Practice Exercises #5-1

Do these in any order. Let your interest be your guide.

1. Add classes for other types of pets, such as Horse and Fish, to the Dog program. Make one or more subclasses for each pet type. For example, GoldFish would be a subclass of Fish.
2. Add a class called Pet to the Dog program and give it some general fields that all pets have (e.g. name, gender). Next, modify the Dog class so that it is a subclass of Pet. Remove any fields from Dog that are already in Pet.  
     
   To test your class structure in setup(), make an array of Pet objects called myZoo. Fill the array with a variety of Pet objects. Include at least one object of each subclass. For example, your zoo might have two dogs, a horse and gold fish. Use a for-loop to call the describe() method on every object in myZoo.
3. Pick another realm that has a hierarchy of categories (e.g. foods, musical instruments) and write an inherited class structure for it, along with a setup procedures that does stuff with it.   
     
   Goal: Make it as cool as possible. Can you think of methods to add besides just **describe**()?  
     
   Example 1: class Apple extends Fruit {…} might have a boolean field called stillOnTree, which is set to true when the fruit object is first constructed. But Apple might have a method called **pick**(), which sets stillInTree to false.  
     
   Apple myApple = new Apple(200, “granny smith”:) *// a 200-gram granny smith apple*

myApple.describe() *//prints “This granny smith apple weighs 200 grams and is still on the tree”*

myApple.pick();

myApple.describe() *//prints “This granny smith apple weighs 200 grams and is no longer on the tree”*

myApple.takeBite()

myApple.describe() *//prints “This granny smith apple weighs 190 grams and is no longer on the tree”*