

1. Basket0 doesn't work. Very buggy. No command works the way it should be intended to.

Basket1 is buggy. It does not take into account null items and duplicate items (different variable name, same item).

Basket2 is very buggy with adding and removing, as it points to null pointers and adds to the ArrayList if what it is pointing to is a null pointer, which means the "if" statements are never true within the "for" loops (both adding and removing methods).

Basket3 is very buggy, as "items" is a static field, but constantly referenced in a non-static manner. Also, the algorithm fails to add cost properly, as it adds only the second element constantly (`this.items.get(1)`).

Basket4 has the proper scope, but everything else is bad. Not only can it handle up to only 10 items, comparing the item using "==" instead of `.equals` is incorrect.

Basket5 is buggy as it struggles with removing items from the list. Also "this.counts.set(index, this.counts.get(index) - 1);" and similar lines with it does not make sense, and doesn't work.

Basket6 passes all tests, and should function properly to a reasonable degree.

Basket7 doesn't account for null variables, but it might not be buggy as this is how the user wants to implement the program.

Basket8 is buggy as it fails to account for different fields with the same properties when items are added.

Basket9 is buggy the same as basket8, except that Basket9 is able to handle null variables.

Basket10 is buggy when similar variable are added, and it struggles with adding and removing items, especially items with different field names with the same properties are added.

Basket11 is when different fields, same property variables are added. Or let's say I try adding 2 "item1" into the basket, and it would function improperly.

Basket12 is buggy when different fields, same property variables are added, and also fails to `removeAllFromBasket(Item i)` properly, as it removes all items, not just the item listed in the parameter.

2. Three baskets I am choosing are Basket8, Basket9, and Basket12. All baskets fail my “addedHasCount2()” test methods, where the method adds 2 identical items, just with different field names. Then, I would check to see if calling the `countItem` method would return a value of 2 with either field names, as they are the same items, just with a different field name. All three baskets failed this test, but Basket8 and Basket9 differ in the fact that Basket9 accounts for “null” items, but Basket8 does not. Unlike Basket8 and Basket9, however, not only does Basket12 fail the identical items test, it also contains a buggy `removeAllFromBasket` method. The proper implementation of the method would be to remove all instances of the item called upon in the parameter. But, Basket12 removes all items in the `ArrayList` with the `removeAll()` method.