database application

Technical Manual

Company Name

Elmer Jr Jubelag

Table of Contents

[Application Usage: Testing Connection 2](#_Toc43195626)

[Application Usage: Create Users Table 3](#_Toc43195627)

[Application Usage: Drop Users Table 4](#_Toc43195628)

[Application Usage: Adding Users 5](#_Toc43195629)

[Application Usage: Updating Users 7](#_Toc43195630)

[Application Usage: Delete Users 9](#_Toc43195631)

[Application Usage: Searching Users 10](#_Toc43195632)

[Application Usage: Adding Roles to Users 11](#_Toc43195633)

[Application Usage: Display current users 13](#_Toc43195634)

[Application Usage: Display User Login 14](#_Toc43195635)

[Application code overview 15](#_Toc43195636)

[Subroutines 15](#_Toc43195637)

[Main Menu 15](#_Toc43195638)

[Manage User 16](#_Toc43195639)

[Manage User Roles 17](#_Toc43195640)

[Display User 17](#_Toc43195641)

[Display User Login 17](#_Toc43195642)

[User Stored Procedures 18](#_Toc43195643)

[Application test 1: Creating Users with Roles 19](#_Toc43195644)

[Application test 2: Updating Error 21](#_Toc43195645)

[Application test 3: Deleting Error 22](#_Toc43195646)

[Application test 4: Searching Error 23](#_Toc43195647)

[Application test 4: Insertion Error 24](#_Toc43195648)

[Application test 5: Maintain Table Errors 25](#_Toc43195649)

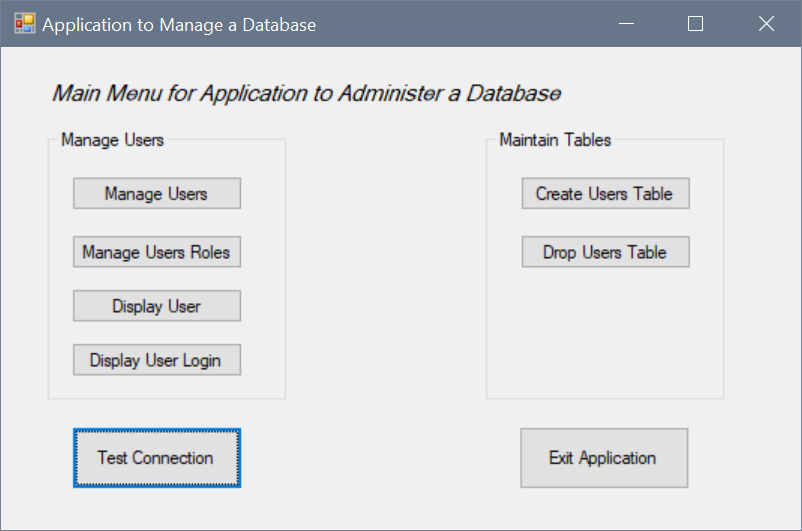
[Application test 6: DisplayUser Errors 25](#_Toc43195650)

[Application test 7: DisplayUserRoles Errors 26](#_Toc43195651)

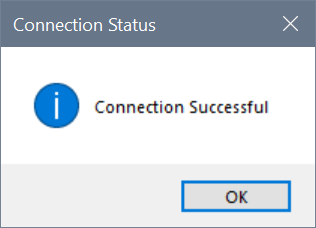
# Application Usage: Testing Connection

This workflow purpose is to test the connection with the database, enabling further use of the functions of this application.

1. Left Click on Test Connection Button



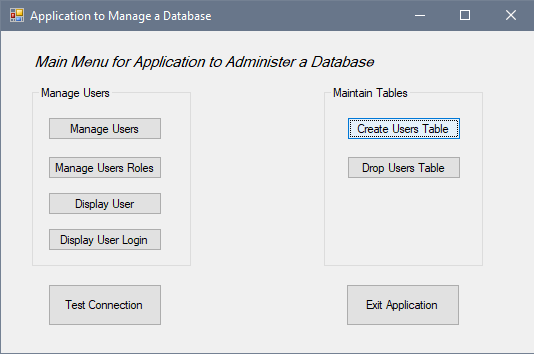
1. Prompt will show “Connection Successful” which indicates a successful connection with the database



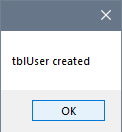
# Application Usage: Create Users Table

This workflow purpose is to create a table called “tblUser” which is needed for further functionality of this application.

1. Click on “Create Users Table” button



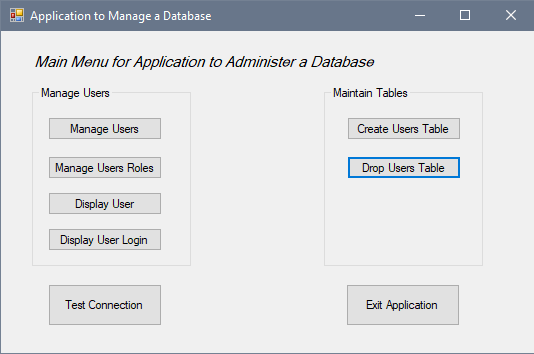
1. Prompt will show “tblUser created” which indicates creation of a table named tblUser in the database



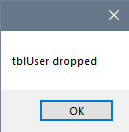
# Application Usage: Drop Users Table

This workflow purpose is to drop a table called “tblUser” and related tables and entries.

1. Click on “Drop User Table” button



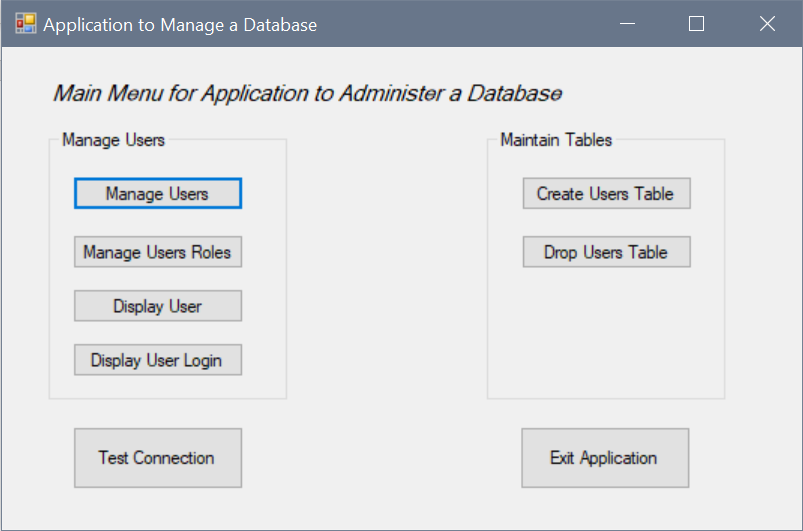
1. Prompt will show “tblUser dropped” which indicates tblUser has been dropped in the database



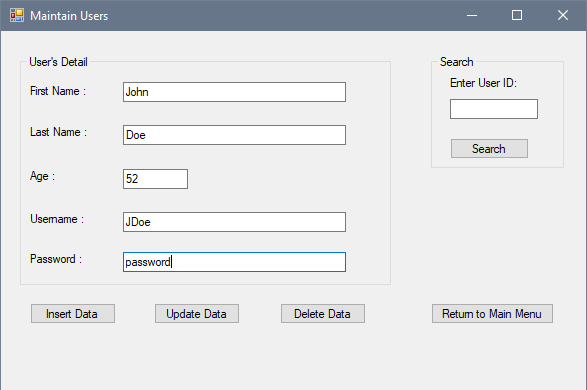
# Application Usage: Adding Users

This workflow purpose is the Add users to the database which adds entries into tblUser and adds Users into the databases authentication system.

