

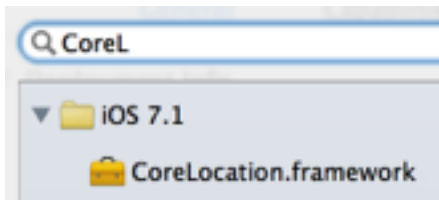
简介

- 在移动互联网时代，移动app能解决用户的很多生活琐事，比如
 - p 导航：去任意陌生的地方
 - p 周边：找餐馆、找酒店、找银行、找电影院
- 在上述应用中，都用到了地图和定位功能，在iOS开发中，要想加入这2大功能，必须基于2个框架进行开发
 - p Map Kit：用于地图展示
 - p Core Location：用于地理定位
- 2个热门专业术语
 - p LBS：Location Based Service
 - p SoLoMo：Social Local Mobile（索罗门）

CoreLocation框架的使用

- CoreLocation框架使用前提

- p 导入框架



- p 导入主头文件

```
#import <CoreLocation/CoreLocation.h>
```

- CoreLocation框架使用须知

- p CoreLocation框架中所有数据类型的前缀都是CL

- p CoreLocation中使用CLLocationManager对象来做用户定位

CLLocationManager

- CLLocationManager的常用操作

- p 开始用户定位

- ✓ – (void)startUpdatingLocation;

- p 停止用户定位

- ✓ – (void) stopUpdatingLocation;

- 当调用了startUpdatingLocation方法后，就开始不断地定位用户的位置，中途会频繁地调用代理的下面方法

- p – (void)locationManager:(CLLocationManager *)manager
didUpdateLocations:(NSArray *)locations;

- ✓ locations参数里面装着CLLocation对象

CLLocation

- CLLocation用来表示某个位置的地理信息，比如经纬度、海拔等等
- `@property(readonly, nonatomic) CLLocationCoordinate2D coordinate;`
 - ✓ 经纬度
- `@property(readonly, nonatomic) CLLocationDistance altitude;`
 - ✓ 海拔
- `@property(readonly, nonatomic) CLLocationDirection course;`
 - ✓ 路线，航向（取值范围是0.0° ~ 359.9°，0.0°代表真北方向）
- `@property(readonly, nonatomic) CLLocationSpeed speed;`
 - ✓ 行走速度（单位是m/s）
- 用- (CLLocationDistance)distanceFromLocation:(const CLLocation *)location方法可以计算2个位置之间的距离

CLLocationManager

- `@property(assign, nonatomic) CLLocationDistance distanceFilter;`
 - p 每隔多少米定位一次
- `@property(assign, nonatomic) CLLocationAccuracy desiredAccuracy;`
 - p 定位精确度（越精确就越耗电）

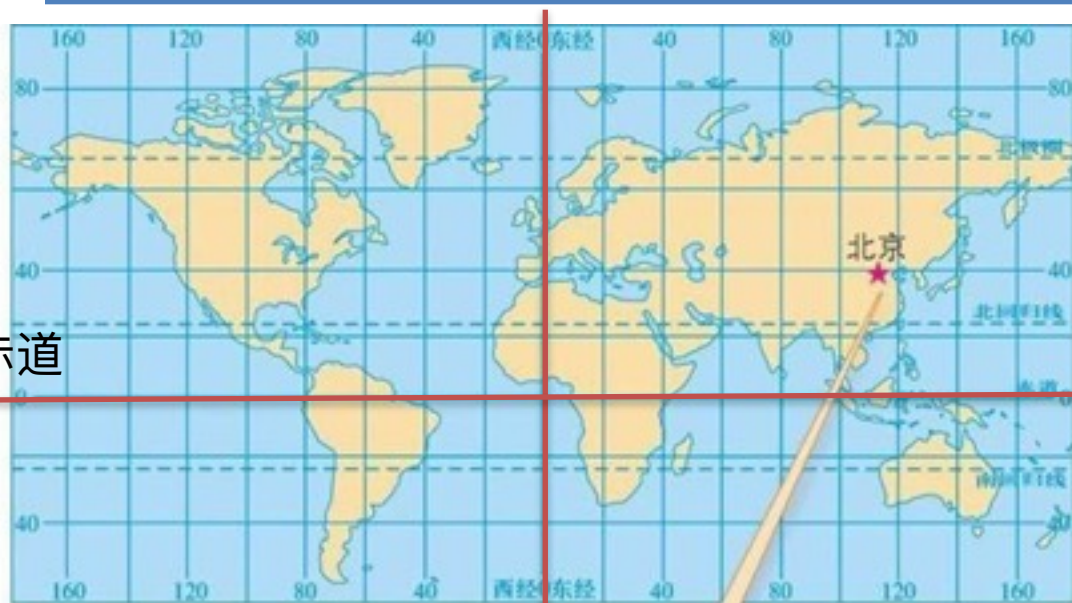
CLLocationCoordinate2D

- CLLocationCoordinate2D是一个用来表示经纬度的结构体，定义如下

```
typedef struct {  
    CLLocationDegrees latitude; // 纬度  
    CLLocationDegrees longitude; // 经度  
} CLLocationCoordinate2D;
```

