MAHARASHTRA STATE BOARD
OF TECHNICAL EDUCATION, MUMBAI.

Government Polytechnic, Osmanabad

Microproject Title
"GUI based Calculator using JAVA Swing"

Submitted By

Roll No.	Student Name	Enrolment No.
35	Mohite Dipak Balu	1901180310

In Guidance of,
Mrs. R. V. Bondage

Submitted to
Department of Computer Engineering,
Government Polytechnic, Osmanabad.
Tuljapur road, Osmanabad – 413501, Maharashtra.

MAHARASHTRA STATE BOARD
OF TECHNICAL EDUCATION, MUMBAI.

Government Polytechnic, Osmanabad

CERTIFICATE

This is certified that,

The microproject entitled **GUI based Calculator using JAVA Swing** Submitted by **Mr. Mohite Dipak Balu** Roll Number **35** of **Fifth** Semester of Diploma in **Computer Engineering (CO)** has completed satisfactory in the course **Advanced Java Programming (22517)** for **academic year 2021-22** as prescribed in the curriculum.

Date   :   /   /

Place   : **Osmanabad**

Enrolment No. : **1901180310**

Exam Seat No. : **206209**

Institute
Seal

Lecturer   **Principal**   **Head of Department**

Index

Acknowledgement

- I take it is an opportunity to thank to all those who have directly and indirectly inspire directed an assisted me towards successful completion of this prosect report.

- I express my sincere thanks to principal **Prof. S. L. Andhare** sir and Head of Department **Prof. P. J. Bansode** sir for having me allowed to submit this report as part of my academics learning.

- I express my sincere thanks to **Mrs. R. V. Bondage** mam lecturer in Computer Engineering department of Government Polytechnic of Osmanabad tor encouragement throughout project report and guideline in designing and working out of this project…!

Micro-Project Proposal

2. Rationale:

- Everyone knows and uses this above-mentioned device named as 'Calculator'. In this microproject I have also created such a real-world calculator using Advanced Java Swing components.
- The overall purpose of this project is to evaluate and analyses the requirement of the customer, design and implement the system, testing the functionality and

maintain the software of a Modern GUI based Calculator using JAVA Swing, consistent with the requirement specification. transformation.

- Anyone who has to perform operations like addition, subtraction, multiplication, division, etc. regularly then they can use this software very easily.

3. Aim/Benefits of the Micro-Project:

- Implement a 'JAVA Swing' program to create a "Graphical User Interface based Calculator". In this microproject I have created a project as similar to a real Calculator using Advanced Java Programming. Aim of this microproject is to Perform all the operation performed by a Real Calculator such as Addition, Subtraction, Multiplication, Division, etc.

- Our project/ Software is able to perform all the operations using Graphical User Interface. The main focus of our microproject is to implement the swing components to create an attractive Calculator that works in with Addition, Subtraction, Multiplication, Division operations.

4. Course Outcome Achieved:

- Developing programs using GUI Framework (AWT and Swing).
- Developing programs to handle events of AWT and Swing Components.
- Implementing different Java Swing Components such as JButton, JLabel, JTextArea.
- Learned that how to implement Listeners such as ActionListener.
- Create a JFrame to contain all this Swing components.
- Implementing the use of Switch case Statement to handle different events generated by the user while using our calculator project.
- Studied that how the Calculator functionality actually works.
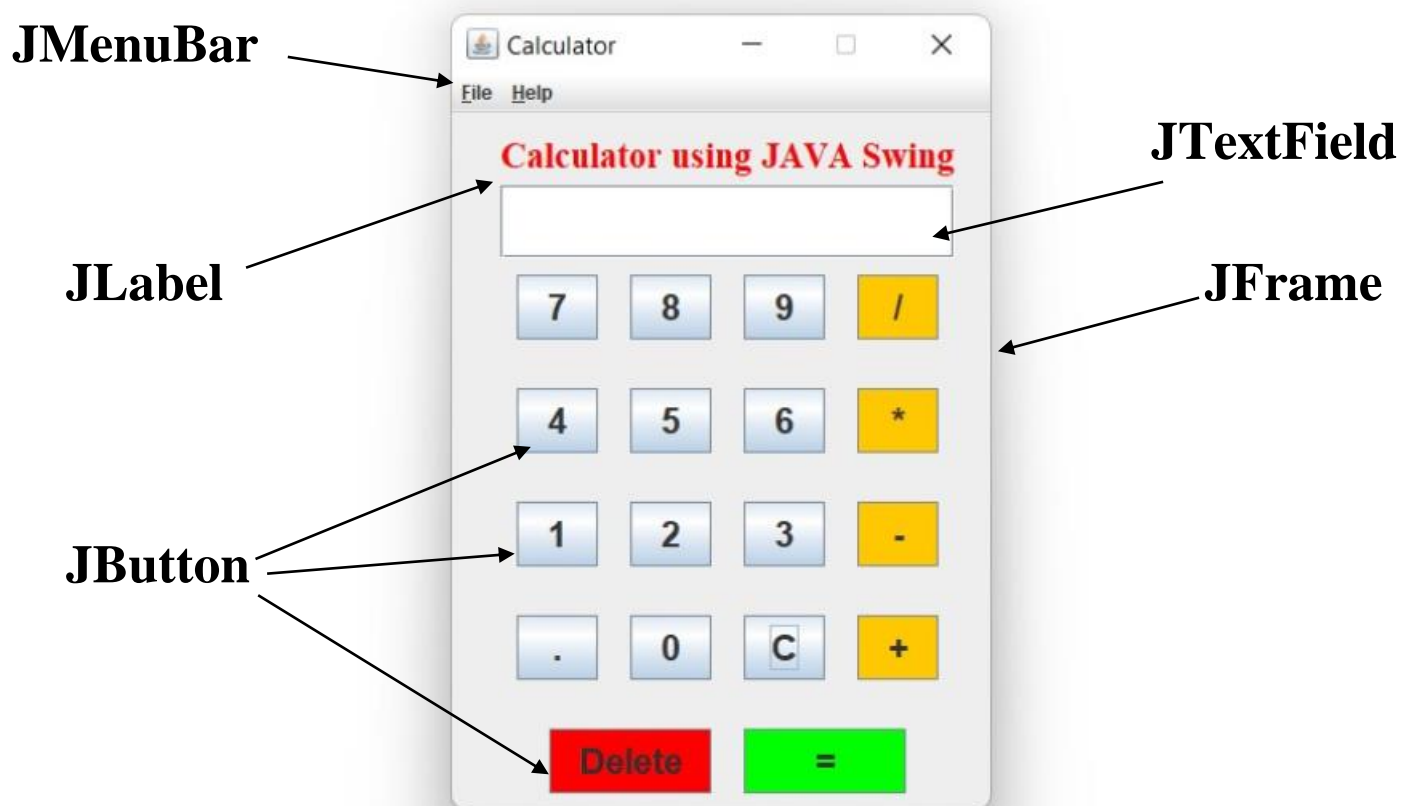- Developing programs to handle events in Java programming.

1. Modeling and Designing :

Step 1 : Create a GUI Design :

In this firstly we have created a GUI design of our Calculator using Swing Components.

The created design using different AWT and Swing components looks like: -

In this diagram I have navigated all the components which are used in this Calculator Software the description of this components is mentioned below:

1. JFrame:

The javax.swing.JFrame class is a type of container which inherits the java.awt.Frame class. JFrame works like the main window where components like labels, buttons, textfields are added to create a GUI.

2. JMenuBar:

The JMenuBar class is used to display menubar on the window or frame. It may have several menus.

The object of JMenu class is a pull down menu component which is displayed from the menu bar. It inherits the JMenuItem class.

3. JButton:

The JButton class is used to create a labeled button that has platform independent implementation. The application result in some action when the button is pushed. It inherits AbstractButton class.

4. JLabel:

The object of JLabel class is a component for placing text in a container. It is used to display a single line of read only text. The text can be changed by an application but a user cannot edit it directly. It inherits JComponent class.

5. JTextField:

The object of a JTextField class is a text component that allows the editing of a single line text. It inherits JTextComponent class.