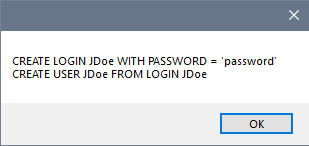
1. Left Click on “Manage Users” button



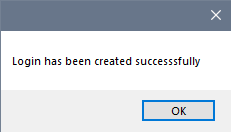
1. Fill the entries in the appropriate format, filling the textbox under the label “Enter User ID” is not needed as the application automatically assigns a User ID for each new entry.



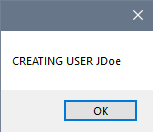
1. Prompt will show the username and password assigned which inform the user that the procedure in the database has ran.



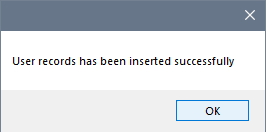
1. Prompt will show “Login has been created successfully” which indicates user has been inserted into the database’s authentication system



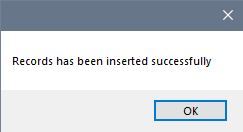
1. Prompt will show the username used that for the database and insert it into tblUser.



1. Prompt shows “User has been created successfully” which indicates that user has been inserted into the database



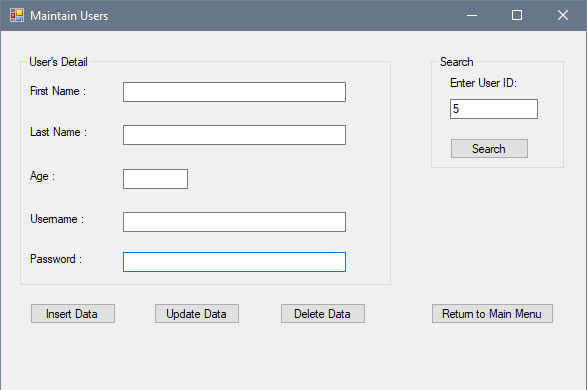
1. Prompt will show the “User records has been inserted successfully” which indicates procedures has ran successfully.



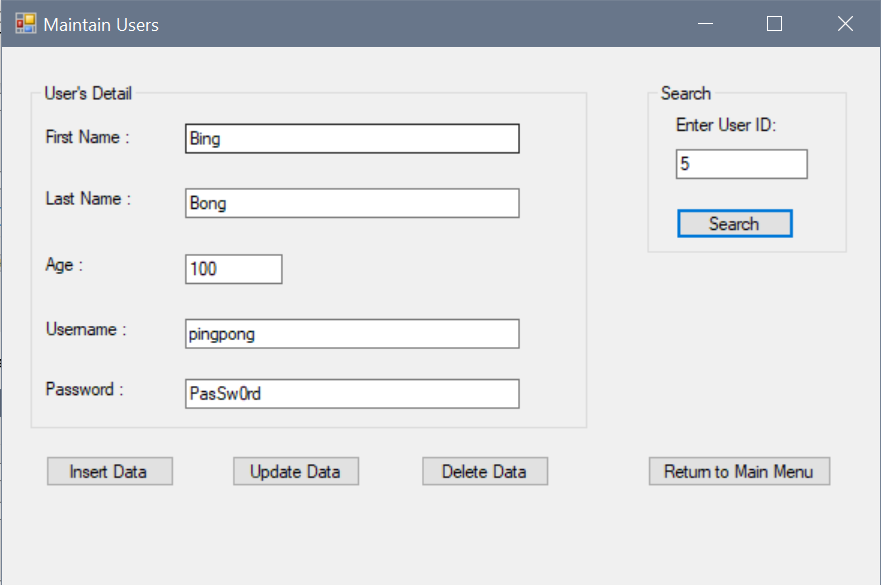
# Application Usage: Updating Users

This workflow purpose is to update user’s information in the database which updates entries from tblUser and Users in the master databases authentication system.

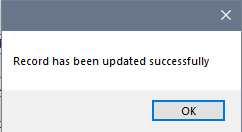
1. Enter the ID of the user information needs to be updated and left click “Search” button to fill



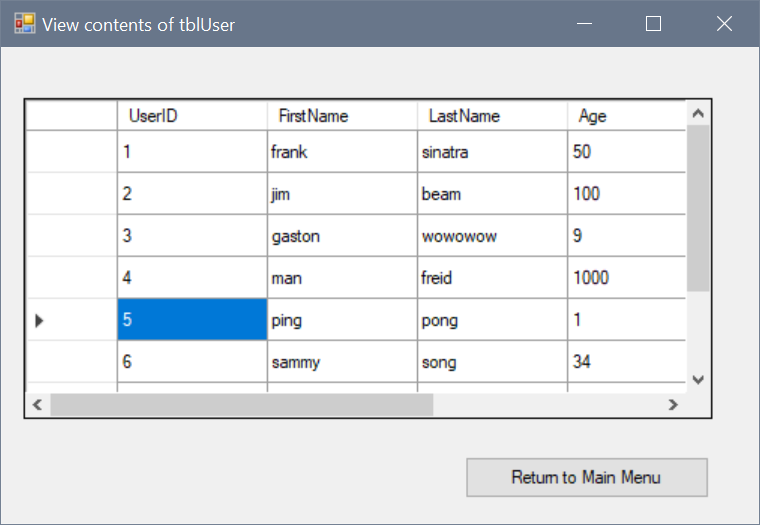
1. Enter the updated information to the appropriate fields and left click “Update Data”



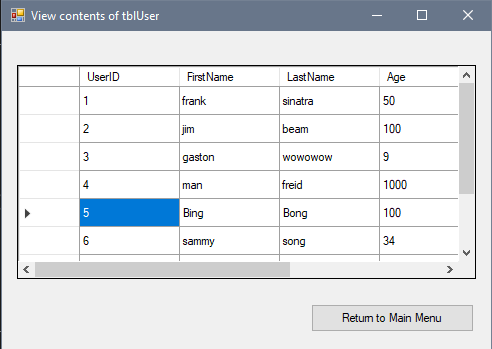
1. Prompt shows “Record has been updated successfully” indicate user records has been updated



1. Before update



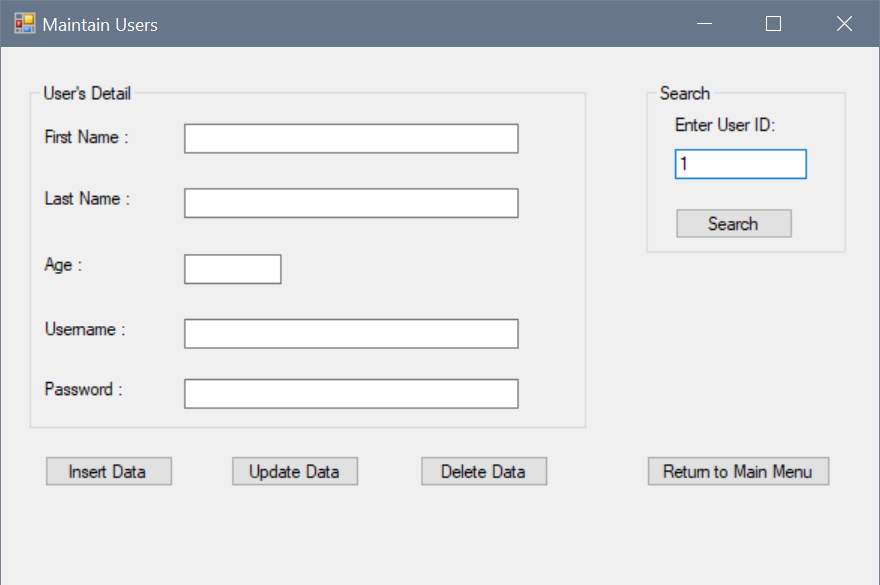
1. After update



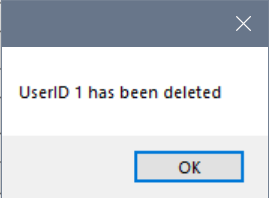
# Application Usage: Delete Users

This workflow purpose is to delete user’s information in the database which deletes entries from tblUser but not Users in the master databases authentication system. (Please contact the database administrator to delete User from master database)

