

## General

At the top of the every document that you create (word processing files or source files) include:

```
/**
 * Description of the class or document.
 *
 * @author YOUR NAME
 * @version CS162 Lab #, mm/dd/yyyy (replace with the last edit date)
 */
```

Submit in this lab via Moodle using the “Lab #” link. Your assignment must be uploaded by the assigned due date and time. Moodle will automatically close the link at that time. It is strongly highly recommend you do not wait until the very last minute to submit your work.

## Concepts

This lab will give you your first experience with designing and programming with Java Inheritance. This lab focuses on inheritance as well as using the skills practice in the last lab - creating classes, defining fields/instance variables, writing methods, and writing a main method to run your program from.

## Background

Read chapter #8 and do the online tutorials on inheritance and the pre-lab for Lab #2. Study the “network” example in chapter #8 as that has similar concepts to what you will be doing for this lab.

## Assignment

Lab instructions:

1. Create a new project and name it Lab2.
2. Define three classes; one class must be the superclass and the other two classes being subclasses. Choose from the following options for ideas or come up with your own. If you come up with your own, **make sure that your assignment of superclass and subclasses pass the “is-a” relationship test described in the text.** Note that all of the following would pass this test.
  - Relative (superclass); Sister, Brother, Aunt, Uncle (subclasses)
  - Parent (superclass); Mom, Dad (subclasses)
  - Appliance (superclass); Stove, Refrigerator, Oven, Dishwasher (subclasses)
  - Animal (superclass); Dog, Cat, Hamster, Tiger (subclasses)
  - Publication (superclass); Book, Magazine, Newspaper (subclasses)

FOR YOUR SUPERCLASS, IMPLEMENT THE FOLLOWING:

3. Your superclass should contain at least **2 - 3 fields** (attributes) with accessor and mutator methods as needed.
4. Your superclass must have a non-default constructor with parameters that initialize all of the fields.
5. Your superclass must have a printDetails( ) method that prints out a description of the object (it’s fields).

FOR ALL OF YOUR SUBCLASSES, IMPLEMENT THE FOLLOWING”

6. Your subclasses should contain **2 - 3 additional** fields (attributes) with accessor and mutator methods as needed.

7. The subclass must have a non-default constructor that uses the proper syntax to call the constructor of its parent class and to initialize the new fields.
8. Your subclasses should also have a “printDetails( )” method that prints out the details of your ALL of the subclass attributes, including the attributes that are inherited. Note that there are several approaches to accomplish this; your choice as long as you meet common coding guidelines (no using public fields!).

NOW CREATE AN ADDITIONAL CLASS (that just inherits from the Object class)

9. This class must have a field of type ArrayList that holds objects of your superclass.
10. Create a method that adds a SINGLE object to your ArrayList. The object instance must be passed as a parameter to this method using a polymorphic parameter of your superclass type (this can hold any of the subtype instances).
11. Again using the concept of a polymorphic variable, write a “printAll( )” method that has a for-each loop that calls the printDetails( ) methods to print the details of each object in your ArrayList.

IMPLEMENT A MAIN

12. Write a main method in the appropriate class that creates the necessary object(s), adds several instances to the list and calls the printAll( ) method. Your main must create at least 5 instances of each of your subclasses, add them to the list using your previously written method that adds an objects to the list, and then calls the printAll( ) method.

Additional lab requirements:

1. Add JavaDoc comments for ALL public METHODS and CLASSES that you write

Hints: You should eliminate both code (methods) and state (fields) that have duplication between the super and sub classes.

## **Submission:**

Submit your entire BlueJ project (zip only, no tar, 7zip, or other file compression types) folder for this lab into the CS162, Lab2 upload link (just as was done in CS161).