Step 2 : Creating Events on used Swing Components:

Steps to perform Event Handling

1. Register the component with the Listener
2. Registration Methods
3. For registering the component with the Listener, many classes provide the registration methods.

2. Program Code:

```java
import javax.swing.*;
import java.awt.event.*;
import java.awt.Color;
import java.awt.Font;
class Calculator implements ActionListener
{
    JMenu jmenuFile, jmenuHelp;
    JMenuItem jmenuitemExit, jmenuitemAbout;
    JFrame f;
    JLabel l;
    JTextField t;
    JButton b1,b2,b3,b4,b5,b6,b7,b8,b9,b0,bdiv,bmul,bsub,badd,bde
c,clr,bdel,eq;
    static double a=0,b=0,result=0;
    static int operator=0;

    Calculator()
    {
        f=new JFrame("Calculator");
        jmenuFile = new JMenu("File");
        jmenuFile.setMnemonic(KeyEvent.VK_F);
```

```java
jmenuitemExit = new JMenuItem("Exit");
jmenuitemExit.setAccelerator(KeyStroke.getKeyStroke( KeyE
vent.VK_X, ActionEvent.CTRL_MASK));
jmenuFile.add(jmenuitemExit);
jmenuHelp = new JMenu("Help");
jmenuHelp.setMnemonic(KeyEvent.VK_H);
jmenuitemAbout = new JMenuItem("About Calculator");
jmenuHelp.add(jmenuitemAbout);
JMenuBar mb = new JMenuBar();
mb.add(jmenuFile);
mb.add(jmenuHelp);
l=new JLabel("Calculator using JAVA Swing");
t=new JTextField();
b1=new JButton("1");
b2=new JButton("2");
b3=new JButton("3");
b4=new JButton("4");
b5=new JButton("5");
b6=new JButton("6");
b7=new JButton("7");
b8=new JButton("8");
b9=new JButton("9");
b0=new JButton("0");
bdiv=new JButton("/");
bmul=new JButton("*");
bsub=new JButton("-");
badd=new JButton("+");
bdec=new JButton(".");
clr=new JButton("C");
bdel=new JButton("Delete");
eq=new JButton("=");
l.setBounds(30,10,290,30);
t.setBounds(30,45,280,45);
```

```
        b7.setBounds(40,100,50,40);
        b8.setBounds(110,100,50,40);
        b9.setBounds(180,100,50,40);
        bdiv.setBounds(250,100,50,40);
        b4.setBounds(40,170,50,40);
        b5.setBounds(110,170,50,40);
        b6.setBounds(180,170,50,40);
        bmul.setBounds(250,170,50,40);
        b1.setBounds(40,240,50,40);
        b2.setBounds(110,240,50,40);
        b3.setBounds(180,240,50,40);
        bsub.setBounds(250,240,50,40);
        bdec.setBounds(40,310,50,40);
        b0.setBounds(110,310,50,40);
        clr.setBounds(180,310,50,40);
        badd.setBounds(250,310,50,40);
        bdel.setBounds(60,380,100,40);
        eq.setBounds(180,380,100,40);

        f.add(l);
        f.add(t);
        f.add(b7);
        f.add(b8);
        f.add(b9);
        f.add(bdiv);
        f.add(b4);
        f.add(b5);
        f.add(b6);
        f.add(bmul);
        f.add(b1);
        f.add(b2);
        f.add(b3);
        f.add(bsub);
```

```java
        f.add(bdec);
        f.add(b0);
        f.add(clr);
        f.add(badd);
        f.add(bdel);
        f.add(eq);
        f.setJMenuBar(mb);

        t.setHorizontalAlignment(JTextField.RIGHT);
        l.setFont(new Font("Serif", Font.BOLD, 22));
        l.setForeground(Color.RED);
        t.setFont(new Font("Arial", Font.BOLD, 22));
        b0.setFont(new Font("Arial", Font.BOLD, 22));
        b1.setFont(new Font("Arial", Font.BOLD, 22));
        b2.setFont(new Font("Arial", Font.BOLD, 22));
        b3.setFont(new Font("Arial", Font.BOLD, 22));
        b4.setFont(new Font("Arial", Font.BOLD, 22));
        b5.setFont(new Font("Arial", Font.BOLD, 22));
        b6.setFont(new Font("Arial", Font.BOLD, 22));
        b7.setFont(new Font("Arial", Font.BOLD, 22));
        b8.setFont(new Font("Arial", Font.BOLD, 22));
        b9.setFont(new Font("Arial", Font.BOLD, 22));
        badd.setFont(new Font("Arial", Font.BOLD, 22));
        bdiv.setFont(new Font("Arial", Font.BOLD, 22));
        bmul.setFont(new Font("Arial", Font.BOLD, 22));
        bsub.setFont(new Font("Arial", Font.BOLD, 22));
        bdiv.setFont(new Font("Arial", Font.BOLD, 22));
        bdiv.setFont(new Font("Arial", Font.BOLD, 22));
        bdel.setFont(new Font("Arial", Font.BOLD, 22));
        eq.setFont(new Font("Arial", Font.BOLD, 22));
        bdec.setFont(new Font("Arial", Font.BOLD, 22));
        clr.setFont(new Font("Arial", Font.BOLD, 22));
```

```java
        badd.setBackground(Color.ORANGE);
        bdiv.setBackground(Color.ORANGE);
        bmul.setBackground(Color.ORANGE);
        bsub.setBackground(Color.ORANGE);
        bdel.setBackground(Color.ORANGE);
        bdel.setBackground(Color.RED);
        eq.setBackground(Color.GREEN);

        f.setLayout(null);
        f.setVisible(true);
        f.setSize(350,500);

        f.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        f.setResizable(false);
        b1.addActionListener(this);
        b2.addActionListener(this);
        b3.addActionListener(this);
        b4.addActionListener(this);
        b5.addActionListener(this);
        b6.addActionListener(this);
        b7.addActionListener(this);
        b8.addActionListener(this);
        b9.addActionListener(this);
        b0.addActionListener(this);
        badd.addActionListener(this);
        bdiv.addActionListener(this);
        bmul.addActionListener(this);
        bsub.addActionListener(this);
        bdec.addActionListener(this);
        clr.addActionListener(this);
        bdel.addActionListener(this);
        eq.addActionListener(this);
    }
```

```java
public void actionPerformed(ActionEvent e)
{
    if(e.getSource()==b1)
    t.setText(t.getText().concat("1"));
    if(e.getSource()==b2)
    t.setText(t.getText().concat("2"));
    if(e.getSource()==b3)
    t.setText(t.getText().concat("3"));
    if(e.getSource()==b4)
    t.setText(t.getText().concat("4"));
    if(e.getSource()==b5)
    t.setText(t.getText().concat("5"));
    if(e.getSource()==b6)
    t.setText(t.getText().concat("6"));
    if(e.getSource()==b7)
    t.setText(t.getText().concat("7"));
    if(e.getSource()==b8)
    t.setText(t.getText().concat("8"));
    if(e.getSource()==b9)
    t.setText(t.getText().concat("9"));
    if(e.getSource()==b0)
    t.setText(t.getText().concat("0"));
    if(e.getSource()==bdec)
    t.setText(t.getText().concat("."));
    if(e.getSource()==badd)
    {
    a=Double.parseDouble(t.getText());
    operator=1;
    t.setText("");
    }
    if(e.getSource()==bsub)
    {
```

```java
a=Double.parseDouble(t.getText());
operator=2;
t.setText("");
}
if(e.getSource()==bmul)
{
a=Double.parseDouble(t.getText());
operator=3;
t.setText("");
}
if(e.getSource()==bdiv)
{
a=Double.parseDouble(t.getText());
operator=4;
t.setText("");
}
if(e.getSource()==eq)
{
b=Double.parseDouble(t.getText());
switch(operator)
{
case 1: result=a+b;
break;
case 2: result=a-b;
break;
case 3: result=a*b;
break;
case 4: result=a/b;
break;
default: result=0;
}
t.setText(""+result);
}
```
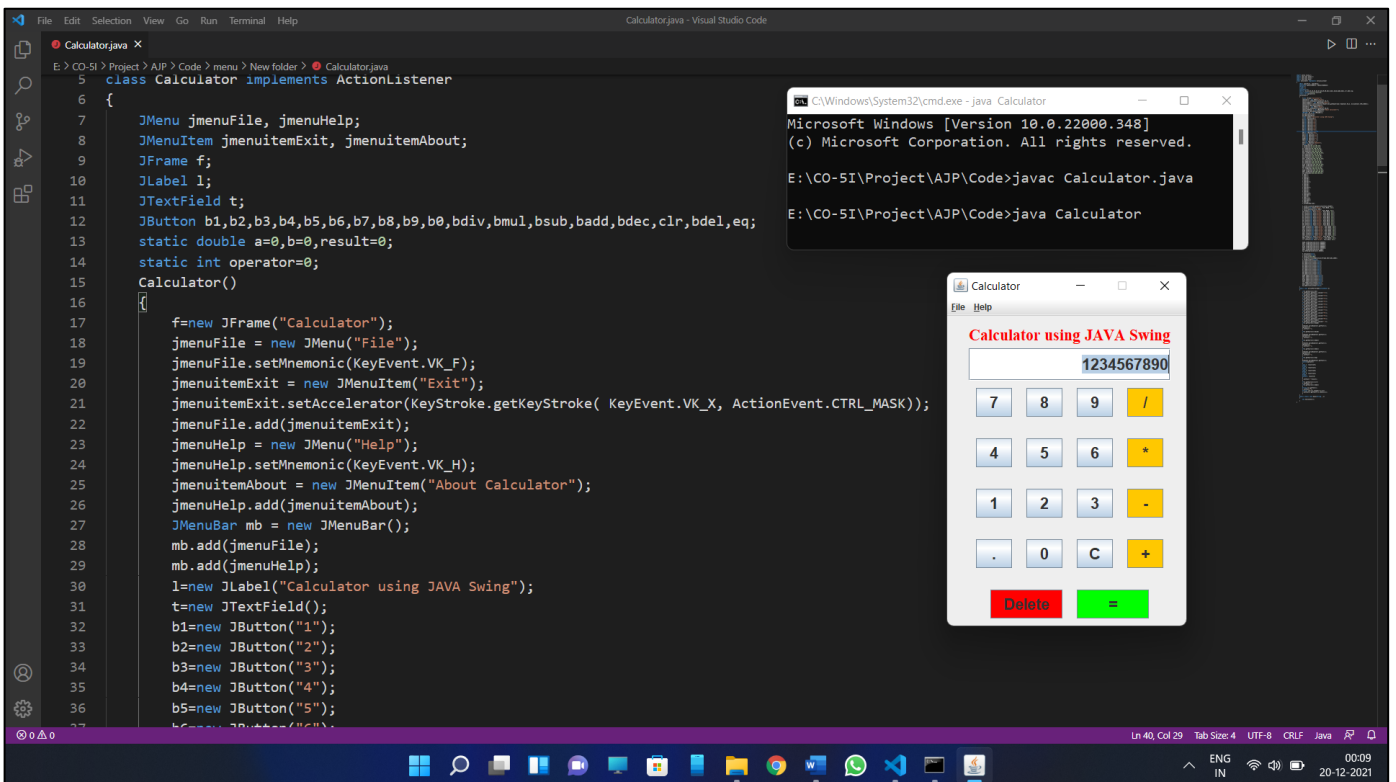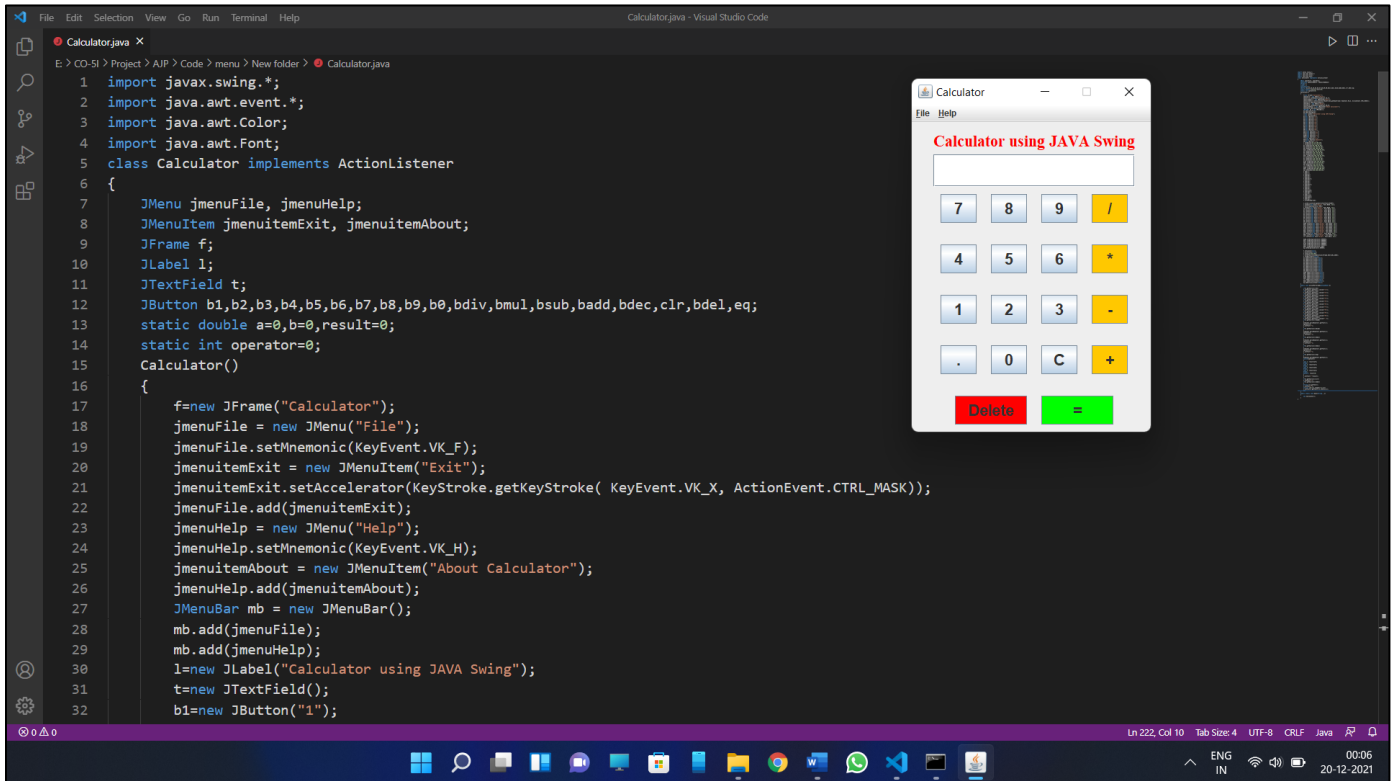
```java
        if(e.getSource()==clr)
        t.setText("");
        if(e.getSource()==bdel)
        {
        String s=t.getText();
        t.setText("");
        for(int i=0;i<s.length()-1;i++)
        t.setText(t.getText()+s.charAt(i));
        }
    }
    public static void main(String...s)
    {
        new Calculator();
    }
}
```

