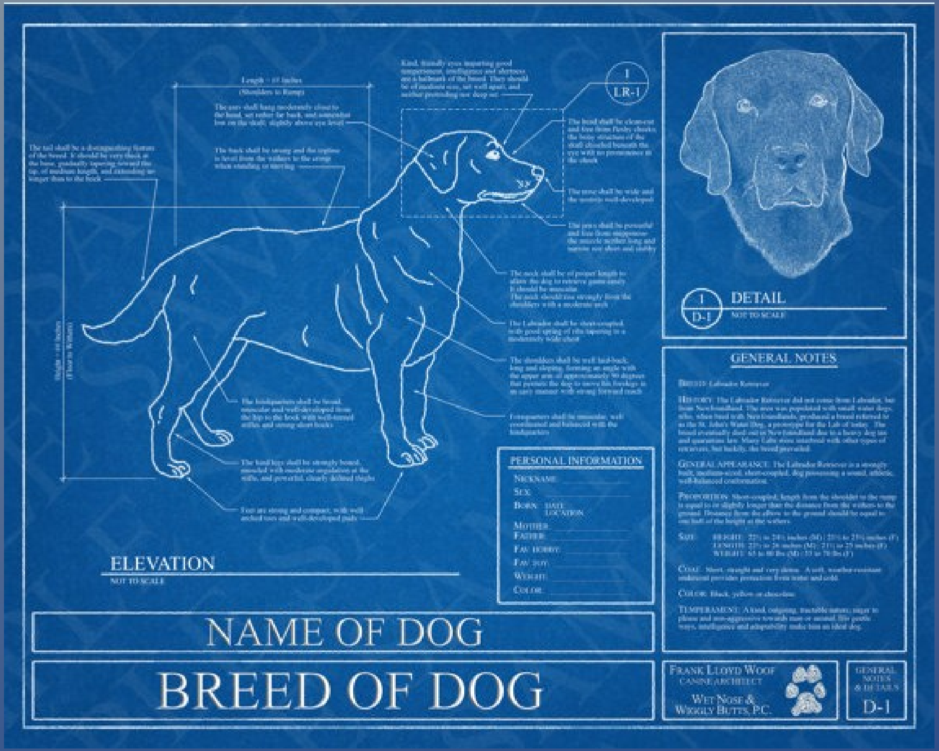
Objects in Java

Basic Concepts

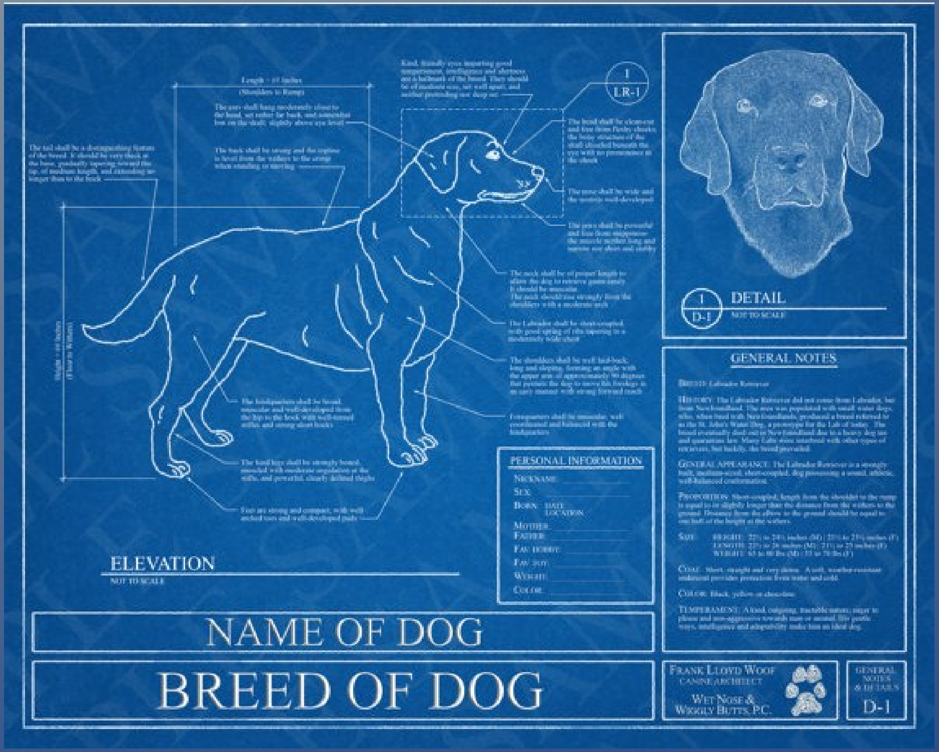
- what is a class?
- defining classes
- creating objects