1. Enter the ID of the user information needs to be deleted



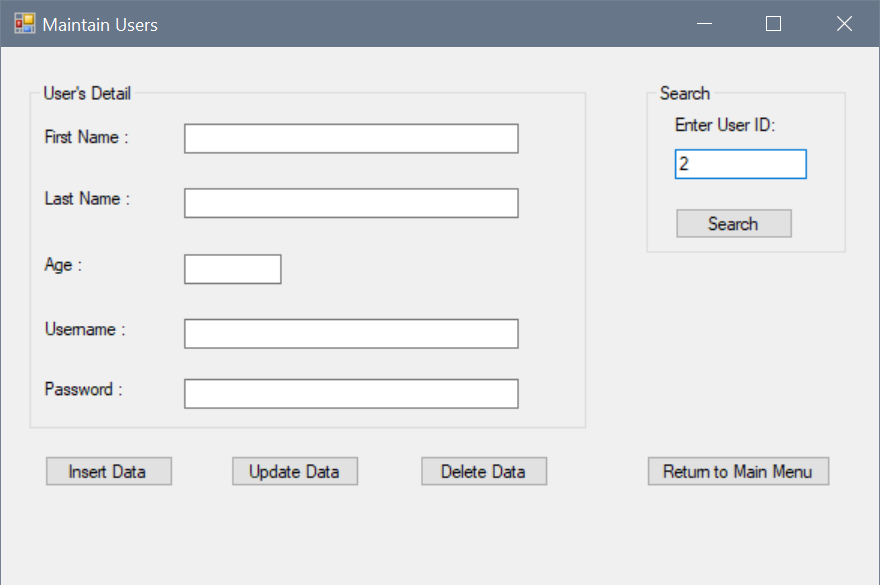
1. Prompt show “UserID X has been deleted” indicate User has been deleted from tblUser



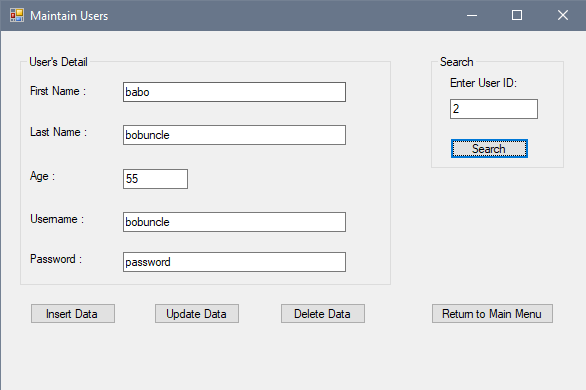
# Application Usage: Searching Users

This workflow purpose is for searching user’s information from the database which fills the appropriate text fields from tblUser

1. Enter the ID of the user information needs to be searched and text field to be filled.



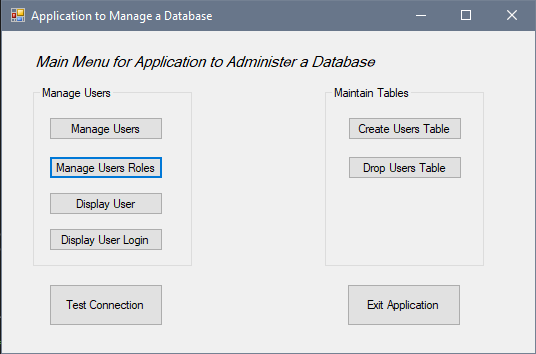
1. Form shows all text field is filled with information from the databases and tblUser



# Application Usage: Adding Roles to Users

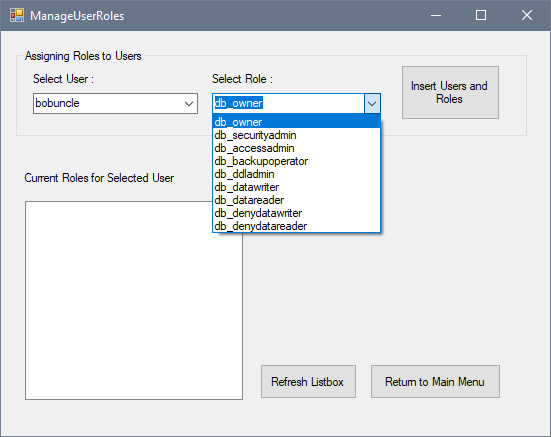
This workflow purpose is to add roles to users in the database which would insert into tblUserRoles and Users role to the master databases authentication system.

1. Left click on “Manage User Roles” button

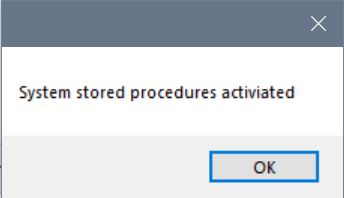


1. Select the desired user in the combo box under the “Select User :” label and Select the desired role in the combo box under the “Select Role :” label

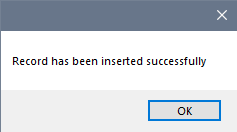
Left click on “Insert Users and Roles” if both combo box is filled.



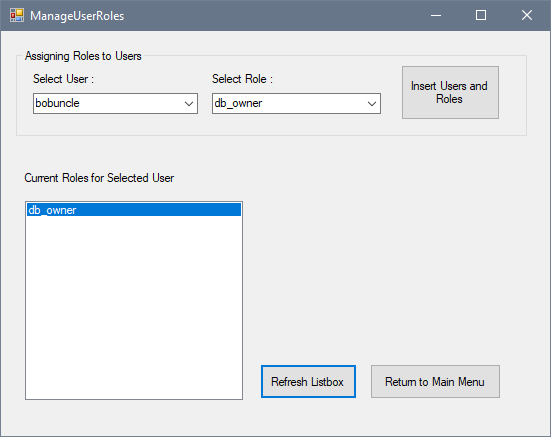
1. Prompt shows “System stored procedures activated” indicates the master database procedures “sp\_addrolemember” ran successfully.



1. Prompt shows “Record has been inserted successfully” indicates the database has recorded the entry into “tblUserRoles”



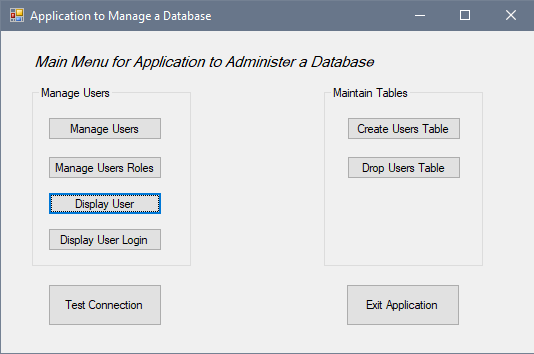
1. List Box should show the inserted role for the selected user.



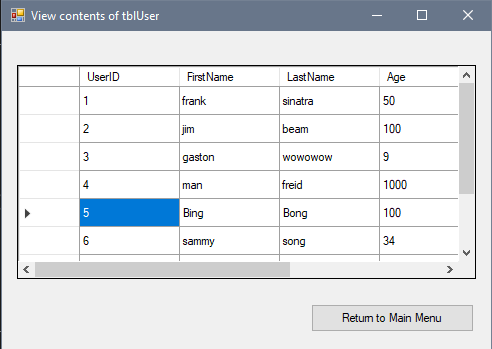
# Application Usage: Display current users

This workflow purpose is to view current users in the database with the related data fields.

