```
import javafx.application.Application;
import javafx.geometry.Insets;
import javafx.geometry.Pos;
import javafx.scene.Scene;
import javafx.scene.control.Button;
import javafx.scene.control.TextField;
import javafx.scene.layout.GridPane;
import javafx.stage.Stage;

public class BasicCalculator extends Application {

    private TextField display;

    private double firstNumber = 0;
    private String operator = "";
    private boolean startNewNumber = true;

    @Override
    public void start(Stage primaryStage) {
        primaryStage.setTitle("Basic JavaFX Calculator");

        // Initialize display
        display = new TextField();
        display.setPrefWidth(240);
        display.setEditable(false);
        display.setAlignment(Pos.CENTER_RIGHT);

        // Create buttons
        Button btn0 = new Button("0");
        Button btn1 = new Button("1");
        Button btn2 = new Button("2");
        Button btn3 = new Button("3");
        Button btn4 = new Button("4");
        Button btn5 = new Button("5");
        Button btn6 = new Button("6");
        Button btn7 = new Button("7");
        Button btn8 = new Button("8");
        Button btn9 = new Button("9");

        Button btnAdd = new Button("+");
        Button btnSubtract = new Button("-");
        Button btnMultiply = new Button("*");
        Button btnDivide = new Button("/");
```

```
    Button btnEquals = new Button("=");
    Button btnClear = new Button("C");

    // Set button sizes
    Button[] buttons = {btn0, btn1, btn2, btn3, btn4, btn5, btn6, btn7, btn8, btn9,
            btnAdd, btnSubtract, btnMultiply, btnDivide, btnEquals, btnClear};
    for (Button btn : buttons) {
        btn.setPrefWidth(50);
    }

    // Number button actions
    btn0.setOnAction(e -> appendNumber("0"));
    btn1.setOnAction(e -> appendNumber("1"));
    btn2.setOnAction(e -> appendNumber("2"));
    btn3.setOnAction(e -> appendNumber("3"));
n(e -> appendNumber("4"));
    btn5.setOnActtn6.setOnAction(e -> appendNumber("6"));
    btn7.setOnAction(e -> appendNumber("7"));
    btn8.setOnAction(e -> appendNumber("8"));
    btn9.setOnAction(e -> appendNumber("9"));

    // Operation button actions
    btnAdd.setOnAction(e -> setOperator("+"));
    btnSubtract.setOnAction(e -> setOperator("-"));
    btnMultiply.setOnAction(e -> setOperator("*"));
    btnDivide.setOnAction(e -> setOperator("/"));

    // Equals button
    btnEquals.setOnAction(e -> calculate());

    // Clear button
    btnClear.setOnAction(e -> clear());

    // Layout setup
    GridPane grid = new GridPane();
    grid.setAlignment(Pos.CENTER);
    grid.setPadding(new Insets(15));
    grid.setVgap(10);
    grid.setHgap(10);

    // Add display
    grid.add(display, 0, 0, 4, 1);

    // Add number buttons to grid
```

```
    grid.add(btn7, 0, 1);
    grid.add(btn8, 1, 1);
    grid.add(btn9, 2, 1);
    grid.add(btnDivide, 3, 1);

    grid.add(btn4, 0, 2);
    grid.add(btn5, 1, 2);
    grid.add(btn6, 2, 2);
    grid.add(btnMultiply, 3, 2);

    grid.add(btn1, 0, 3);
    grid.add(btn2, 1, 3);
    grid.add(btn3, 2, 3);
    grid.add(btnSubtract, 3, 3);

    grid.add(btn0, 0, 4);
    grid.add(btnClear, 1, 4);
    grid.add(btnEquals, 2, 4);
    grid.add(btnAdd, 3, 4);

    // Set scene and show
    Scene scene = new Scene(grid);
    primaryStage.setScene(scene);
    primaryStage.show();
}

private void appendNumber(String num) {
    if (startNewNumber) {
        display.setText(num);
        startNewNumber = false;
    } else {
        display.setText(display.getText() + num);
    }
}

private void setOperator(String op) {
    try {
        firstNumber = Double.parseDouble(display.getText());
        operator = op;
        startNewNumber = true;
    } catch (NumberFormatException e) {
        display.setText("Error");
    }
}
```

```
    private void calculate() {
        try {
            double secondNumber = Double.parseDouble(display.getText());
            double result = 0;

            switch (operator) {
                case "+":
                    result = firstNumber + secondNumber;
                    break;
                case "-":
                    result = firstNumber - secondNumber;
                    break;
                case "*":
                    result = firstNumber * secondNumber;
                    break;
                case "/":
                    if (secondNumber == 0) {
                        display.setText("Error");
                        return;
                    } else {
                        result = firstNumber / secondNumber;
                    }
                    break;
                default:
                    return;
            }
            display.setText(String.valueOf(result));
            startNewNumber = true;
        } catch (NumberFormatException e) {
            display.setText("Error");
ion(e -> appendNumber("5"));
        b       }
    }

    private void clear() {
        display.setText("");
        firstNumber = 0;
        operator = "";
        startNewNumber = true;
    }

    public static void main(String[] args) {
        launch(args);
```

```
    }
}
```