Note: - This program must be saved as Calculator.java
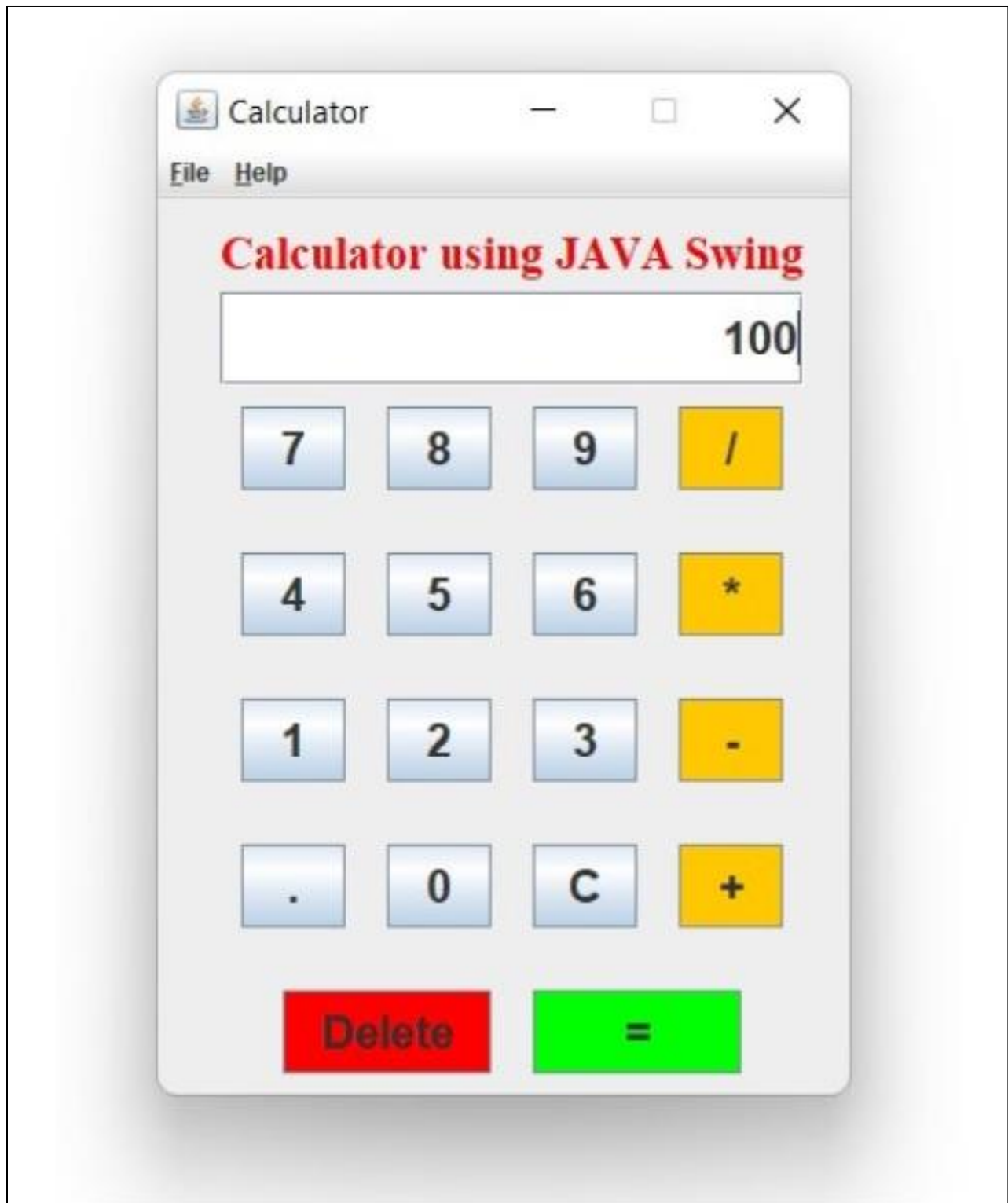
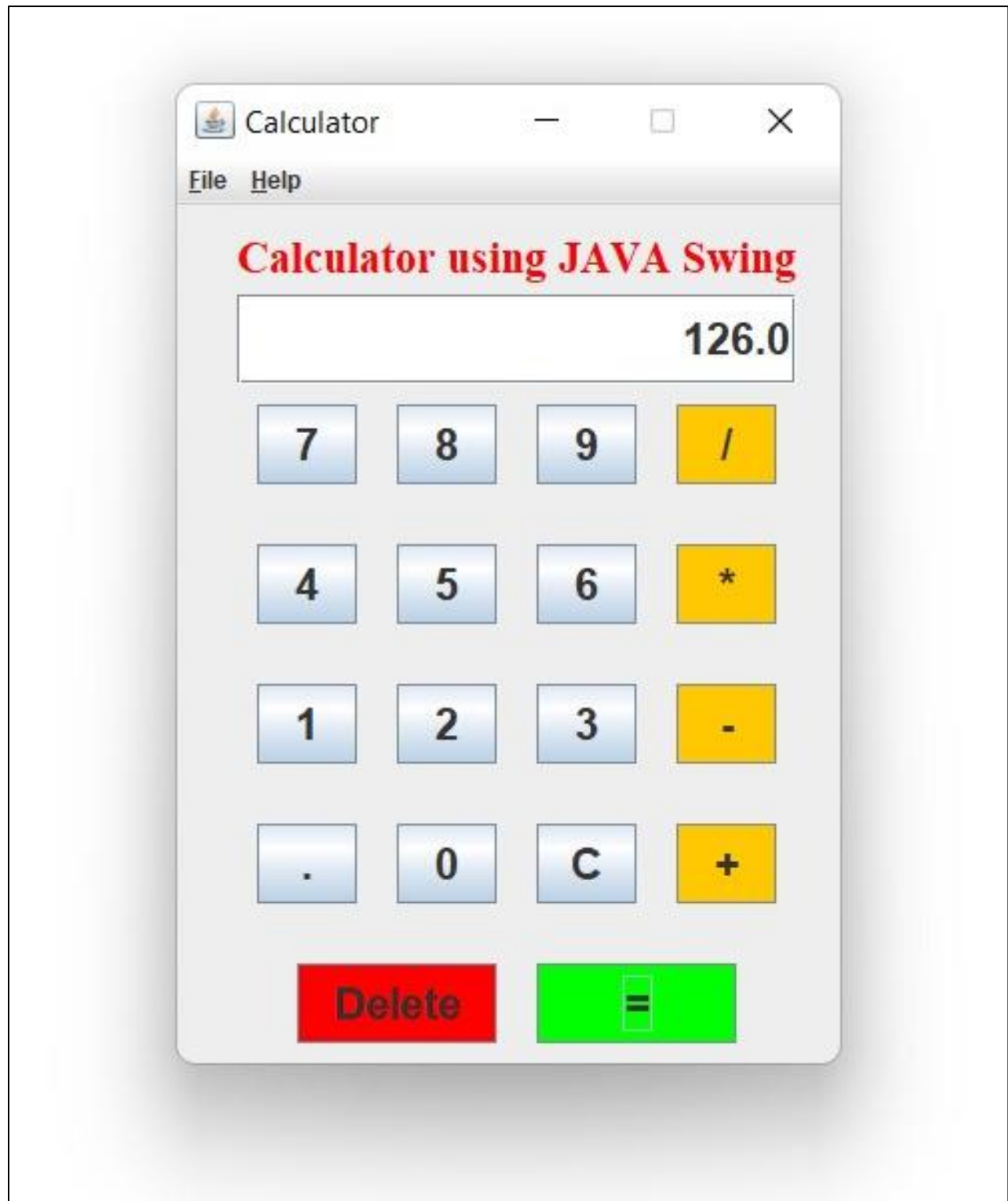# 6. Output of Program Code:

