Name: Vinayak Kumar Singh **Subject**: Java Programming Lab

Register No: 23MCA1030

1. Create a JavaFX program using Eclipse that displays a basic user interface for a simple login screen. The UI should include:

- Labels for "Username" and "Password".
- Text fields for users to input their username and password.
- A "Login" button.
- Properly aligned layout using JavaFX layouts such as VBox, HBox, or GridPane.
- When the user enters a username and password and clicks the "Login" button, display a message indicating whether the login was successful or not.

Ensure that the UI is visually appealing and user-friendly. You can experiment with different JavaFX layouts to achieve the desired look and feel.

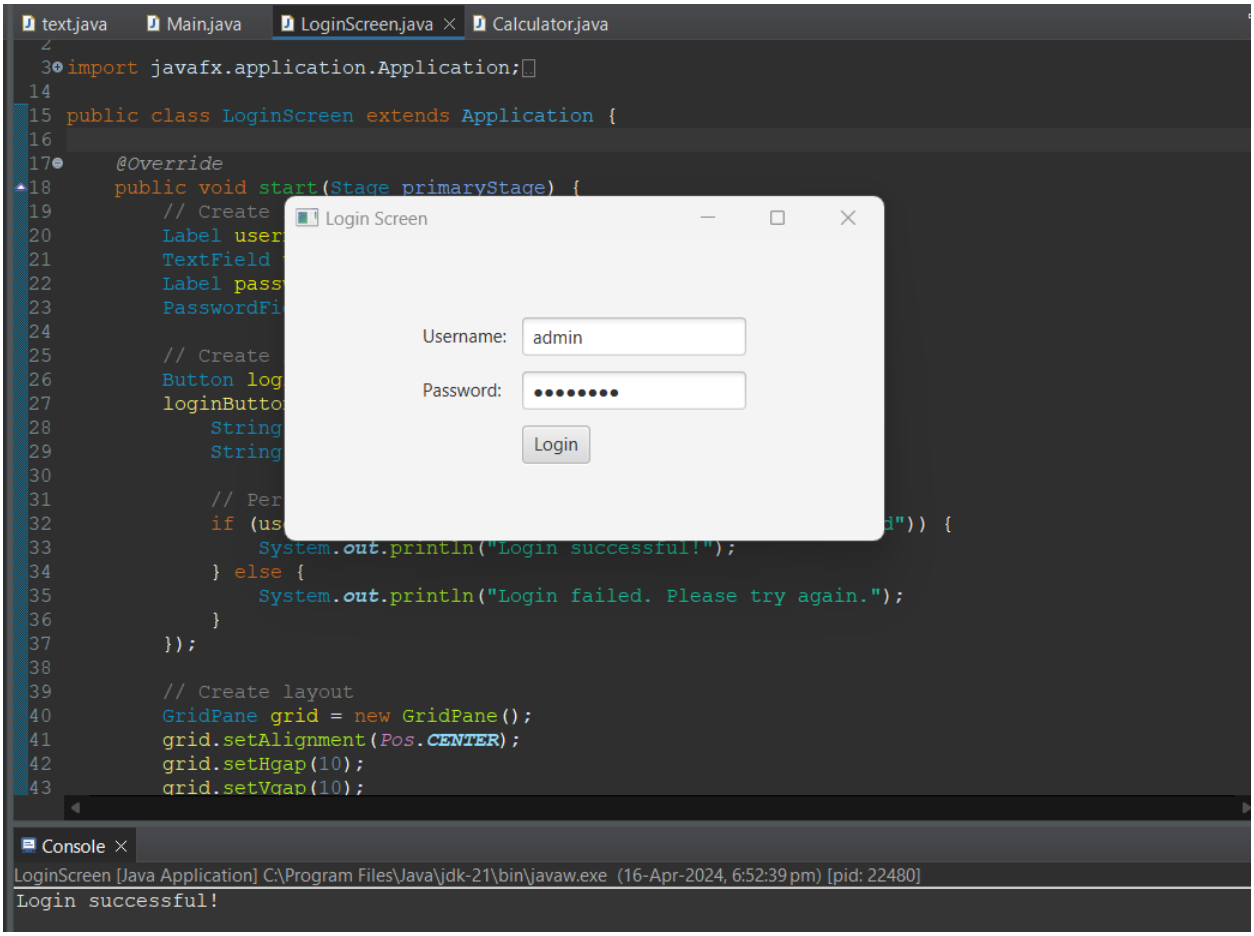
Code:

```
package application;


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.Label;

import javafx.scene.control.PasswordField;

import javafx.scene.control.TextField;

import javafx.scene.layout.GridPane;

import javafx.scene.layout.VBox;

import javafx.stage.Stage;
```

```
public class LoginScreen extends Application {
    @Override
    public void start(Stage primaryStage) {
        // Create labels and text fields
        Label usernameLabel = new Label("Username:");
        TextField usernameField = new TextField();
        Label passwordLabel = new Label("Password:");
        PasswordField passwordField = new PasswordField();

        // Create login button
        Button loginButton = new Button("Login");
        loginButton.setOnAction(event -> {
            String username = usernameField.getText();
            String password = passwordField.getText();

            // Perform login logic here
            if (username.equals("admin") && password.equals("password")) {
                System.out.println("Login successful!");
            } else {
                System.out.println("Login failed. Please try again.");
            }
        });
```

```
        // Create layout
        GridPane grid = new GridPane();
        grid.setAlignment(Pos.CENTER);
        grid.setHgap(10);
        grid.setVgap(10);
        grid.setPadding(new Insets(20, 20, 20, 20));

        grid.add(usernameLabel, 0, 0);
        grid.add(usernameField, 1, 0);
        grid.add(passwordLabel, 0, 1);
        grid.add(passwordField, 1, 1);
        grid.add(loginButton, 1, 2);
        VBox root = new VBox(grid);
        root.setAlignment(Pos.CENTER);
        // Create scene and set stage
        Scene scene = new Scene(root, 400, 200);
        primaryStage.setScene(scene);
        primaryStage.setTitle("Login Screen");
        primaryStage.show();
    }

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

Output:



2. Create a Java program using Eclipse that implements a basic calculator GUI. The GUI should include:

- Two text fields to input numbers.
- Buttons for addition, subtraction, multiplication, and division operations.
- Another text field to display the result of the operation.
- Proper labeling for each input field and button.

When the user enters two numbers and clicks on one of the operation buttons, the program should perform the corresponding arithmetic operation and display the result in the designated text field.

Additionally, handle edge cases such as division by zero and non-numeric inputs gracefully by displaying appropriate error messages in the result text field or using dialog boxes.

This exercise will help you practice designing GUI components, handling user input, implementing event listeners, performing arithmetic operations, and handling exceptions within Eclipse.

Code:

