|  |  |
| --- | --- |
| **CSE1206 : Object Oriented Programming Lab Fall 2018** | **Set- A** |

**Online: 2 Date: 24 January, 2019 Group: C1 Time: 35 minutes**

**ID:**

**Marks**

|  |  |
| --- | --- |
| 1. Create a Java Project Named **‘ItemStore’**. Inside this project (folder) create a new class named **‘Item’**. (There should be two classes: **ItemStore** [the default class], **Item** [the newly created class]. And the two classes should be in the same package). | **1** |
| 1. Inside the **Item** class declare **3** **private** variables: **itemName (String), itemPrice (double), itemDiscount (double).** | **1** |
| 1. Declare **2 public** Constructorsin **Item Class:**     1. A consturctor that takes no parameter and initializes the variables as:   itemName = “Null”  itemPrice= -1  itemDiscount = -1   * 1. Another Constructor which takes all the variables as parameter and assigns those parameters to the class variables. | **2+2** |
| 1. Declare necessary **public** getter() and setter() methods for all the 3 variables. Set the **itemPrice** using its **setter method** in such a way that it can never be less than or equal to zero. If zero or less than zero is assigned then set it to 1. | **1+2** |
| 1. Declare a **public** method called **calculateDiscount()** in **Item Class** that returns a **double** value and takes no parameters. The method calculates the final price of the item after applying the discount and returns it. For example: If price of an item is 800 and the discount is 50% then discounted price is 400. | **3** |
| 1. Declare another **public** method called **compareDiscount()** in **Item Class** which takes the **Item object** as parameter and returns a **String** variable. This method compares the final price of two separate items after calculating the discount and returns the following Strings:  * If price of both items are same then return “equal” * If price of item-1 is greater than item-2 then return “greater” * If price of item-1 is lesser than item-2 then return “lesser”   (Item-1 = Item Object of Current Class , Item-2 = Item Object of the Class in the Parameter) | **3** |
| 1. Now test the methods of **Item** Class in the **ItemStore** Class. Take user input for item Names, item Prices and item Discounts. Check the final prices obtained after applying discount to two items and print those with the particular item Names.   Then set the discount of the item-1 to 37% using the setter method and again compare the discounted price of item-1 and item-2. | **5** |

**Total: 20**

|  |  |
| --- | --- |
| Sample Input 1 | Sample Output 1 |
| Enter Item Name 1: Bag  Enter Item Price 1: 1500  Enter Item Discount 1: 40  Enter Item Name 2: Shoe  Enter Item Price 2: 3000  Enter Item Discount 2: 65 | Discounted Price for Bag is 900.0  Discounted Price for Shoe is 1050.0  The price of Bag is lesser than Shoe  After Changing the Discount for Bag:  New Discounted Price for Bag is 945.0.  The price of Bag is still lesser than Shoe. |

|  |  |
| --- | --- |
| Sample Input 2 | Sample Output 2 |
| Enter Item Name 1: Bag  Enter Item Price 1: 1500  Enter Item Discount 1: 40  Enter Item Name 2: Shoe  Enter Item Price 2: 2000  Enter Item Discount 2: 55 | Discounted Price for Bag is 900.0  Discounted Price for Shoe is 900.0  The price of Bag is equal to Shoe.  After Changing the Discount for Bag:  New Discounted Price for Bag is 945.0.  The price of Bag is greater than Shoe. |

**Hints:**

\*user Scanner class to take input:

for string use nextLine() , for double use nextDouble()

\*When you wrtie String a= sn.nextLine()

String b=sn.nextLine()

After taking the first input when you press enter the **enter (“\n”)** that is the newline will be taken as the input for String b which you don’t want. To solve this problem you can insert another sn.nextLine() between them like this:

String a= sn.nextLine()

Sn.nextLine()

String b=sn.nextLine()

So after you take the first input and press enter the middle nextLine() will take the newline input and then won’t effect your String b input. Do this after each String if you have more inputs.