1. Start-up Screen of our Calculator:
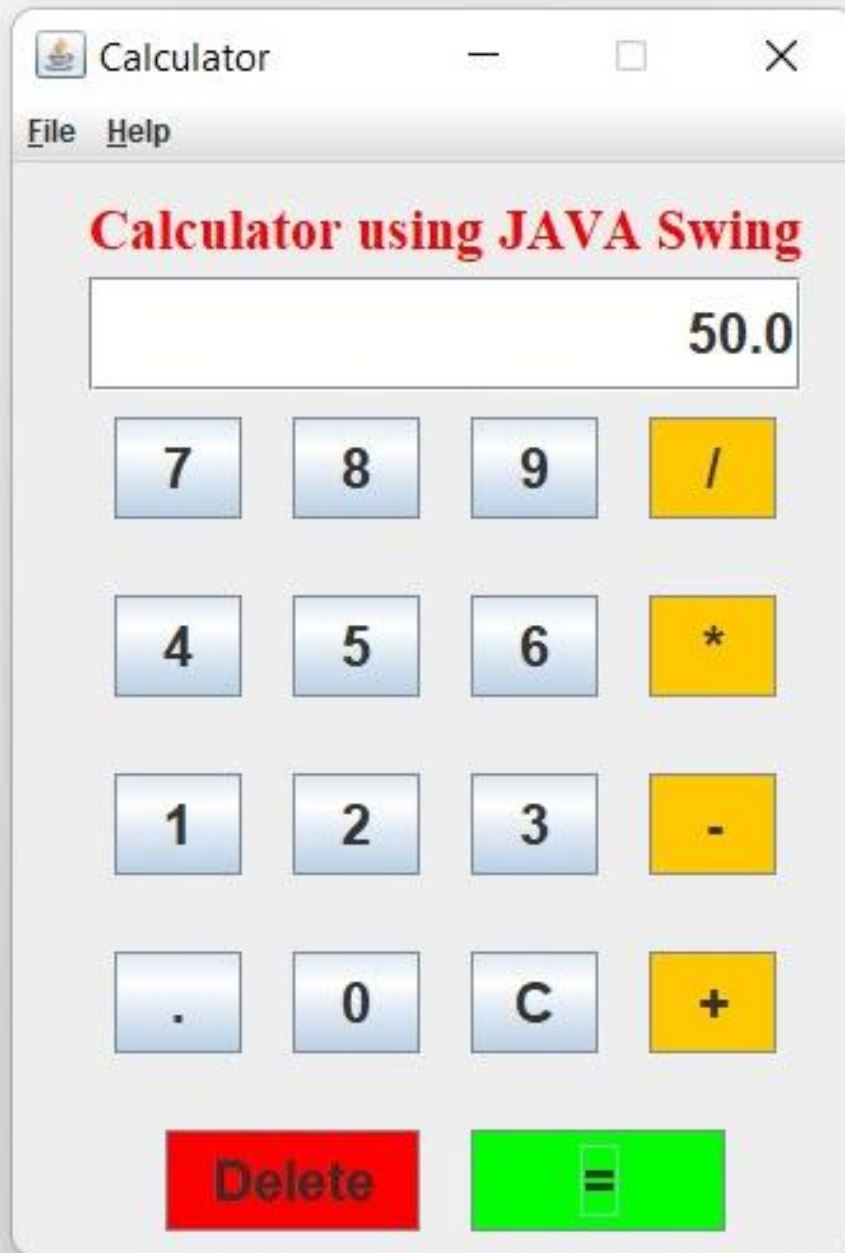
2. Entering 1ˢᵗ Operand:
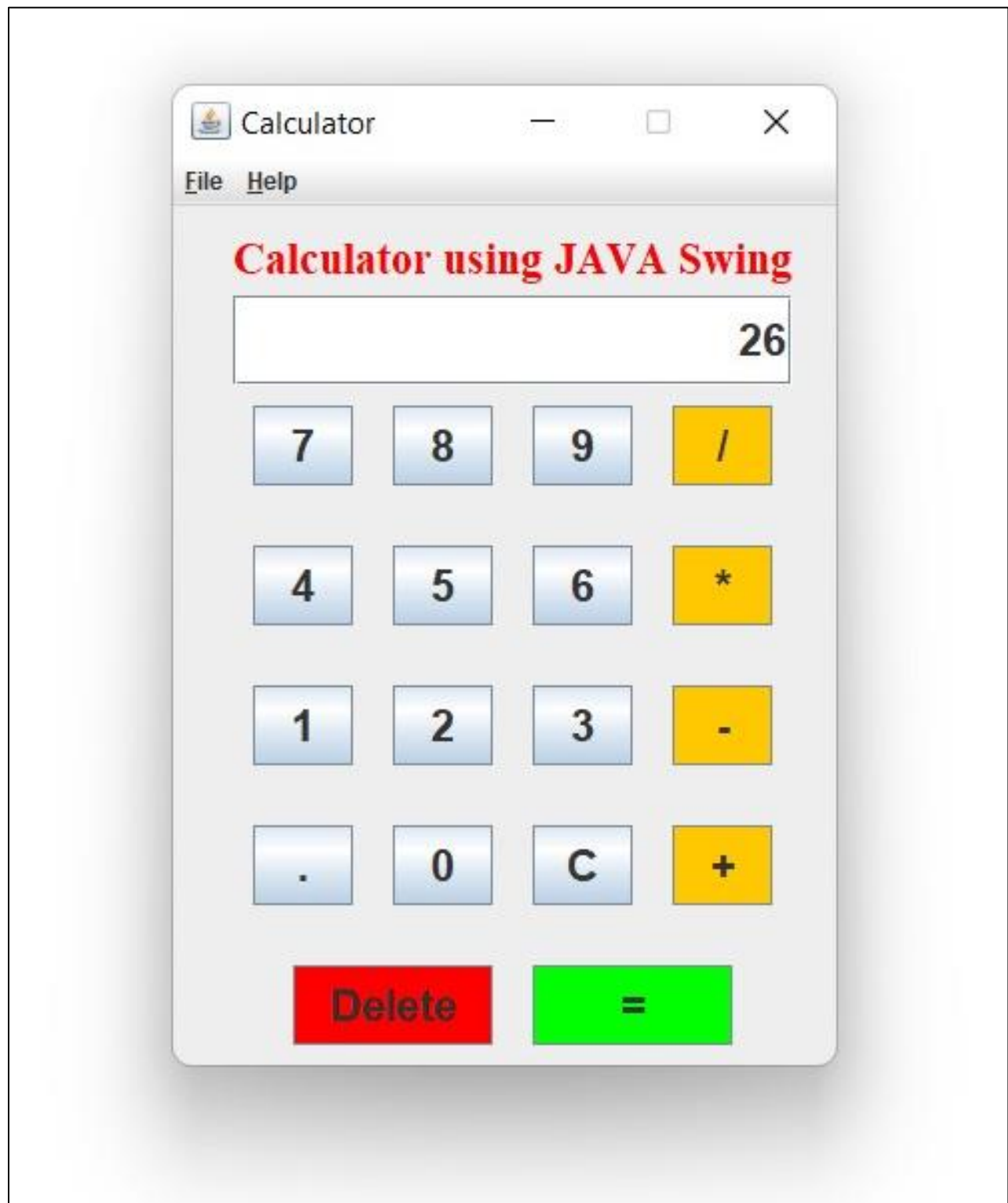
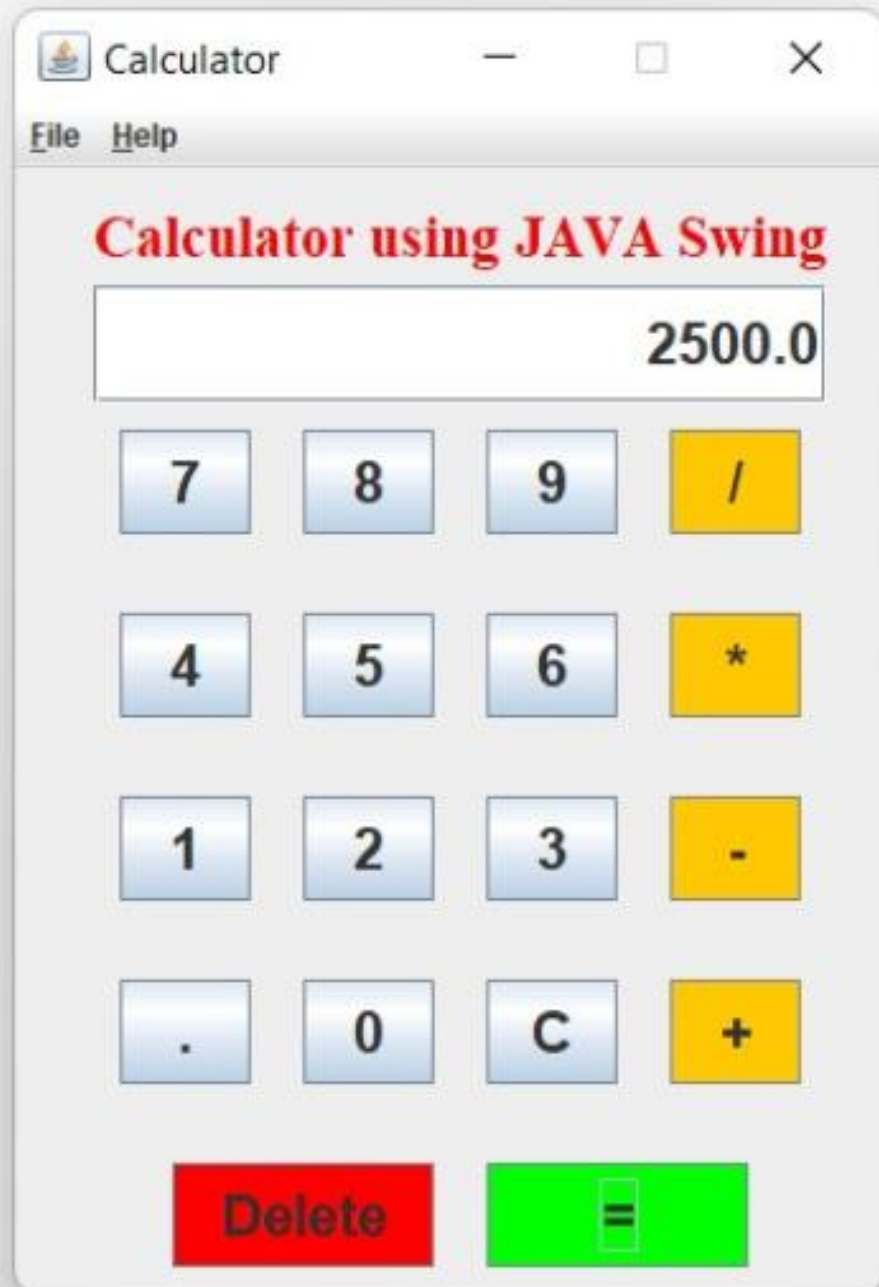# 3. Selected Addition (+) Operation and Entering 2nd Operand:

