Abstract Methods

Abstract Definition

abstract: existing in thought or as an idea but not having a physical or concrete existence

abstract method: A method name, it's arguments, and return type, but no Java instructions. Here's a method that should exist, but doesn't have a concrete set of instructions yet.

Abstract Method Syntax

```
abstract class classname {
    abstract modifiers returnType method_name (parameters);
}
```

- Abstract methods can only occur inside abstract classes
- Java will not create a new object for an abstract class
- An abstract method can be inherited… if so, the child class must be abstract.
- An abstract method can be overridden by a concrete method
- If all abstract methods are overridden , the child class does not have to be abstract, and child objects can be created

Abstract Method Implications

- An "abstract method" can be invoked
  - Need a reference to invoke. Need a concrete method override to get a concrete class to make an object. With dynamic dispatch, you will invoke the concrete method!
- No static abstract methods
  - abstract means "no functionality", but static means "functionality without an object"… contradiction.
- A parent uses abstract methods to force their children to either be abstract and un-implementable, or live up to their responsibility, and provide the method.
  - The only use of an abstract class is to be the supertype of a concrete sub-class

Problem: Shapes

- Keep track of Rectangles, Circles, and right Triangles in a cartesian coordinate system
- Need a Point (x,y), a min point, a max point, a perimeter, and an area for each kind of shape
- Each shape should be movable

Or, a "bad" concrete function

```
class Shape {
   public Point max() {
      throw new UnsupportedOperationException(
         "I have no idea what the max is for this shape.");
   }
   …
}
```