The Class definition is like a blueprint or design or specification



- here we specify the properties and actions of our class
- properties
  - name
  - breed
- actions
  - bark
  - wag tail

Create objects using class design



Create objects using class design

File: Dog.java

```
class Dog
{
    String name;
    int age;
    String breed;
    int tailPosition = 0;

    void bark()
    {
        println("woof!");
    }

    void wagTail()
    {
        tailPosition = -5;
        delay(1); // 1 second
        tailPosition = +5;
        delay(1); // 1 second
        tailPosition = 0;
    }
}
```

File: DogPark.java

```
public static void main(...)
{
    // create some dog objects
    Dog dog1 =
        new Dog("Fido", "Lab");

    Dog dog2 =
        new Dog("Rex", "Boxer");

    Dog dog3 =
        new Dog("Buddy", "Spaniel");

    dog1.bark();

    dog2.wagTail();

    dog3.bark();
    dog3.wagTail();
}
```

Defining a Class

- define a class in a separate file with the same name as the class
  - Person.java
- define fields to hold data, or properties, of the class
  - name, age

```
class Person
{
   String name;
   int age;
}
```

Using a Class to Create Objects

- a class is just an idea
- an object is that idea made into something "real"
- create and use objects in a separate file from the class
  - TestPerson.java
- the "new" keyword asks for space in memory for object

```
// a regular old variable
int count = 0;

// more complex variables
String msg = "Hello";

double[] grades =
     new double[4];

// create a new person
Person p1 = new Person();
```

Object Data Fields

- data fields contain the properties of individual objects
- each object will have its own copies of its own data
- data fields can store basic data types, arrays, or even other objects

```
// create a new person
Person p1 = new Person();
Person p2 = new Person();

p1.name = "Arthur Dent";
p1.age = 44;

p2.name = "Ford Prefect";
p2.age = 32;

println(p1.name);
// output is "Arthur Dent"

println(p2.age);
// output is 32
```

What is an Object

- the simplest version is like a complex variable
- holds multiple pieces of data
- data can be different types
  - int, double, boolean, char, String
- this type of data structure is sometimes called a record
- data for a Person might include:
  - name (String)
  - age (int)
  - married (boolean)

Source Code: Person Class

```java
// file1: Person.java

class Person
{
   String name;
   int age;
   boolean isMarried;
}
```

```java
// file2: TestPerson.java

class TestPerson
{
   public static void main(String[] args)
   {
      int x;
      double y;

      Person p = new Person();

      p.name = "Fred";
      p.age = 25;
      p.isMarried = false;
   }
}
```

Source Code: Person Class

```
// file1: Person.java

class Person
{
   String name;
   int age;
   boolean isMarried;
}
```

- the class (or record) is defined in its own file
- each part of the record is then defined within the class

Source Code: Person Class

- to use the record, make an object which has the properties of the class definition
- create a variable to identify the object
- use the "new" keyword to create the object in RAM

```
// file2: TestPerson.java

class TestPerson
{
  public static void main...
  {
    int x;
    double y;

    Person p = new Person();

    p.name = "Fred";
    p.age = 25;
    p.isMarried = false;
  }
}
```