1. The Menu Class

| Menu |
| --- |
| static Scanner JObject  static String input  static String NewUserName, NewPassword  final static String HEADING1  static boolean exitTime  static final String DB\_URL  static final String USER  static final String PASS |
| Public static void main(String[] argos)  Public static login  Public static void signUp |

A1 The Main Method

START

Declare integer userOption

Declare boolean exitTime = false

While exitTime is false:

Display menu options:

"1. Login"

"2. Sign Up"

"3. Quit"

Get user input and store it in input

Convert input to integer and store it in userOption

Switch on userOption:

Case 1:

Call login() function

Break

Case 2:

Call signUp() function

Break

Case 3:

Set exitTime to true

Break

End Switch

End While

STOP

A2 The Login Method

START

Declare String Username, Password

Prompt user to enter Username and store it in Username

Prompt user to enter Password and store it in Password

Try:

Create a database connection with DriverManager.getConnection(DB\_URL, USER, PASS)

Create a statement with conn.createStatement()

Execute a query to select from the sdb\_user table where Usr\_Name is Username and Usr\_Pass is Password

Store the result in resultSet

If resultSet has next row:

Display "Login Successful!" message

Set exitTime to true

Create an instance of LoggedInScreen and call its logIn() method

Else:

Display "USERNAME OR PASSWORD INCORRECT" message

STOP

A3 The SignUp Method

START

Try:

Create a database connection with DriverManager.getConnection(DB\_URL, USER, PASS)

Get database metadata with conn.getMetaData()

Get the list of tables/views in the "secdb" database

Store the result in resultSet

Initialize StringBuilder tables

While resultSet has next row:

Append the table name to tables

Close resultSet

Display a dialog to select a table from the list:

Title: "Add Data to Table"

Message: "Select a table to add data to:"

Options: List of tables from tables

Store the selected table name in selectedTable

If selectedTable is not null and not empty:

Get column names by executing a SELECT \* FROM selectedTable and retrieving metadata

Initialize StringBuilder columnNames

Loop through column names:

Append each column name to columnNames

Prompt the user to input data for each column:

Loop through column count:

Display input dialog to enter value for each column

Initialize String array inputData with size equal to column count

Loop through column count:

Store user input values in inputData array

Initialize StringBuilder insertStatement to construct SQL INSERT statement:

Append "INSERT INTO selectedTable (" to insertStatement

Loop through column count:

Append column name to insertStatement

If not the last column, append ", "

Append ") VALUES ("

Loop through inputData:

Append each value surrounded by single quotes to insertStatement

If not the last value, append ", "

Append ")"

Execute the INSERT statement with stmt.executeUpdate(insertStatement)

Display "Data added successfully to selectedTable" message

Else:

Display "No table selected." message

Catch SQLException and print the stack trace if an exception occurs

STOP

1. LoggedInScreen Class

| LoggedInScreen |
| --- |
| Static Scanner JObject  Static String input  Final static String HEADING1  Static boolean exitTime |
| Public void logIn |

B1 The logIn method

START

Declare integer userOption

While exitTime is false:

Display menu options:

"1. Display"

"2. Create"

"3. Modify"

"4. Remove"

"5. Quit"

Get user input and store it in input

Convert input to integer and store it in userOption

Switch on userOption:

Case 1:

Create an instance of ViewsecDB and call its View() method

Break

Case 2:

Create an instance of CreatesecDB and call its Create() method

Break

Case 3:

Create an instance of ModifysecDB2 and call its Modify() method

Break

Case 4:

Create an instance of DeletesecDB and call its Delete() method

Break

Case 5:

Set exitTime to true

Break

End Switch

End While

STOP

1. DeletesecDB Class

| DeletesecDB |
| --- |
| Static final String DB\_URL  Static final String USER  Static final String PASS |
| Public void Delete |

C1 The Delete method

START

Try:

Create a database connection with DriverManager.getConnection(DB\_URL, USER, PASS)

Get database metadata with conn.getMetaData()

Get the list of tables/views in the "secdb" database

Store the result in resultSet

Initialize StringBuilder tables

While resultSet has next row:

Append the table name to tables

Close resultSet

Display a dialog to select a table from the list:

Title: "Delete Data from Table"

Message: "Select a table to delete data from:"

Options: List of tables from tables

Store the selected table name in selectedTable

If selectedTable is not null and not empty:

Get column names by executing a SELECT \* FROM selectedTable and retrieving metadata

Initialize StringBuilder columnNames

Loop through column names:

Append each column name to columnNames

Prompt the user to input the primary key value for deletion:

Display input dialog to enter the primary key value

Construct the SQL DELETE statement:

Formulate "DELETE FROM selectedTable WHERE firstColumnName = 'primaryKeyValue'"

Execute the DELETE statement with stmt.executeUpdate(deleteStatement)

