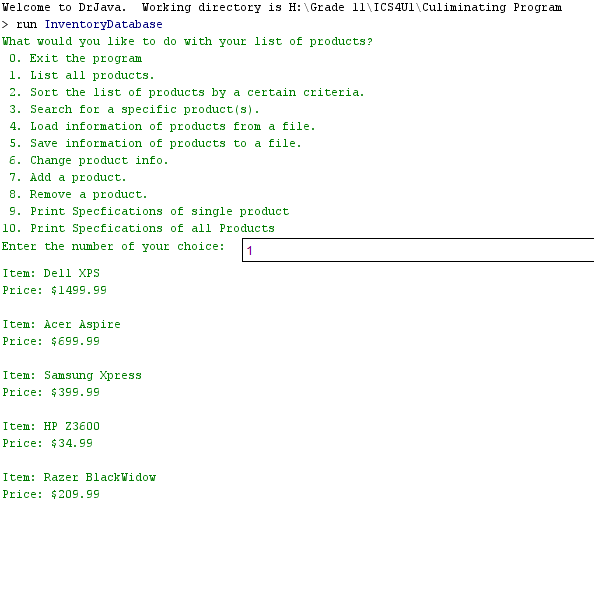
4.A) System Testing

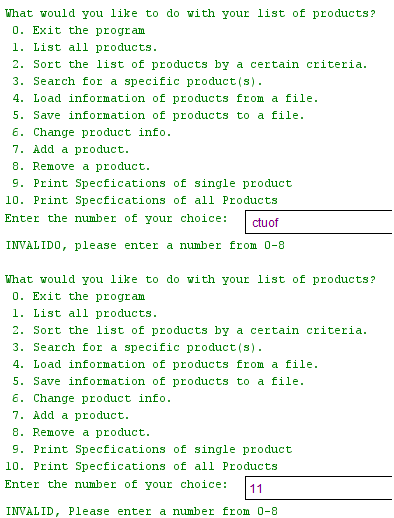
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| # | Functionality | Purpose | Input | Expected Output | Output |
| 1. | SortHighToLow | sort price highest to lowest | Inventory list | The inventory is sorted by prices from highest to lowest, if two prices were the same value, order is whichever came first | The output was the same expected output |
| 2. | SortLowToHigh | sort price lowest to highest | Inventory list | All prices in the inventory list is sorted highest to lowest, if two prices were the same value, order is whichever came first | The output was the same expected output |
| 3. | SortBrand | sort all brands in alphabetical order | Inventory list | All prices in inventory list is sorted in alphabetical order | The output was the same expected output |
| 4. | SortName | sort the items by name in alphabetical order | Inventory list | All products in the list are sorted by name in alphabetical order | The output was the same expected output |
| 5. | searchTag | search specific tag from all product | Inventory list, String [] tags | Return all products that have the tag that is prompted | The output was the same expected output |
| 6. | findTag | checks to see if a product has a certain tag | Product, String tag | Returns a true or false value to see if it has the tag or not | The output was the same expected output |
| 7. | searchName | search for a product with a specific name | Inventory list, String name | Return the product that has the same name | The output was the same expected output |
| 8. | searchBrand | search for all products with a specific brand | Inventory list, String brand | Return all products that have the same brand | The output was the same expected output |
| 9. | searchPrice | search for all products with a specific price | Inventory list, double price | Return all products that have the same price | The output was the same expected output |
| 10. | Load | load file from a text | Inventory list and a txt file | list was loaded in | Information was not loaded |
| 11. | Save | save a txt file | Inventory list | the txt file was saved | A run time error occured |
| 12. | Add Product | add a new product to the inventory | inventory list | parameters for a new parameter is prompted and then the item is added | The output was the same expected output |
| 13. | Remove product | removes product from the inventory | Inventory list | product is removed from inventory arraylist | The output was the same expected output |
| 14. | Print Specifications | prints the specifications of a product | Inventory list, name | it finds the product given its name, then prints its specifications | The output was the same expected output |
| 15. | ChangeProductInfo | given a certain product it would change the values of it’s parameters | Inventory list, name | it finds the product, then reprompts for the new values | The output was the same expected output |
| 16. | PrintProducts | given a list of products, it should print the name, brand and price of each product | Inventory list | it goes through every product, and prints the name, brand, and price | The output was the same expected output |

4. B)

This program is a database of products such as computers, laptops, printers, keyboard, and mouses. The program starts off at a menu in which the user enters the number of their choice. For example, it might be listed 1. Print all products 2. Sort products 3. Search for a product and so on, to sort the products the user would enter 2. The user is advised to enter proper input since if they do not, they will not receive a proper response. The user is told how to enter certain things correctly. The other section of the program for searching and sorting is similar in the sense of having a main menu and a similar number based choice system. Before running the program ensure all files are saved in the same location. Refer to design document for file format if you wish to load your own products into the program.

Sample Output of main menu

Sample output of when user enters invalid data



4. C) Project Review

**Did it work?**

Yes, the program worked quite well, there were no huge errors

**Did it address the goals and objectives?**

Yes, except for the sale function

**Did it work for some datasets, but not others?**

It worked for all datasets except we could not use a boolean instance field as it would cause a runtime in the load and save methods

**Does the program in its current form have any limitations?**

The inventory can not be too big because it prints all products vertically

**What additional features could the program have?**

A sale function

**Was the initial design appropriate?**

Yes, except for the boolean instance fields, the initial design had little flaw to it

**Did we meet all the deadlines?**

Yes, all of our stages were handed in before the due date

**Could we have divided the work in a better way?**

No, the work was divided pretty evenly. no one had more work than another

**Could we have manage the project components better?**

no, because we had worked pretty efficiently on assignment