Task 1: Simple Calculator Application.

Code:

```java
package calapp;

import javax.swing.*;
import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;

public class CalculatorApp extends JFrame implements ActionListener {

private JTextArea display;
private JPanel buttonPanel;
private JButton[] buttons;
private String[] buttonLabels = {
"7", "8", "9", "/",
"4", "5", "6", "*",
"1", "2", "3", "-",
"0", "C", "=", "+",
".", "DEL"
};

private StringBuilder expression = new StringBuilder();
private boolean resultDisplayed = false;

public CalculatorApp() {
setTitle("Calculator");
setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
setLayout(new BorderLayout());

// Display
display = new JTextArea();
display.setFont(new Font("Arial", Font.PLAIN, 36));
display.setWrapStyleWord(true);
display.setLineWrap(true);
display.setOpaque(false);
display.setEditable(false);
JScrollPane scrollPane = new JScrollPane(display);
scrollPane.setVerticalScrollBarPolicy(ScrollPaneConstants.VERTICAL_SCROLLBAR_AS_NEEDED);
scrollPane.setPreferredSize(new Dimension(400, 100));
add(scrollPane, BorderLayout.NORTH);

// Button panel
buttonPanel = new JPanel();
buttonPanel.setLayout(new GridLayout(5, 4, 10, 10));
buttons = new JButton[buttonLabels.length];
```

```java
        for (int i = 0; i < buttonLabels.length; i++) {
buttons[i] = new JButton(buttonLabels[i]);
buttons[i].setFont(new Font("Arial", Font.PLAIN, 24));
buttons[i].addActionListener(this);
buttonPanel.add(buttons[i]);
}
add(buttonPanel, BorderLayout.CENTER);

pack();
setLocationRelativeTo(null);
}

public void actionPerformed(ActionEvent e) {
String command = e.getActionCommand();
if(Character.isDigit(command.charAt(0))||(command.equals(".")&&
!expression.toString().contains("."))) {
if (resultDisplayed) {
expression.setLength(0);
resultDisplayed = false;
}
expression.append(command);
} else if (command.charAt(0)=='C') {
expression.setLength(0);
} else if (command.charAt(0)=='=') {
try {
double result = evaluateExpression();
expression.setLength(0);
expression.append(result);
resultDisplayed = true;
} catch (ArithmeticException ex) {
expression.setLength(0);
expression.append("Error: Division by zero");
resultDisplayed = true;
}
} else if (command.equals("DEL")) {
if (expression.length() > 0) {
expression.deleteCharAt(expression.length() - 1);
}
} else {
if (resultDisplayed) {
expression.setLength(0);
expression.append(display.getText());
resultDisplayed = false;
}
expression.append(command);
}
display.setText(expression.toString());
}

private double evaluateExpression() {
```

```java
String expressionWithoutSpaces = expression.toString().replaceAll("\\s", "");
String[] parts = expressionWithoutSpaces.split("[-+*/]");
if (expressionWithoutSpaces.startsWith("-")) {
parts[0] = "-" + parts[0];
}

double num1 = Double.parseDouble(parts[0]);
char operator = expressionWithoutSpaces.charAt(parts[0].length());
double num2 = Double.parseDouble(parts[1]);

switch (operator) {
case '+':
return num1 + num2;
case '-':
return num1 - num2;
case '*':
return num1 * num2;
case '/':
if (num2 != 0) {
return num1 / num2;
} else {
throw new ArithmeticException("Division by zero");
}
}
return 0;
}

public static void main(String[] args) {
SwingUtilities.invokeLater(() -> {
CalculatorApp calculator = new CalculatorApp();
calculator.setVisible(true);
});
}
}
```
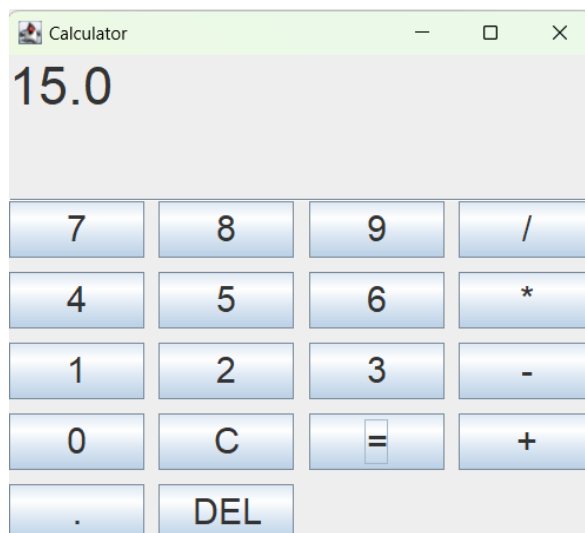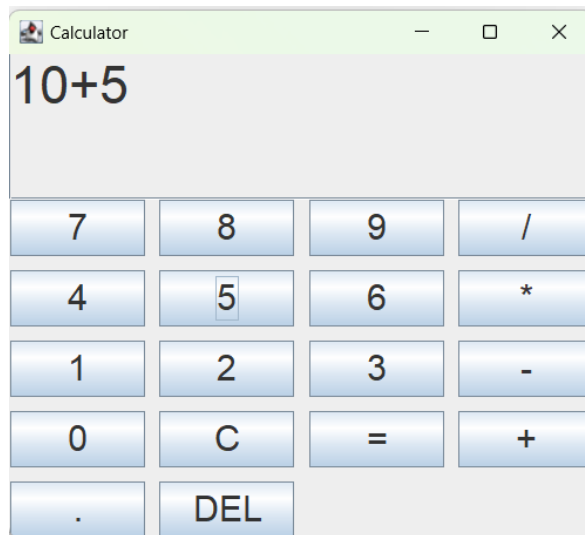
OUTPUT:

Task 2: Number Guessing Game.

Code:

```java
package guess;
import javax.swing.*;

public class NumberGuess {
public static void main(String[] args) {
int computerNumber = (int) (Math.random()*100 + 1);
int userAnswer = 0;
int count = 1;

while (userAnswer != computerNumber)
{
String response = JOptionPane.showInputDialog(null,
"Enter a guess between 1 and 100", "Guessing Game", 3);
userAnswer = Integer.parseInt(response);
JOptionPane.showMessageDialog(null,""+determineGuess(userAnswer,
computerNumber, count));
count++;
}
}

public static String determineGuess(int userAnswer, int computerNumber, int
count){
if (userAnswer <=0 || userAnswer >100) {
return "Your guess is invalid";
}
else if (userAnswer == computerNumber ){
return "Correct!\nTotal Guesses: " + count;
}
else if (userAnswer > computerNumber) {
return "Your guess is too high, try again.\nTry Number: " + count;
}
else if (userAnswer < computerNumber) {
return "Your guess is too low, try again.\nTry Number: " + count;
}
else {
return "Your guess is incorrect\nTry Number: " + count;
}
}
}
```

OUTPUT:

Guessing Game   ✕

?   **Enter a guess between 1 and 100**

    [   ]

        OK   **Cancel**

Message   ✕

ⓘ   **Your guess is too low, try again.**

     Try Number: 1

        OK

Message   ✕

ⓘ   **Your guess is too high, try again.**

     Try Number: 5

        OK

Message   ✕

ⓘ   **Correct!**

     Total Guesses: 9

        OK