Get the number of rows affected by the deletion

If rowsAffected is greater than 0:

Display "Row deleted successfully from selectedTable" message

Else:

Display "No row with primary key 'primaryKeyValue' found in selectedTable" message

Else:

Display "No table selected." message

Catch SQLException and print the stack trace if an exception occurs

STOP

1. ModifysecDB2 Class

| ModifysecDB2 |
| --- |
| Static final String DB\_URL  Static final String USER  Static final String PASS |
| Public void Modify |

D1 Modify method

START

Try:

Create a database connection with DriverManager.getConnection(DB\_URL, USER, PASS)

Get database metadata with conn.getMetaData()

Get the list of tables/views in the "secdb" database

Store the result in resultSet

Initialize StringBuilder tables

While resultSet has next row:

Append the table name to tables

Close resultSet

Display a dialog to select a table from the list:

Title: "Modify Data in Table"

Message: "Select a table to modify data in:"

Options: List of tables from tables

Store the selected table name in selectedTable

If selectedTable is not null and not empty:

Get column names by executing a SELECT \* FROM selectedTable and retrieving metadata

Initialize StringBuilder columnNames

Loop through column names:

Append each column name to columnNames

Prompt the user to input the primary key value for modification:

Display input dialog to enter the primary key value

Initialize StringBuilder newValues

Loop through column count:

Prompt the user to input the new value for each column:

Display input dialog to enter new value for the column

Append column name and its new value to newValues

If not the last column, append ", "

Construct the SQL UPDATE statement:

Formulate "UPDATE selectedTable SET column1 = 'newValue1', column2 = 'newValue2', ... WHERE primaryKeyColumn = 'primaryKeyValue'"

Execute the UPDATE statement with stmt.executeUpdate(updateStatement)

Get the number of rows affected by the update

If rowsAffected is greater than 0:

Display "Row updated successfully in selectedTable" message

Else:

Display "No row with primary key 'primaryKeyValue' found in selectedTable" message

Else:

Display "No table selected." message

Catch SQLException and print the stack trace if an exception occurs  
STOP

1. CreatesecDB Class

| CreatesecDB |
| --- |
| Static final String DB\_URL  Static final String USER  Static final String PASS |
| Create |

E1 The Create Method

START

Try:

Create a database connection with DriverManager.getConnection(DB\_URL, USER, PASS)

Get database metadata with conn.getMetaData()

Get the list of tables/views in the "secdb" database

Store the result in resultSet

Initialize StringBuilder tables

While resultSet has next row:

Append the table name to tables

Close resultSet

Display a dialog to select a table from the list:

Title: "Add Data to Table"

Message: "Select a table to add data to:"

Options: List of tables from tables

Store the selected table name in selectedTable

If selectedTable is not null and not empty:

Get column names by executing a SELECT \* FROM selectedTable and retrieving metadata

Initialize StringBuilder columnNames

Loop through column names:

Append each column name to columnNames

Initialize String array inputData with size equal to column count

Loop through column count:

Prompt the user to input data for each column:

Display input dialog to enter value for each column

Store user input values in inputData array

Construct the SQL INSERT statement:

Formulate "INSERT INTO selectedTable (column1, column2, ...) VALUES ('value1', 'value2', ...)"

Execute the INSERT statement with stmt.executeUpdate(insertStatement)

Display "Data added successfully to selectedTable" message

Else:

Display "No table selected." message

Catch SQLException and print the stack trace if an exception occurs  
STOP

1. ViewsecDB Class

| ViewsecDB |
| --- |
| Static final String DB\_URL  Static final String USER  Static final String PASS |
| View |

F1 The View Method

START

Try:

Create a database connection with DriverManager.getConnection(DB\_URL, USER, PASS)

Get database metadata with conn.getMetaData()

Get the list of tables/views in the "secdb" database

Store the result in resultSet

Initialize StringBuilder tables

While resultSet has next row:

Append the table/view name to tables

Close resultSet

Display a dialog to select a table/view from the list:

Title: "Table/View Viewer"

Message: "Select a table/view to view:"

Options: List of tables/views from tables

Store the selected table/view name in selectedTable

If selectedTable is not null and not empty:

Get column names by executing a SELECT \* FROM selectedTable and retrieving metadata

Initialize StringBuilder columnNames

Loop through column names:

Append each column name to columnNames

Execute a SELECT \* FROM selectedTable query to retrieve table data

Initialize StringBuilder tableInfo with column names

While tableData has next row:

Loop through column count:

Retrieve column value for each column

Calculate padding to align data

Append column value with padding to tableInfo

Append new line to tableInfo

Close tableData

Display a message dialog showing selected table/view name and formatted table data

Else:

Display "No table/view selected." message

Catch SQLException and print the stack trace if an exception occurs

STOP