

CST8110 – Introduction to Programming

Assignment #2 – Using interacting classes

DUE: FRIDAY October 25 at 6pm SHARP – see submission instructions below. Late submissions receive a grade of 0.

Problem Description:

Generate for the following problem:

- a) test plan AND
 - b) write and test the program code in Java.
- This problem will build on our assignment which simulates producing an invoice for a Internet Providing company (ie a company that provides Internet connection and services).
 - Use your solution from Lab #3 and Lab #4 (or my posted solution to Assignment #1) as the starting point for this assignment. For now, the invoice will only handle one month for a client.
 - The new functionality to be added to this program is indicated in red.
 - All of the calculations/functionality from Assignment #1 will remain.
 - You will **WRITE** a **OurDate** class to handle the date functionality for this invoice – don't use the Java class. NOTE – we will use this class in a totally different program in another assignment (which will illustrate the advantage to using classes – reusability of code). Date data will consist of day, month and year. The class does NOT need to handle leap years. We will then use two objects of this OurDate class in our IPBilling class – today's date and invoice date.
 - Your program should prompt the user to enter today's date and the invoice date then the invoice should display the number of days the invoice has been due to be paid. If the number of days overdue is over 60, you should add 10% interest to the bill (calculated after all the bill has been calculated – including taxes).

Suggested design for Date class:

- **Three private integer fields to store year, month (1-12) and day (1-31)**
- **Methods:**
 - **Constructor – default date of Jan 1, 1900**
 - **Constructor – using three integers to set initial values for year, month and day**
 - **setYearFromUser – prompt user to enter a year and edit it; don't allow them to enter an invalid year**
 - **setMonthFromUser – prompt user to enter a month and edit it; don't allow them to enter an invalid month**
 - **setDayFromUser – prompt user to enter a day and edit it; don't allow them to enter an invalid day**
 - **displayDate – will display the date in mm/dd/yyyy format**
 - **calcDays – returns an int which is the number of days from 1/1/0000**

Sammples Output #1: (blue indicate user entered information)

Enter your name: joe blow

Enter your address: 92 algonquin ave ottawa

Enter the invoice date: Enter a valid year: 2012

Enter the month: 3

Enter the day: 23

Enter todays date: Enter a valid year: 2012

Enter the month: 3

Enter the day: 24

Enter the number of gigabits per month you wish to purchase: 10

Date:2012/3/24

Invoice Date: 2012/3/23

Client Info: joe blow 92 algonquin ave ottawa

Billing for 10.0GB

Invoice Amount: \$62.50

Sales Tax: \$8.13

Interest: \$0.00

Total Due: \$70.63

Sample Output #2: (blue indicate user entered information)

Enter your name: Linda Crane

Enter your address: 88a main street

Enter the invoice date: Enter a valid year: -56

Error - Enter a valid year: 2012

Enter the month: 13

Error - enter the month: 0

Error - enter the month: 1

Enter the day: 32

Error - enter the day: 15

Enter todays date: Enter a valid year: 2012

Enter the month: 3

Enter the day: 24

Enter the number of gigabits per month you wish to purchase: 0

Date:2012/3/24

Invoice Date: 2012/1/15

Client Info: Linda Crane 88a main street

Billing for 150.0GB

Invoice Amount: \$50.00

Sales Tax: \$6.50

Interest: \$5.00

Total Due: \$61.50

Note: you were charged 10% interest for overdue account

Submission Requirements:

- You must create a .zip file that contains ONLY the following:
 - Your program code - .java files - (with your name, section, lab teacher listed in comments in the header of the program)
 - Your compiled code - .class files
 - A document created with either Notepad, Wordpad or Word named Assign4.docx or Assign4.txt with your test plan - note this should contain your name, section and lab teacher listed at the top.
- The .zip file must have the following as it's name
 - Your last name, your first name, the word assign and the assign number
Example CraneLindaAssign2.zip
- Submit the .zip file through the Assignment feature which has been enabled in the CST8110 Blackboard course. This should be directly under the Assignment description and will be published as soon as the due date for Assignment #2 has passed.
- Marks will be given for correct submission (ie marks will be deducted for incorrect submission!)