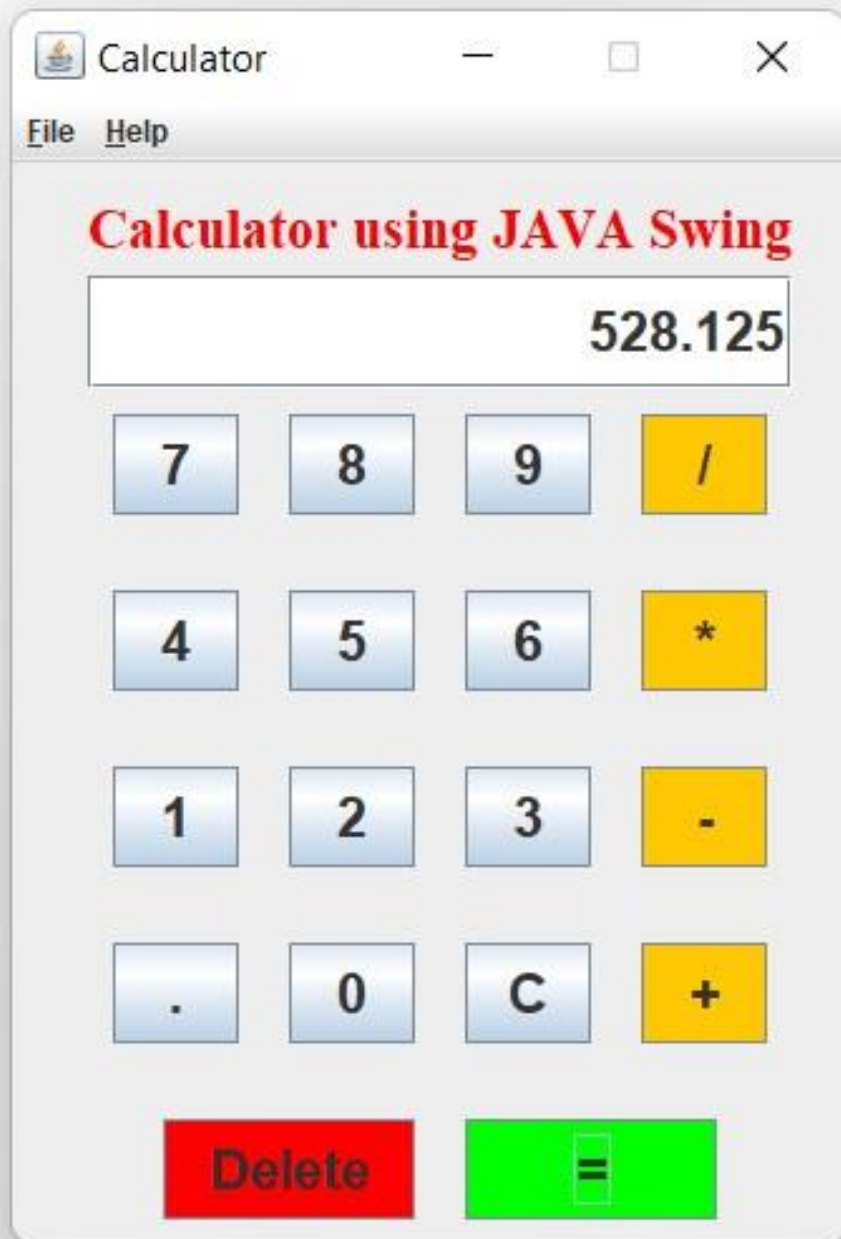# 4. Result after pressing '=' Button:

