A

Micro project

On

“Calculator”

Submitted By

Narayan Malvankar(31)

Tejas Argade (46)

Harshwardhan Kharat(35)

Guided By

Ms. S. S. Kawale

Diploma Course in Computer Technology

(As per directives of I Scheme, MSBTE)

Sinhgad Institutes

Sinhgad Technical Education Society’s

SOU.VENUTAI CHAVAN POLYTECHNIC

PUNE - 411041

ACADEMIC YEAR 2022-2023

Maharashtra State Board of Technical Education

Certificate

This is to certify that **Mr. Narayan Malvankar** with Roll No. **31** of Semester **IV** of Diploma in Computer Technology of Institute Sou. Venutai Chavan Polytechnic (Code: 0040) has successfully completed the Micro-Project in **Java Programming (22412)** for the academic year 2022-2023 as prescribed in the curriculum.

Program Code: CM

Course Code: CM/4/I

Place: Pune

Enrolment No: 2100400101

Date: _____

Exam Seat No:169297

Ms. S. S. Kawale	**Mrs.A.V.Kurkute**	**Dr.(Mrs.)M.S.Jadhav**
Course teacher	HOD of Department	Principal

Maharashtra State Board of Technical Education

Certificate

This is to certify that **Mr. Tejas Argade** with Roll No. **46** of Semester **IV** of Diploma in Computer Technology of Institute Sou. Venutai Chavan Polytechnic (Code: 0040) has successfully completed the Micro-Project in **Java Programming (22412)** for the academic year 2022-2023 as prescribed in the curriculum.

Program Code: CM

Course Code: CM/4/I

Place: Pune

Enrolment No: 2100400142

Date: _____

Exam Seat No: 169334

Ms. S. S. Kawale	**Mrs.A.V.Kurkute**	**Dr.(Mrs.)M.S.Jadhav**
Course Teacher	HOD of Department	Principal

Maharashtra State Board of Technical Education

Certificate

This is to certify that **Mr. Harshwardhan Kharat** Roll No.**35** of Semester **IV** of Diploma in Computer Technology of Institute Sou. Venutai Chavan Polytechnic (Code: 0040) has successfully completed the Micro-Project in **Java Programming (22412)** for the academic year 2022-2023 as prescribed in the curriculum.

Program Code: CM

Course Code: CM/4/I

Place: Pune Enrolment No: 2100400109

Date: _____ Exam Seat No:169305

Mss. S. S. Kawale	**Mrs.A.V.Kurkute**	**Dr.(Mrs.)M.S.Jadhav**
Course Teacher	HOD of Department	Principal

INDEX

Annexure - I

Micro-Project Proposal

“Calculator”

1.0 Aim of the Micro-Project:

The aim of the Micro-project is to develop a Calculator in Java.

2.0 Intended Course Outcomes:

a) Develop programs using Object-Oriented methodology in Java.

b) Apply concept of inheritance for code reusability.

3.0 Proposed methodology:

a) Study the concept of JAVA programming.
b) Study various syntax and functions of JAVA.
c) Study to create small programs using JAVA.
d) Study to create packages and developing a java file in it and importing it.
e) Study to calling the methods of different packages.
f) Make program for given criteria.
g) Prepare the final report.

4.0 Action Plan:

Sr. No.	Details of Activity	Planned Start Date	Planned Finish Date	Name of responsible Team members
1	Identify the requirements of the project.			Tejas Argade
2	Design the structure of the project.			Tejas Argade
3	Develop the program using in JAVA.			Narayan Malvankar
4	Debug code and eliminate errors occurred.			Narayan Malvankar
5	Test the project.			Harshwardhan Kharat
6	Prepare the final report.			Harshwardhan Kharat

5.0 Resources Required:

S. No.	Resources required	Specifications
1	Computer system	Intel i7 -12800P CPU, RAM 16 GB
2	Operating System	Windows 11, 64 Bit Operating System
3	Software	Java JDK 15.02, Notepad, Command Prompt, Visual Studio Code

6.0 Team members:

S. No.	Roll. Number	Name of Student
1	31	Narayan Malvankar
2	46	Tejas Argade
3	35	Harshwardhan Kharat

Annexure - II

Micro-Project Report

1.0 Rationale:

Java, the very popular language for desktop as well as mobile application development, has some unique features, also it's an object-oriented programming language.

