



# iOS SDK公开课

## 第二课 附近搜索

# 重新发现



# 目录

- 了解POI搜索
- 附近搜索
  - 添加附近搜索按钮
  - 请求附近搜索
  - 显示搜索结果
- 为POI点添加annotation

# 了解POI搜索

兴趣点 ( POI , Point of Interest )

## 搜索POI接口

```
- (void)AMapPlaceSearch:(AMapPlaceSearchRequest *)request;
```

## AMapPlaceSearchRequest关键属性

```
@property (nonatomic, assign) AMapSearchType searchType;  
// 默认为AMapSearchType_PlaceKeyword
```

## 四种搜索类型

AMapSearchType_PlaceID	按POI ID搜索
AMapSearchType_PlaceKeyword	按关键字搜索
AMapSearchType_PlaceAround	按关键字进行附近搜索
AMapSearchType_PlacePolygon	按关键字进行自定义范围搜索

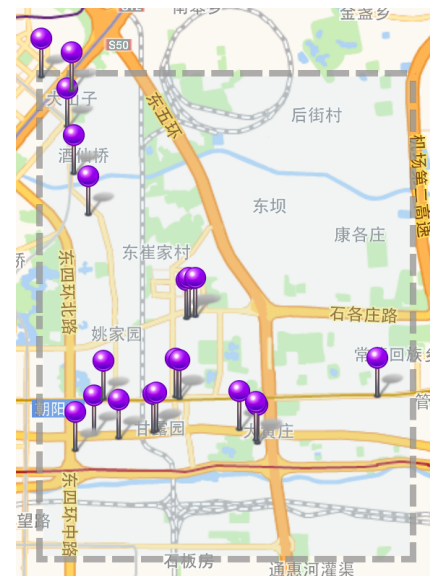
## 按关键字进行周边搜索

以某一位置为圆心，搜索方圆n米内的POI点



## 按关键字进行自定义范围搜索

自定义搜索范围进行POI点搜索，自定义范围可为任意凸多边形



# 了解POI搜索

## AMapPlaceSearchRequest 用法

### 按POI ID搜索      AMapSearchType\_PlaceID

根据POI ID搜索，一般是在使用其他搜索方式得到POI后，取POI ID做进一步搜索，获取POI的详细信息  
搜索时，设置requireExtension为YES。

### 按关键字搜索      AMapSearchType\_PlaceKeyword

需要设置以下参数

```
@property (nonatomic, copy) NSString *keywords;           // 查询关键字，多个关键字用“|”分割，“空格”表示与，“双引号”表示不可分割
@property (nonatomic, strong) NSArray *city;               // 城市数组，可选值：cityname（中文或中文全拼）、citycode、adcode
//若city为空，则返回结果在suggestion中，返回多个城市的poi数据
```

### 附近搜索      AMapSearchType\_PlaceAround

需要设置以下参数

```
@property (nonatomic, copy) NSString *keywords;           // 查询关键字，多个关键字用“|”分割，“空格”表示与，“双引号”表示不可分割
@property (nonatomic, copy) AMapGeoPoint *location;        // 中心点坐标
@property (nonatomic, assign) NSInteger radius;             // 查询半径，单位：米 [default = 3000]
```

### 自定义范围搜索      AMapSearchType\_PlacePolygon

需要设置以下参数

```
@property (nonatomic, copy) NSString *keywords;           // 查询关键字，多个关键字用“|”分割，“空格”表示与，“双引号”表示不可分割
@property (nonatomic, copy) AMapGeoPolygon *polygon;       // 自定义搜索范围
```

## 附近搜索

### 添加附近搜索按钮

```
- (void)initControls
{
    //.....
    UIButton *searchButton = [UIButton buttonWithType:UIButtonTypeRoundedRect];
    searchButton.frame = CGRectMake(80, CGRectGetHeight(_mapView.bounds) - 80, 40, 40);
    searchButton.autoresizingMask = UIViewAutoresizingFlexibleRightMargin |
    UIViewAutoresizingFlexibleTopMargin;
    searchButton.backgroundColor = [UIColor whiteColor];
    [searchButton setImage:[UIImage imageNamed:@"search"] forState:UIControlStateNormal];

    [searchButton addTarget:self action:@selector(searchAction)
    forControlEvents:UIControlEventTouchUpInside];

    [_mapView addSubview:searchButton];
}
```