5. Subtraction after entering 100 as first operand and 50 as second operand:
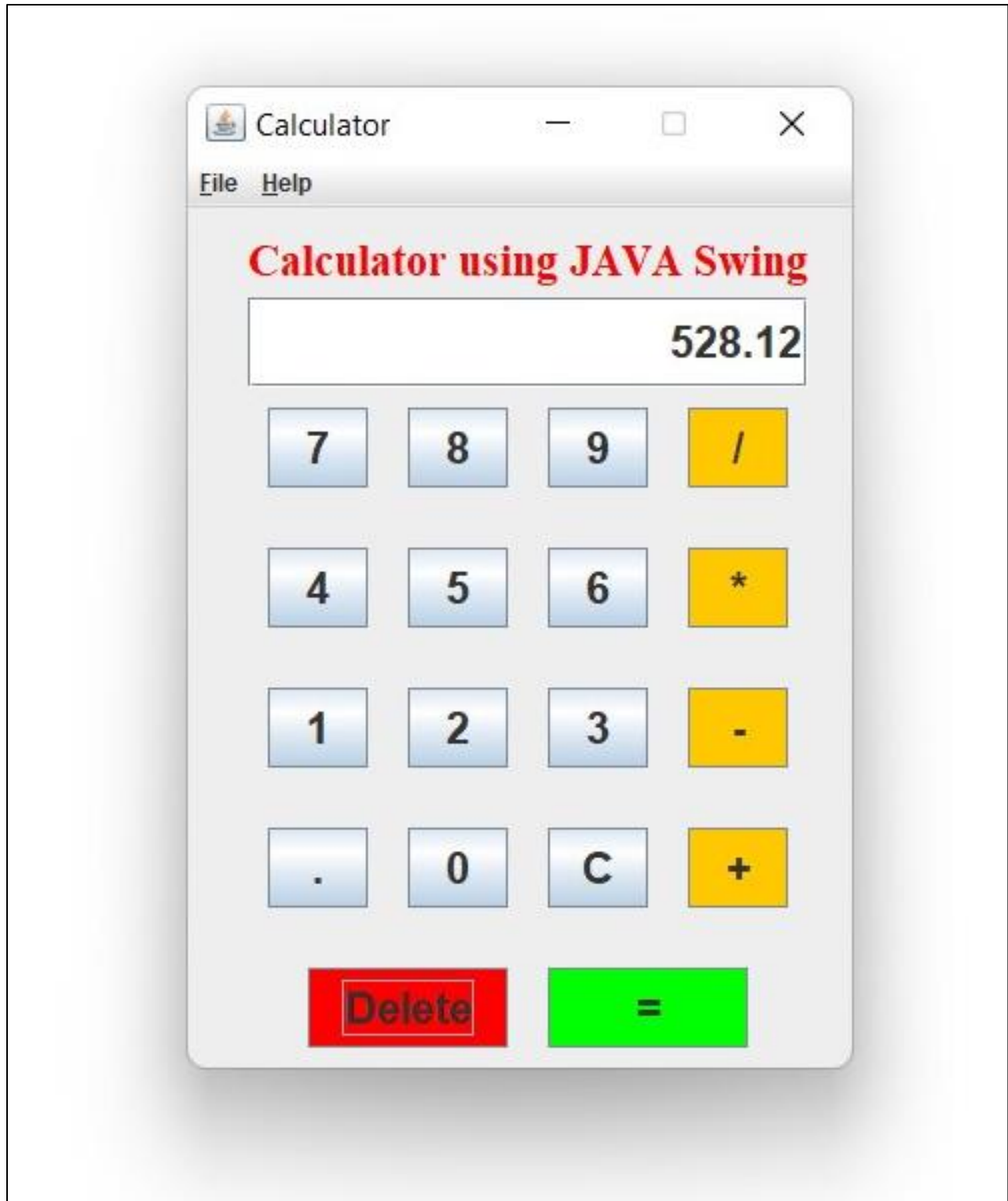
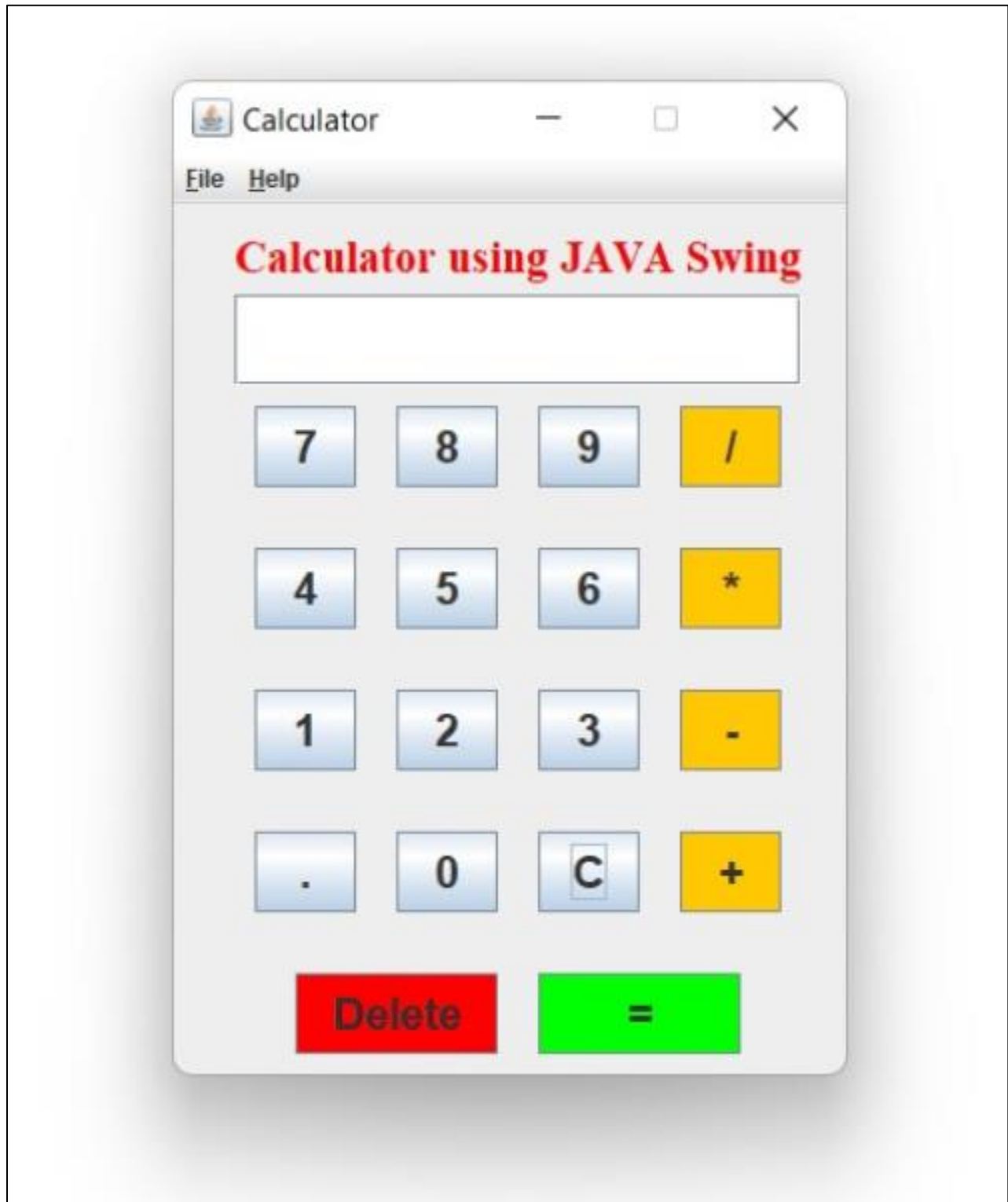# 6. Multiplication after entering 50 as first operand and 50 as second operand:

