1. Create a shopping cart program as detailed below:
2. class item as follows:

* def \_\_init\_\_(self, n, p, q) where n is the name of the item (e.g., apple, bread, etc), p is the price of the item and q is the quantity of the item
* def show(self) which prints the name and price of the item
* def setPrice(self, g) because you may want to change the price of an item later on
* def setQuantity(self, q) because you may purchase more than one unit of a given item (i.e., you may want to change the quantity of an item later on)
* def getName(self) – accessor method
* def getPrice(self) – accessor method
* def getQuantity(self) – accessor method

1. class shoppingCart as follows:

* def \_\_init\_\_(self, items) where items is a **list** of objects from class item. Note that this is a list of objects**, NOT** a list of n-tuples. If you are not sure, **ASK!!!!**
* def show(self) – which prints all the items (and their respective prices & quantities) that are in the shopping cart. Note that you will need to process the list – if not, you did something wrong. If you are not sure, **ASK!!!!**
* def findItem(self, i) – checks to see if the given item is in the list of items. If so, it returns the position of the item in the list (so that, for example, we can add more of that item to the shopping cart). If not, it returns -1 (or an error message). There are alternative ways to do this, which are acceptable as well but check with me as well.
* def addItem(self, i) – checks to see if the given item i is in the list of items. If not, this method will then append the item object to the list items. If it is, it adds the quantity of item i to the existing quantity of the item in the list.
* def deleteItem(self, i) - checks to see if the given item is in the list of items. If it is, this method will then delete the item from the list of items. If it is not, it will print a specific error message.
* def checkOut(self) – calculates the cost of all items in the shopping cart and prints that cost out.