Program 1: 40 points: Inheritance Due: Thursday, September 17th, 11:59 p.m.

**Objectives: The focus of this assignment is the use of inheritance. You will need to understand and implement the ideas of basic inheritance.**

**Program Description:**

This project will have you create a simple class hierarchy building upon the Employee example discussed in class.

A total of five classes are required.

**Employee** – A class which the other employee types inherit from

**HourlyEmployee** – An employee whose pay is based upon an hourly wage and hours worked

**SalaryEmployee** – An employee whose pay is based upon a yearly salary

**CommissionEmployee** – An employee whose pay is based upon a commission rate and sales amount

**EmployeeTest** – Contains main method. Creates one of each sub type of Employee and performs actions upon them. This is provided to you. DO NOT MAKE CHANGES

**UML DIAGRAM FOR AND DISCUSSION FOR Employee**

|  |
| --- |
| Employee |
| - String firstName  - String lastName  - char middleInitial  - boolean fulltime  - char gender  - int employeeNum |
| <<constructor>> Employee (fn : String, ln : String, m : char, g : char, empNum : int, ft : boolean )  + getEmployeeNumber() : int  + setEmployeeNumber(empNum : int)  + getFirstName() : String  + getLastName() : String  + getMiddleInitial() : char  + getGender() : char  + setFirstName(fn: String)  + setLastName(ln : String)  + setMiddleI(m : char)  + setGender(g : char)  + equals(e2 : Object) : boolean  + toString() : String |

Notes on Data Members

Your constructor should call upon the set methods for the data members it is initializing. This is especially important for employeeNum and gender since there is validation required.

The employeeNum member must be between 10000 and 99999, inclusive. If an invalid value is passed, the Employee class should immediately ask for another number until an acceptable one is given. This should be handled in setEmployeeNumber.

If an invalid value for gender is given (not ‘M’ or ‘F’, *specifically uppercase*) it should default to ‘F’. This should be handled in setGender.

Notes on Methods

equals() – Overrides Object equals(). Returns true if the employeeNum of the two instances are equal, false otherwise.

toString() – Overrides Object toString(). Returns as String of the Employee in the following format:

12345

Doe, John M.

Gender: M

Status: Full Time

(Note that Status doesn’t say true or false, rather Full Time or Part Time)

**DISSCUSSION ON Employee SUBCLASSES**

**HourlyEmployee**

|  |
| --- |
| HourlyEmployee extends Employee |
| - double wage  - double hoursWorked |
| <<constructor>> HourlyEmployee (fn : String, ln : String, m : char, g : char, empNum : int, ft : boolean, w : double )  + increaseHours(hours : double)  + toString() : String  + calculateWeeklyPay() : double  + annualRaise()  + holidayBonus() : double  + resetWeek() |

Additional Data Members:

* double wage
* double hoursWorked

Methods

Constructor accepts all that an Employee requires as well as a double for wage, hoursWorked set at 0.0.

Override toString(), returns a String of the HourlyEmployee in the following format:

12345

Doe, John M.

Gender: M

Status: Full Time

Wage: 3.40

Hours Worked: 0.00

calculateWeeklyPay() – Return amount earned in the week using wage and hoursWorked, any hours worked over 40 give double pay

annualRaise() – Wage is increased by 5%, round down to 2 decimal places

holidayBonus() – Return amount of 40 hours worked (40\*wage)

resetWeek() – Resets hours worked to 0

increaseHours() - This class also needs the ability to increase the hours worked. Requesting to increase by a negative value should give no change, and report an error to the user.

**SalaryEmployee**

|  |
| --- |
| SalaryEmployee extends Employee |
| - salary : double |
| <<constructor>> SalaryEmployee (fn : String, ln : String, m : char, g : char, empNum : int, ft : boolean, s : double )  + toString() : String  + calculateWeeklyPay() : double  + annualRaise()  + holidayBonus() : double  + resetWeek() |

Additional Data Members

* double salary

Methods

Constructor accepts all that an Employee requires as well as a double for salary.

Override toString(), returns a String of the SalaryEmployee in the following format:

12345

Doe, John M.

Gender: M

Status: Full Time

Salary: 50000.00

calculateWeeklyPay() – Return amount earned in the week by dividing salary by 52

annualRaise() – Salary is increased by 6%, round down to 2 decimal places

holidayBonus() – Return 3% of salary

resetWeek() – No change

**CommissionEmployee**

|  |
| --- |
| CommissionEmployee extends Employee |
| - sales : double  - rate : double |
| <<constructor>> CommissionEmployee (fn : String, ln : String, m : char, g : char, empNum : int, ft : boolean, r : double )  + increaseSales(sales : double)  + toString() : String  + calculateWeeklyPay() : double  + annualRaise()  + holidayBonus() : double  + resetWeek() |

Additional Data Members

* double sales
* double rate (stored as a percent, eg. 3.5% would be stored as 3.5)

Methods

Constructor accepts all that an Employee requires as well as a double for rate, sales set to 0.0.

