Programming Assessed Exercise 1

**Paul McHard – 2085227M**

**Final State of Program**

The program provides the full functionality required by the exercise specification. It has been thoroughly tested and I believe it adheres precisely to the specification required and fully satisfies what is asked for in the exercise.

**Assumptions**

Mathematically, three assumptions were made. Firstly, it was assumed that for processing transactions it is acceptable to keep both the balance and sale/return price as unbound doubles during calculation, and then formatting results to two decimal places as Strings when output to the text fields.

Secondly, it was assumed that output of the cost of the last transaction should always display as positive, irrespective of whether a sale or return occurred; the ‘Last Cost’ text field’s purpose is to display the scalar value of the previous transaction, rather than showing a positive or negative value, denoting whether the transaction was a sale or a return.

Lastly, it is assumed, and specified in the input box, that the user will enter their initial balance as credit positive. It was intuitive that a user would enter the amount they had available to spend in their account (credit) as positive naturally. The program expects and asks for this input and handles it accordingly.

It was also assumed that the scope and use of some variables are unlikely to change with further program changes. This manifests itself in AssEx1.java , where I have neglected to create pointers for the LWMGUI and CustomerAccount objects, or the username and balance variables, instead choosing to create these objects directly as they are needed. I felt constructors were unnecessary here; the balance and username come from methods which return a double and String respectively, so can be called directly in the calling CustomerAccount constructor, which can in turn be created directly in the construction of the LWMGUI, which itself does not need a pointer, as the main script’s sole purpose is to load the GUI; the main script executes no further commands. I continue this reduction of pointers where it is throughout the program where it is efficient to do so, but this would however be simple to alter if need be.

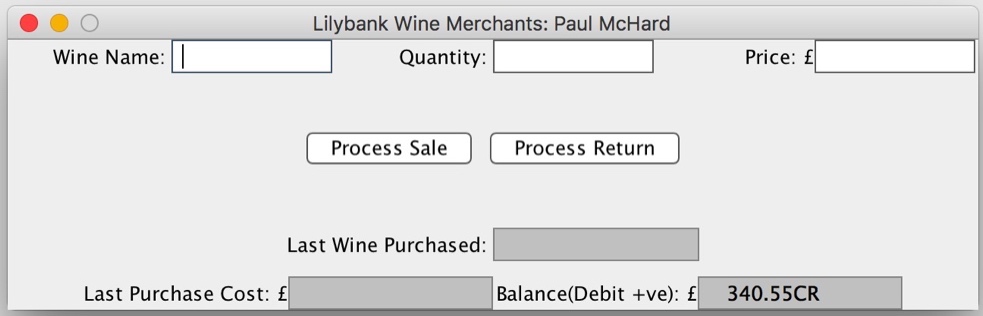
**Deficiencies**

The program has no known deficiencies and has been completed as per the specification. The code is properly indented and well commented.

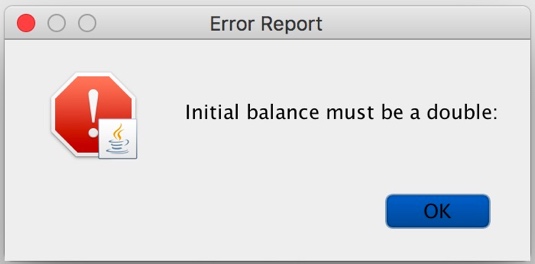
**Test Data**

|  |  |
| --- | --- |
| *Function* | *Proof* |
| Program gets customer name and initial balance from input dialog boxes. | Image 1 |
| Program validates user input for initial balance. | Image 2 |
| Program correctly validates the amount for one transaction and displays the amount and updated balance correctly in the GUI | Image 3,4 |
| Program correctly handles negative account balances as credit and formats output accordingly. | Image 1 |
| Program clears input boxes following a transaction. | Image 4 |
| Program correctly handles invalid input from user in regards to type of input (cost must be double, quantity must be integer, name must not be null.) | Image 5,6,7 |
| Program correctly handles invalid input from user in regards to zero value of quantity or price. | Image 8,9 |

