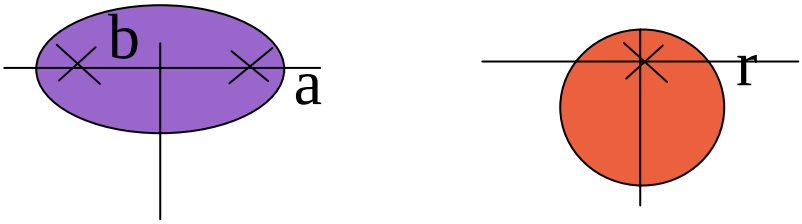
Cmp Sci 187: Programming with Data Structure Hierarchy II

*1. IS-A and Has-A **Relationship:***

A mail order business sells catalog merchandise all over the country. The rules for taxation on the merchandise vary from state to state. Also, the rate of taxation can be different based on the type of merchandise. For simplicity, we will consider three types of merchandise, clothing, pharmaceutical, and beauty products. Suggest a class hierarchy to model the tax on a merchandise.

2. Circle and Ellipse



An ellipse is a somewhat flattened circle with the property that it is the locus of points such that the sum of distance from two fixed points to the point on the locus is a constant. For an ellipse with center at the origin it can be represented by the two intercepts on the axes, a and b. A circle at the origin with radius r can be represented by $\mathbf{x^2+y^2 = r^2}$. An ellipse on the other hand can be represented by $\mathbf{x^2/a^2 +y^2 /b^2 = 1}$ Mathematically, a Circle is a special case of an Ellipse with the two fixed points collapsed into one, the center of circle. How would you model them in a Class hierarchy?

Cmp Sci 187: Programming with Data Structure Hierarchy II

3. Serializable, Externalizable:

For the following code snippet, class **App** contains a field of type **Y**. How would you save the application data to a file so that you can recreate **yobj** from it later?

```
public class X {
  int x;
}

public class Y {
  int y;
  X xobj;
}


public class App {
  Y yobj;
}
```

Cmp Sci 187: Programming with Data Structure Hierarchy II

4. Solution for 1 based on Composition:

```
public abstract class Merchandise {
 Tax tax;
 public int getCost() {}
 public int getTax(int zipCode) {return tax.getTax(zipCode);}
}

public class Clothing extends Merchandise {
  public Clothing () { tax = new ClothingTax(this);}
  public int getCost() {}
}

public abstract class Tax {
 Merchandise article;
 public int getTax(int zipCode);
}

public class ClothingTax extends Tax {
 // imagine a zipcode indexed table for looking up taxation
 public ClothingTax(Clothing article) { this.article = article;}
 public int getTax(int zipCode);
}
```