**CA3 Documentation**

Description

Food distribution LTD. Is a food supplier. It delivers food related products to businesses, such as a container of 3oz burgers. Their database contains four tables: products, customers, invoices and users. The first three are involved with the business side while the user table is only used to give people authorization to use the system. Each of the different tables contains multiple fields to provide information for the firm.

When the user first runs the program, the login screen is displayed. The user must enter the correct username and password corresponding to an entry on the user table, if the user enters wrong data and presses confirm an error pops up. If the user enters correct data, the main menu window opens. If a user wants to create an account they would need to sign up, by clicking the sign-up button.

When the user clicks the signup button, the signup window pops up. At the top of the screen, they can see the company's logo. Underneath they have multiple fields, combo boxes and radio buttons, that they use to fill in their data. Underneath it they can find 2 different buttons used to either confirm or cancel their sign up.

The Main Menu contains 4 tabs. Each of these tabs is related to a table on the database. Each of these tabs contains a table, several buttons and fields. The user can select an entry on the table and then either perform a delete or update operation. A user can also add an entry. A user can search the table for a specific entry using any of its data types. When the user clicks the add, update or delete button the other half of the tab is revealed. This tab contains all the fields required to view or edit the data. Also 2 buttons underneath are used to confirm or cancel a specific operation. The fields become editable or non-editable depending on the chosen operation.

Most of the tables only provide information on one of the tables except for invoices. The invoice table was joined with product and customer table to provide additional information.

Most of the fields in the program contain some sort of validation, that prevents the user from entering incorrect data and causing an error. The fields have some sort of error handling that causes an error message to appear once an error has been detected.

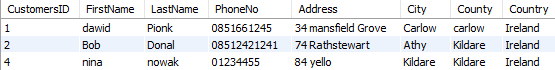
Requirements

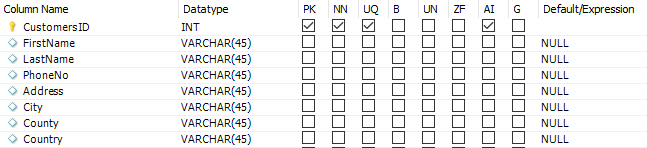
List of Requirements:

* Let the user add, update, search or delete from the product table.
* Let the user add, update, search or delete from the customer table.
* Let the user add, update, search or delete from the invoice table.
* Let the user add, update, search or delete from the user table.
* Let the user sign up and login.