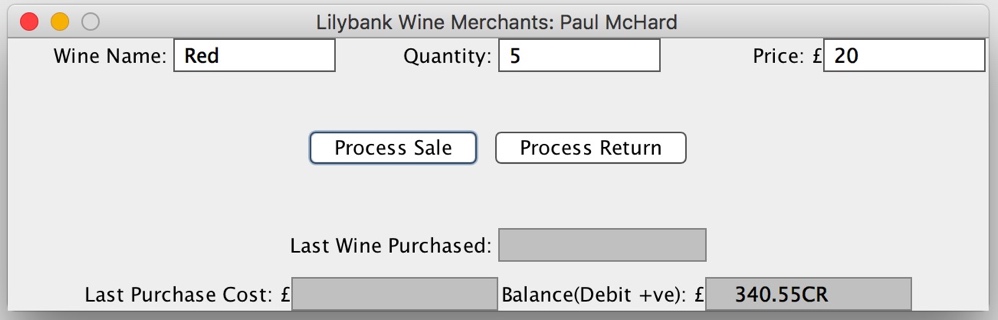
**Screenshots**

**

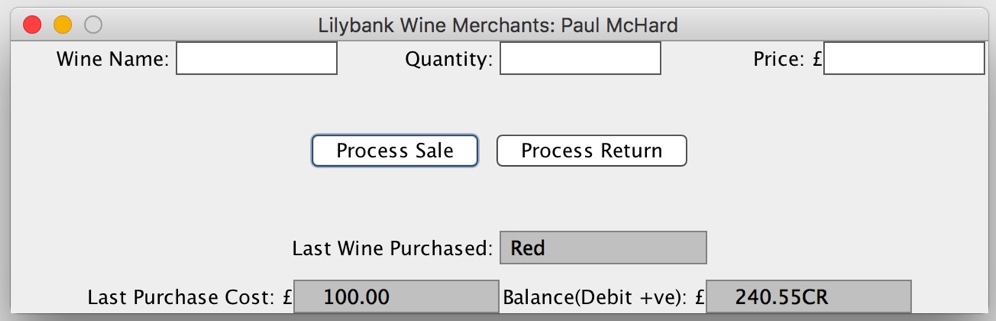
*Image 1*

**

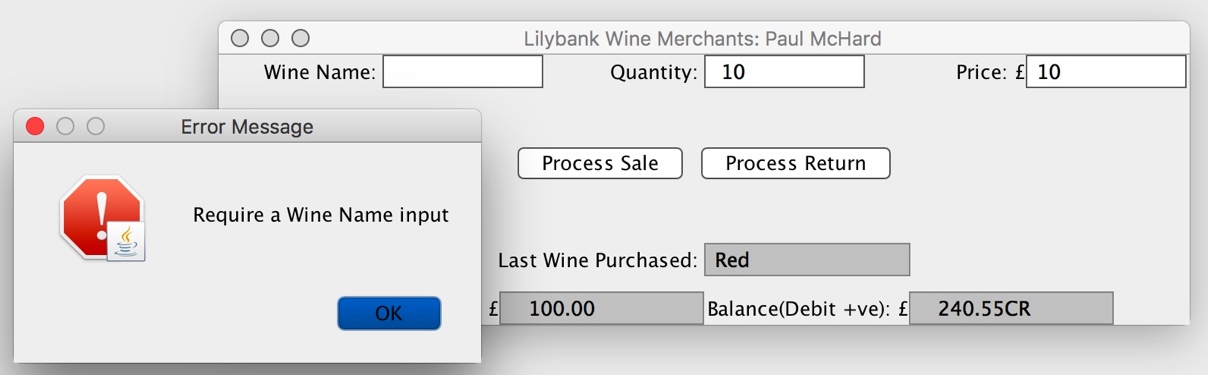
*Image 2*

**

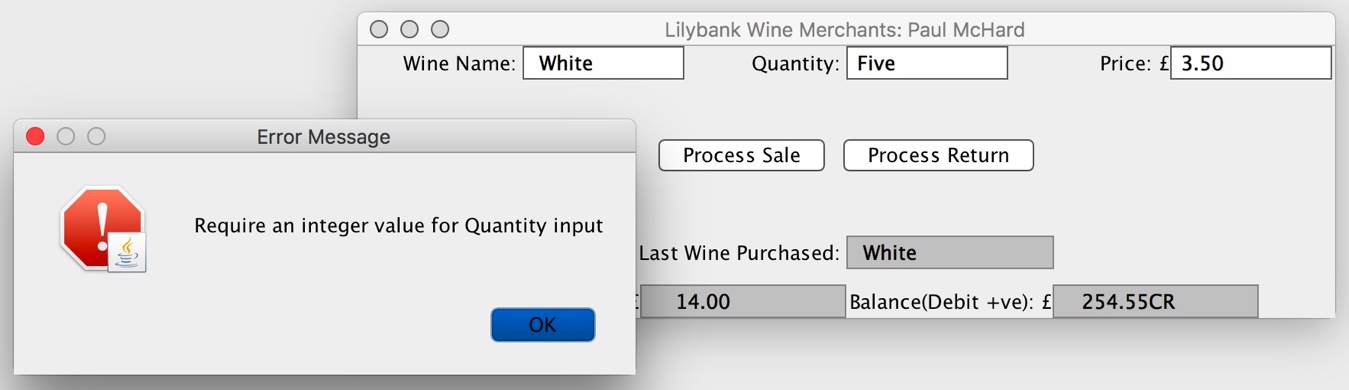