- 一般用CLLocationCoordinate2DMake函数来创建CLLocationCoordinate2D

经纬度



0°经线，本初子午线

- 本初子午线
 - 穿过英国伦敦格林文治天文台
 - 往东边（右边）走，是东经（E）
 - 往西边（左边）走，是西经（W）
 - 东西经各180°，总共360°
- 赤道
 - 往北边（上边）走，是北纬（N）
 - 往南边（下边）走，是南纬（S）
 - 南北纬各90°，总共180°
- 横跨经度\纬度越大（1° ≈ 111km）
 - 表示的范围就越大
 - 在地图上看到的東西就越小

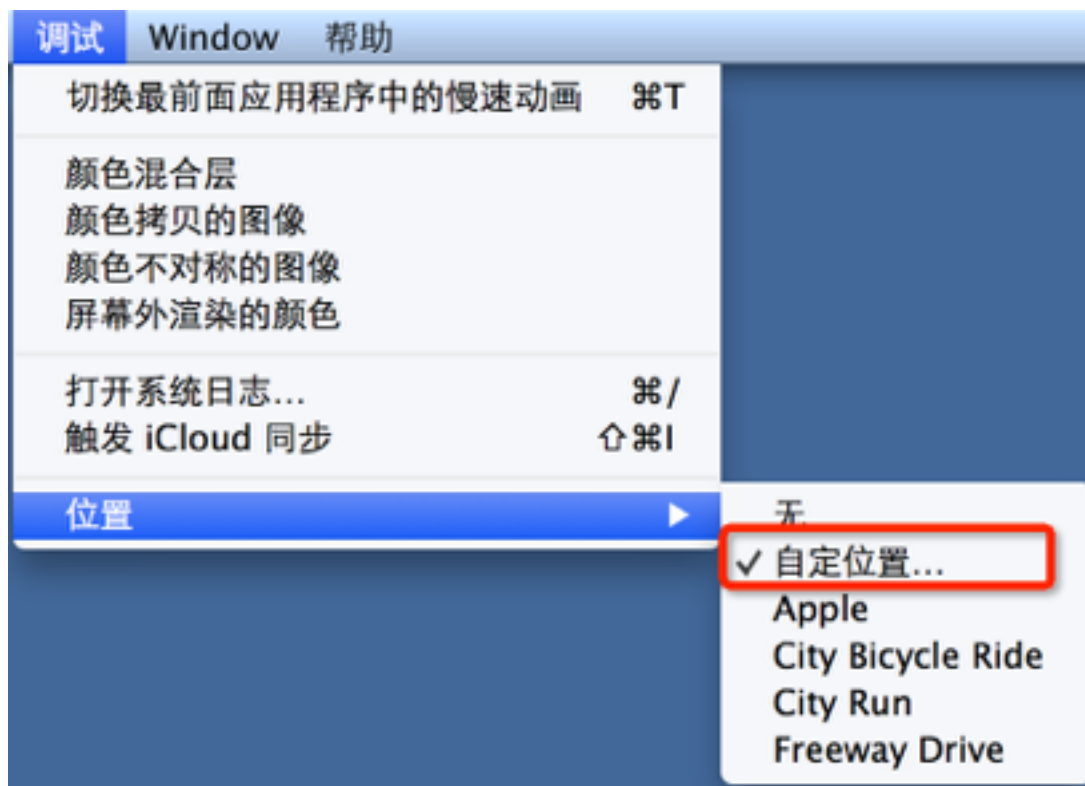
中国经纬度

- 中国的经纬度范围
 - 纬度范围：N 3°51' ~ N 53°33'
 - 经度范围：E 73°33' ~ E 135°05'
- 部分城市的经纬度

城市	经度	纬度
北京	E116°28'	N39°54'
广州	E113°15'	N23°08'
成都	E104°05'	N30°39'
上海	E121°29'	N31°14'
深圳	E113°46'	N22°27'
武汉	E114°21'	N30°37'
郑州	E113°42'	N34°48'