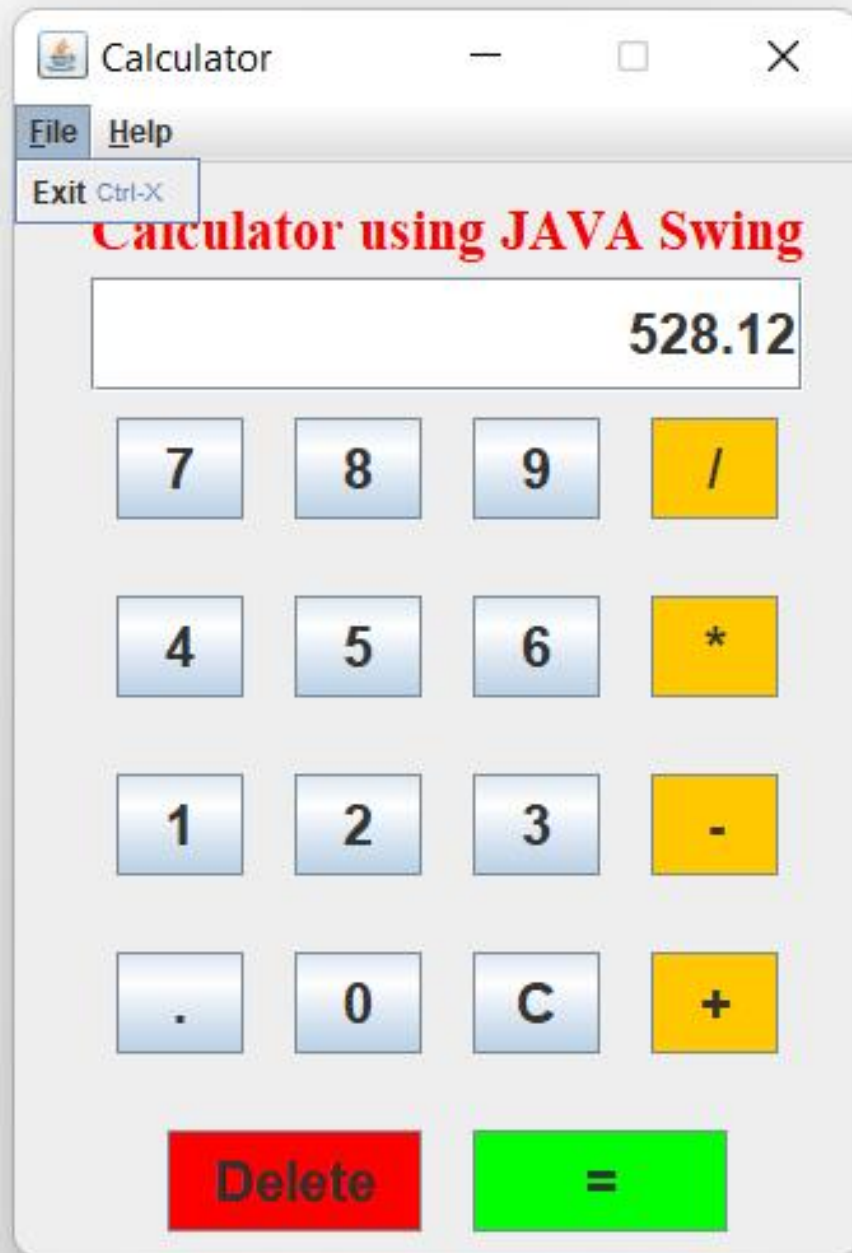# 7. Division after entering 8450 as first operand and 16 as second operand:
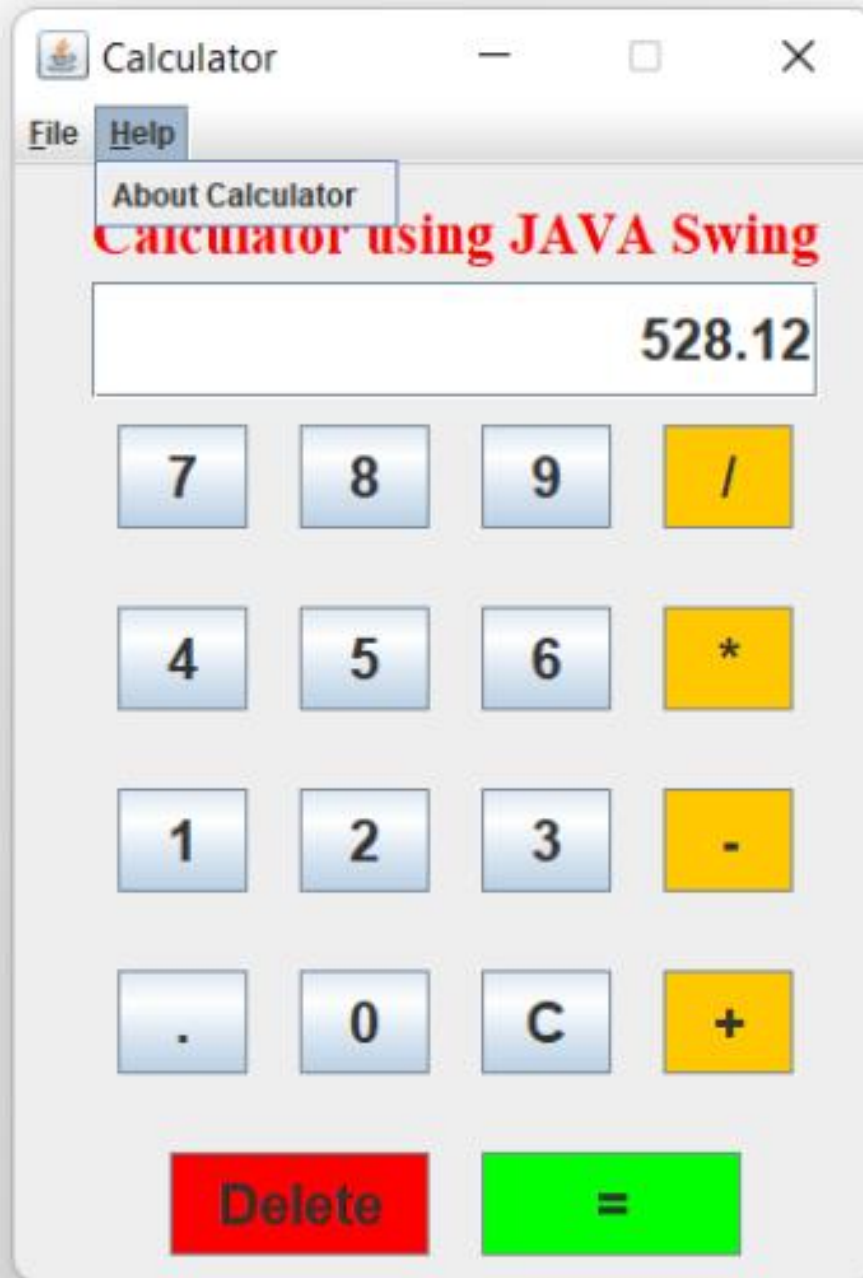
8. Delete Button:

