

CSC171 — Lab 3

Objects

The goal of this assignment is to give you experience with object-oriented programming: designing and using classes that represent real-world objects. Because this assignment requires you to write three different Java files, you should **submit your work as a single zipfile which contains three top-level source files: `Person.java`, `Element.java`, and `SmartPhone.java`**. Read these requirements carefully and ensure that your submission meets them for full credit.

For the following questions, “define a class” means the following:

- Create an appropriately-named file containing the Java `class` definition, including the following:
 - A block comment describing the file (and hence the class). Your block comment should include your name, the assignment number (e.g., HW3), and netid/email, as always.
 - Any instance variables, as required by the question
 - Accessor (getter) and mutator (setter) methods for any instance variables
 - Constructors as appropriate or as required by the question
 - A `toString` method that prints instances of the class informatively
 - Any additional methods as required by the question. All methods other than getters and setters must have a descriptive comment before their definition.
 - For this assignment, each class should also contain a `main` method that tests your class by doing the following
 - * Create at least two instances of the class.
 - * Change at least one of them using the class’ methods.
 - * Print all the objects informatively

Questions

1. Define a class **Person** that represents a person. People have a name, an age, and a phone number. Since people always have a name and an age, your class should have a constructor that has those as parameters. Define a `toString()` method which displays all of the relevant information using a format like this: “**Jim Lahey - Age 57 - Phone 9025555555**”.
2. Define a class representing physical elements (like hydrogen, helium, lithium, etc.). Elements have a name, a symbol, an integer atomic number, and an atomic weight (which may be fractional). Your class should have a constructor that sets all the properties. In your test program, create and print several real elements using the real values for their properties. For example, for the element hydrogen your program might print something like this: “**Hydrogen (H) - Atomic No 1 - Atomic Weight 1.008**”. You are free to choose a different output format, as long as all four pieces of information are presented.
3. Define a class that represents a smartphone and in particular its battery. The battery has a capacity in mAH (milliamp-hours). This would be a parameter to your constructor. Several features of the phone consume battery power: the screen, voice calling, wifi, and bluetooth. Additionally, phones will gradually discharge the battery at a small standby rate even when all features are off. Use the following table of power ratings:

screen	500 mA
voice	300 mA
wifi	200 mA
bluetooth	100 mA
standby	10mA

The battery life (in hours) of the phone is its capacity (in mAH) divided by the total power use of all the features that are turned on (in mA). Your class should have a method that computes and returns the battery life given the current settings of the features. The purpose of this question is for you to get practice with objects having both *state* and *behavior*. Hint: you should use boolean instance variables to represent the states of each of the four features outlined above.

Look up a phone's battery capacity (**for example, 2200 mAH**) and write a short main method which demonstrates the features of your class. Your main method should create an instance of a phone with that capacity and print the battery life with all features off. It should then turn some features on and/or off and print the battery life with those settings.