```
package application;


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.Label;

import javafx.scene.control.TextField;

import javafx.scene.layout.GridPane;

import javafx.stage.Stage;


public class Calculator extends Application {

    private TextField num1Field, num2Field, resultField;


    @Override

    public void start(Stage primaryStage) {

        // Create input fields

        Label num1Label = new Label("Number 1:");

        num1Field = new TextField();
```

```
Label num2Label = new Label("Number 2:");
num2Field = new TextField();

// Create operation buttons
Button addButton = new Button("+");
Button subtractButton = new Button("-");
Button multiplyButton = new Button("*");
Button divideButton = new Button("/");

// Create result field
Label resultLabel = new Label("Result:");
resultField = new TextField();
resultField.setEditable(false);

// Set event handlers for buttons
addButton.setOnAction(event -> performCalculation('+'));
subtractButton.setOnAction(event -> performCalculation('-'));
multiplyButton.setOnAction(event -> performCalculation('*'));
divideButton.setOnAction(event -> performCalculation('/'));

// Create layout
GridPane grid = new GridPane();
grid.setAlignment(Pos.CENTER);
grid.setHgap(10);
grid.setVgap(10);
```

```
    grid.setPadding(new Insets(20, 20, 20, 20));

    grid.add(num1Label, 0, 0);
    grid.add(num1Field, 1, 0);
    grid.add(num2Label, 0, 1);
    grid.add(num2Field, 1, 1);
    grid.add(addButton, 0, 2);
    grid.add(subtractButton, 1, 2);
    grid.add(multiplyButton, 0, 3);
    grid.add(divideButton, 1, 3);
    grid.add(resultLabel, 0, 4);
    grid.add(resultField, 1, 4);

    // Create scene and set stage
    Scene scene = new Scene(grid, 400, 300);
    primaryStage.setScene(scene);
    primaryStage.setTitle("Calculator");
    primaryStage.show();
}

private void performCalculation(char operation) {
    try {
        double num1 = Double.parseDouble(num1Field.getText());
        double num2 = Double.parseDouble(num2Field.getText());
        double result;
```

```
    switch (operation) {
        case '+':
            result = num1 + num2;
            break;
        case '-':
            result = num1 - num2;
            break;
        case '*':
            result = num1 * num2;
            break;
        case '/':
            if (num2 == 0) {
                resultField.setText("Error: Division by zero");
                return;
            }
            result = num1 / num2;
            break;
        default:
            resultField.setText("Error: Invalid operation");
            return;
    }

    resultField.setText(Double.toString(result));
} catch (NumberFormatException e) {
```

```
            resultField.setText("Error: Invalid input");
        }
    }

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

Output: