch(NumberFormatException e) {
    System.out.println(e.getMessage());
}

try {
    System.out.println("Enter the file:");
    String fn = sc.next();
    File f = new File(fn);
    Scanner s = new Scanner(f);
    System.out.println("file content:"+s.nextLine());
}
catch(FileNotFoundException e) {
    System.out.println(e.getMessage());
}
catch(Exception e) {
    System.out.println(e.getMessage());
}
}
}

```

### Pro 3 Access specifiers:

**package** accMod;

```
public class AccessModifier {  
    public String publicVar = "This is a public variable";  
    protected String protectedVar = "This is a protected variable";  
    String defaultVar = "This is a default variable (package-private)";  
    private String privateVar = "This is a private variable";  
  
    public void publicMethod() {  
        System.out.println("This is a public method");  
    }  
  
    protected void protectedMethod() {  
        System.out.println("This is a protected method");  
    }  
  
    void defaultMethod() {  
        System.out.println("This is a default method (package-private)");  
    }  
  
    private void privateMethod() {  
        System.out.println("This is a private method");  
    }  
  
    public static void main(String[] args) {  
        AccessModifier example = new AccessModifier();  
  
        System.out.println("Public Variable: " + example.publicVar);  
        example.publicMethod();  
  
        System.out.println("Protected Variable: " + example.protectedVar);  
        example.protectedMethod();  
  
        System.out.println("Default Variable: " + example.defaultVar);  
        example.defaultMethod();  
  
        System.out.println("Private Variable: " + example.privateVar);  
        example.privateMethod();  
    }  
}
```

## Pro 4 Swing Application:

```
package swing_application;
```

```
import javax.swing.*;
```

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

```
public class MetricConversionApp extends JFrame {  
    private JTextField cmTextField, inchTextField;
```

```
    public MetricConversionApp() {  
        setTitle("Metric Conversion");  
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);  
        setLayout(new FlowLayout());  
        setSize(300, 150);  
        setVisible(true);
```

```
        cmTextField = new JTextField(10);  
        inchTextField = new JTextField(10);  
        JButton convertButton = new JButton("Convert");
```

```
        convertButton.addActionListener(e -> {  
            try {  
                double inches = Double.parseDouble(cmTextField.getText()) / 2.54;  
                inchTextField.setText(String.format("%.2f", inches));  
            } catch (NumberFormatException ex) {  
                JOptionPane.showMessageDialog(null, "Invalid input. Please enter a  
valid number.");  
            }  
        });
```

```
        add(new JLabel("Centimeters:"));  
        add(cmTextField);  
        add(new JLabel("Inches:"));  
        add(inchTextField);  
        add(convertButton);  
    }
```

```
    public static void main(String[] args) {  
        SwingUtilities.invokeLater(() -> new MetricConversionApp());  
    }  
}
```

## Pro 6 message communication:

```
package server;
```

```
import java.io.*;
```

```
import java.net.*;
```

```
public class MyServer {
```

```
    public static void main(String[] args) throws Exception {  
        ServerSocket serverSocket = new ServerSocket(3334);  
        Socket clientSocket = serverSocket.accept();
```

```
        DataInputStream din = new  
DataInputStream(clientSocket.getInputStream());  
        DataOutputStream dout = new  
DataOutputStream(clientSocket.getOutputStream());
```

```
        BufferedReader br = new BufferedReader(new  
InputStreamReader(System.in));
```

```
        String clientMessage = "";  
        String serverResponse = "";
```

```
        while (!clientMessage.equals("stop")) {
```

```
            // Read client's message
```

```
            clientMessage = din.readUTF();  
            System.out.println("Client says: " + clientMessage);
```

```
            serverResponse = br.readLine();  
            dout.writeUTF(serverResponse);  
            dout.flush();
```

```
        }
```

```
        din.close();  
        dout.close();  
        clientSocket.close();  
        serverSocket.close();
```

```
    }
```

```
}
```

## Client

```
package client;
```

```
import java.io.*;
```

```
import java.net.*;
```

```
public class MyClient {
```

```
    public static void main(String[] args) throws Exception {  
        Socket clientSocket = new Socket("localhost", 3334);
```

```
        DataInputStream din = new  
        DataInputStream(clientSocket.getInputStream());
```

```
        DataOutputStream dout = new  
        DataOutputStream(clientSocket.getOutputStream());
```

```
        BufferedReader br = new BufferedReader(new  
        InputStreamReader(System.in));
```

```
        String clientMessage = "";  
        String serverResponse = "";
```

```
        while (!clientMessage.equals("stop")) {  
            // Read user's input  
            clientMessage = br.readLine();  
            dout.writeUTF(clientMessage);  
            dout.flush();  
  
            // Receive and display server's response  
            serverResponse = din.readUTF();  
            System.out.println("Server says: " + serverResponse);  
        }
```

```
        din.close();  
        dout.close();  
        clientSocket.close();  
    }  
}
```

**In client cnsole:**

**Type hai and press enter**

**To stop it from client : stop**



## Pro 9 JDBC:

```
import java.sql.*;
```

```
public class MyJdbc {  
    public static void main(String[] args) {  
        String jdbcUrl = "jdbc:mysql://localhost:3306/Mca"; // Update the  
        database name if necessary  
        String username = "root";  
        String password = "Elan@27";  
  
        try {  
            Class.forName("com.mysql.cj.jdbc.Driver");  
            try (Connection con = DriverManager.getConnection(jdbcUrl,  
username, password);  
                Statement stmt = con.createStatement();  
                ResultSet rs = stmt.executeQuery("SELECT * FROM emp")) {  
                while (rs.next()) {  
                    System.out.println(rs.getInt("id") + " " + rs.getString("name") + "  
" + rs.getInt("age"));  
                }  
            }  
        } catch (Exception e) {  
            System.out.println(e);  
        }  
    }  
}
```

```
C:\Users\Dell>mysql -u root -p  
Enter password: *****(Mca@30)
```

```
mysql> CREATE DATABASE Mca;  
Query OK, 1 row affected (0.01 sec)
```

```
mysql> use Mca;  
Database changed  
mysql> CREATE TABLE emp(id INT, name VARCHAR(40), age INT);  
Query OK, 0 rows affected (0.02 sec)
```

```
mysql> INSERT INTO emp (id, name, age) VALUES (1, 'Elanchezhian M',  
21);  
Query OK, 1 row affected (0.00 sec)  
mysql> exit
```

## Pro 10 Java Bean:

### PersonBean.java

```
package javaBean;

import java.io.Serializable;

public class PersonBean implements Serializable {
    private String name;
    private int age;

    public PersonBean() {
        // Default constructor
    }

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

    public String getName() {
        return name;
    }

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

    public int getAge() {
        return age;
    }

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

## *JavaBean.java*

```
package javaBean;

public class JavaBean {
    public static void main(String[] args) {
        PersonBean person = new PersonBean();
        person.setName("MCA");
        person.setAge(30);

        System.out.println("Name: " + person.getName());
        System.out.println("Age: " + person.getAge());
    }
}
```