Object-Oriented Programming II (CS 116 - 03)
Class Project[1]
This is an individual assignment. You will work on this assignment by yourself
Design and code due in Blackboard Thursday, April 30th, 11:59 PM

---

1 Grade

1.1 Design (20 Points)

- The design consists of a pdf document that should be included with the source code and the compiled files.
- See further down for instructions on Design Document Content. Due together with the source code and compiled code files.

1.2 Source Code (80 Points)

Source code and compiled files are due April 30th, 2020 by 11:59pm on Blackboard.

1.3 Submission Instructions

zip all files and name it `CS116_LastName_FirstName_Project.zip` then upload on Blackboard under Project assignment folder.

Description

2 Office Supply Order System

You will design a system to handle office orders. A supply order consists of the following attributes:

- Unique order ID (numeric)
- Customer ID (alpha)
- Product ID (alphanumeric)
- Date of the order
- Order amount

Note that there are two types of orders, namely "one-time" or "repeated" order.

- For "repeated" orders, a supply order also has a period (the number of days between repeats), and an end date for the repeating.

The `Office Supply Order System` needs to allow a `Order Manager` to manage a set of supply orders:

- to be able to add an order (and assign a unique order ID),
- to delete an order (given the order ID),
- to list all the orders for a particular customer ID in increasing date sorted order.

[1] Matthew Bauer & George Koutsogiannakis, Illinois Institute of Technology

Finally, the `Office Supply Order System` needs to be able to calculate and report for each month on the inventory needed for each product ID based on the current supply orders.

Each class for the `Office Supply Order System` needs its own individual test program. Each class for the `Office Supply Order System` needs its own individual exception class and should throw exceptions for any invalid action. Other classes need to catch these exceptions where appropriate.
Create a `UserInterface` class for your `Office Supply Order System`. This should be the project `void main` that allows the user these options:

1. specify a file name of orders to load
2. specify the information to add an order or repeated order
3. specify an orderID to delete
4. specify a CustomerID to display a list of orders for that customer in increasing date sorted order
5. calculate and output an order inventory report (sorted by year, month and productID).

Below is a sample input file for the project `orderssmall.txt` with these fields comma delimited. `O` or `R` (onetime or repeated order), customerID, productID, orderDate, orderAmount, and if a repeated order, period and endDate.

```
O,Orange Inc.,A1,6/25/2004,5
O,Macrohard Inc.,C2,8/2/2004,1
R,Peak Enterprises,A2,11/18/2006,10,7,12/21/2006
R,Orange Inc.,B1,11/25/2004,4,30,12/28/2004
R,HAL Industries,B1,9/23/2009,1,7,9/24/2009
R,Peak Enterprises,C3,11/4/2006,5,7,12/28/2006
R,HAL Industries,B1,5/11/2007,1,7,6/8/2007
O,Orange Inc.,C2,1/2/2004,1
R,Peak Enterprises,A1,2/15/2002,4,7,3/25/2002
O,Macrohard Inc.,A1,6/3/2003,2
R,Orange Inc.,A2,2/20/2007,2,30,4/20/2007
R,HAL Industries,C1,8/21/2002,8,7,8/23/2002
R,Giggle Industries,A2,2/16/2008,6,1,2/24/2008
R,Giggle Industries,B1,2/3/2003,8,1,2/24/2003
R,Giggle Industries,C2,3/11/2004,6,1,4/3/2004
R,HAL Industries,C2,6/22/2003,6,7,7/29/2003
O,Macrohard Inc.,A2,3/24/2009,2
R,Giggle Industries,A2,11/9/2005,4,1,11/30/2005
R,Orange Inc.,A2,10/6/2006,6,30,11/7/2006
O,Macrohard Inc.,A1,2/15/2005,2
O,HAL Industries,C2,4/21/2005,10
R,Orange Inc.,B1,3/22/2001,6,30,4/25/2001
O,Orange Inc.,A2,10/26/2005,4
R,Peak Enterprises,C2,5/26/2003,10,7,7/24/2003
R,HAL Industries,A2,12/20/2008,1,7,2/11/2009
O,Orange Inc.,B1,11/20/2008,3
R,Giggle Industries,A1,6/11/2004,9,1,6/27/2004
O,Orange Inc.,A1,12/14/2002,4
O,HAL Industries,B1,8/6/2003,10
O,HAL Industries,A1,11/2/2002,1
O,Orange Inc.,C3,4/24/2007,7
O,Macrohard Inc.,C1,6/25/2001,6
R,Giggle Industries,A1,9/25/2002,7,1,11/22/2002
O,HAL Industries,A2,6/1/2005,6
R,Peak Enterprises,A1,1/7/2003,1,7,1/17/2003
R,Peak Enterprises,A1,11/13/2001,10,7,1/4/2002
```

You will need to write an input file generator that will generate `random` input files. This should be its own "void main". Assume:

- customerIDs comes from the current NYSE Company `companylist.txt`
- productIDs are all a single capital letter followed by a single numeric digit
- orderDates are all in 2019
- orderAmounts are all integers less than 150
- repeatedOrderPeriods are all 1 to 30 days
- repeatedOrderEndDates all in 2019

2.1 Design Requirements

Create the Class Diagram for the above requirements. Your design document is pdf document with the name `CS116Design.pdf`. It should have the following information:

- A list of all classes and their type.
- Interfaces if any.
- For each class give a short description of what the class does. Identify the class as abstract if it is abstract. For each abstract class identify the abstract methods in that class.
- In interfaces, identify the abstract methods.
- A relationships diagram.
  - Draw a box for each class and then draw a line or an arrow from that box to another classes follows:
    - If there is an inheritance via extends draw an arrow from the superclass to the subclass and label the arrow as extends or implements.
    - If one class uses an object of another class (without an inheritance relationship) draw a line between them and label the line as uses.

2.2 Coding Requirements

Code and test your `Office Supply Order System`

2.3 Extra Credit

You will have up to `20%` extra credit bonus if you implement you project using Graphical User Interface.