系统定位

1. iOS8以前使用CLLocationManager：
2. 导入头文件<CoreLocation/CoreLocation>
3. 创建位置管理者CLLocationManager，并添加到属性
4. 设置代理，遵守协议、实现协议方法，再代理方法中获取位置信息
5. 调用开始更新位置方法
6. 设置每隔多远定位一次和精确度，精确度越高越耗电，定位时间越长
7. 请求授权，iOS6后，苹果开始加强保护用户隐私，在info.plist文件中定义key来提醒用户，提高用户允许定位的概率

Privacy-Location Usage Description (String，Ios8以前定位使用说明)

1. 如果要后台定位，需打开后台模式：Background Modes => Location updates
2. iOS8之后使用CLLocationManager
3. Ios8以后，苹果进一步加强隐私保护，不会主动弹出对话框，需实现request方法，并在plist中设置相应的key，才会弹出对话框
4. 当程序当前授权状态为未决定时，在前台时请求定位服务许可时使用requestWhenInUseAuthorization。需在plist文件中设置一个key：NSLocationWhenInUseUsageDescription，若不设置，系统忽略定位请求
5. 当用户授权when-in-use时，程序在前台可启用大部分定位服务，若想后台定位，需开启后台定位模式，但在状态栏会出现蓝条提示用户程序正在定位
6. 当程序当前授权状态为未决定时，请求前后台定位服务授权时使用requestAlwaysAuthorization，必须在plist中设置一个key:NSLocationAlwaysUsageDescription
7. 注意：
8. iOS8之后，想要定位，必须调用其中一个request授权方法。
9. 如果两个授权方法都执行，会有如下情况：

when-in-use写前面，第一次打开程序请求授权，如果勾选了后台模式，进入后台会出现蓝条提示正在定位。第二次启动程序时always请求授权，之后进入后台就不会出现蓝条了；always写前面，只会请求一次授权，因为when-in-use不会执行，因为always授权已获得所有定位能力。

1. 判断是否开启了定位服务

在定位前要先判断是否开启了定位服务，还有用户是否允许定位。

1. 适配版本号的方法

when-in-use和always是Ios8之后的方法，在Ios7会crash，此时需作判断：

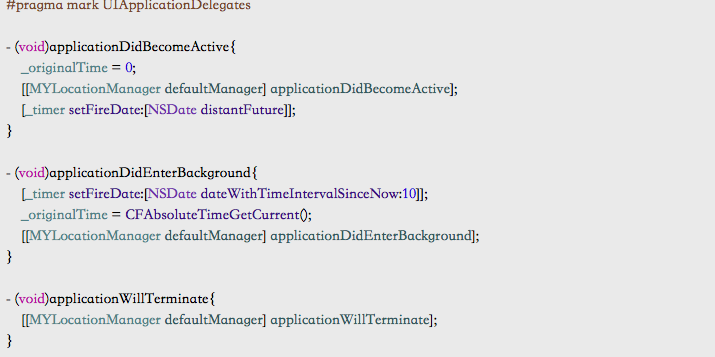
1)[[UIDevice currentDevice].systemVersion floadValue] >= 8.0

2)[\_locationMa responseToSelector:@selector(requestWhenInUseAuthorization)]

1. 代理方法的回调信息

当位置管理器获取到位置后，调用locationManager:didUpdateLocations方法，返回location信息

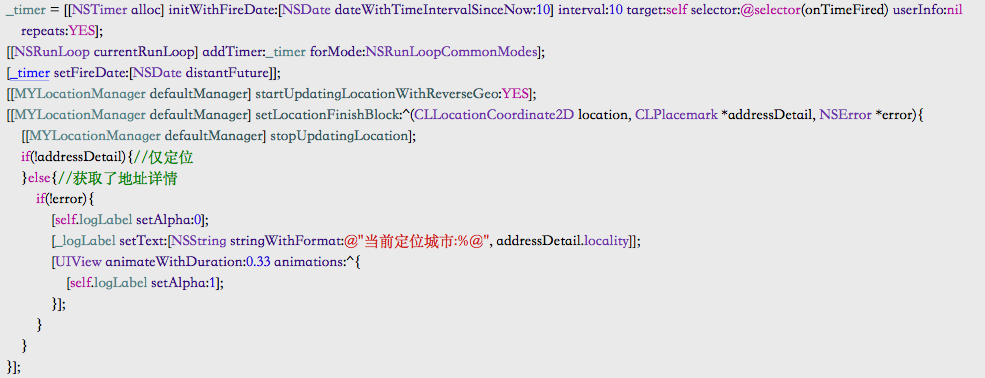
1. coordinate:当前位置坐标 latitude:纬度 longitude:精度
2. altitude:海拔，高度
3. horizontalAccuracy:纬度和经度的精度
4. verticalAccuracy:垂直精度(获取不到海拔时为负数)
5. course:行进方向，0为真北
6. speed:以米/秒为单位的行进速度
7. description:位置描述信息
8. iOS9.0之后使用CLLocationManager
9. iOS9之后有一种新的请求定位方法requestLocation，作用是：按照定位精确度从低到高排序，逐个进行定位，如果取到的位置不是精确度最高的那个，也会在定位超时后，通过代理告诉外界。
10. requestLocation必须实现didUpdateLocation和didFailWithError方法，但只会调用一次
11. 使用requestLocation就不能同时使用startUpdateLocation
12. 实现requestWhenInUseAuthorization或requestAlwaysAuthorization方法，并设好相应的key
13. 默认在前台授权模式下不能后台定位，即使已勾选后台模式，还需设置allowsBackgroundLocationUpdates属性为YES，需使用-responsedToSelector判断，当定位完成时，设置为NO，并且不再定位跟踪
14. 后台定位练习及MYLocationManager管理类封装
15. 后台定位
16. 导入CoreLocation静态库，引入<CoreLocation/CoreLocation.h>
17. 工程文件=>Target=>Background Modes开启Location updates服务
18. info.plist添加铅后台定位授权声明：NSLocationAlwaysUsageDescription
19. 在AppDelegate代理方法里做相应定位处理



1. 开启定时器NSTimer，添加到运行循环
2. 实现定时器触发方法



1. 设置定位回调：



1. MYLocationManager管理类封装