1. Left click “Display User” button.



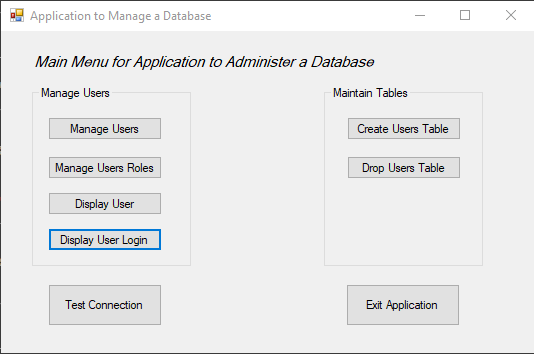
1. Form should the current users in the database and entries in tblUser



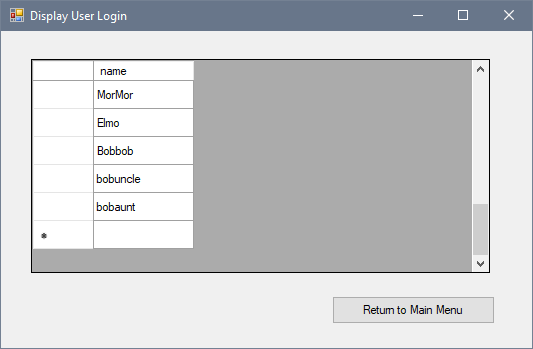
# Application Usage: Display User Login

This workflow purpose is to view User Logins in the master database showing current user’s names.

1. Left click “Display User Login” button.



1. Form should show the current users in the master database



# Application code overview



## Subroutines

### Main Menu

**CreateUserTable()**

Calls a class object from System.Data.SqlClient to be used on a subroutine within class object to a string (“Connection1”) as a ConnectionString, passing a SqlClient.Commandtext which calls a procedures “uspCreateUserTable” from a the connecting database through SqlClient. Creating tblUser with the procedures.

**DropUserTable()**

Calls a class object from System.Data.SqlClient and used on a subroutine within class object to a string (“Connection1”) as a ConnectionString, passing a SqlClient.Commandtext which calls a procedures “uspDropUserTable” from a the connecting database through SqlClient. Dropping tblUser and tblUserRole with the procedures.

**TestConBtn\_Click()**

Call a class object from System.Data.SqlClient to open a connection from the database to the application. Showing a message box if connection was established or failed.

### Manage User

**LoadData()**

A query contained in a string “SELECT \* FROM tblUsers” was input into SqlClient class object is then connected to the database which returned the result of the query into the form’s text boxes.

**AddLogin()**

A query with username and password contained in a string was input into SqlClient class object which has a connection to the master database allowing the creation of a login in the master database. A message box confirms to the users their chosen username and password.

**AddUser()**

A message box prompts the user the chosen username and passes a query contained in a string, it is then passed into SqlClient class object which create a user in the database from the login in the master database. Prompting the users of the success or failure of the subroutines.

**InsertNewRecord()**

Text in the form text field is parsed to the SqlClient object(sqlComm.Connection), opening a connection with the database and placing all parse values as input for the called user stored procedure (uspInsertUserData) which would store all parsed information into “tblUsers”

**UpdateRecord()**

Calls a user stored procedures (uspUpdateUserData) and passes parsed text from the text field. A prompt will show if the subroutine is successfully or if it failed.

**DeleteRecord()**

Calls a user stored procedures (uspDeleteUserData) and passes parse integer into it, deleting the records of UserID that is equal to txtSearch.Text input number.

**btnSearch\_Click()**

A query contained in a string “SELECT \* FROM tblUsers WHERE UserID = Integer.Parse(UserID)” was input into SqlClient class object is then connected to the database which returned the result of the query into the form’s text boxes if the UserID is equal the form’s text box.

### Manage User Roles

**DisplayUser()**

A query contained in a string “SELECT \* FROM tblUsers” was input into SqlClient class object is then connected to the database which returned the result of the query into the form’s user combo box which accepts entries with columns “Username” and arranged by their “UserID”

**DisplayRoles()**

A query contained in a string “SELECT \* FROM tblUsers” was input into SqlClient class object is then connected to the database which returned the result of the query into the form’s role combo box which accepts entries with columns “RoleDesc” and arranged by their “UserID”

**DisplayListBox()**

Calls a user stored procedures (uspselectRoles) and passes the UserID from the combo box selection and returns the roles that is related to the selected user.

**InsertNewUserRole()**

A query contained in a string “INSERT INTO tblUserRoles(RoleID, UserID) VALUES ('" & role & "','" & user & "')” is input into SqlClient class object, prompts then inform user of the success of failure to assign a role the selected user.

**addUserRole()**

Calls a stored procedures (sp\_addrolemember) and passes membername as Username and rolename as roledesc from the combo box selection

### Display User

A query contained in a string “SELECT \* FROM tblUsers” was input into SqlClient class object is then connected to the database which returned the result of the query into a SqlDataApdater which fill a DataTable, filling up the gridview.

### Display User Login

A query contained in a string “SELECT \* FROM master.sys.sql\_login” was input into SqlClient class object is then connected to the database which returned the result of the query into a SqlDataApdater which fill a DataTable, filling up the gridview.

## User Stored Procedures

**uspCreateUserTable**

This Procedure runs 2 “CREATE TABLE ” SQL commands where it adds tblUsers and tblUserRoles to a database, it contains the all the required columns and data type for the insert, update and deletion from the application

**uspDropUserTable**

This Procedure runs 2 “DROP TABLE” SQL commands where it drops tblUsers and tblUserRoles from the database, used to remove all entries from the database.

**uspInsertUserData**

This Procedures requires input at the correct data type to run, it runs an “INSERT INTO” SQL command into tblUser.

**uspUpdateUserData**

This Procedures requires input at the correct data type to run and existing data in the database, it runs an “UPDATE” SQL command into tblUser, updating selected existing entries.

**uspDeleteUserData**

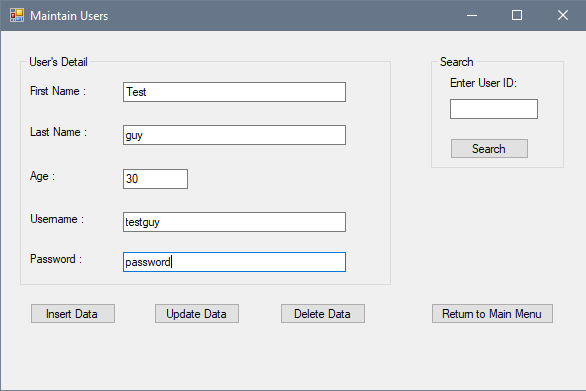
This Procedures requires input at the correct data type to run and existing data in the database, it runs an “DELETE” SQL command into tblUser, deleting selected existing entries.

