Homework 5 (Shopping) Test Cases - (Juliana Carvalho)

Test Case Name	Scenario	Steps to Recreate	Expected Results	Actual Outcome	Output Sample	Bugs
Enter a list of products	- Verify that user can enter items regardless if it's in upper case or lower case. - Verify that program will not end if input is an integer. - Verify that only 7 items is asked from the user - Verify that there are no duplicate products	- Program will prompt the user to enter an item name one-by-one > Add a value in String or Integer form > Prompt will end after 7 items (NON DUPLICATE)ite ms have been added	- Program will accept a total of seven items, without any errors. - Duplicates items will be discarded - Program should ignore case	- Programs accepts seven items without any errors. - Program accepts items containing String and Integers - Program ignores duplicate items	Hello! Let's Add Some Products to Our List Enter an Item APPLE Enter an Item apple Enter an Item Apple Enter an Item Banana Enter an Item W@ater Enter an Item Grapes Enter an Item Apple Enter an Item 0ranges123 You Have Entered: [APPLE, apple, Apple, Banana, W@ater, Grapes, 0ranges123]	- Program will ignore duplicates ONLY if the item is entered exactly as the one before. - Apple was entered 4 times (Upper Case and Lower Case) but the program recognizes upper case as being a separate product.
Assign a priority to each item and allow duplicate priorities	- Verify that user can assign an integer value as the product priority - Verify that the user can enter a priority in the required order	- Program will take the product name, and ask the user to enter a priority for that product > user will enter a priority for seven given items.	- User will be allowed to enter a priority in integer form without any errors. - User will be asked to enter priority for 7 items only	- User is allowed to enter a priority in integer form without any errors. - User is prompt to enter the priority for seven products only	Enter a priority for: APPLE 1 Enter a priority for: apple 1 Enter a priority for: Apple 2 Enter a priority for: Banana 3 Enter a priority for: W@ater 3 Enter a priority for: Grapes 4 Enter a priority for: 0ranges123 5	None
Error handling if priority is not in integer form	- Verify that if a user enters any value that is not an integers, the program will handle an Input Mismatch Exception	Enter products > User will be prompt to enter product priorities > Enter any value that is NOT an integer	- Program will print a warning saying "Invalid Input … Program Will End" - Program will exit	The program will end if any value that is not an integer is entered.	Enter a priority for: APPLE 1 Enter a priority for: apple 2 Enter a priority for: Apple 1 Enter a priority for: Banana A Invalid Input...Program Will End Goodbye	- None - the program will end if a string or a special character is entered.
Enter a Budget Amount	Verify that user will be able to enter a budget that is $58.00 or less	User will be prompt to enter a budget amount > Enter any amount that is 58.00 or less > Program should accept without any errors	- User is allowed to enter a value that is 58.00 or less. Program should move on.	- Program accepts any value that is 58.00 or less	Bank Account $ 58.0 - At this point the program will proceed to print a report with the list of products, priorities and price.	None
Enter a Budget Amount - with Conditionals and Error Handling	- If the user input is not an integer or double, the program will prevent an exception error and end the program. - Verify that the user input is 58.00 or less.	- User will be prompt to enter a budget amount > enter a string or special character > the program should print an error message and end the program. - User will be prompt to enter a budget amount > On the first try, enter any value beyond $58.00 > A warning message will print, reminding the user that budget amount must be below $58.00 > If on the second try, the user still enters a budget greater than $58.00 > End the program	- If a string or special character is entered in place of an integer or double , then end the program. - If the user enters a budget amount greater than $58.00 on the second try, end the program. - If a valid amount is entered on the second try, then the program will proceed correctly	- Program will end if a special character or string is entered in place of a double or integer value.	- **1st Scenario** Enter A Budget @ Invalid Input. System Will Now Exit **- 2nd Scenario** Enter A Budget 76 Budget Must Be Below $58.00 Try One More Time . . .Enter A Budget 59 Sorry! Application Will Now End	- None - program will enter if a string or special character is entered. - None - The user will get two chances to enter a budget amount. On the second try, if a valid value is entered, the program will proceed.

Test Case Name	Scenario	Steps to Recreate	Expected Results	Actual Outcome	Output Sample	Bugs
Display list of products	An auto-generated report will print listing all the products priorities and price	- Report will generate after entering the budget value	Report will print with the following columns: - Priority - Product - Price - Report will display product priorities and price will match product name when displayed on the report - Report should display priorities in ascending order and the correct corresponding product next to the priority.	- Priority column is sorting correctly based on the bubble on bubble sort algorithm. - Product array fails to sort in parallel to the priority array. - The product array will print in the order that it was entered.	``` --------------------------------------- Priority Product Price --------------------------------------- 1 a $ 3.64 2 b $ 3.1 3 c $ 4.01 4 d $ 5.09 5 e $ 3.74 5 f $ 4.9 6 g $ 6.77 --------------------------------------- ``` - In this scenario, the priority values were NOT entered in the order that it printed in this report. So the priority column is sorting and printing on the correct order. - The product column is not sorting correctly. In this scenario, the product list was printed in the order that it was entered, and not in the priority order.	- Bubble sort works just for the priority column. - Product column won't sort correctly.
Adding products to shopping bag	- Verify that user can enter a products to shopping cart, based on the priority. - Verify that user can enter quantity	User will be prompt to enter a priority > the program will match the priority to the product and add to a shopping list > program will print and display the item price, shopping cart items shopping cart total and Remaining Bank Account Value > program will ask for a priority until the shopping cart total is $59.00 or less	- Product added to the shopping bag, correspond s to the correct product priority determined by the user. - Program will ask the user to enter an item until the shopping bag total is $59.00 or less. - Shopping Cart should support adding products based on quantity.	- Program adds the correct product to the shopping bag, based on the user determined product priority. - The while loop fails to stop once the total shopping bag reaches $59.00. Currently the loop allows users to exceed the shopping bag limit. - Program fails to add products to shopping cart based on quantity or display how many of each products the user will be purchasing.	``` Choose a Priority 6 Choose a Quantity 1 ITEM PRICE: $ 48.54 SHOPPING CART: [butter, eggs] SUBTOTAL: 58.66 REMAINING BANK ACCOUNT : -26.2 ```	- Program fails to add products to shopping list based on quantity. **For Example:** - 2 apples Then shopping cart should be: SHOPPING CART: [apple, apple] - Shopping Cart allows user to enter products, even though shopping cart value is beyond $58.00
Verify that priority exist	User should be allowed to enter priority values between 1 - 7 BUT if another value is entered, program should warn the user that the value does not exist	Once user is prompt to enter a value > enter a number that IS NOT between 1 - 7> Message should print: *"Cannot Locate Priority in the Array, Please Try Again"*	- If the value entered is greater than 7, than the following message will print: *Priority Does Not Exits* - Program will not end.	- Message is printed correctly.	``` Choose a Priority 8 Choose a Quantity 1 Priority Does Not Exist Choose a Priority 2 ```	None - if priority doesn't exist than a message will print back to the user
Printing a Text File report	- After all products are purchased, a text file report will display all the products that were purchased.	After all items within the allowed price range is entered > If the filename is correct, the program will display a message saying that a text report will be printed.	Once the program is finished, a text file should be found in the project file path	- Program correctly prints a summary of the user purchase	Message that will be displayed `A Report of Your Purchase Can be Found in: ProductList1.txt` **Text Report** ``` Product Price --------------------- eggs $ 33.37 oranges $ 29.29 --------------------- BAG TOTAL: $26.2 ```	- If more than 7 are added by the user, the product printer will run into an index out of range exception. - Exception happens in the productPrinter class and on line 41.