3D

Core Animation 帮助你从二维的世界自由浮动;尽管它不是一个真正的三维框架, Core Animation 来帮助你设置二维物体在三维空间位置。换句话说,图层和动画仍然发生在两个维度,但你可以旋转和每个元素的2 d平面位置在三维空间一样:

CATransform3D中将使用一个特殊的性质:即转换层。CATransform3D 和CGAffineTransform类似,但除了缩放、倾斜和翻转在x和y方向,它还带来了第三维度:z。

简单的3D动画

原来你可以在Layer层做动画，现在你可以在三维扩展层中旋转、缩放、翻转等。

结合骨架、3D动画、MVC、单例、Storyboard综合

1、侧边栏同样是APP的一种常规的布局方式：优点：留出更多的空间给用户，将不常用的交互隐藏起来；

缺点：对于有些操作不能够直观的操作。

2、将对常用的APP设计进行模块化拆分、整合、总结。

3、拆分：

第一：一个tableview列表就可以实现侧边栏的选择，首先创建一个tableview，storyboard中拉入一个UItableview，关联一个SideMenuViewController.swift类，来进行交互；其中将MVC拆分开，这个因为业务简单，就将View的展示放在控制层，模型负责形成模型，采用单例的形式保存，保证初始化一次，不过多使用内存；

**全局常量**

let \_SingletonSharedInstance = Singleton()

class Singleton  {

    class var sharedInstance : Singleton {

        return \_SingletonSharedInstance

    }

}

嵌入结构体

structs do support static constants. By using a nested struct we can leverage its static constant as a class constant.

class Singleton {

    class var sharedInstance : Singleton {

        struct Static {

            static let instance : Singleton = Singleton()

        }

        return Static.instance

    }

}

多线程方式（OC 的方式）

**dispatch\_once**

class Singleton {

    class var sharedInstance : Singleton {

        struct Static {

            static var onceToken : dispatch\_once\_t = 0

            static var instance : Singleton? = nil

        }

        dispatch\_once(&Static.onceToken) {

            Static.instance = Singleton()

        }

        return Static.instance!

    }

}

此时侧边栏已经独立形成，采用MVC的形式；