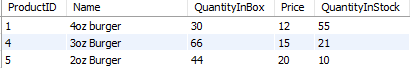
Database Screenshots

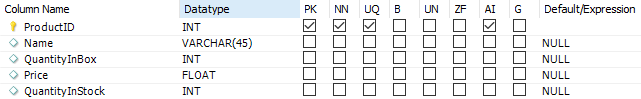
*Customers Table*



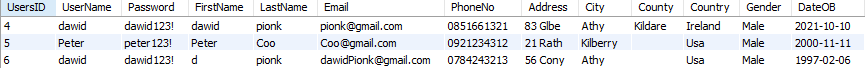


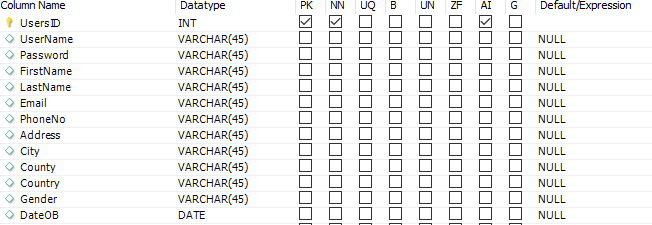
*Products Table*



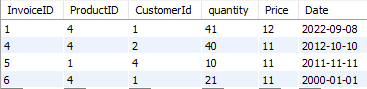


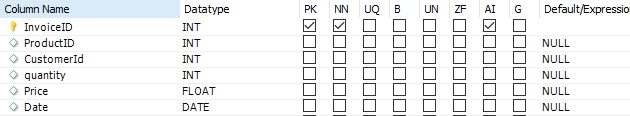
*Users Table*



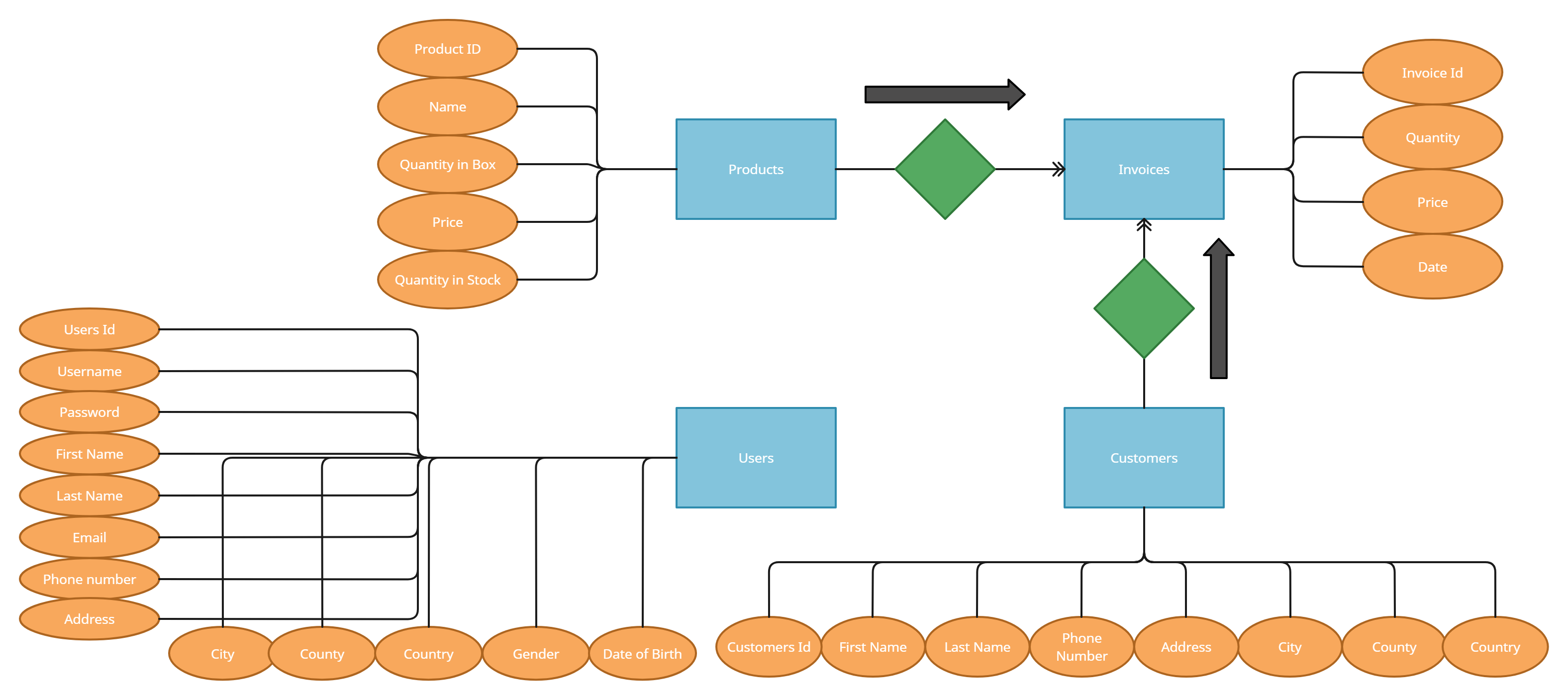


*Invoice Table*





ER Diagram



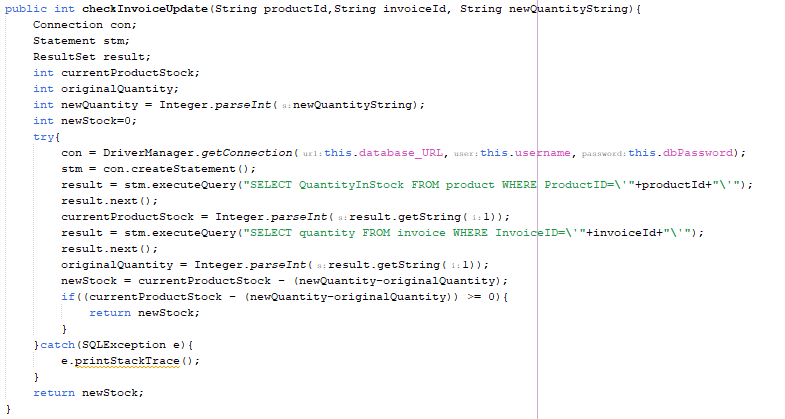
Interesting Code Snippets

Set Invoice Table



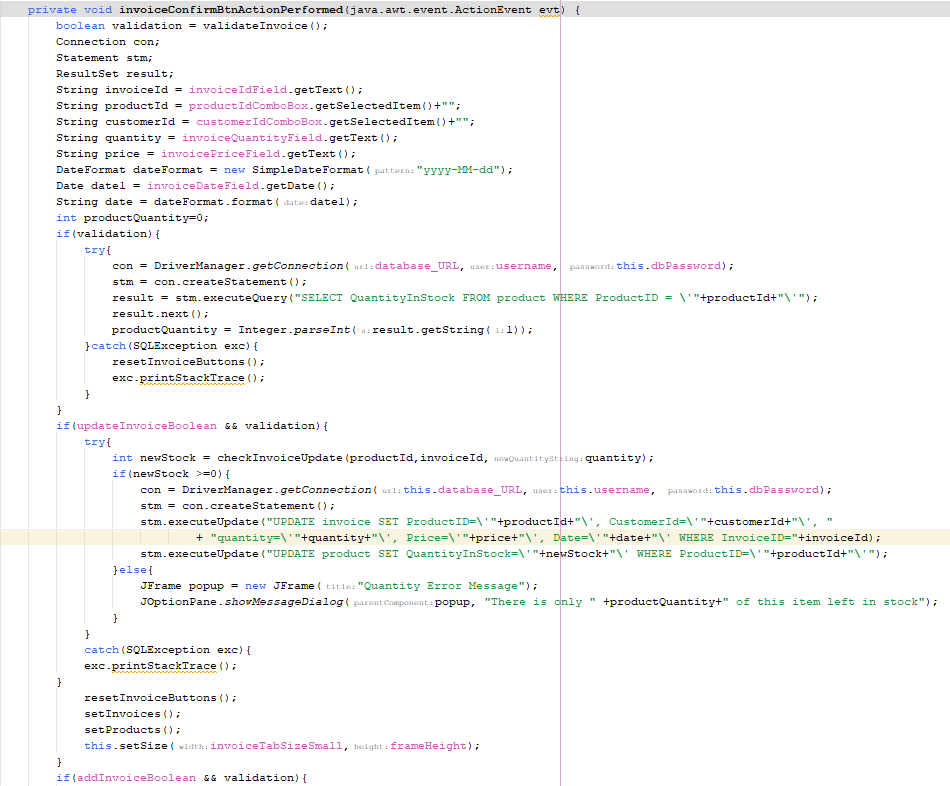
The snippet above shows how the invoice table is set. The invoice table doesn’t contain all the fields that need to be displayed, so INNER JOIN was used to combine 3 tables to display all the required data.

Check Invoice Update



This function shows how the program handles stock change. When an invoice is updated, the quantity in the stock field on the product table is changed as well. This function decides whether the update can go through, or