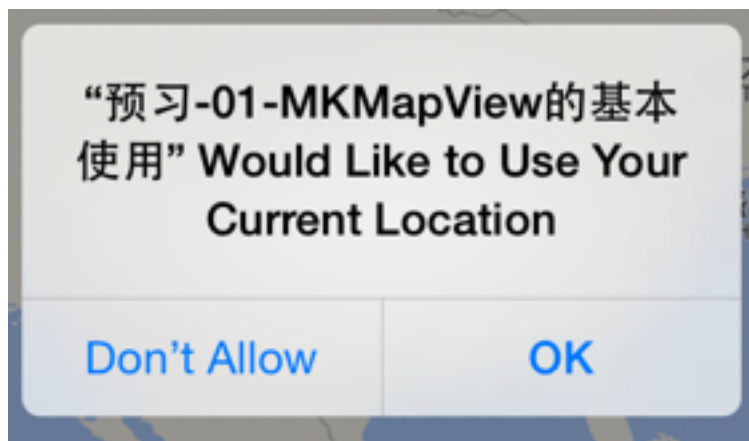
模拟位置

- 如果是模拟器，需要设置模拟位置（经纬度）
- 帝都的经纬度是：北纬40°，东经116°



用户隐私的保护

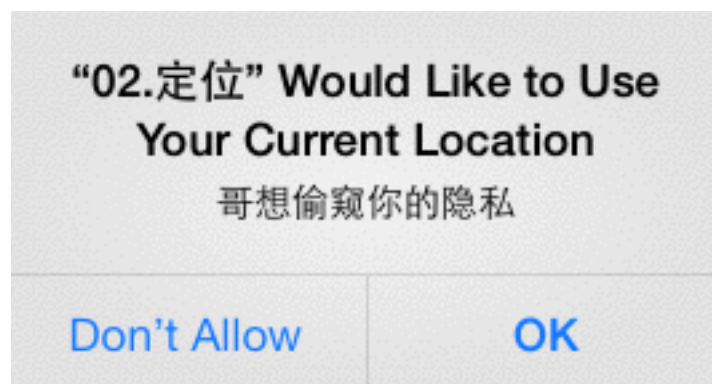
- 从iOS 6开始，苹果在保护用户隐私方面做了很大的加强，以下操作都必须经过用户批准授权
 - p 要想获得用户的位置
 - p 想访问用户的通讯录、日历、相机、相册等等
- 当想访问用户的隐私信息时，系统会自动弹出一个对话框让用户授权



用户隐私的保护

- 开发者可以在Info.plist中设置NSLocationUsageDescription说明定位的目的(Privacy - Location Usage Description)

▼ Information Property List	Dictionary	(15 items)
Privacy - Location Usage Description	String	哥想偷窥你的隐私
Localization native development region	String	en
Bundle display name	String	\${PRODUCT_NAME}



- 一旦用户选择了“Don't Allow”，意味着你的应用以后就无法使用定位功能
- 为了严谨起见，最好在使用定位功能之前判断当前应用的定位功能是否可用
- CLLocationManager有个类方法可以判断当前应用的定位功能是否可用
- ✓ + (BOOL)locationServicesEnabled;

CLGeocoder

- 使用CLGeocoder可以完成“地理编码”和“反地理编码”
 - p 地理编码：根据给定的地名，获得具体的位置信息（比如经纬度、地址的全称等）
 - p 反地理编码：根据给定的经纬度，获得具体的位置信息
- 地理编码方法
 - p – (void)geocodeAddressString:(NSString *)addressString
completionHandler:(CLGeocodeCompletionHandler)completionHandler;
- 反地理编码方法
 - p – (void)reverseGeocodeLocation:(CLLocation *)location
completionHandler:(CLGeocodeCompletionHandler)completionHandler;

CLGeocodeCompletionHandler

- 当地理\反地理编码完成时，就会调用CLGeocodeCompletionHandler
- `typedef void (^CLGeocodeCompletionHandler)(NSArray *placemarks, NSError *error);`
- 这个block传递2个参数
- ✓ `error` : 当编码出错时（比如编码不出具体的信息）有值
- ✓ `placemarks` : 里面装着CLPlacemark地标对象

CLLocationacemark

- CLLocationacemark的字面意思是地标，封装详细的地址位置信息

p @property (nonatomic, readonly) CLLocation *location;

✓ 地理位置

p @property (nonatomic, readonly) CLRegion *region;

✓ 区域

p @property (nonatomic, readonly) NSDictionary *addressDictionary;

✓ 详细的地址信息

p @property (nonatomic, readonly) NSString *name;

✓ 地址名称

p @property (nonatomic, readonly) NSString *locality;

✓ 城市

