API

* public class Object
  + public boolean equals(Object x) – Returns true if this object is shallow-equal to x. In subclasses, should be overridden to return true if this object is deep-equal to x.
  + public int hashCode() – Returns an int that is different for every object. In subclasses, should be overridden to support efficient hashing, while obeying the equals/hashcode contract.
* public class Exception
  + public Exception(String s) – Constructor that stores s as the exception’s message. s may be retrieved by calling getMessage() on the exception object.
* public class String
  + public int length() – Returns the length of the string.
* public interface Comparable<T>
  + public int compareTo(T that) – Returns a negative number if this object “comes before” or is “less than” that object. Returns a positive number if this object “comes after” or is “greater than” that object. Returns zero if the objects are deep-equal.
* public interface Set<T>
  + public boolean add(T t) – Adds t to the set, unless the set already contains a member that is deep-equal to t. Returns true if the set changed, otherwise returns false.
  + public boolean remove(T t) – Removes from the set an object that is deep-equal to t, if such an object is in the set. Returns true if an object was removed, otherwise returns false.
  + public boolean addAll(Collection<T> addUs) – Adds all members of addUs to the set. Returns true if the set changed.
  + public boolean removeAll(Collection<T> removeUs) – Removes all members of removeUs from the set. Returns true if the set changed.
  + public boolean contains(T t) – Returns true if the set contains an object that is deep-equal to t.
* public interface Map<K, V>
  + public V put(K key, V val) – Maps key to val. Returns val.
  + public V get(K key) – Returns the value associated with key.
  + public Set<K> keyset() – Returns a set containing the maps keys. The type of the set depends on the type of the map.

This ***doesn’t mean*** that you only need to understand these classes, interfaces, and methods!