changes how much stock should be increased or decreased. It is also used for update invoice validation, to check if the data entered for quantity is correct.

Event listener for the invoice confirm button

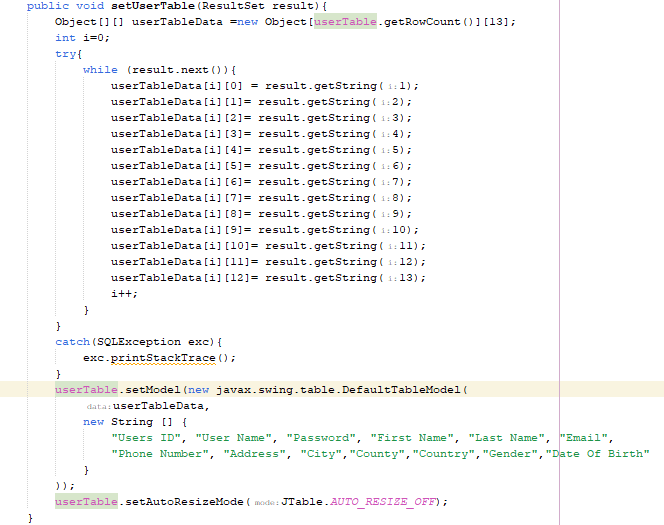




The above code shows how the program handles update, delete or add operations for the invoice table. Booleans are used to detect which button was pressed previously. All the data entered fields and combo boxes are extracted into several variables. Above you can also see the different types of SQL queries used.

Search User





The above code shows how the search user operation is executed. When the search users' button is pressed, the top function activates. This function extracts all the data from the combo box and from the search field. Afterwards it executes an SQL statement and places the results in a variable which is sent into a function that sets the user table with only the data that matched the search criteria.

