Class/Object Lab

1. Following is a function that draws a happy face. Use this function as the basis of a new class called HappyFace. The HappyFace class constructor should take x, y and diam variables that will be stored in the class as fields. The class should have a method called display() that draws itself on the sketch window. The body of the display() method should closely follow the following happyFace() function. Test the class by creating a new instance of HappyFace positioned at the center of the sketch window and calling its display() method.

```
// Draw happy face
void happyFace(float x, float y, float diam){
  //Face
  fill(255, 255, 0);
  stroke(0);
  strokeWeight(2);
  ellipseMode(CENTER);
  ellipse(x, y, diam, diam);

  // Smile
  float startAng = 0.1*PI;
  float endAng = 0.9*PI;
  float smileDiam = 0.6*diam;
  arc(x, y, smileDiam, smileDiam, startAng, endAng);

  // Eyes
  float offset = 0.2*diam;
  float eyeDiam = 0.1*diam;
  fill(0);
  ellipse(x-offset, y-offset, eyeDiam, eyeDiam);
  ellipse(x+offset, y-offset, eyeDiam, eyeDiam);
}
```

2. Declare all the necessary classes in order to make the following driver program work properly (steps have been broken down for you in the sub-parts). For example, this is a sample output that is acceptable:

Generally, a Dolphin can be found in water, it can not lay eggs, and is often overheard saying 'ak, ak, ak, ak'
Generally, a Platypus can be found on land, it can lay eggs, and is often overheard saying 'errrr'
Generally, a Human can be found on land, it can not lay eggs, and is often overheard saying 'I'll take a grande latte with a double-shot of espresso'
Generally, a CSStudent can be found on land, it can not lay eggs, and is often overheard saying 'I love programming!'

```
void setup(){
  Mammal[] mammals = new Mammal[4];
  mammals[0] = new Dolphin();
  mammals[1] = new Platypus();
  mammals[2] = new Human();
  mammals[3] = new CSStudent();

  for (int i=0; i< mammals.length; i++){
    print("Generally, a " + mammals[i].getName());
    print(" can be found ");
    if(mammals[i].livesInWater() == false){
      print("on land, ");
    }
    else {
      print("in water, ");
    }

    print("it can ");
    if(mammals[i].laysEggs() == false) {
       print("not ");
    }
    print("lay eggs, and is often overheard saying '");
    mammals[i].speak();
    println("'");
  }
}
```

3. Declare a new class called Mammal with the following members:
    - Two String fields called ‘name’ and ‘sound’
    - A constructor that accepts two String parameters (‘name’ and ‘sound’) and saves values in fields
    - A void method called ‘speak()’ that prints the object’s sound to the console area,
    - A boolean method called ‘laysEggs()’ that returns false
    - A boolean method called ‘livesInWater()’ that returns false.
    - A ‘getter’ String method called ‘getName()’ that returns the object’s name field;
4. Declare a new class called Platypus that extends Mammal. Override methods as appropriate.
5. Declare a new class called Dolphin that extends Mammal. Override methods as appropriate.
6. Declare a new class called Human that extends Mammal. Override methods as appropriate.
7. Declare a new class called CSStudent that extends Human. Override methods as appropriate.

8. Trace the following code. (Draw a table with the appropriate variables and fields.) What are the intermediate values of each of the array elements? Show what is printed in the correct order.

```
class A {
  int x;
  void b(int i){};
  A(int x) {
     this.x = x;
  }
}
class B extends A {
  B(int y) {
    super(y * y);
    println(x);
  }
  void b(int i){
    x = x + i;
  };
}

A[] array;

void setup() {
 array = new A[3];
 for (int i = 0; i < array.length; ++i) {
   if (i % 2 == 0) {
     array[i] = new A(i);
   } else {
     array[i] = new B(i);
   }
 }
 for (int j = 10; j < 13; j++) {
   int k = j % array.length;
   array[k].b(j);
   println(array[k].x);
 }
}
```