In this course of microproject we are going to learn about one of Java's unique feature i.e., Package creation and importing it in a program to use its function.

By default, in Java the main class runs in a default java package, but one can create our own package and can implement classes in and methods, so that we can import that package to use its functions.

2.0 Aim of the Micro-Project:

To create a Airline and develop a java file in it, and to create a method for displaying list of seats.

3.0 Course Outcomes Achieved:

- Read and elementary modification to java program that solves the real world problems.
- Use an integrated development environment to write,compile,run,and test simple object oriented java programs.

4.0 Literature Review:

1. A Java class is a blueprint or template for creating objects that have certain characteristics, such as attributes (fields) and behaviors (methods).
2. Each Java class should have a unique name, and it can be either public or non-public (default).
3. A public class can be accessed from other packages, while a non-public class can only be accessed within its own package.
4. To create an object from a class, you need to use the "new" keyword and call the class's constructor method.
5. Inheritance is a fundamental concept in Java classes, where a subclass can inherit the attributes and methods of a superclass.
6. The Object class is the parent class of all other Java classes, and it provides some basic methods such as equals(), hashCode(), and toString().
7. The access modifiers (public, private, protected) can be used to control the visibility and accessibility of class members (fields and methods).
8. Java classes can also implement interfaces, which define a set of methods that the implementing class must provide.
9. Java supports nested classes, which are classes defined inside other classes and can be either static or non-static.
10. The Java standard library provides many pre-defined classes, such as Math, String, ArrayList, and HashMap, that you can use in your code.

Advantages of Java Class

1. Platform Independence: Java code can be run on any platform that has a JVM, making it highly portable.
2. Object-Oriented Programming (OOP): Java is a fully object-oriented programming language, making it easy to write modular and reusable code.
3. Robustness: Java has features like automatic memory management and exception handling, which make it more reliable and less prone to crashes.

4. Large Community: Java has a large and active community of developers, which means that there are many resources available for learning and troubleshooting.
5. Scalability: Java is well-suited for building large, complex applications that can handle high levels of traffic and data processing.

6.The aim of the Micro-project is to develop aAirline management system in Java.

5.0 Actual Methodology Followed:

I. Study the concept of JAVA programming.

II. Study various syntaxes and functions of JAVA.

III. Study to create small programs using JAVA.

IV. Study to create packages and defining classes in it and importing it. V. Study to calling the methods of different packages.

VI. Make program for given criteria.

VII. Prepare the final report.

6.0 Actual Resources Used:

S. No.	Resources required	Specifications
1	Computer system	Intel(R) Pentium CPU, RAM 8 GB
2	Operating System	Windows 11, 64 Bit Operating System
3	Softwares	Java JDK 15.02, Notepad++, Command Prompt, Visual Studio Code

Java Source code:

