Memory Management

* The goal of the memory management is to reduce the memory footprint of a program by controlling the life time of all its objects.
* The object ownership is implemented using reference counting system.
* Reference count refers to the number of owners each object has.
* The reference count increases when the number of owners increase and it is decreased when the object gets destroyed.
* As soon as the reference count reaches zero the operating system allows to destroy the object.
* Earlier developers controlled the reference cont manually called manual Retain Release, Xcode 4.2 introduced Automatic Reference Counting, which automatically controls the reference count.

1. Manual Retain Release

* In this method we use the special memory related methods, those are:

1. alloc : Create the object and claim the ownership.
2. retain : Claim the ownership of existing object.
3. copy : Copy an object claim its ownership.
4. release : Relinquish the ownership of an object and destroy it immediately.
5. autorelease : Relinquish the ownership of an object and defer the destruction.

* In this method developer has to relinquish the ownership of an object after using it, means every alloc, retain and copy should be balanced with release and autorelease on a same object.
* If the developer forgets to release the memory, the memory is not freed and the memory leak may occurs.
* If the release is used on the object many times, dangling pointer will occur and it gives an invalid address if u try to access it.

2. Automatic Reference Counting

* It works similar to MRR but the reference count is set automatically and inserts the appropriate memory management methods.
* The ARC inserts the retain and release calls automatically depending on object's lifetime, by analyzing the code.
* No need of calling the retain, release and autorelease methods.
* ARC works by analyzing your code to figure how what the ideal lifetime of each object should be, then inserts the necessary retain andrelease calls automatically.
* The algorithm requires complete control over the object-ownership in your entire program, which means you’renot allowed to manually call retain, release, or autorelease.
* The only memory-related methods you should ever find in an ARC program are alloc and copy.
* ARC introduces new @property attributes.
* The strongattribute can be used in place of retain, and weak in place of assign.
* No need to release instance variables as we did in [The dealloc Method](http://rypress.com/tutorials/objective-c/memory-management#the-dealloc-method)—ARC does this.
* The superclass’s dealloc is automatically called.