**Assignment 2**

**Instructions**

For this assignment, you will be creating a program that simulates an eCommerce

store. You will be creating classes to represent the different types of items that the

store sells and the customers who purchase them. Your program should have the

following classes:

1. Item: This is the base class that all other item classes will inherit from. It should have the following properties:

* name: A string representing the name of the item
* price: A float representing the price of the item.
* quantity: An integer representing the number of items in stock.
* category: A string representing the category the item belongs to (e.g. "clothing", "electronics", etc.).

2. Customer: This class should have the following properties:

* name: A string representing the name of the customer.
* email: A string representing the customer's email address.
* purchases: A list of Purchase objects representing the items the customer has purchased.

3. Purchase: This class should have the following properties:

* item: An Item object representing the item that was purchased.
* quantity: An integer representing the number of items that were purchased.

Your program should also have the following five reports:

1. **TopSellingItemsReport**: This report should display the top 10 selling items, sorted by quantity sold in descending order.
2. **TopCustomersReport**: This report should display the top 10 customers, sorted by total amount spent in descending order.
3. **CategorySalesReport**: This report should display the total sales for each category, sorted by total sales in descending order.
4. **CustomerPurchaseHistoryReport**: This report should display the purchase history for a given customer, including the name of each item purchased and the total amount spent on each item.
5. **InventoryReport**: This report should display the current inventory levels for all items in the store, sorted by quantity in stock in descending order.

**Each report should be created as a separate static class that will expose method generate().**

**All classes should be covered with unit tests.**

**Each report should be created with threads as well.**