Concordia University
Comp 249 - Object Oriented programming II
Winter 2015
Assignment 3

Deadline:	By 11:59pm, Friday March 20, 2015.
Evaluation:	5% of your final grade.
Late Submission:	No late submission.
Teams:	The assignment can be done individually or in teams of 2 persons (from the same lecture section). Submit only one assignment per team.
Purpose:	The purpose of this assignment is to apply in practice the notions of file I/O, exception handling and abstract classes. In addition, the student will also reinforce the notions of inheritance and arrays.

- **Problem specification**

The file ***payroll.txt*** stores payroll information about the employees of the company *Harbour Industries* for the year 2014. Each line of the file specifies the following fields ***: employee number*** *(long type)*, ***employee first name*** (*String type*), ***employee last name*** (*String type*), ***hours worked*** (*double type*), and ***hourly wage*** (*double type*). Each field is separated by at least a space in the input file.

The payroll file compiles the average number of hours worked per week during the year by each employee and it needs to be processed in order to calculate the weekly gross salary, net salary and tax deductions. Tax deductions are for provincial income tax, federal income tax, RRQ (régime des rentes du Québec), EI (employment insurance), and QPIP (Québec Parental Insurance Plan).

- **Implementation**

You are required to write a Java Program that calculates the salaries and deductions for the employees, and that produces reports.

At start up, the program reads the input file and saves the input fields ***employee number***, ***employee first name, employee last name*** and the calculated field ***annual gross salary*** in an array of employee objects (i.e. you must create a class Employee). The weekly gross salary (*number of hours worked multiplied by the hourly wage*) must be multiplied by 52 in order to obtain an annual gross salary. Therefore, all access to the payroll file will now be achieved using this array.

Input lines of the payroll file with invalid data must be written to an **error file** (*say payrollError.txt*) and thus are not part of the array. Invalid inputs are those with inappropriate input types and hourly wages less than $10.35.

Inappropriate input types should be handled with the specific **Java Exception classes**. You must also implement an exception class (e.g. **MinimumWageException**) to validate hourly wages that are under the legal minimum wage.

Once the array is populated with the payroll file data, the program then calculates the appropriate deductions for each employee. Deductions must be implemented as a base class from which the classes for ***ProvincialTax***, ***FederalTax, RRQ***, ***EI*** and ***QPIP*** are derived. The ***Deductions class*** must be ***abstract*** and must include at least the ***calculateTax abstract method***. Each derived class must therefore implement the ***calculateTax*** method because each deduction has a different calculation method.

- Calculating EI premiums.
  - The EI premium rate is 1.53 for each $100 of gross salary.
  - The maximum premium deducted is $743.58 and is based on an annual gross salary of $48 600.
- Calculating QPIP premiums.
  - The QPIP rate is 0.559% for salaried workers, 0.782% for employers and 0.993% for self-employed workers. Harbour industries only have salaried employees.
  - The maximum premium of $385.71 is based on an annual gross salary of $69 000.
- Calculating RRQ premiums.
  - The RRQ rate is 5.175% multiplied by the gross salary.
  - The maximum premium is $2535.75 based on an annual gross salary of $52 500.
- Calculating provincial income tax.
  - 16% rate applies to annual taxable income of $41,495 or less;
  - 20% rate applies to annual taxable income of more than $41,495, but not more than $82,985;
  - 24% rate applies to annual taxable income of more than $82,985, but not more than $100,970;
  - 25.75% rate applies to annual taxable annual income of more than $100,970.
- Calculating federal income tax.
  - 15% on the first $43,953 of annual taxable income;
  - 22% on the next $43, 954 of annual taxable income (on the portion of taxable income over $43,953 up to $87,907);
  - 26% on the next $48,363 of annual taxable income (on the portion of taxable income over $87,907 up to $136270);
  - 29% of annual taxable income over $136,271.

Once the deductions are calculated, the program is now ready to print a **report file** (say payrollReport.txt). The report must be formatted with columns and column headings. Each data line must display the ***employee number***, ***first name***, ***last name***, ***annual gross salary***, ***Deductions*** (Quebec income tax + Federal income tax + RRQ + QPIP + EI) and ***net salary***.

- **Test cases**

> Opening file payroll …

>Reading file payroll …

>Error lines found in file payroll
52726 MITCHELL HACKETT 23,7 12.05
#9331 MARIO TODARO 35.0 17.00
22310 CLAUDIA O'HARE 30.5 9.80
26734 EDITH ROOSEVELT 20.O 25.50
41024 EMILE BOURGEOYS 8.5 6.45
43018 JAMES WALKER 20.0 8.00
00812 VICTORIA PORTER 36.0 9.50
9201(DAVID BROCK 29.0 22.50

> 71 lines read from file payroll

> 8 lines written to error file

> Calculating deductions

> Writing report file

> Report file

Harbour Industries

Employee Number	First name	Last Name	Gross salary	Deductions	Net salary
31718	PHILLIP	LENNOX	$57 200.00	$22 886.15	$34 313.85
11528	NANCY	TROOPER	$21 736.00	$8 317.06	$13 418.94
…….	……..	……..	……..	…….	…….

- **Evaluation criteria.**

Criteria	Marks
Programming style and comments	5%
Employee class	10%
Abstract base class Deductions	5%
Derived classes of class Deductions	25%
Main class	15%
Processing of payroll file, payrollError file and payrollReport file	15%
Exception handling (file open, input data validation), MinimunWageException class	10%
Output files (payrollError and payrollReport)	15%

- **Required documents**
    - Payroll, error and report files.
    - Test cases.
    - Java class files.

- **Submission.**
    - Create one zip file, containing the necessary files (.java, payroll.txt, payrollError.txt, payrollReport.txt, test cases). If the assignment is done individually, your file should be called *a3_studentID*, where a3 is the number of the assignment and *studentID* is your student ID number. If the work is done in a team of 2 people, the zip file should be called *a3_studentID1_studentID2* where *studentID1* and *studentID2* are the student ID numbers of each student.
    - Upload your zip file on eas (electronic assignment submission) at the URL: https://fis.encs.concordia.ca/eas/ as *Programming Assignment 3* before midnight on the due date.