IT Workshop -II (CS251)

Lab Assignment : 5

Array of Objects: You can create an array of objects of a class (say Student) with the following code: Student[] studentArray = new Student[size];

After that you can use any object of the array like a normal object. As for example, if you want to call a method getGrade() of 3rd object of the array write: studentArray[2].getGrade();

Q-1) Create the following classes:

public class **Items**

`int itemCode;`
`String itemName;`
`double unitPrice;`
`int stockRemaining;`
`int itemLimit;`

+ `public Items(int itemCode, String itemName, double unitPrice, int stockRemaining, int itemLimit)` + public String toString()

public class **User**
`String name;` `double budget;`
+ `public User(String name, double budget)`

Create a menu driven application for an electronics shop by doing the following operations:

I. Create a class Shop and do all your work from main method of this class.
II. Create an array of **Items** objects with following description:

itemCode	itemName	unitPrice	stockRemaining	itemLimit
1	TV	10000	3	2
2	Mouse	1000	7	3
3	HeadPhone	2000	1	6
4	Mobile	5000	5	1

III. Create a **User** object with any name and any initial budget.
IV. Give the following options to the user (and do this operation in a loop):

1. Display Items
2. Buy Item
3. Exit

Note : Throw **User Defined Exception** (with appropriate message) where ever necessary.

V. If user selects option 1., then display all items information **implicitly** by calling their toString() method and go to step IV.

VI. If user selects option 2. then ask him for item code and quantity (to be bought). Decrement the item stock and user budget accordingly and go to step IV.

VII. If user selects option 3. then say "Thank You" and exit the application.

User Defined Exceptions:

Handle the following **user defined exceptions** in Step IV (with option 2.)

1. **ItemNotFound:** If the item doesn't exist in the shop i.e. item code entered by the user is not present in the item list
2. **OverBudget:** If user doesn't have the required money to buy all quantity of the entered item.
3. **ItemLimit**: if user wants to buy a quantity of an item which is greater than prescribed item limit.
4. **OutofStock:** if user wants to buy a quantity of an item which is greater than it's remaining stock.