CSF202 – Lab Class 9 – Wednesday 18/11/2020

This lab class involves getting some hands-on experience with coding an example of the observer design pattern in Java.

□ Task 9.1

For this task a working program is provided. Download it from the following link:
https://github.com/tomblanch118/CSF202/blob/master/Code/Labs/Lab%209/Lab9.zip

This zip file contains a small Basket/Item program that currently partially implements the observer pattern. Your task is to complete the implementation.

The main idea:

- A basket is created.
- Items are added to a basket.
- As each item is added, the basket keeps track of the total (price) of the items in the basket.
- **The problem:** If an item changes its price while it is inside the basket then the basket's internal total value will be out-of-sync and contain invalid data.
- **The solution:** The observer pattern. The basket can observe items. An item can notify observers when its name or price changes. When a basket gets a notification of a change in an item it can recalculate the total price (hence keeping the value in sync).

The Basket class already has some implementation of the observer pattern. The Item class has none. A Main class already contains a main method.

You do NOT need to create your own Observer/Observable interface/class. Please use the Java's built-in versions.

What you should do:

1. First, you should compile and run the program. Understand the output. Notice that after an item is added the total of the basket is out-of-sync and is incorrect.
2. Understand the program.
3. Make Item extend the Observable class
    - You will need to use **notifyObservers()** and **setChanged()** to notify observers when the Item's state has changed).
4. Make the basket observe each item added to the basket. (You should probably stop observing items if they are removed from the basket too).
5. Implement the Baskets update method.
    - This will involve looping through all the items and recalculating the totalPrice.
6. Run the program and check the output. The baskets total should now stay in sync after items change their price.