Inheritance

Inheritance is one of the core concepts of object-oriented programming (OOP) languages. It is a mechanism where you can derive a class from another class for a hierarchy of classes that share a set of attributes and methods. You can use it to declare different kinds of exceptions, add custom logic to existing frameworks, and even map your domain model to a database.

Write a program that creates *Employee* and *ProductionWorker* Classes. The Employee class keeps data attributes for the following pieces of information:

- Employee name
- Employee number

The ProductionWorker class is a subclass of Employee class. The ProductionWorker class should keep data attributes for the following information:

- Shift number (an integer, such as 1, 2, and 3)
- Hourly pay rate

The workday is divided into two shifts: day and night. The shift variable will hold an integer value representing the shift that the employee works. The day shift is shift 1, and the night shift is shift 2. Provide a constructor and the appropriate accessor and mutator functions for the class. Demonstrate the classes by applying a Python list of ProductionWorker objects in your main program.

Extend your program above by adding the following requirements:
In a particular factory, a shift supervisor is a salaried employee who supervises a shift. In addition to a salary, the shift supervisor earns a yearly bonus when his or her shift meets production goals. Add a *ShiftSupervisor* class that is derived from the Employee class. The ShiftSupervisor class should have a member variable that holds the annual salary and a member variable that holds the annual production bonus that a shift supervisor has earned. Provide appropriate constructor, accessor, and mutator functions for the class. Demonstrate the class by applying a Python List of ShiftSupervisor objects in your main program. Make up all values.

Save your *Employee, ProductionWorker*, and *ShiftSupervisor* classes in a file called *myLib.py*. Obviously, your main program will have to import *MyLib* to access those classes.

Create a UML diagram and submit the completed diagram to the class blackboard.

Write a Report Summary
Using Microsoft Word, answer the following eight questions.

1. Did you complete your assignment and did it run without errors?
2. Did your program produce the correct result?
3. Did you test your program thoroughly?
4. How much time did you spend completing your assignment?
5. Did you write the program yourself? Did you get any help from anyone?

6. How did you resolve the issues when you encountered obstacles to completing your program? Did you use Google to get help? Describe how Google was abled or not able to assist you.
7. What did you learn from doing this assignment?
8. Any other information you would like to share with your instructor?

What to submit

1. Submit all your program files (.py file)
2. Submit your program output files
3. Submit your learning report summary