远程推送原理:

苹果的推送服务通知是由自己专门的推送服务器APNs （Apple Push Notification service）来完成的，其过程是 :APNs 接收到我们自己的应用服务器发出的被推送的消息，将这条消息推送到指定的 iOS 的设备上，然后再由 iOS设备通知到我们的应用程序，我们将会以通知或者声音的形式收到推送回来的消息。

iOS 远程推送的前提是，装有我们应用程序的 iOS 设备，需要向 APNs 服务器注册，注册成功后，APNs 服务器将会给我们返回一个 devicetoken，我们获取到这个 token 后会将这个 token 发送给我们自己的应用服务器。当我们需要推送消息时，我们的应用服务器将消息按照指定的格式进行打包，然后结合 iOS 设备的 devicetoken 一起发给 APNs 服务器。我们的应用会和 APNs 服务器维持一个基于 TCP 的长连接，APNs 服务器将新消息推送到iOS 设备上，然后在设备屏幕上显示出推送的消息。

证书及推送实现过程:

1.新建Certificate Signing Request(也就是CSR)的请求文件，绑定你的电脑的

2. 配置AppID

Certificates,identifiers&Profiles --> Identifiers --> App IDs, +号创建

3.创建证书: 一种是开发的、一种是发布的，开发的是做测试用的。下载好双击安装

选择Development 点击右上角的+号,创建证书, 选择(1)创建的 CSR 文件

4. 添加 Devices (可添加100个设备)

5. 查找设备的 UDID

6. 生成配置文件(下载下来，双击安装，和手机绑定)

二进制数，前缀是0b

八进制数，前缀是0o

十六进制数，前缀是0x

类型别名:使用 typealias给现有类型定义另外一个名字，

typealias myInt = Int

元组:（tuples）:把多个值组合成一个复合值。元组内的值可以是任意类型, 通过下标(从0开始)来访问元组中的单个元素

可以将一个元组的内容分解（decompose）成单独的常量和变量:

let (statusCode, statusMessage) = (404, "Not Found")

可选类型（？）：处理值可能缺失的情况

nil不能用于非可选的常量和变量,

let a = nil报错，应该let a: Sting? =nil

注意：Swift 的 nil 和 OC 中的 nil 并不一样。在 OC 中，nil 是一个指向不存在对象的指针。在 Swift 中，nil 不是指针——它是一个确定的值，用来表示值缺失。任何类型的可选状态都可以被设置为 nil，不只是对象类型

当确定可选类型有值后，用感叹号（！）来取值---强制解析（注意：

使用 ! 来获取一个不存在的可选值会导致运行时错误）

错误处理: 一个函数可以通过在声明中添加throws关键词来抛出错误消息

对负数求余时，符号会被忽略， 9 % -4 =》1，-9 % 4= -1

两个元组的元素相同，且长度相同的话，元组就可以被比较