Week 11 Lab 8-1 – Inheritance

Do a simple inheritance exercise plus Chapter 8 Practice Program **3** (write and test a set of ***Vehicle*** and ***Truck*** classes.

A. **Create a set of three classes that use inheritance:**

1. A **Fruit** class that has two instance variables, *private String name* and *private double weight*. Fruit also has a 2-parameter constructor to set those and two *getter* methods for them, plus a *toString* method that returns this:
   Fruit: *name*, weight: *weight* → this overrides **Object**'s *toString*.
2. A **ColoredFruit** class that extends **Fruit** and adds *private String color*, the color of the fruit. ColoredFruit has a 3-parameter constructor that calls Fruit's constructor using **super()** and then sets the color, a *getter* for the color, and a *toString* method that overrides Fruit's *toString*, adding at the end:
   , color: *color* → call Fruit's *toString* & add this extra *String* to it.
3. Finally, an **Apple** class that extends **ColoredFruit** but does not add any instance variables or methods. Apple only has a 1-parameter constructor that takes a *double weight* parameter; Apple's constructor calls ColoredFruit's 3-parameter constructor passing in *name* **"Apple"**, the *weight*, and *color* **"red"**.

Finally, write a test class to create & print objects from these 3 classes.

B. **Do Chapter 8 Practice Program 3, create Vehicle and Truck classes:**

1. Create a base class called *Vehicle* that has the manufacturer's name (type *String*), # of cylinders in the engine (type *int*), and owner (type *Person* from **Week 11 Chapter 8 Source Code** folder).
2. Then create a class called *Truck* that is derived from *Vehicle* and has additional properties: the load capacity in tons (type *double*, since it may contain a fractional part) and towing capacity in tons (type *double*).
3. Give your classes a reasonable complement of constructors (Vehicle: 0- and 3-parameter constructors; Truck: 0- and 5-parameter constructors) and accessor methods (*getters* and *setters* for all instance variables in each class). Truck's constructors should call Vehicle's constructors using **super()**.
4. Write a driver program (no pun intended) that tests all your methods. It will be easier to test the constructors if you write a *toString* method for both classes. You can test the *setters* by calling them in the constructors, and the *getters* by calling them in the *toString* methods; use Person's *getName* to get their name in the Vehicle *toString* method.
   ***Hint*: if you use all of the *setters* in the constructors and all of the *getters* in the *toString* methods then your driver program only has to create objects using the two constructors in each class and print those objects in order to test everything.**