# IOS面试题目

1.定义属性时，什么时候用 assign、retain、copy 以及它们的之间的区别。

\* assign:普通赋值,一般常用于基本数据类型,常见委托设计模式, 以此来防止循环引用。(我们称之为弱引用).

\* retain:保留计数,获得到了对象的所有权,引用计数在原有基础上加1.

\* copy:一般认为,是在内存中重新开辟了一个新的内存空间,用来 存储新的对象,和原来的对象是两个不同的地址,引用计数分别为1。但是当copy对象为不可变对象时,那么copy 的作用相当于retain。因为,这样可以节约内存空间

2.MVC设计模式是什么？ 你还熟悉什么设计模式？

答案：略

3.什么是KVO和KVC?

答案：KVC:键 – 值编码是一种间接访问对象的属性使用字符串来标识属性，而不是通过调用存取方法，直接或通过实例变量访问的机制；

KVO:键值观察机制，他提供了观察某一属性变化的方法，极大的简化了代码；

4.KVO底层实现原理是什么？

答：KVO 底层实现原理是系统给当前类创建子类 , 在子类 setter 方法调用父类的 setter 方法﻿﻿。

5.frame和bounds有什么不同?

答:frame指的是：该view在父view坐标系统中的位置和大小。(参照点是父亲的坐标系统)

bounds指的是：该view在本身坐标系统中 的位置和大小。(参照点是本身坐标系统)

6.什么时候使用NSMutableArray，什么时候使用NSArray?

答：当数组在程序运行时，需要不断变化的，使用NSMutableArray，当数组在初始化后，便不再改变的，使用NSArray。需要指出的是，使用NSArray只表明的是该数组在运行时不发生改变，即不能往NSAarry的数组里新增和删除元素，但不表明其数组內的元素的内容不能发生改变。NSArray是线程安全的，NSMutableArray不是线程安全的，多线程使用到NSMutableArray需要注意。

7.tableView 的重用机制

答：UITableView 通过重用单元格来达到节省内存的目的: 通过为每个单元格指定一个重用标识符(reuseIdentifier),即指定了单元格的种类,以及当单元格滚出屏幕时,允许恢复单元格以便重用.对于不同种类的单元格使用不同的ID,对于简单的表格,一个标识符就够了.

8. 什么是延迟加载?

答：懒汉模式，只在用到的时候才去初始化。

一个延时加载，避免内存过高，一个异步加载，避免线程堵塞。

9.类别的作用?继承和类别在实现中有何区别?

答：category 可以在不获悉，不改变原来代码的情况下往里面添加新的方法，只能添加，不能删除修改，并且如果类别和原来类中的方法产生名称冲突，则类别将覆盖原来的方法，因为类别具有更高的优先级。

类别主要有3个作用：

1).将类的实现分散到多个不同文件或多个不同框架中。

2).创建对私有方法的前向引用。

3).向对象添加非正式协议。

继承可以增加，修改或者删除方法，并且可以增加属性。

10.在一个对象的方法里面：self.name= “object”；和 \_name =”object” 有什么不同吗?

答：self.name =”object”：会调用对象的setName()方法；

\_name = “object”：会直接把object赋值给当前对象的name属性。

11.View和View之间传值方式？

对象的property属性传值；

方法参数传值；

NSUserDefault传值；

块传值。

12.#import 与@class 的区别

#import不会引起交叉编译,确保头文件只会被导入一次；

@class 的表明,只定 义了类的名称,而具体类的行为是未知的,一般用于.h 文件；

@class 比#import 编译效率更高。

此外@class 和#import 的主要区别在于解决引用死锁的问题。

13.定时器与线程的区别

\* 定时器;可以执行多次，默认在主线程中；

\* 线程：只能执行一次。

14.如何进行真机调试？

1.首先需要用钥匙串创建一个钥匙（key）；

2.将钥匙串上传到官网，获取iOS Development证书；

3.创建App ID即我们应用程序中的Boundle ID；

4.添加Device ID即UDID；

5.通过勾选前面所创建的证书：App ID、Device ID；

6.生成mobileprovision文件；

7.先决条件：申请开发者账号 99美刀