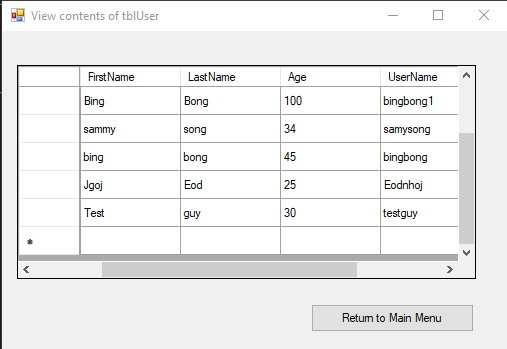
**uspselectRoles**

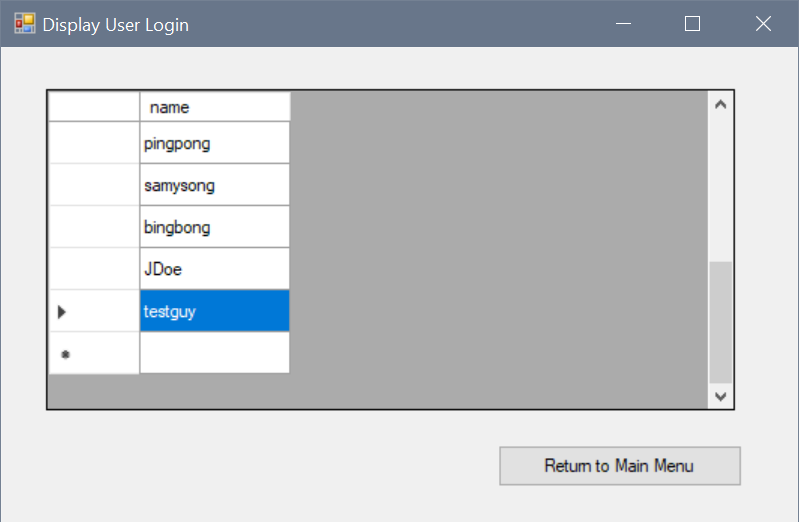
This Procedures requires input at the correct data type to run and existing data in the database, it runs a “SELECT” SQL command from tblUser, tblRoles and tblUserRoles.

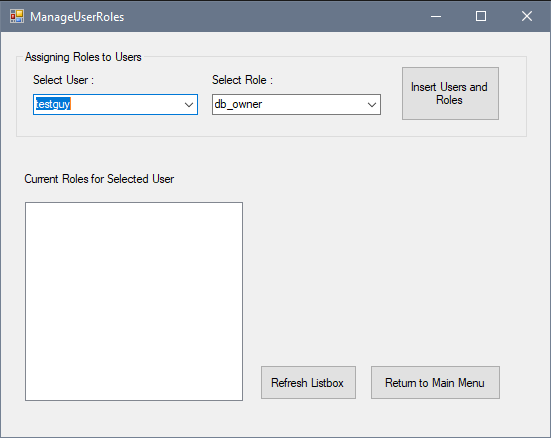
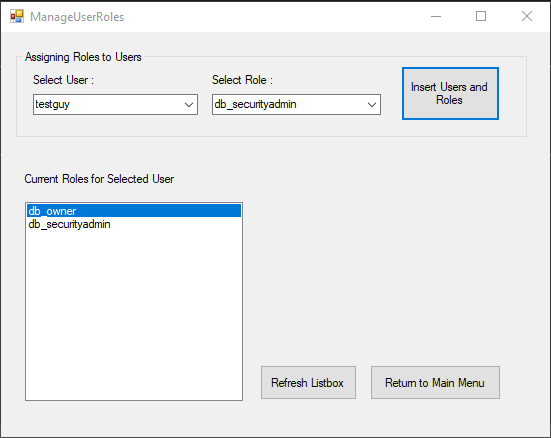
# Application test 1: Creating Users with Roles

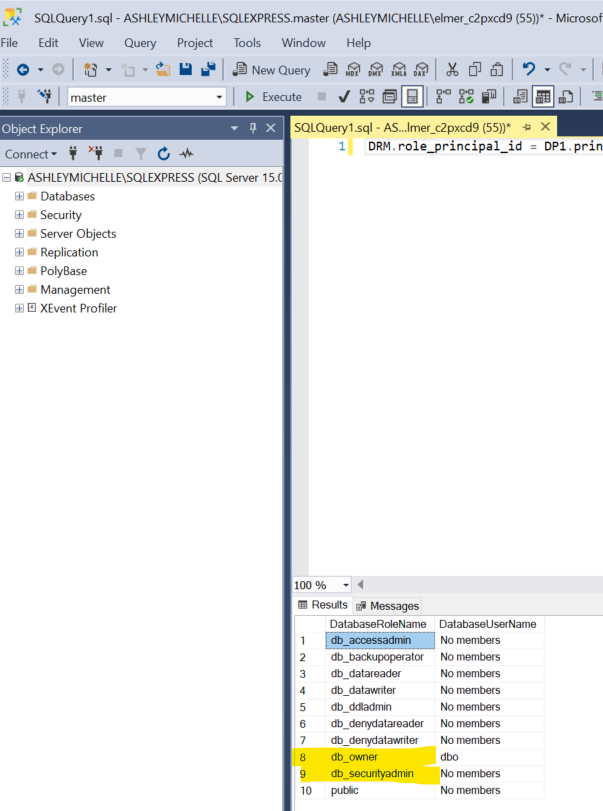
Testing insertion of a user and roles into database and master database