9. Clear Screen (Clear) Button:

10. Menu - File:

11. Menu - Help:

7. Actual Resources Used:

Sr. No.	Name of Resource	Specification	Quantity
1.	Computer System	Intel Core-i3 10th gen with 8GB RAM, 512GB SSD	1
2.	Operating System	Windows 11	1
3.	Software	Visual Studio Code (Version 1.62), Command Prompt	1

8. Skill Developed/ Learning Outcome:

a) Learned about Java Swing Components.

b) Studied that how create an Event on Buttons.

c) Learned implementation of ActionEvent Listener.

d) Studied that how to do proper use of setBounds() method.

e) Learned that how to set the colors to different components of the Swing components.

f) Learned to create a JFrame and place different components on Containers/JFrame.

9. Advantages of Calculator:

i. Most handheld or online calculators are relatively cheap, various designs and functionality suitable for various budgets.

ii. They are also inexpensive to maintain once handled with care. As a professional, in any field, **the Calculator save time, energy and increases efficiency** in workplace.

iii. When teachers realize that the mathematical capacity of some of their pupils are not so great, using this handheld device or the online calculator platforms, helps them manage addition, subtraction, multiplication and division problems in an efficient manner. Teachers understand that the student would not have learned mathematical table of 20x, or would be well versed in solving the problem without the help of a mathematical device or paper.

iv. They would make fewer mistakes and will be more comfortable in solving tough problems. If a student is confident about his or her ability, then the problem seems halfway solved.

10. Disadvantages of Calculator:

i. Dependency:

Even though calculators can do the basic operations instantly, students should not use it all the time. When a student is always dependent on a calculator for solving problems of addition, subtraction, multiplication and division, it may make them incapable of solving these basic operations manually. This may cause them real trouble during their tests and even at the later stages of their life where they will need these skills to solve complex problems.

ii. Cheating:

The availability of graphic calculators has made it easier for students to cheat during their tests. Besides letting you perform basic mathematical operations, these calculators allow you to store information like notes, formulas etc. This gives students an opportunity to store unauthorized information to cheat on the test. This is one of the major disadvantages of a calculator.

iii. High cost:

While the cost of a simple calculator is not that high, calculators like the graphing calculator is of high cost. In addition to that, these calculators need batteries to operate which is again a recurring cost. However, you have the option to use online calculators for the purpose which gives you access to all the functionality of a physical calculator.

11. Application of this Micro-project:

i. Calculator in our today's daily life it is booming, and modern innovations and this technology has changed our lives.

ii. This Calculator allows students to solve complicated problems quickly and in an efficient manner. Additionally, it can reduce the problem to simpler tasks and allows the student to devote more time in understanding the problem. Secondly, they are saved from monotonous calculations and the same boring procedure.

iii. Anyone who has to perform operations like addition, subtraction, multiplication, division, etc. regularly then they can use this software very easily.

iv. The overall purpose of this project is to evaluate and analyses the requirement of the customer, design and implement the system, testing the functionality and maintain the software of a Modern GUI based Calculator using JAVA Swing, consistent with the requirement specification. transformation.

v. Calculator helps students to avoid boredom, and it does not demoralize their mathematical understanding. If mathematics is not made entertaining, pupils can feel bored, and it can demoralize their mathematical understanding.

11. Conclusion:

I have created a Calculator Software using JAVA Swing for performing different arithmetic operations, this software is created using software such as Visual Studio Code, notepad++, Command prompt. Here every used software has a different use of it, like visual studio is used for code typing as Code editor in same manner notepad++ is also used. Using Command prompt, we have compiled and run our program. For compilation we have used command: javac, Ex.: javac Calculator.java and for run the java program we used java command as, Ex.: java Calculator. After clicking the ENTER our software runs successfully.

12. References:

1. Textbook: Advanced Java Programming (22517)
2. Our Subject Teacher Mrs. R. V. Bondage Mam.
3. Book: Head First Java - by Bert Bates & Kathy Sierra.
4. https://www.javatpoint.com/java-swing
5. https://www.javatpoint.com/event-handling-in-java
6. https://docs.oracle.com/javase/tutorial/uiswing/events/intro.html
7. https://www.tutorialspoint.com/awt/awt_event_handling.htm
8. https://www.geeksforgeeks.org/java-swing-jpanel-with-examples/

THANK YOU