```
package quiz;
///*
//* Aim : Write a program to find square of a
given number using applet.
//* Name : Sayyad Mohamed Samar
//* UIN : 231P082
//* Div : A.
//*/
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
public class SquareApplet extends JApplet implements ActionListener {
 // Declaring components
 private Label label;
 private TextField inputField;
 private Button squareButton;
 private String result = "";
 @Override
 public void init() {
 // Setting layout and adding components
 setLayout(new FlowLayout());
 label = new Label("Enter a number: ");
 add(label);
 inputField = new TextField(10);
 add(inputField);
 squareButton = new Button("Find Square");
 add(squareButton);
// Adding action listener to the button
```

```
squareButton.addActionListener(this);
}
// Action performed when the button is clicked
@Override
public void actionPerformed(ActionEvent e) {
try {
// Get the input from the text field and calculate the square
int number = Integer.parseInt(inputField.getText());
int square = number * number;
result = "Square of " + number + " is " + square;
} catch (NumberFormatException ex) {
// Handle invalid input
result = "Please enter a valid number.";
}
// Repaint the applet to display the result
repaint();
}
@Override
public void paint(Graphics g) {
// Display the result
g.drawString(result, 50, 150);
}
public static void main(String[] args) {
// Create a JFrame to hold the applet
JFrame frame = new JFrame("Square Applet");
SquareApplet applet = new SquareApplet();
// Initialize the applet (same as if it were run in a browser)
applet.init();
```

```
 applet.start();
 // Add the applet to the frame
 frame.add(applet);
 frame.setSize(400, 200);
 frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
 frame.setVisible(true);
 }
}
```

```
package quiz;
///*
//* Aim :Write a program to implement calculator using Actionlistener.
//* Name : Sayyad Mohamed Samar
//* UIN : 231P082
//* Div : A.
//*/

import javax.swing.*;
import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
public class Calculator implements ActionListener {
 JFrame frame;
 JTextField display;
 JButton[] numberButtons;
 JButton addButton, subButton, mulButton, divButton, equButton, delButton, clrButton;
 JPanel panel;
 double num1 = 0, num2 = 0, result = 0;
 char operator;
 public Calculator() {
 // Frame settings
 frame = new JFrame("Calculator");
 frame.setSize(400, 600);
 frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
 frame.setLayout(null);
 // Display panel
```

```
display = new JTextField();
display.setBounds(50, 25, 300, 50);
display.setFont(new Font("Arial", Font.BOLD, 24)); // Increased font size
display.setBackground(Color.LIGHT_GRAY); // Background color
display.setEditable(false);
frame.add(display);
// Number buttons
numberButtons = new JButton[10];
for (int i = 0; i < 10; i++) {
numberButtons[i] = new JButton(String.valueOf(i));
numberButtons[i].setFont(new Font("Arial", Font.BOLD, 24));
numberButtons[i].addActionListener(this);
}
// Operation buttons
addButton = new JButton("+");
subButton = new JButton("-");
mulButton = new JButton("*");
divButton = new JButton("/");
equButton = new JButton("=");
delButton = new JButton("Delete");
clrButton = new JButton("Clear");
// Set button fonts
addButton.setFont(new Font("Arial", Font.BOLD, 24));
subButton.setFont(new Font("Arial", Font.BOLD, 24));
mulButton.setFont(new Font("Arial", Font.BOLD, 24));
divButton.setFont(new Font("Arial", Font.BOLD, 24));
equButton.setFont(new Font("Arial", Font.BOLD, 24));
delButton.setFont(new Font("Arial", Font.BOLD, 24));
```

```
clrButton.setFont(new Font("Arial", Font.BOLD, 24));
// Add action listeners to operation buttons
addButton.addActionListener(this);
subButton.addActionListener(this);
mulButton.addActionListener(this);
divButton.addActionListener(this);
equButton.addActionListener(this);
delButton.addActionListener(this);