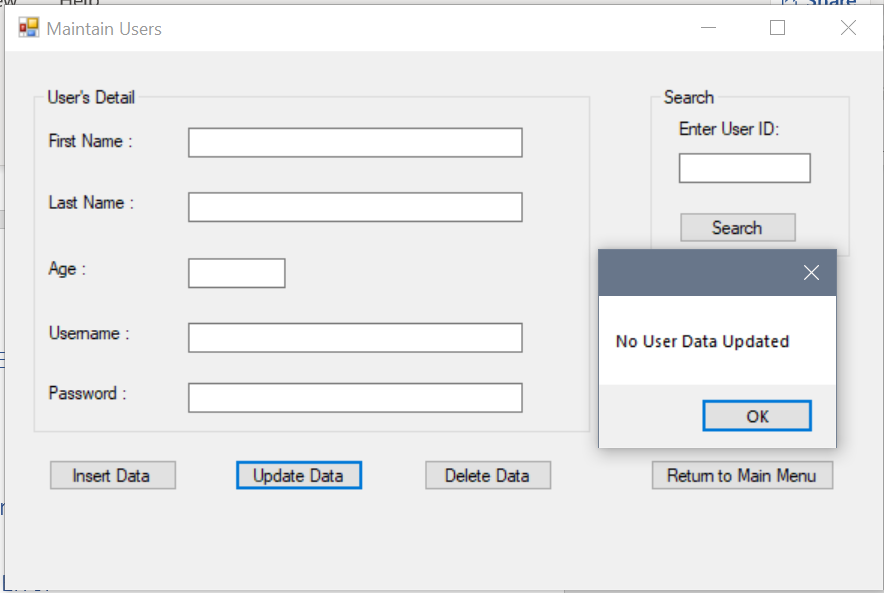


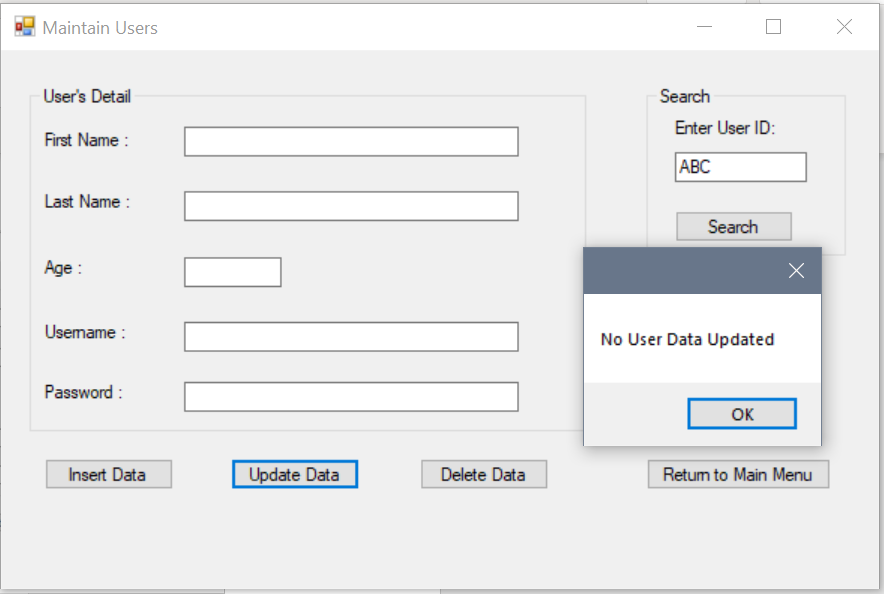


# Application test 2: Updating Error

**Empty UserID text field**

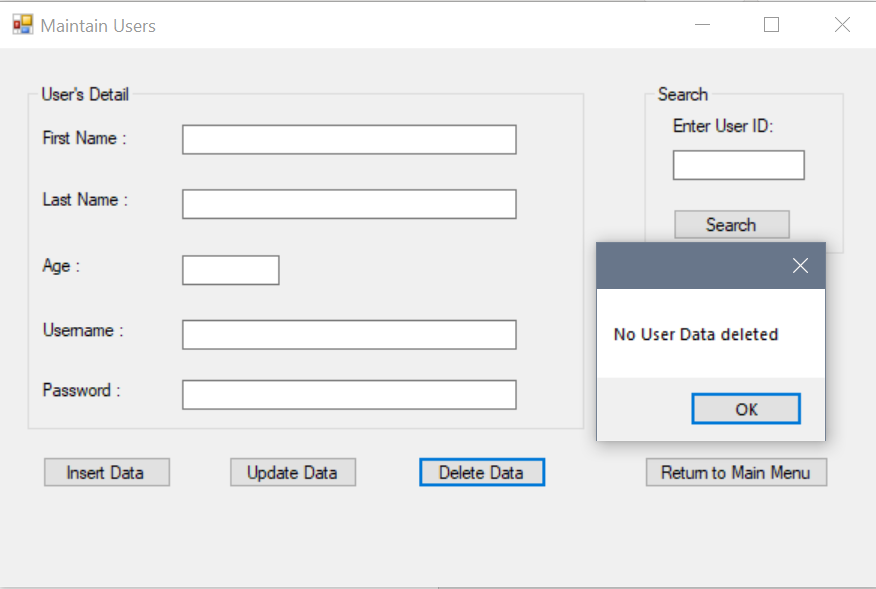
****

**UserID that doesn’t exist**

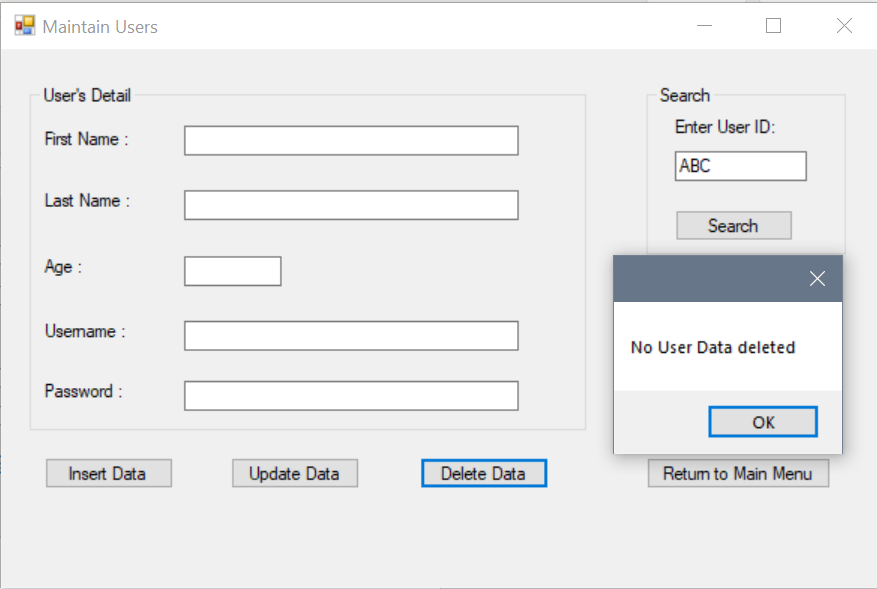
****

# Application test 3: Deleting Error

**Empty UserID text field**

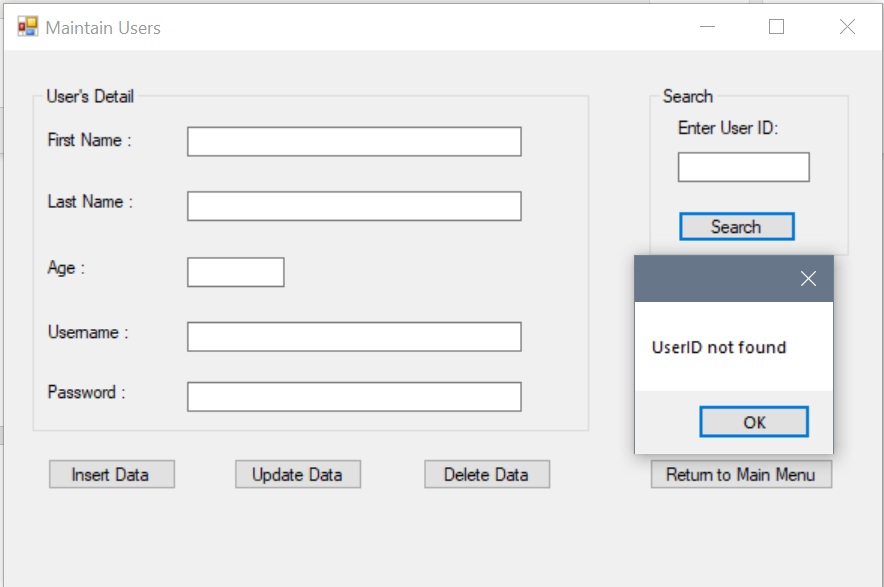