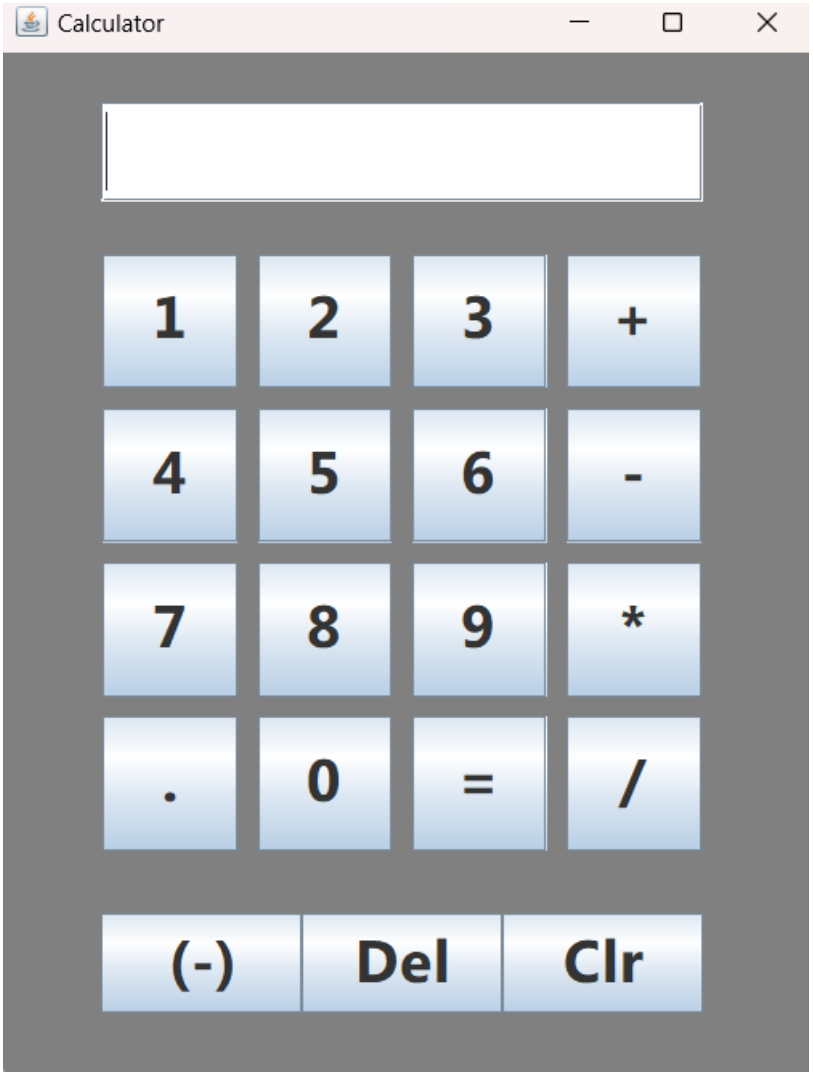
```
import java.awt.*;
import java.awt.event.*;

public class Calculator implements ActionListener{

	JFrame frame;
	JTextField textfield,textfield1;
	JButton[] numberButtons = new JButton[10];
	JButton[] functionButtons = new JButton[9];
	JButton addButton,subButton,mulButton,divButton,decButton,
equButton, delButton, clrButton,     negButton;
	JPanel panel;



	Font myFont = new Font("Microsoft PhagsPa",Font.BOLD,30);

	double num1=0,num2=0,result=0;
	char operator;
	String x,y;

	Calculator(){

		frame = new JFrame("Calculator");
		frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
		frame.setSize(420, 550);
		frame.setLayout(null);
		frame.getContentPane().setBackground(Color.gray);


		textfield = new JTextField();
		textfield.setBounds(50, 25, 300, 50);
		textfield.setFont(myFont);
		textfield.setEditable(true);




		addButton = new JButton("+");

		subButton = new JButton("-");
		mulButton = new JButton("*");
		divButton = new JButton("/");
		decButton = new JButton(".");
		equButton = new JButton("=");
		delButton = new JButton("Del");
		clrButton = new JButton("Clr");
		negButton = new JButton("(-)");

		functionButtons[0] = addButton;
		functionButtons[1] = subButton;
```

```
                functionButtons[2] = mulButton;
                functionButtons[3] = divButton;
                functionButtons[4] = decButton;
                functionButtons[5] = equButton;
                functionButtons[6] = delButton;
                functionButtons[7] = clrButton;
                functionButtons[8] = negButton;



                for(int i =0;i<9;i++) {

                      functionButtons[i].addActionListener(this);
                      functionButtons[i].setFont(myFont);
                      functionButtons[i].setFocusable(false);
                }

                for(int i =0;i<10;i++) {

                      numberButtons[i] = new JButton(String.valueOf(i));
                      numberButtons[i].addActionListener(this);
                      numberButtons[i].setFont(myFont);
                      numberButtons[i].setFocusable(false);
                }

                negButton.setBounds(50,430,100,50);
                delButton.setBounds(150,430,100,50);
                clrButton.setBounds(250,430,100,50);

                panel = new JPanel();
                panel.setBounds(50, 100, 300, 300);
                panel.setLayout(new GridLayout(4,4,10,10));
                panel.setBackground(Color.gray);

                panel.add(numberButtons[1]);
                panel.add(numberButtons[2]);
                panel.add(numberButtons[3]);
                panel.add(addButton);
                panel.add(numberButtons[4]);
                panel.add(numberButtons[5]);
                panel.add(numberButtons[6]);
                panel.add(subButton);
                panel.add(numberButtons[7]);
                panel.add(numberButtons[8]);
                panel.add(numberButtons[9]);
                panel.add(mulButton);
                panel.add(decButton);
                panel.add(numberButtons[0]);
                panel.add(equButton);
                panel.add(divButton);

