MaxValue

```
import java.util.ArrayList;


public class MaximumFinder {
   public static Integer findMaximum(ArrayList<Integer> numbers) {
      if (numbers == null || numbers.isEmpty()) {
         return null;
      }

      Integer largest = numbers.get(0);
      for (Integer number : numbers) {
         if (number > largest) {
            largest = number;
         }
      }
      return largest;
   }
}
```

SortList

```
import java.util.ArrayList;
import java.util.Collections;


public class ListSorter {
   public static void arrangeAscending(ArrayList<Integer> numbers) {
```

```
        if (numbers != null) {
            Collections.sort(numbers);
        }
    }
}
```

GroceryItemOrder

```
public class ShoppingItem {
    private String itemName;
    private int itemQuantity;
    private double unitPrice;

    public ShoppingItem(String itemName, double unitPrice) {
        this.itemName = itemName;
        this.unitPrice = unitPrice;
        this.itemQuantity = 1; // Default quantity is 1
    }

    public double calculateCost() {
        return itemQuantity * unitPrice;
    }

    public void updateQuantity(int itemQuantity) {
        this.itemQuantity = itemQuantity;
```

```
    }

    public String getItemName() {
        return itemName;
    }

    public int getItemQuantity() {
        return itemQuantity;
    }

    public double getUnitPrice() {
        return unitPrice;
    }
}
```

GroceryListTest

```
public class ShoppingListDemo {
    public static void main(String[] args) {
        // Create shopping list
        ShoppingList list = new ShoppingList();

        // Create shopping items
        ShoppingItem milk = new ShoppingItem("Milk", 1.50);
        milk.updateQuantity(2);
```

```
        ShoppingItem bread = new ShoppingItem("Bread", 2.00);
        bread.updateQuantity(3);
        ShoppingItem eggs = new ShoppingItem("Eggs", 0.50);
        eggs.updateQuantity(12);

        // Add items to the shopping list
        list.addItem(milk);
        list.addItem(bread);
        list.addItem(eggs);

        // Display the total cost of the shopping list
        System.out.printf("Total cost of the shopping list: $%.2f%n", list.calculateTotalCost());
    }
}
```