UNIVERSITY AT ALBANY
State University of New York

COLLEGE OF ENGINEERING AND APPLIED SCIENCES
DEPARTMENT OF COMPUTER SCIENCE

ICSI201 Introduction to Computer Science

Lab 07 Created by Qi Wang

Goals:

- *Dog* class

Notice that students are expected to start the lab as soon as the description is available, seek feedback during the lab and submit all required documents on time.

Work will be rejected with no credit if

- The work is late.
- The work is not submitted properly (Blurry, wrong files, not in required format, crashed files, etc.).
- The work is a copy or partial copy of others' work (such as work from another person or the Internet).

Labs are contiguous study of the lecture or used as stepping-stones for the projects. Skipping lab activities would impact the learning significantly.

Submissions (100 points):

- A UML diagram (a Violet Class Diagram)
- Dog.java
- Driver.java

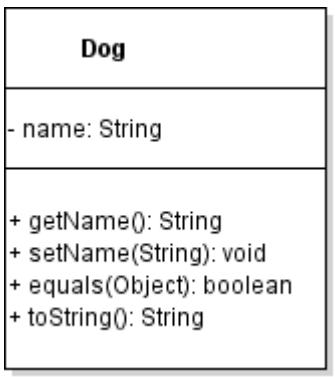
Instructions:

For this lab, you will implement the *Dog* design and test it in a *driver* program. Use the following two files as stater files and complete the source codes. Submit the diagram, *Dog.java* and *Driver.java* after you complete them.

Dog.java

```
/**
 * Defines a dog that consists of a name.
 * @author
 * @version
 */
public class Dog{
     /**
      * The name of this dog
      */


     /**
      * Constructs a dog with a default name.
      */
     public Dog(){}
```



```
    /**
     * Constructs a dog with a name.
     * @param name A reference to a name
     */
    public Dog(String name){}

    /**
     * Returns the name of this dog.
     * @return A string specifying this dog's name
     */
    public String getName(){}

    /**
     * Changes the name of this dog.
     * @param name A reference to a name
     */
    public void setName(String name){}

    /**
     * Indicates if this dog is equal to some other object. If the other object
     * is not a dog, this dog is not equal to the other object. If the other
     * object is a dog and has the same name, this dog is equal to the other
     * object.
     * @return A boolean value specifying whether this dog is equal to some
     *         other object
     * @param obj A reference to some other object
     */
    public boolean equals(Object obj){}

    /**
     * Returns a String representation of this dog. The string consists of
     *       - the type of this object
     *       - the instance variable value(s) of this object.
     *
     * @return A string representation of this dog
     */
    public String toString(){}
}
```

Driver.Java:

```
/**
 * Test the dog design.
 * @author
 * @version
 */
public class Driver{
    /**
     * Creates dog instances and uses them to test the instance methods.
     * @param args A reference to a string array containing command-line
     *             arguments
     */
    public static void main(String[] args){
        //Test constructors.
```

```
        //Create a default dog, dog1.
        //Create a dog with a name "Buddy" or a name of your choice, dog2.
        //Create a string object, string1.

        //Test toString.
        //Print dog1.
        //Print dog2.

        //Test getName.
        //Retrieve dog1's name. Print the name.
        //Retrieve dog2's name. Print the name.

        //Test setName.
        //Change dog1's name. Print dog1.
        //Change dog2's name. Print dog2.


        //Test equal.
        //Compare dog1 with dog2, print the comparison result.
        //Compare dog1 with string1, print the comparison result.
    }
}
```