Joshua Atienza (NetID: jra165), Kyle Lee (NetID: kl781)

Test Document for GroceryBag Class

Test Case #	Purpose for test case	Input Data	Expected Output
0	Test the constructor for instantiating an instance of ShoppingBag. The instanceof method will be used to compare the instance with the ShoppingBag object. Test case prints true if instance is an instanceof ShoppingBag.	N/A (ShoppingBag instances: **traderJoes**, **wegmans**, **target**)	**true** \n **true** \n **true**
1	Test the .add() method, which adds a GroceryItem to the bag. ● *Case 1*: GroceryItem added to bag and bag size stays the same, Test case prints "GroceryItem added to the bag." ● *Case 2*: GroceryItem added to bag and bag capacity increases by 5. Test case "GroceryItem added to the bag" and the new capacity of the bag.	*Case #1* (ShoppingBag instance: **traderJoes**) - Instance #1 of (GroceryItem): **"milk", 2.50, true** *Case #2* (ShoppingBag instance: **traderJoes**) - Instance #2 of (GroceryItem): **"eggs", 1.99, false** - Instance #3 of (GroceryItem): **"toast", 3.14, true** - Instance #4 of (GroceryItem): **"meat", 7.99, true** - Instance #5 of (GroceryItem): **"bread", 3.99, true**	*Case #1* prints **"milk added to the bag"** *Case #2* prints **"eggs added to the bag."** \n **"toast added to the bag."** \n **"meat added to the bag."** \n **"bread added to the bag."** \n **"BAG CAPACITY: 10"**
2	Test the .grow() method, which increases bag size by 5. Test case prints the new bag capacity and current number of GroceryItems.	N/A (uses same ShoppingBag instance **traderJoes** in previous test case)	**"BAG CAPACITY: 10"** \n **"CURRENT # ITEMS IN BAG: 5"**
3	Test the .find() method, which finds a target GroceryItem in the bag ● *Case 1*: GroceryItem found, returns index of item ● *Case 2*: GroceryItem not found, returns -1	*Case #1* (ShoppingBag instance: **traderJoes**) - Instance #1 (of GroceryItem): **"eggs", 1.99, false** Case #2 (ShoppingBag instance: **traderJoes**) - Instance #2 (of GroceryItem): **"jelly", 3.99, true**	*Case #1* prints **"Index of eggs: 1"** *Case #2* prints **"Index of jelly: -1"**

4	Test the .remove() method, which removes a GroceryItem from the bag • *Case 1*: GroceryItem successfully removed and number of items in bag decreases by 1, returns true • *Case 2*: bag is empty, return false • *Case 3*: GroceryItem does not exist in bag, returns false	*Case #1* (ShoppingBag instance: **traderJoes**) - Instance #1 (of GroceryItem): **"eggs", 1.99, false** *Case #2* (ShoppingBag instance: **traderJoes**) - Instance #2 (of GroceryItem): **"jelly", 3.99, true** *Case #3* (ShoppingBag instance: **wegmans**) - Instance #3 (of GroceryItem) : **"jelly", 3.99, true**	• *Case #1* prints **"eggs 1.99 removed."** \n **"STATUS: true"** \n **"CURRENT # ITEMS IN BAG: 4"** • *Case #2* prints **"Unable to remove, this item is not in the bag."** \n **"STATUS: false**" \n **"CURRENT # ITEMS IN BAG: 4**" • *Case #2* prints **"Unable to remove, this item is not in the bag."** \n **"STATUS: false"** \n **"CURRENT # ITEMS IN BAG: 0"**
5	Test the .salesPrice() method, which returns sum of the prices in the bag	N/A (tested with **traderJoes** ShoppingBag instance)	**"Sales Tax: $1.17"**
6	Test the .salesTax() method, which returns sales tax total of taxable GroceryItems in the bag	N/A (tested with **traderJoes** ShoppingBag instance)	**"Sales Price: $17.62"**
7	Test the .print() method, which prints the string representation of each GroceryItem in the bag • *Case 1*: bag is empty, which prints "The bag is empty!" • *Case 2*: 1 GroceryItem is in the bag, which prints "You have 1 item in the bag..." • *Case 3*: Multiple GroceryItems in the bag, which prints "You have xx items in the bag..."	*Case #1* Instance #1 (of ShoppingBag): **target** *Case #2* Instance #2 (of ShoppingBag): **wegmans** *Case #3* Instance #3 (of ShoppingBag): **traderJoes**	*Case #1* prints **"The bag is empty!"** *Case #2* prints **"You have 1 item in the bag:"** \n **"jelly: $3.99 : is taxable"** \n **"**End of List"** *Case #3* prints **"You have 4 items in the bag:"** \n **"milk: $2.50 : is taxable"** \n **"bread: $3.99 : is taxable"** \n **"toast: $3.14 : is taxable"** \n **"meat: $7.99 : is taxable"** \n **"**End of list"**