* TriangleAreaApp - see the powerpoint for the java project that demonstrates the program's design
  + ScannerObject.hasNext()
    - Returns a boolean value that is true if the scanner has more things to read and false if it doesn't

* When you have a large process, break it down to smallest possible steps and make small helper functions to achieve these functions, and then call these from the main function.

* To make sure that a process happens only once, setup a Boolean value that is equal to true, put a If(boolean == true) in front of the function, and then after the function is done, change the boolean to false.
  + This will make sure that the function won't be performed more than once
* Non Object Oriented Approach
  + All functions are static
  + Variables are declared locally or passed in as parameters
  + Static class functions simply act as a library of helper functions used by your application
* When you think about the three points and their x&y coordinates, ax, ay, bx, by, cx, cy
  + They represent a specific triangle to US as humans, but the compiler doesn't know any of these relationships
  + It is on the human programmer to ensure that these points actually represent a triangle and are valid
* Object Oriented Programming was developed so that you can define and express ways of creating relationships between information
  + Formalize the collective meaning of this information as an abstraction
  + By defining an object, you can create properties and functions that make it clear what information is necessary to represent the object
  + In Java, create a class that has a name that represents the object
  + We will assume each class will be a public class, and each class will be in its own file
* Step 1 for Object Oriented Programming:
  + Name and create the class
* Step 2 - declare its fields
  + declare variables that will define it
  + Here you make design decisions: you decide if a field is an int or a double, which has implications on how to deal with this piece of data

* Step 3 - define a constructor
  + Constructors create and initialize a new instance of your object
  + Constructors are just functions that are named to match the class type
    - No return value
  + Within a constructor, the keyword **'this'** is a reference to the object being constructed
    - Helps to refer to the specific object you are creating when defining stuff in a constructor
    - Any information needed can be passed in as parameters
  + To call a constructor, use the **'new'** keyword
    - Calling this will create a new instance/object of this class
* Step 4 - define instance methods
  + Instance methods are called without 'static', which means you have to create an instance of the object and call the method on that
  + Within instance methods, the keyword 'this' refers to the object itself. Only use this if a variable is ambiguous, or if the name of the field and the name of the parameters are the same
* Classes are like factories that contain a blueprint for an object
  + The fields are the pieces of data
  + The methods are what it can do
  + Anything marked as static in this class belongs to this class, but doesn't belong to any certain one object of this class
    - Meaning the factory wants its own functionality, rather the blueprint having functionality
* Immutable object: an object that's state won't change after you create it
* **Make sure you are thoughtful when designing objects**