### 请求附近搜索

```
- (void)searchAction
{
    if (_currentLocation == nil || _search == nil)
    {
        NSLog(@"search failed");
        return;
    }

    AMapPlaceSearchRequest *request = [[AMapPlaceSearchRequest alloc] init];
    request.searchType = AMapSearchType_PlaceAround;
    request.location = [AMapGeoPoint locationWithLatitude:_currentLocation.coordinate.latitude
    longitude:_currentLocation.coordinate.longitude];

    request.keywords = @"餐饮";

    [_search AMapPlaceSearch:request];
}
```

## 附近搜索

显示搜索结果 以tableView的方式

```
_tableView变量, 实现UITableViewDataSource, UITableViewDelegate
@interface ViewController ()<MAMapViewDelegate, AMapSearchDelegate, UITableViewDataSource,
UITableViewDelegate>
{
    //...
    UITableView *_tableView;
    NSArray *_pois;
}
初始化_tableView
- (void)initTableView
{
    CGFloat halfHeight = CGRectGetHeight(self.view.bounds) * 0.5;

    _tableView = [[UITableView alloc] initWithFrame:CGRectMake(0, halfHeight,
CGRectGetWidth(self.view.bounds), halfHeight) style:UITableViewStylePlain];
    _tableView.delegate = self;
    _tableView.dataSource = self;

    [self.view addSubview:_tableView];
}
```

在搜索回调方法中得到搜索结果

```
- (void)onPlaceSearchDone:(AMapPlaceSearchRequest *)request response:
(AMapPlaceSearchResponse *)response
{
    NSLog(@"request: %@", request);
    NSLog(@"response: %@", response);
    if (response.pois.count > 0)
    {
        _pois = response.pois;

        [_tableView reloadData];
    }
}
```

## 附近搜索

### 利用tableView显示搜索结果

```
#pragma mark - UITableViewDataSource

- (UITableViewCell *)tableView:(UITableView *)tableView cellForRowAtIndexPath:(NSIndexPath *)indexPath
{
    static NSString *cellIdentifier = @"cellIdentifier";

    UITableViewCell *cell = [tableView dequeueReusableCellWithIdentifier:cellIdentifier];

    if (cell == nil)
    {
        cell = [[UITableViewCell alloc] initWithStyle:UITableViewCellStyleSubtitle reuseIdentifier:cellIdentifier];
    }

    AMapPOI *poi = _pois[indexPath.row];

    cell.textLabel.text = poi.name;
    cell.detailTextLabel.text = poi.address;

    return cell;
}

- (NSInteger)tableView:(UITableView *)tableView numberOfRowsInSection:(NSInteger)section
{
    return _pois.count;
}
```



## 附近搜索



艾尚比萨宅急送

望京

方恒假日酒店大宴会厅

阜通东大街6号院3号楼方恒假日酒店2层

锦绣餐厅(北京方恒假日酒店北)

望京阜通东大街6号院3号楼北京方恒假日酒店6层

盛世唐宫(方恒购物中心)

望京阜通东大街6号院4号楼F3层

火宴山(方恒购物中心)

望京阜通东大街6号院4号楼F4层

食尚坊美食汇

阜通东大街6号方恒购物中心负1层(物美生活超市望...

泰油能餐吧

# annotation标记

## MAAnnotation

用作地图上的标记 只提供数据

## MAAnnotationView

是地图上标记的样式 只用于显示

## 用法

1 添加annotation

MAMapView中

```
-(void)addAnnotation:(id<MAAnnotation>)annotation;
```

2 实现MAMapView的回调方法

```
-(MAAnnotationView *)mapView:(MAMapView *)mapView  
viewForAnnotation:(id<MAAnnotation>)annotation;
```