Override toString(), returns a String of the CommissionEmployee in the following format:

12345

Doe, John M.

Gender: M

Status: Full Time

Rate: 3.50

Sales: 0.00

calculateWeeklyPay() – Return rate percentage of sales

annualRaise() – Rate percentage increased .2% *example, if rate was 2.5, it becomes 2.7*

holidayBonus() – No bonus

resetWeek() – Reset sales to 0.0

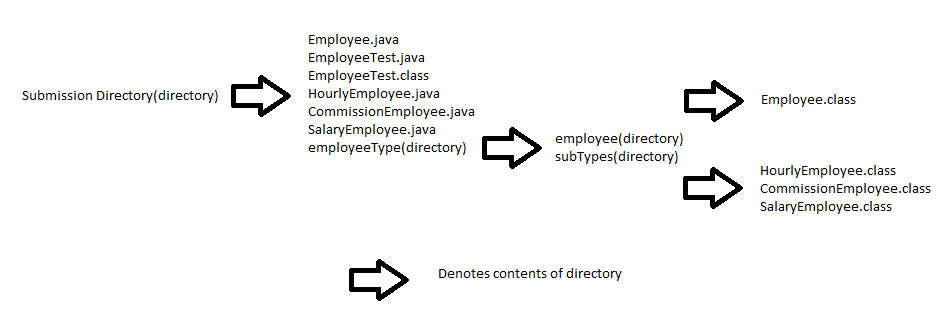
increaseSales() - This class also needs the ability to increase the sales. Requesting to increase by a negative value should give no change and report the error.

**Package Structure:**

The classes you create must follow the given package structure –

* Employee belongs to the employeeType.employee package
* HourlyEmployee, SalaryEmployee, and CommissionEmployee belong to the employeeType.subTypes package
* EmployeeTest does not declare any package

These packages as well as all source files should exist in your submission directory. After all is compiled your directory structure should look like this:



**Other Notes:**

* Make sure monetary values are formatted correctly
* Stored monetary amounts are forced to be 2 decimal places
  + Keep this in mind when applying raises, for example if a wage is 9.50, increasing by 5% gives 9.975, this should be rounded DOWN to 9.97
* Make proper use of super

**Summary of Files**

You will need to five files for this assignment.

* Employee.java
* HourlyEmployee.java
* SalaryEmployee.java
* CommissionEmployee.java
* EmployeeTest.java (provided)

**Required Elements:**

* All outputs to the user must include identifying text.
* The user should be prompted for all inputs.
* Your program file must meet the programming standards defined for this course and contain the appropriate header defined for this course.
* Include Honor Pledge on all files and document all methods

# **Submitting Your Program Files:**

The assignment is automatically collected from your account on the “Loki” machine on the due date. You must put your source code in the correct directory in your account in order for this to work. Your directory name includes the course, section, semester and the assignment number. This is case-sensitive; **all alphas are upper-case**. This directory must be created in your home directory.

Submission Directory:

**CSCI-1620-1-F15-A1**

**Expected Output (Input Underlined):**

Creating HourlyEmployee…

12345

Rogers, Steve A.

Gender: M

Status: Full Time

Wage: 15.34

Hours Worked: 0.00

Creating SalaryEmployee...

54321

Pryde, Kitty X.

Gender: F

Status: Full Time

Salary: 75000.00

Creating CommissionEmployee...

Employee Number invalid, please re-enter:

33333

33333

Storm, Johnny F.

Gender: F

Status: Part Time

Rate: 2.50

Sales: 0.00

Increasing Hourly's hours worked by 50.

Increasing Commissions's sales by 150,000.

Hourly Payout: 920.40

Salary Payout: 1442.31

Commission Payout: 3750.00

Finding total bonus payout...

Bonus Payout is 2863.60

Applying annual raises and resetting week...

12345

Rogers, Steve A.

Gender: M

Status: Full Time

Wage: 16.10

Hours Worked: 0.00

54321

Pryde, Kitty X.

Gender: F

Status: Full Time

Salary: 79500.00

33333

Storm, Johnny F.

Gender: F

Status: Part Time

Rate: 2.70

Sales: 0.00