*Image 3*

**

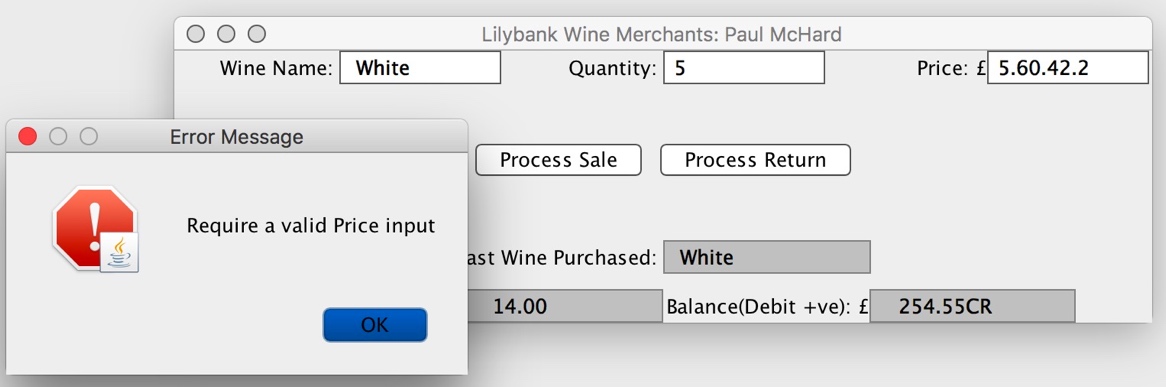
*Image 4*

**

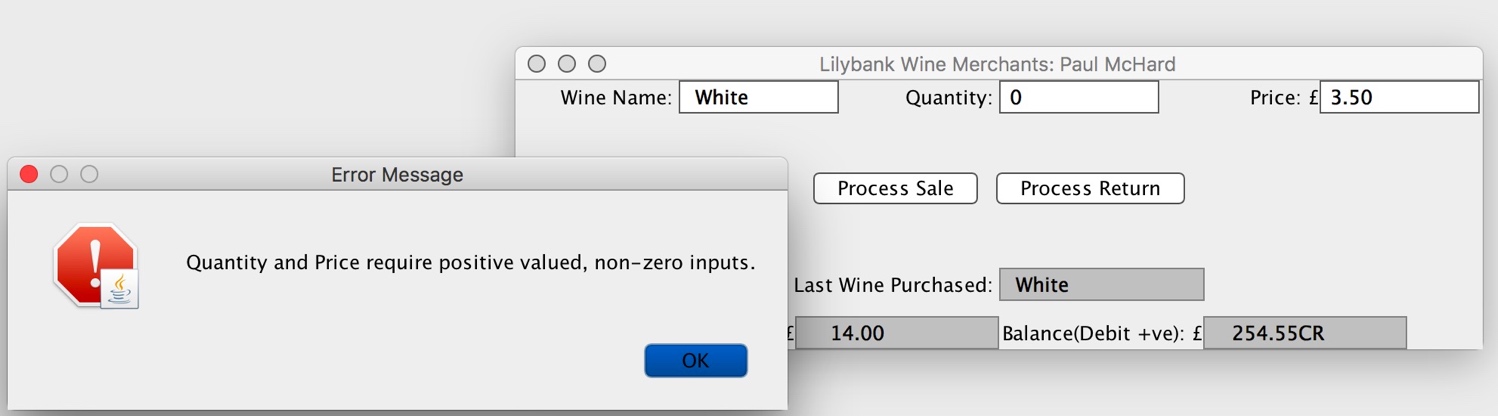
*Image 5*

**

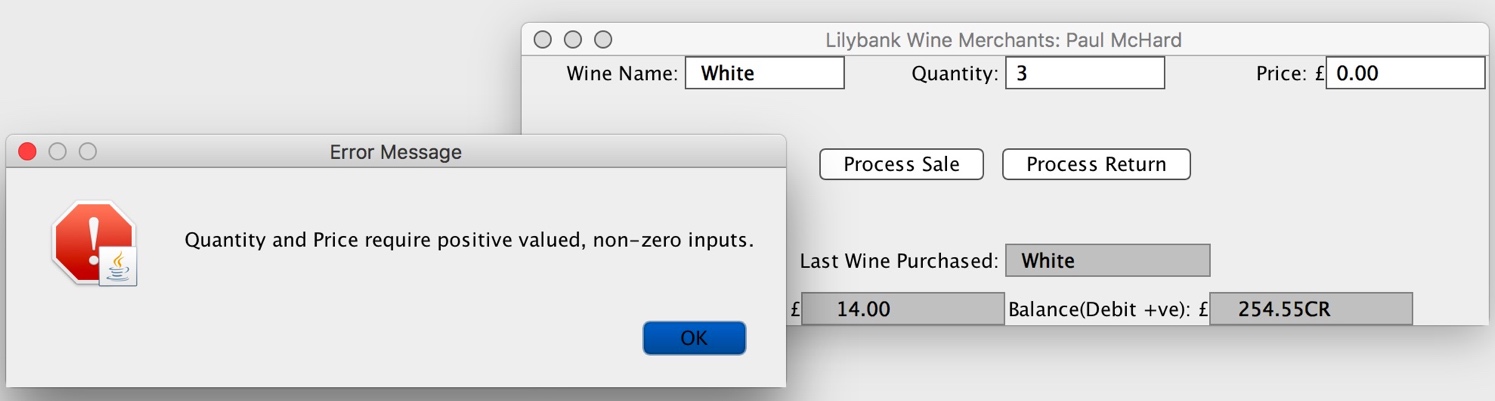
*Image 6*

**

*Image 7*

**

*Image 8*

**

*Image 9*