Testing

User Login Screen (Login.java) - Testing

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | TC 001 | | | |
| **Requirements** | The system allows a user with an account on the database to login and use the system. | | | |
| **Preconditions** | The user has already made an account, that is stored on the database.  The user has initiated the program and is on the login screen. | | | |
| **Test Run** | **Test Data** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1. | Username: dawid  Password: dawid123! | Successful login | Successful login | Pass |
| 2. | Username: dawid  Password: dawid | Popup: Username and password are incorrect | Popup: Username and password are incorrect | Pass |

User Sign up Screen (SignUp.java) - Testing

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | TC 002 | | | |
| **Requirements** | The system allows the user to sign up, and created an account on the database | | | |
| **Preconditions** | The user doesn’t have an account on the database.  The user has opened the sign up screen after opening the login window. | | | |
| **Test Run** | **Test Data** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1. | Username: dawid  Password : dawid123!  Confirm Password: dawid123!  FirstName: dawid  Last Name: dawid  Email: [dawid@gmail.com](mailto:dawid@gmail.com)  Confirm Email: dawid  Phone Number: 0851221234  Address: 14 Rathstewart  City: Athy  Country: Ireland  County: Kildare  Gender: Male  Date of birth: 10/02/2001 | Successful signup | Successful signup | Pass |
| 2. | Username: dawid  Password : dawid123!  Confirm Password: dawid12  FirstName: dawid  Last Name: dawid  Email: [dawid@gmail.com](mailto:dawid@gmail.com)  Confirm Email: dawid  Phone Number: 0851221234  Address: 14 Rathstewart  City: Athy  Country: Ireland  County: Kildare  Gender: Male  Date of birth: 10/02/2001 | Popup: Passwords Don’t match | Popup: Passwords Don’t match | Pass |

Add Product Screen (MainMenu.java) - Testing

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | TC 003 | | | |
| **Requirements** | The system adds an entry to the product database using the data provided by the user | | | |
| **Preconditions** | The program has connected to the database  The user has logged in through the login screen | | | |
| **Test Run** | **Test Data** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1. | Name: 3oz Burger  Quantity In Box: 44  Price: 4.50  Quantity In Stocsk: 55 | Entry added to the product table | Entry added to the product table | Pass |
| 2. | Name: 3oz Burger  Quantity In Box: hello  Price: 4.50  Quantity In Stocsk: 55 | Popup: Tells the user to enter a number in the quantity box field | Popup: Tells the user to enter a number in the quantity box field | Pass |

Update Customer Screen (MainMenu.java) - Testing

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | TC 003 | | | |
| **Requirements** | The system changes the chosen entries data and updates the database. | | | |
| **Preconditions** | The program has connected to the database.  The user has logged in through the login screen.  The user has selected a customer to update | | | |
| **Test Run** | **Test Data** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1. | First Name: Stephen  Last Name: Donald  Phone number: 0121221234  Address: 94 Rathstewart  City: Houston  Country: USA | Entry updated on the customer table | Entry updated on the customer table | Pass |
| 2. | First Name:  Last Name: Donald  Phone number: 0121221234  Address: 94 Rathstewart  City: Houston  Country: USA | Popup: Tells the user data on the first name field | Popup: Tells the user data on the first name field | Pass |

Delete User Screen (MainMenu.java) - Testing

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | TC 004 | | | |
| **Requirements** | The system allows the user to delete another users entry from the database | | | |
| **Preconditions** | The program has connected to the database.  The user has logged in through the login screen.  The user has selected a customer to delete | | | |
| **Test Run** | **Test Data** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1. | Users ID: 4  Username: dawid  Password : dawid123!  Confirm Password: dawid123!  FirstName: dawid  Last Name: dawid  Email: [dawid@gmail.com](mailto:dawid@gmail.com)  Confirm Email: dawid  Phone Number: 0851221234  Address: 14 Rathstewart  City: Athy  Country: Ireland  County: Kildare  Gender: Male  Date of birth: 10/02/2001 | Entry has been deleted from the database | Entry has been deleted from the database | Pass |

Search Invoice Screen (MainMenu.java) - Testing

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | TC 005 | | | |
| **Requirements** | The system allows the user to delete another users entry from the database | | | |
| **Preconditions** | The program has connected to the database.  The user has logged in through the login screen. | | | |
| **Test Run** | **Test Data** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| 1. | Search Combo Box: Invoice ID  Search Field: 1 | The invoice entry matching the data provided appears | The invoice entry matching the data provided appears | Pass |
| 2. | Search Combo Box: Invoice ID  Search Field: | Popup: Tells the user to enter search criteria | Popup: Tells the user to enter search criteria | Pass |