**Garbage Collection**

* Garbage Collection is the process of destroying unused objects from heap memory.
* Garbage collection is done by Garbage collector.
* JVM calls garbage collector randomly.
* Programmer can request JVM to run garbage collector.
* Unused are eligible for garbage collection.

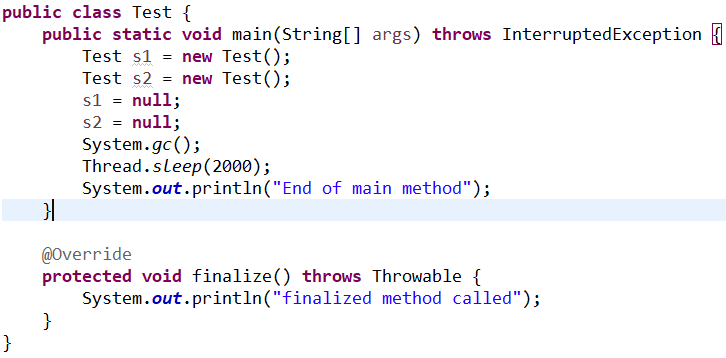
**Different ways to make object eligible for garbage collection**

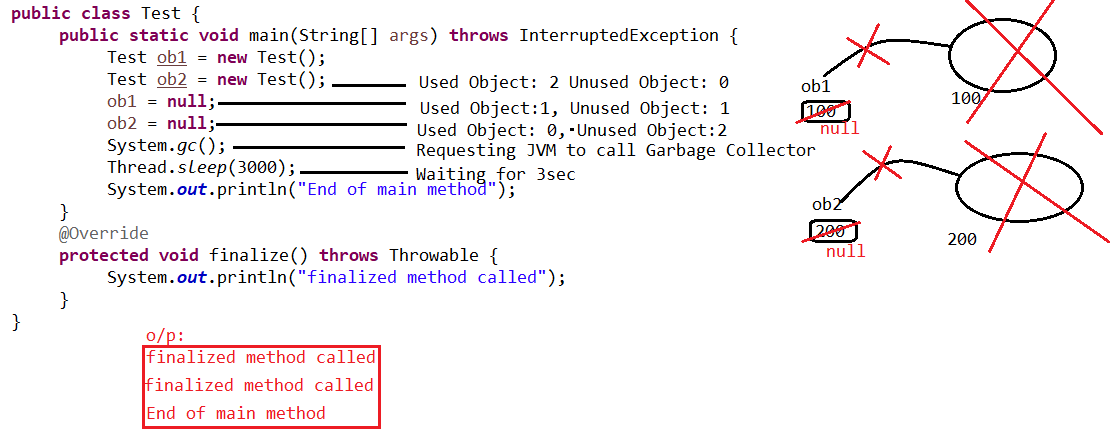
* Nullifying a reference variable
* Re-assigning a reference variable
* Island of isolation

**Nullifying a reference variable**

* If an object no longer required then assign null to all its reference variables, this approach is called nullifying a reference variable.

