```java
import javax.swing.*;

import java.awt.*;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;


public class FinanceManagementWithGUI extends JFrame implements ActionListener {

    private JTextField incomeField, expensesField;

    private JTextArea resultArea;


    public FinanceManagementWithGUI() {

        // Set the title and default close operation for the GUI window

        setTitle("Finance Management");

        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);


        // Set the initial size and layout of the window

        setSize(400, 300);

        setLayout(new BorderLayout());


        // Create an input panel with labels, text fields, and a button

        JPanel inputPanel = new JPanel();

        inputPanel.setLayout(new GridLayout(3, 2));


        // Labels for income and expenses

        JLabel incomeLabel = new JLabel("Monthly Income: ₹");

        incomeField = new JTextField(10);

        JLabel expensesLabel = new JLabel("Monthly Expenses: ₹");

        expensesField = new JTextField(10);


        // Button to trigger the calculation
```

```java
        JButton calculateButton = new JButton("Calculate");

        // Add components to the input panel
        inputPanel.add(incomeLabel);
        inputPanel.add(incomeField);
        inputPanel.add(expensesLabel);
        inputPanel.add(expensesField);
        inputPanel.add(calculateButton);

        // Add the input panel to the top of the window
        add(inputPanel, BorderLayout.NORTH);

        // Create a text area for displaying the calculation result
        resultArea = new JTextArea();
        resultArea.setEditable(false);

        // Add a scroll pane for the text area and add it to the center of the window
        add(new JScrollPane(resultArea), BorderLayout.CENTER);

        // Register the action listener for the calculate button
        calculateButton.addActionListener(this);

        // Make the window visible
        setVisible(true);
    }

    @Override
    public void actionPerformed(ActionEvent e) {
        double income, expenses;
```

```java
try {
    // Parse the values entered in the income and expenses fields
    income = Double.parseDouble(incomeField.getText());
    expenses = Double.parseDouble(expensesField.getText());
} catch (NumberFormatException ex) {
    // Display an error message if the input is not a valid number
    resultArea.setText("Please enter valid income and expenses.");
    return;
}

double savings = income - expenses;
String message = "Your monthly savings are ₹" + savings + "\n";

if (savings > 0) {
    // Ask the user if they want to invest in a Fixed Deposit (FD)
    String investmentChoice = JOptionPane.showInputDialog("Do you want to invest in Fixed Deposit (FD)? (yes/no)").toLowerCase();
    if (investmentChoice.equals("yes")) {
        // If yes, gather information about the FD and calculate the maturity amount
        double principal = Double.parseDouble(JOptionPane.showInputDialog("Enter the principal amount for FD: ₹"));
        double rate = 5.0; // Fixed FD interest rate of 5%
        int tenure = Integer.parseInt(JOptionPane.showInputDialog("Enter the FD tenure (in months): "));
        double maturityValue = (principal * rate * tenure) / 100;
        message += "FD Maturity Amount: ₹" + maturityValue + "\n";
        double totalAmount = savings + maturityValue;
        message += "Total Amount in the Bank after FD maturity: ₹" + totalAmount + "\n";
    } else {
        message += "No FD investment made.\n";
```

```java
        }
    } else if (savings < 0) {

        // If savings are negative, offer the option to take a loan

        int readyForLoan = JOptionPane.showConfirmDialog(null, "Are you ready to take a loan?",
"Confirm Loan", JOptionPane.YES_NO_OPTION);

        if (readyForLoan == JOptionPane.YES_OPTION) {

            int confirm = JOptionPane.showConfirmDialog(null, "Do you accept 5% interest for the loan?",
"Confirm Interest Rate", JOptionPane.YES_NO_OPTION);

            if (confirm == JOptionPane.YES_OPTION) {

                // If accepted, gather loan information and calculate the total repayment

                double loanPrincipal = Double.parseDouble(JOptionPane.showInputDialog("Enter the loan
principal amount: ₹"));

                int loanTenure = Integer.parseInt(JOptionPane.showInputDialog("Enter the loan tenure (in
months): "));

                double loanAmount = (loanPrincipal * 5 * loanTenure) / 100;

                double totalLoanRepayment = loanPrincipal + loanAmount;

                message += "Loan Details:\n";

                message += "Loan Amount: ₹" + loanPrincipal + "\n";

                message += "Total Repayment: ₹" + totalLoanRepayment + "\n";

            } else {

                message += "No loan taken due to interest rate rejection.\n";

            }

        } else {

            message += "No loan taken.\n";

        }

    }


    // Display the calculation result in the text area

    resultArea.setText(message);

}
```

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

    // Create and display the FinanceManagementWithGUI window

    SwingUtilities.invokeLater(() -> new FinanceManagementWithGUI());

}
}
```