Programming Assignment #9: Inheritance & Polymorphism

Program Problem: Demonstrate your understanding of Programmer written classes, Inheritance, and over-riding. (problem 11.2)

Write: (The Person, Student, Employee, Faculty, and Staff classes) Design a class named Person and its two subclasses named Student and Employee. Make Faulty and Staff subclasses of Employee. A person has a name, address, phone number, and e-mail address. A student has a class status (freshman, sophomore, junior, or senior). Define the status as a constant. An employee gas an office, salary, and date hired. Use the MyDate class (defined in Chapter 10 exercise 14) to create an object for date hired. A faculty member has office hours and rank. A staff member has a title. Override the toString method in each class to display the class name and the person's name.

Draw the UML diagram for the classes and implement them. Write a test program that creates a Person, Student, Employee, Faculty, and Staff, and invokes their toString() methods

Documentation: You must use a readable, logical, and coherent set of style and formatting rules. You are to stick to the "structured approach" in programming**.** Be sure to comment your code in addition to the required header. Each submission must have a block comment area that includes: Your first and last name, program exercise title, program due date, and the program description.

Submission Details: All submissions are electronic. When you turn in a programming assignment, you must send me a compilable and correctly working copy of the assigned program source code. I will, at my discretion, compile and run (on my own test input) the programs you submit electronically. This is a part of my grading procedure. Your program must work. That means it must compile correctly, run according to specifications, and give correct results. Generally, a program that works will receive at least 40-50 percent of full credit. The rubric used for scoring is visible to you so please review it before you submit your assignment. Submit your source file for your test program this means you are giving me your test program and class files not a link to an online compiler, text file, or executable file. In addition to your java file and class file, include a word document that has your UML diagram (this should be the first page) and screen snips for your test runs name that file **<lastnameInheritance.docx>.**

For full credit, your program must also meet the following criteria:

- Good design, including good algorithms.
- Good form, including documentation, and readability.
- Adequate testing, especially the testing of data boundaries and special cases.

You need to do a good job on all the criteria to receive an "A" on your program.