**Example#1**

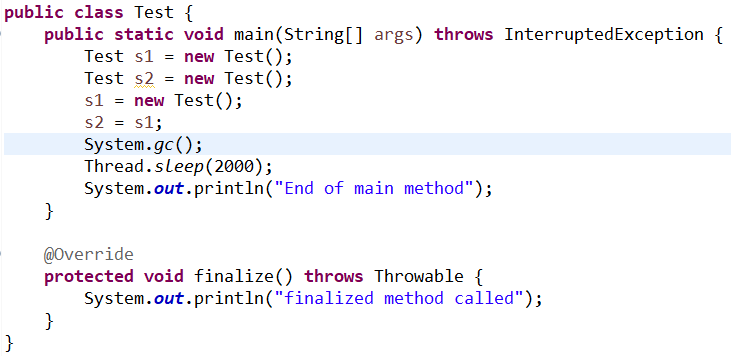


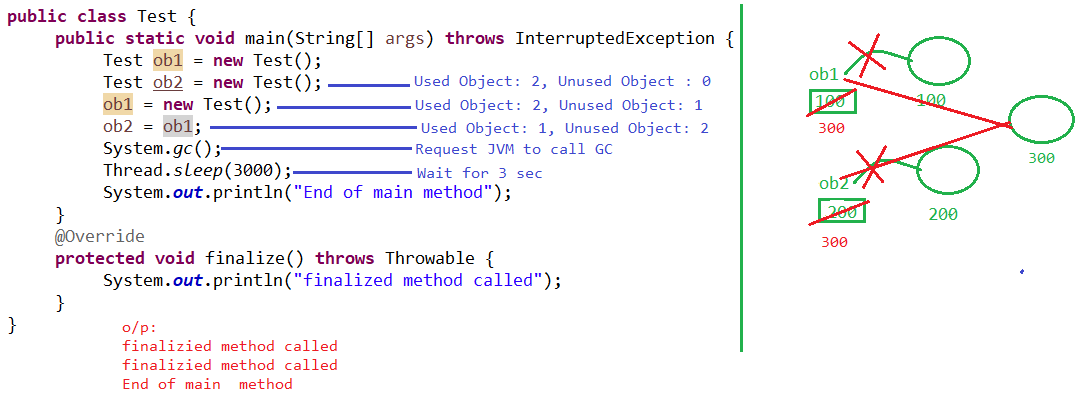


**Re-assigning a reference variable**

* If an object no longer required then reassign it's reference variable to some other object.

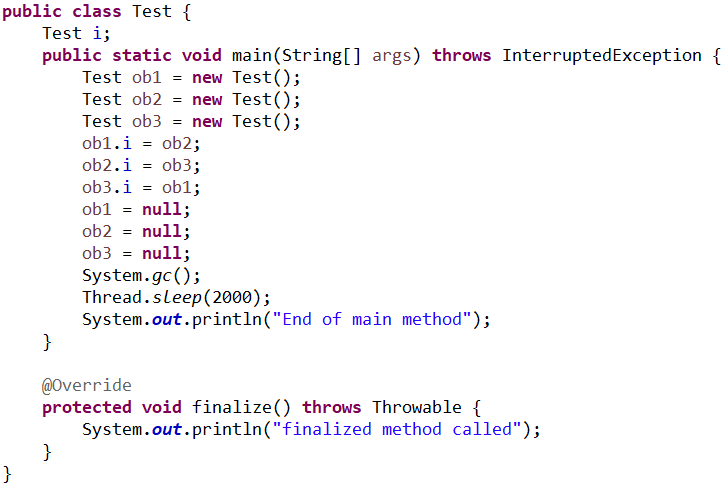
**Example#2**

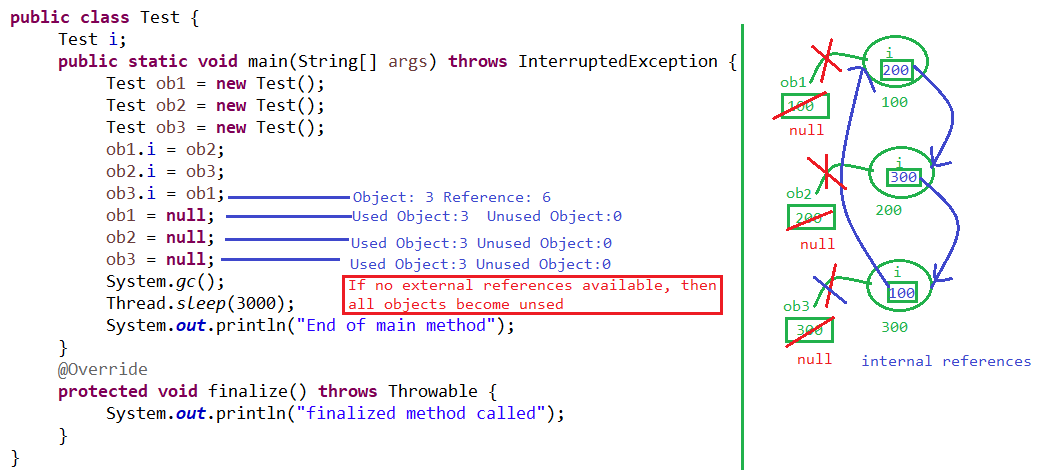


****

**Island of isolation**

* A group of objects that reference each other but are not referenced by any active/external object in the application.

****

****

**Requesting JVM to run garbage collection**

* By Using System.gc()
* By Using Runtime.getRuntime().gc()

**Finalization**

* Garbage collector calls finalize() method before destroying object to perform cleanup operation.
* finalize() method present in object class with following declartion

protected void finalize()