ATM SIMULATION SYSTEM

```
import java.awt.*;
import java.awt.event.*;

public class ATMExampleAWT extends Frame implements ActionListener {
    // Initialize account details
    private int accountNumber = 123456; // example account number
    private int pin = 1234; // example PIN
    private int balance = 100000; // Initialize balance

    // UI components
    Label label1, label2, label3, messageLabel;
    TextField accountField, pinField, inputField;
    Button loginButton, withdrawButton, depositButton, checkBalanceButton,
exitButton, confirmButton;
    Panel loginPanel, menuPanel, actionPanel;

    private String currentAction = ""; // To track if we are withdrawing or depositing

    public ATMExampleAWT() {
        // Set up the frame
        setTitle("ATM Example");
        setSize(400, 300);
        setLayout(new CardLayout());

        // Login panel
        loginPanel = new Panel(new GridLayout(4, 2));
        label1 = new Label("Account Number:");
        label2 = new Label("PIN:");
        accountField = new TextField();
        pinField = new TextField();
        pinField.setEchoChar('*');
        loginButton = new Button("Login");
        loginButton.addActionListener(this);
        messageLabel = new Label("", Label.CENTER);

        loginPanel.add(label1);
        loginPanel.add(accountField);
```

```
loginPanel.add(label2);
loginPanel.add(pinField);
loginPanel.add(new Label(""));
loginPanel.add(loginButton);
loginPanel.add(messageLabel);

// Menu panel
menuPanel = new Panel(new GridLayout(5, 1));
withdrawButton = new Button("Withdraw");
depositButton = new Button("Deposit");
checkBalanceButton = new Button("Check Balance");
exitButton = new Button("Exit");

withdrawButton.addActionListener(this);
depositButton.addActionListener(this);
checkBalanceButton.addActionListener(this);
exitButton.addActionListener(this);

menuPanel.add(withdrawButton);
menuPanel.add(depositButton);
menuPanel.add(checkBalanceButton);
menuPanel.add(exitButton);

// Action panel for deposit/withdraw inputs
actionPanel = new Panel(new FlowLayout());
label3 = new Label("Enter Amount:");
inputField = new TextField(10);
confirmButton = new Button("Confirm");
confirmButton.addActionListener(this);

actionPanel.add(label3);
actionPanel.add(inputField);
actionPanel.add(confirmButton);
actionPanel.setVisible(false);

// Add panels to the frame
add(loginPanel);
```

```
    add(menuPanel);
    add(actionPanel);

    menuPanel.setVisible(false);
    setVisible(true);

    // Handle window closing
    addWindowListener(new WindowAdapter() {
        public void windowClosing(WindowEvent e) {
            dispose();
        }
    });
}

@Override
public void actionPerformed(ActionEvent e) {
    String action = e.getActionCommand();

    if (action.equals("Login")) {
        int enteredAccountNumber = Integer.parseInt(accountField.getText());
        int enteredPin = Integer.parseInt(pinField.getText());

        if (enteredAccountNumber == accountNumber && enteredPin == pin) {
            loginPanel.setVisible(false);
            menuPanel.setVisible(true);
        } else {
            messageLabel.setText("Invalid Account Number or PIN");
        }
    } else if (action.equals("Withdraw")) {
        currentAction = "Withdraw";
        actionPanel.setVisible(true);
        menuPanel.setVisible(false);
        label3.setText("Enter Amount to Withdraw:");
        inputField.setText("");
    } else if (action.equals("Deposit")) {
        currentAction = "Deposit";
        actionPanel.setVisible(true);
```

```
            menuPanel.setVisible(false);
            label3.setText("Enter Amount to Deposit:");
            inputField.setText("");
        } else if (action.equals("Check Balance")) {
            showMessage("Balance: " + balance);
        } else if (action.equals("Exit")) {
            System.exit(0);
        } else if (action.equals("Confirm")) {
            int amount = Integer.parseInt(inputField.getText());

            if (currentAction.equals("Withdraw")) {
                if (balance >= amount) {
                    balance -= amount;
                    showMessage("Withdrawal successful. Collect your cash.");
                } else {
                    showMessage("Insufficient Balance");
                }
            } else if (currentAction.equals("Deposit")) {
                balance += amount;
                showMessage("Deposit successful.");
            }

            // Reset to menu
            actionPanel.setVisible(false);
            menuPanel.setVisible(true);
        }
    }

    private void showMessage(String message) {
        Dialog dialog = new Dialog(this, "Message", true);
        dialog.setLayout(new FlowLayout());
        Label messageLabel = new Label(message);
        Button okButton = new Button("OK");
        okButton.addActionListener(e -> dialog.setVisible(false));
        dialog.add(messageLabel);
        dialog.add(okButton);
        dialog.setSize(300, 100);
```

```
        dialog.setVisible(true);
    }

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