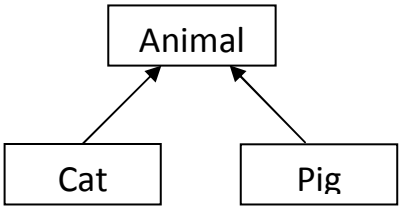
Unit 5 – Worksheet 02 More With Inheritance

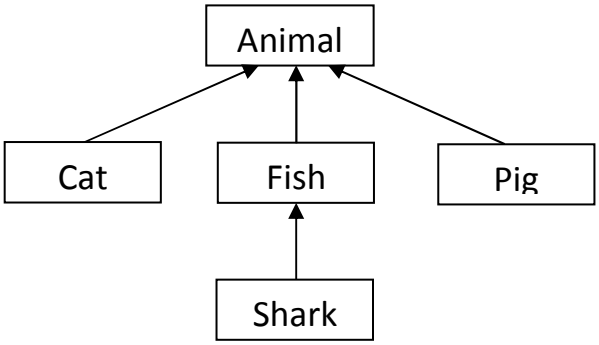
Name: _____

Code Used in the Classroom can be found on page 2:

The goal of this assignment is to build off of the code that was discussed in class. The code represents the following Inheritance Hierarchy:



Problem #1 – Create two new classes: one called Fish and the other called Shark (a total of 6 classes located in the same package). Create these two classes so that the following Inheritance Hierarchy exists:



Problem #2 – Edit the Animal class so that it also includes a String variable that tracks the animal's color. When an animal is constructed, the color should be set to " ".

Problem #3 – Add a Mutator method to the Animal class called setColor( ). This method needs to have a parameter so that the user can set the color of the animal to their desired color.

Problem #4 – Add an accessor method to the Animal class called getColor( ) so that the color of all animals can be accessed when needed.

Problem #5 – Edit the Cat class so it now also tracks what breed of cat it is (Siamese, Persian, etc.). When a cat is constructed the default breed should be " ".

Problem #6 – Add a Mutator method to the Cat class called setBreed( ) that uses a parameter to allow the user to set the breed of the cat.

Problem #7 – Add an accessor method to the Cat class called getBreed( ) so that the breed can be looked up.

Problem #8 – Edit the Fish class so that the type of fish can be stored (default " ").

Problem #9 – Add a Mutator method to the Fish class so that the user can edit the type of fish represented.

Problem #10 – Add an accessor method to the Fish class called getType( ) so that the type of fish can be found.

Problem #11 – Add a method to the Fish class called swim( ). This method should output the following statement when called: The fish is swimming…

Problem #12 – Add a method to the Fish class called fishNoise( ). This method should output a statement that represents some sort of fish noise.

Problem #13 – Add an instance variable to the Shark class called attack. This variable will track whether or not the shark is attacking at the moment (boolean).

Problem #14 – Add a mutator method to the Shark class that changes the current attacking state of a shark.

Problem #15 – Add an accessor method to the Shark class that checks to see if a shark is currently attacking.

Problem #16 – Return to your main class. Write code to test that you are able to successfully create and manipulate all items in the Inheritance Hierarchy: Animal, Cat, Fish, Pig, and Shark. Also verify that all methods in all classes are working properly.

Code From A Classroom:

```
public class Animal {
   public boolean living, awake;
   public Animal()  {
      living=true;
      awake=false;
   }
   public void wakeUp()  {
      awake=true;
   }
   public void goToSleep()  {
      awake=false;
   }
   public void getSleepStatus()  {
      if(this.awake==true)
         System.out.println("This animal is awake!");
      else
         System.out.println("This animal is asleep.”);
   }
   public void death()  {
      living=false;
   }
   public boolean getLivingStatus()  {
      return living;
   }
}
```

```
public class Cat extends Animal{
   public boolean meowing;
   public Cat(){
      super();
      meowing=false;
   }
   public void makeMeow(){
      meowing=true;
   }
   public void stopMeow(){
      meowing=false;
      super.goToSleep();
   }
   public boolean getMeowStatus(){
      return meowing;
   }
}
```

```
public class Pig extends Animal{
   public boolean oinking;
   public Pig(){
      super();
      oinking=false;
   }
   public void makeOink(){
      oinking=true;
   }
   public void stopOink(){
      oinking=false;
      super.goToSleep();
   }
   public boolean getOinkStatus(){
      return oinking;
   }
}
```

```
public class animalMain {
   public static void main(String[] args) {
      Animal a1 = new Animal();
      System.out.println("Animal 1 is alive ... "+a1.living);
      System.out.println("Animal 1 is awake ... "+a1.awake);
      a1.wakeUp();
      System.out.println("Animal 1 is alive ... "+a1.living);
      System.out.println("Animal 1 is awake ... "+a1.awake);
      a1.death();
      System.out.println("Animal 1 is alive ... "+a1.living);
      System.out.println("Animal 1 is awake ... "+a1.awake);
      Pig piggy1 = new Pig();
      System.out.println("Piggy 1 "+piggy1.awake+" "
            +piggy1.living);
         }
      }
```