                frame.add(panel);
                frame.add(negButton);
                frame.add(delButton);
                frame.add(clrButton);
                frame.add(textfield);
                frame.setVisible(true);
```

```
	}


	public static void main(String[] args) {

		Calculator calc = new Calculator();
	}


	public void actionPerformed(ActionEvent e) {

		for(int i=0;i<10;i++) {
			if(e.getSource() == numberButtons[i]) {

textfield.setText(textfield.getText().concat(String.valueOf(i)));
			}
		}
		if(e.getSource()==decButton) {
			textfield.setText(textfield.getText().concat("."));
		}
		if(e.getSource()==addButton) {
			num1 = Double.parseDouble(textfield.getText());
			operator ='+';
			textfield.setText(x);
		}
		if(e.getSource()==subButton) {
			num1 = Double.parseDouble(textfield.getText());
			operator ='-';
			textfield.setText("");
		}
		if(e.getSource()==mulButton) {
			num1 = Double.parseDouble(textfield.getText());
			operator ='*';
			textfield.setText("");
		}
		if(e.getSource()==divButton) {
			num1 = Double.parseDouble(textfield.getText());
			operator ='/';
			textfield.setText("");
		}
		if(e.getSource()==equButton) {
			num2=Double.parseDouble(textfield.getText());

			switch(operator) {
			case'+':
				result=num1+num2;
				break;
			case'-':
				result=num1-num2;
				break;
			case'*':
				result=num1*num2;
				break;
			case'/':
				result=num1/num2;
				break;
			}
```

```
                        textfield.setText(String.valueOf(result));
                        num1=result;
                }
                if(e.getSource()==clrButton) {
                        textfield.setText("");
                }
                if(e.getSource()==delButton) {
                        String string = textfield.getText();
                        textfield.setText("");
                        for(int i=0;i<string.length()-1;i++) {

    textfield.setText(textfield.getText()+string.charAt(i));
                        }
                }
                if(e.getSource()==negButton) {
                        double temp = Double.parseDouble(textfield.getText());
                        temp*=-1;
                        textfield.setText(String.valueOf(temp));
                }
        }
    }
```

8.0 Output:



9.0 Skills Developed:

During the course of this micro-project, we learnt to create a Calculator and define a class in which we created a method for Calculation.

a) We learnt various syntaxes of Java language.

b) We learnt to create package and imported it.

c) We also learnt to call methods of other packages by creating object of its class.

10.0 Applications of this Micro-project:

This micro-project finds its application in:

a) For creation of Calculator importing it for use in main file.

b) For defining classes in another package.

c) For accessing methods of classes from different packages.

11.0 Conclusion:

The code provides a basic implementation of Calculator in Java, demonstrating various concepts such as object-oriented programming, user input, and serialization. With additional development in these areas, the code could provide a more comprehensive solution.