Java Day 8

Garbage Collection

Remember that Java handles memory management

* Garbage Collection - is up to Java -> and it's when an object gets removed from memory
  + finalize(): method invoked by the JVM when it realizes an object should be garbage collected.
    - its main purpose is to release resources used by objects before they're removed from memory
  + gc() : the garbage collection method

**NEVER** use these!! The time at which garbage collector calls finalizers is dependent on the JVM's implementation and the system's condition, which are out of our control.

* They are costly.
* gc() does **not** trigger garbage collection - just a hint to the JVM to start GC.
* The JVM knows better when to do this.

Generics

* All Java Collections use "Generics" -> <Type>
* Generics act as a specifier (and limiter) for a type of data to be used within a class or interface.
* Generics can be used when writing classes, interfaces, and methods.
  + To use generics in a method, the containing class or interface must use generics / be generic
* Generic types must be an Object (primitives are not allowed -> Wrapper Classes)
* "I don't know what type this will be, but when this class is instantiated, a type will be provided"  
  List<String> stringList = new ArrayList<String>();