****

**UserID that doesn’t exist**

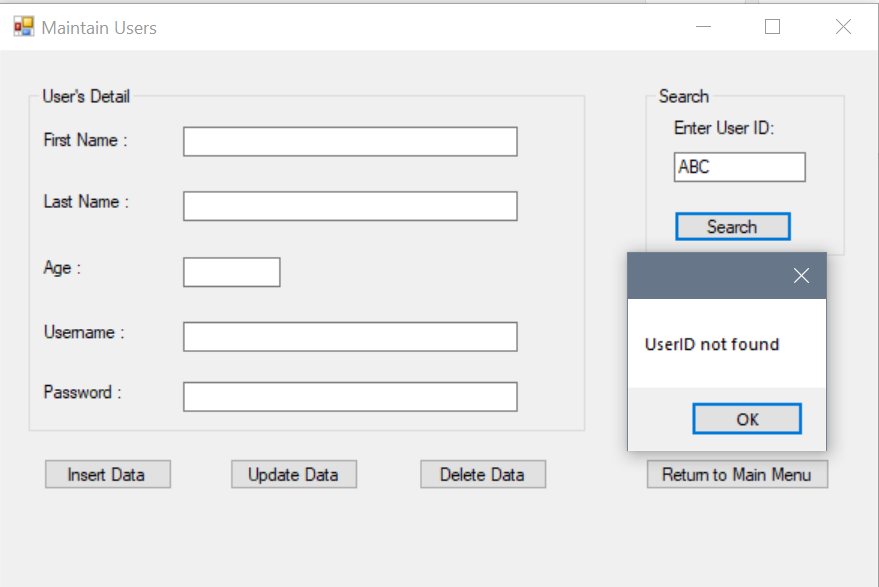
****

# Application test 4: Searching Error

**Empty UserID text field**

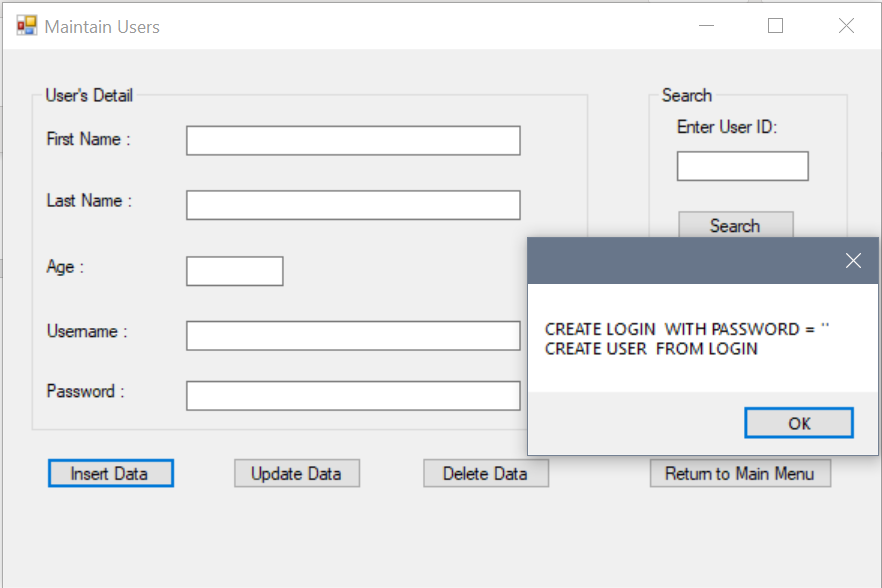
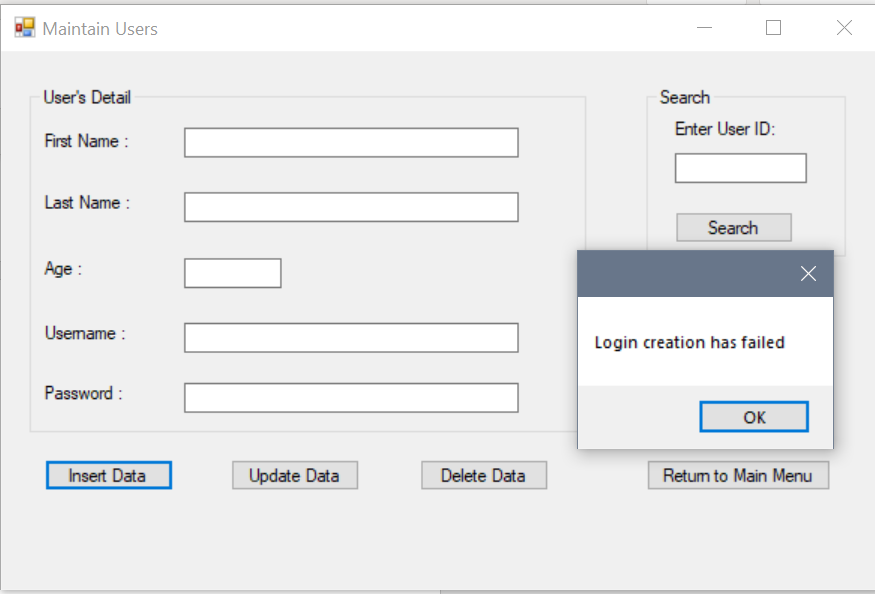
****

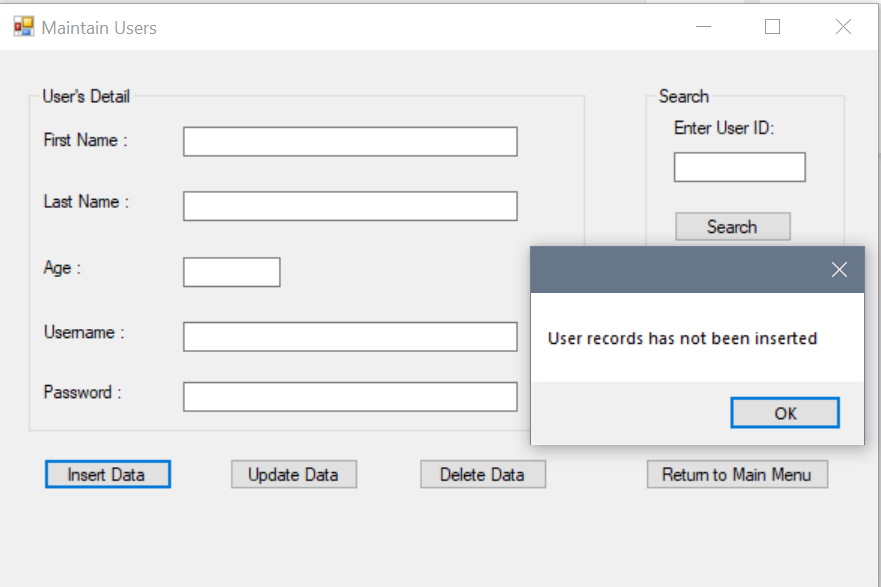
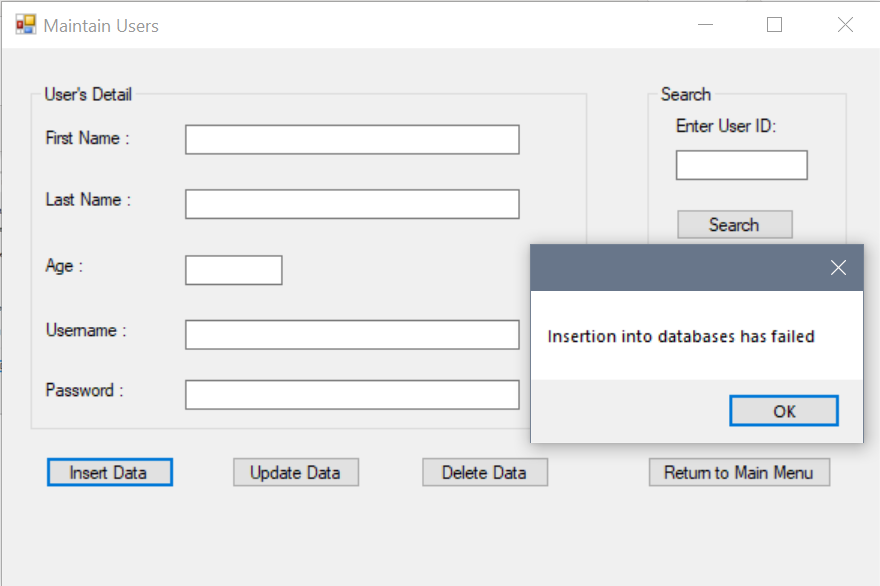
**UserID that doesn’t exist**