clrButton.addActionListener(this);
// Panel settings
panel = new JPanel();
panel.setBounds(50, 100, 300, 400);
panel.setLayout(new GridLayout(4, 4, 10, 10)); // Adjusted gaps
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(clrButton);
panel.add(numberButtons[0]);
panel.add(delButton);
```

```
panel.add(equButton);
panel.add(divButton);
frame.add(panel);
frame.setVisible(true);
}
public void actionPerformed(ActionEvent e) {
for (int i = 0; i < 10; i++) {
if (e.getSource() == numberButtons[i]) {
display.setText(display.getText().concat(String.valueOf(i)));
}
}
// Check if display is not empty before parsing
if (e.getSource() == addButton) {
if (!display.getText().isEmpty()) {
num1 = Double.parseDouble(display.getText());
operator = '+';
display.setText("");
}
}
if (e.getSource() == subButton) {
if (!display.getText().isEmpty()) {
num1 = Double.parseDouble(display.getText());
operator = '-';
display.setText("");
}
}
if (e.getSource() == mulButton) {
if (!display.getText().isEmpty()) {
```

```
num1 = Double.parseDouble(display.getText());
operator = '*';
display.setText("");
}
}
if (e.getSource() == divButton) {
if (!display.getText().isEmpty()) {
num1 = Double.parseDouble(display.getText());
operator = '/';
display.setText("");
}
}
if (e.getSource() == equButton) {
if (!display.getText().isEmpty()) {
num2 = Double.parseDouble(display.getText());
switch (operator) {
case '+':
result = num1 + num2;
break;
case '-':
result = num1 - num2;
break;
case '*':
result = num1 * num2;
break;
case '/':
if (num2 != 0) { // Prevent division by zero
result = num1 / num2;
```

```
} else {
display.setText("Error");
return;
}
break;
}
display.setText(String.valueOf(result));
num1 = result;
}
}
if (e.getSource() == delButton) {
String str = display.getText();
display.setText(str.length() > 0 ? str.substring(0, str.length() - 1) : "");
}
if (e.getSource() == clrButton) {
display.setText("");
num1 = num2 = result = 0;
}
}
public static void main(String[] args) {
new Calculator();
}
}
```

```
package quiz;
///*
//* Aim :Write a program to implement calculator using Actionlistener.
//* Name : Sayyad Mohamed Samar
//* UIN : 231P082
//* Div : A.
//*/

import javax.swing.*;
import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
public class LargestNumberCalculator extends JFrame implements ActionListener {
 private JTextField num1Field, num2Field, num3Field, resultField;
 private JButton findLargestButton;
 public LargestNumberCalculator() {
 // Set up the frame
 setTitle("Largest Number Calculator");
 setSize(300, 200);
 setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
 setLayout(new FlowLayout());
 // Create text fields
 num1Field = new JTextField(10);
 num2Field = new JTextField(10);
 num3Field = new JTextField(10);
 resultField = new JTextField(10);
resultField.setEditable(false); // Result field is not editable
 // Create button
```

```
findLargestButton = new JButton("Find Largest");
findLargestButton.addActionListener(this);
// Add components to the frame
add(new JLabel("Number 1:"));
add(num1Field);
add(new JLabel("Number 2:"));
add(num2Field);
add(new JLabel("Number 3:"));
add(num3Field);
add(findLargestButton);
add(new JLabel("Largest Number:"));
add(resultField);
}
@Override
public void actionPerformed(ActionEvent e) {
// Get numbers from text fields
try {
double num1 = Double.parseDouble(num1Field.getText());
double num2 = Double.parseDouble(num2Field.getText());
double num3 = Double.parseDouble(num3Field.getText());
// Find the largest number
double largest = Math.max(num1, Math.max(num2, num3));
resultField.setText(String.valueOf(largest));
} catch (NumberFormatException ex) {
resultField.setText("Invalid input");
}
}
public static void main(String[] args) {
```

```
// Create the GUI
SwingUtilities.invokeLater(() -> {
LargestNumberCalculator calculator = new LargestNumberCalculator();
calculator.setVisible(true);
});
}
}
```