CS 449 Assignment #2

This assignment focuses on refactoring and automated testing. The problems are based on the Java code given below. It is part of a sales system that supports discount policies.

```
public class Sale {

    private static double DISCOUNT_RATE = 0.05;

    private ArrayList<Item> itemList = new ArrayList<Item>();

    public void addItem(Item item) {
        itemList.add(item);
    }

    public double getDiscountRate(){
        return DISCOUNT_RATE;
    }

    public double getDiscount(){
        return getSubTotal()*getDiscountRate();
    }

    public double getSubTotal() {
        double subTotal = 0.0;
        for (Item item: itemList) {
            subTotal += item.getItemTotal();
        }
        return subTotal;
    }

    public double getDiscountedSubTotal() {
        return getSubTotal() – getDiscount();
    }

}
```

This assignment requires the use of a unit test framework (e.g., JUnit). For each problem, you must provide a description about your work in addition to the source code. Handwritten answers will not be accepted. The source code for each problem should be provided as a separate package. Turn in your work as a single zip file via Canvas.

If you want to use a different object-oriented programming language, it is your responsibility to translate the given code without changing the functionality.

1. Complete the Item class to make the program runnable. The instance variables of Item should at least include title, original unit price, and quantity **(5 points)**. Also write a test class for

each of the methods *getDiscount*, *getSubTotal*, *getDiscountedSubTotal* so that the Sale class is fully tested (**5 points).**

2. Update the Sale class to support different discount rates (e.g., 10% and 15%) for seniors and preferred customers, in addition to the discount rate in (1) for any customer **(15 points)**. Update your test code with appropriate new tests (**5 points**). Your program must use switch statement(s), otherwise you will not receive any credit for the problems (3)-(5).

3. Update your code for (2) to support that the senior's discount is only on Tuesday **(15 points)**. Update your test code with appropriate new tests **(5 points)**. Your program must keep the switch statement(s), otherwise you will not receive any credit for the problems (3)-(5).

4. Refactor your code for (3) to eliminate the switch statement(s) by using class hierarchy and polymorphism (**30 points)**. You will receive no credit if any switch statement is simply changed to if-then-else statements.

5. Update your code for (4) to handle the new discount policy "buy one, get the second (and the rest of the same item) 50% off" for certain items **(15 points)**, and add new tests for this new functionality. (**5 points)**