Lecture
Chapter 9
Java Again

Inheritance

- **Ability of a child class to obtain,**
  and possibly to modify,
  the characteristics of a parent class

```
public class Sphere
{
    private double radius;
    public static final double DEFAULT_RADIUS = 1.0;

   public Sphere( )
   {
      setRadius(DEFAULT_RADIUS);
   }
   public Sphere(double initialRadius)
   {
      setRadius(initialRadius);
   }
        …
 }
```

```
public class Ball extends Sphere
{
    private String name;
    public static final double DEFAULT_RADIUS = 1.0;

   public Ball( )
   {
      setName("unknown");
   }
   public Ball(double initialRadius, String initialName)
   {
      super(initialRadius);    // call superclass's constructor
      setName(initialName);
   }
        …
 }
```

Superclass 's private data fields cannot be revised nor can their names be reused in a subclass.

Superclass's private methods can be overridden in a subclass. A method in in a subclass overrides a method in a superclass if the two methods have the same declarations and different bodies.

Annotations explicitly notify the compiler of exceptional uses, e.g., overrides

```
@Override
Public void displayStatistics( )
{
  Systrem.out.println( … );
  Super.displayStatistics( );
}
```

Java Access Modifiers

See Lecture 9_b #10

Is-A Relationships

Object type compatability
Superclass-subclass
Extends

Has-A Relationships

Containment

```
public class pen
{
   private Ball point;
        …

}
```

For more details see Lectures 9_a, 9_b & 9_c