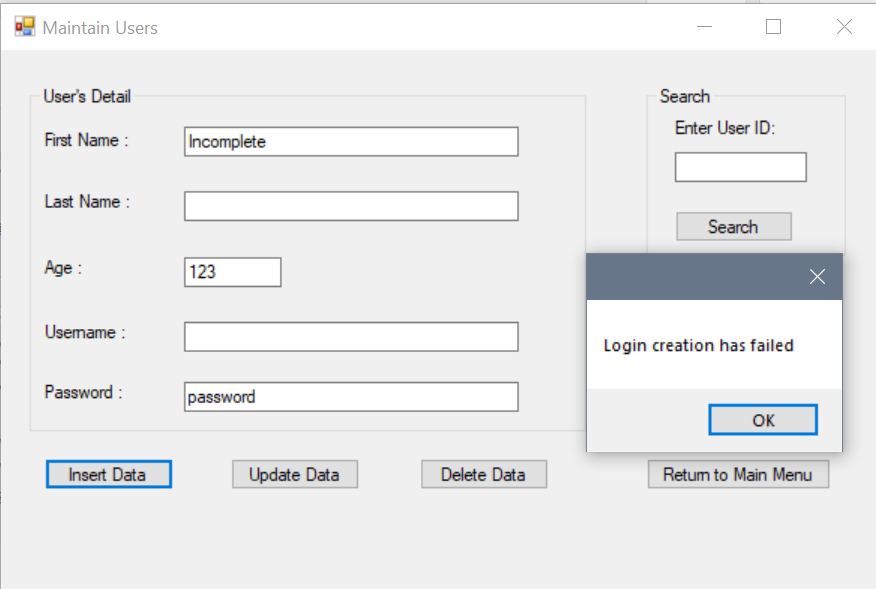
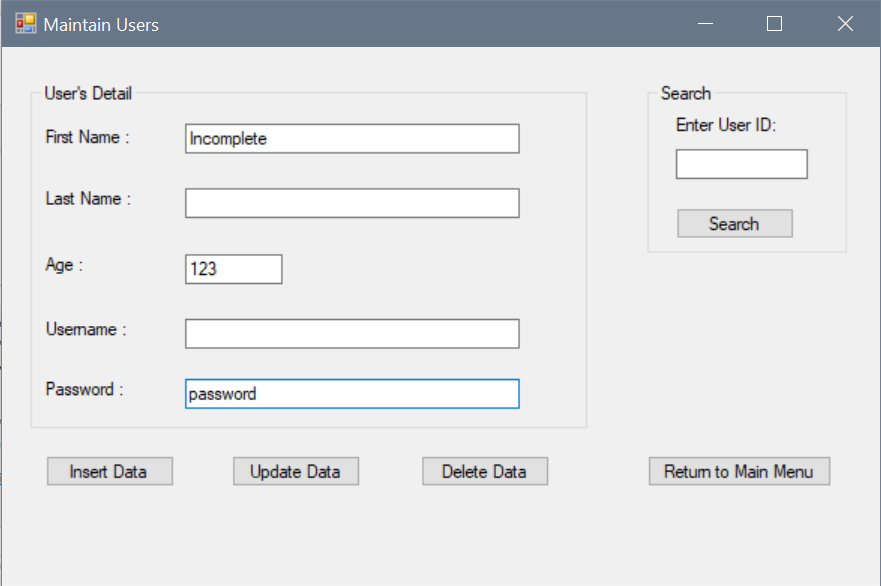
# Application test 4: Insertion Error

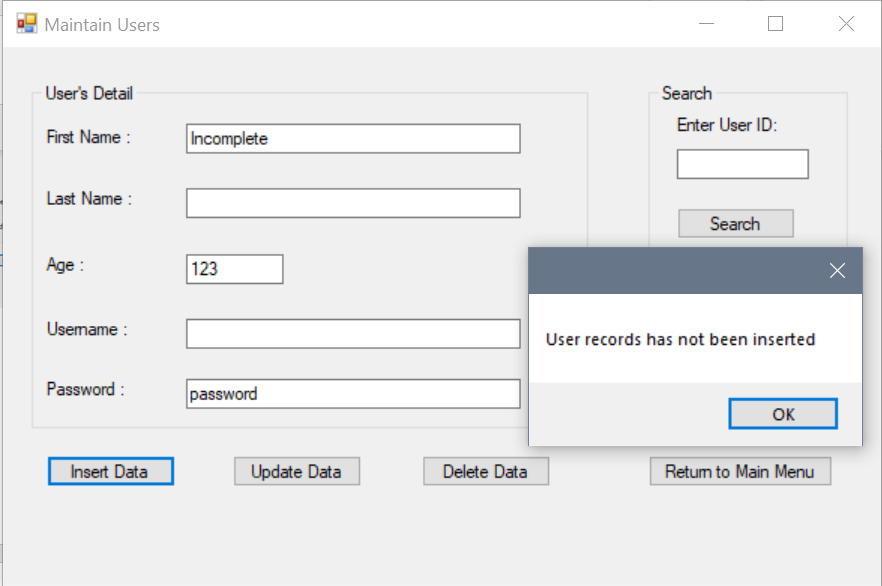
**Empty text fields**

**** ****

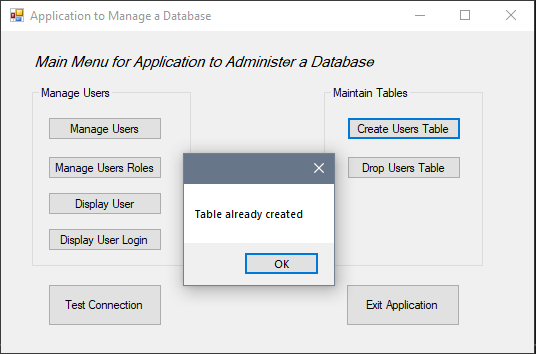
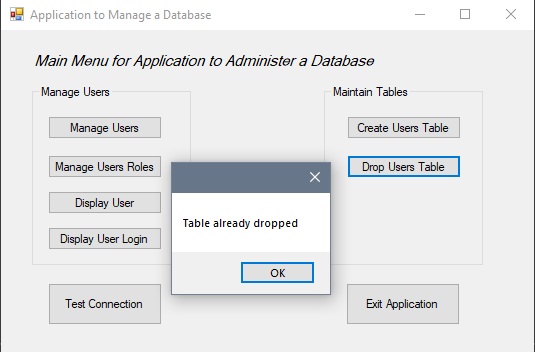
 

**Incomplete text fields**



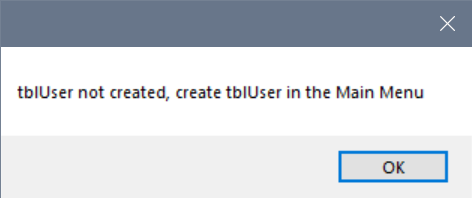
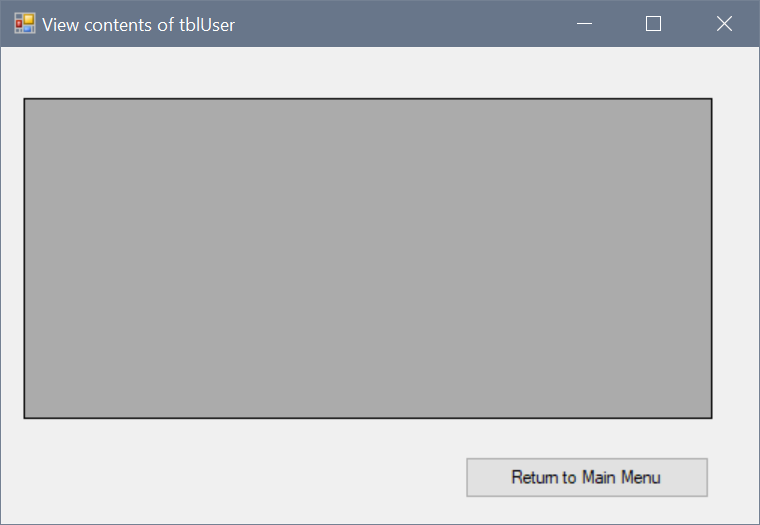


# Application test 5: Maintain Table Errors

# Application test 6: DisplayUser Errors

**Open frmDisplayUsers when tblUser is not created**

# Application test 7: DisplayUserRoles Errors

**tblUser doesn’t exist in the database**

