GC feature provided by the CLR to clean up the manages object to reclaim the memory. Unused managed object. Continuously checks if the memory need to be cleaned or not

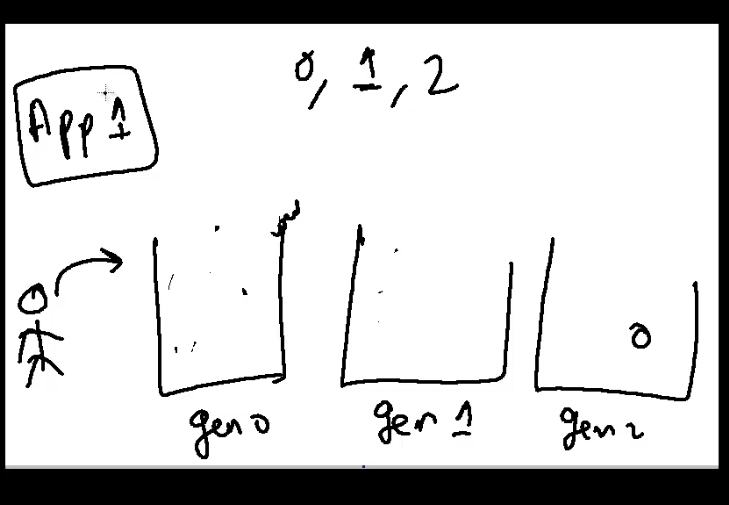
Background thread nobody will comes to know when he was visited the system.

Generations 0,1,2:

App1 has 2 object created will be moved to bucket called generation0

Gc visits and comes to know 2 objects are unused So he cleans those 2 object and other used one are moved to generation 1 bucket

Again 2 objects created it will be moved to generation 1 bucket



Generation indicates how long the objects are used in application.

As a generation increases GC less likely to go there and check if the object is active or not

GC perform clean up for only managed resources.

So we need a way to perform clean up for unmanages resources.

We have some thing called destructor.

Name same as class name With ~ symbol

When GC run it checks if the class has destructor code. It will reserve the job for it an move the object to next generation even those Gc can be done.

This can be eliminated with the help of Dispose finalize pattern.

Implement Idisposibel interface and Dispose method

Call GC.suppressFinalizer(true)

Indicates Gc that go ahead with finalization even if I have finalizer.