## 松耦合

MAAnnotation和MAAnnotationView是松耦合的。

## 原因

MAAnnotationView有重用机制，能够大大降低内存使用。

dequeue

重用，当annotation出现在地图可视区域内

recycle

回收，当annotation移出地图可视区域内

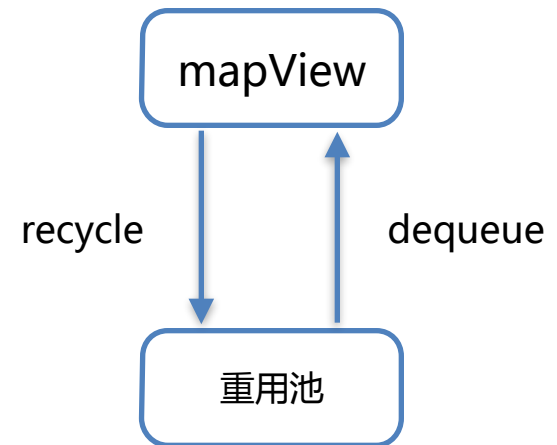
## 怎么使用重用机制

在实现MAMapView的回调方法时，

1 从重用池中取reuseIndetifier对应的AnnotationView;

2 若没有可重用的AnnoationView，再创建一个新的AnnotationView。

```
static NSString *reuseIndetifier = @"annotationReuseIndetifier";  
MAPinAnnotationView *annotationView = (MAPinAnnotationView*)[mapView  
dequeueReusableAnnotationViewWithIdentifier:reuseIndetifier];  
if (annotationView == nil)  
{  
    annotationView = [[MAPinAnnotationView alloc] initWithAnnotation:annotation  
reuseIndetifier:reuseIndetifier];  
}
```



# 为POI点添加annotation标记

在选中TableView某一行POI时，在地图上添加标记

1 使用TableView的delegate回调函数，得到annotation坐标

```
- (void)tableView:(UITableView *)tableView didSelectRowAtIndexPath:(NSIndexPath *)indexPath
{
    [tableView deselectRowAtIndexPath:indexPath animated:YES];

    // 为点击的poi点添加标注
    AMapPOI *poi = _pois[indexPath.row];

    MAPPointAnnotation *annotation = [[MAPPointAnnotation alloc] init];
    annotation.coordinate = CLLocationCoordinate2DMake(poi.location.latitude, poi.location.longitude);
    annotation.title = poi.name;
    annotation.subtitle = poi.address;

    [_mapView addAnnotation:annotation];
}
```

添加annotation

2 在MAMapView的回调函数中，取得或创建annotationView，并配置annotationView的属性

```
- (MAAnnotationView *)mapView:(MAMapView *)mapView viewForAnnotation:(id<MAAnnotation>)annotation
{
    if ([annotation isKindOfClass:[MAPPointAnnotation class]])
    {
        static NSString *reuseIdentifier = @"annotationReuseIdentifier";
        MAPinAnnotationView *annotationView = (MAPinAnnotationView*)[mapView
        dequeueReusableAnnotationViewWithIdentifier:reuseIdentifier];
        if (annotationView == nil)
        {
            annotationView = [[MAPinAnnotationView alloc] initWithAnnotation:annotation
            reuseIdentifier:reuseIdentifier];
        }

        annotationView.canShowCallout = YES;

        return annotationView;
    }

    return nil;
}
```

重用或创建  
annotationView

设置属性

# 为POI点添加annotation标记

在获得新的搜索结果时，需要清空原有的annotation

记录annotation数组

```
@interface ViewController ()<MAMapViewDelegate, AMapSearchDelegate, UITableViewDataSource, UITableViewDelegate>
{
    NSMutableArray *_annotations;
}
- (void)initAttributes
{
    _annotations = [NSMutableArray array];
    _pois = nil;
}
- (void)viewDidLoad {
    [super viewDidLoad];
}
//...
[self initAttributes];
}

- (void)tableView:(UITableView *)tableView didSelectRowAtIndexPath:(NSIndexPath *)indexPath
{
    //...
    MAPointAnnotation *annotation = [[MAPointAnnotation alloc] init];
    annotation.coordinate = CLLocationCoordinate2DMake(poi.location.latitude, poi.location.longitude);
    //...
    [_annotations addObject:annotation];
}
}
```

在搜索回调中清空annotation

```
- (void)onPlaceSearchDone:(AMapPlaceSearchRequest *)request response:(AMapPlaceSearchResponse *)response
{
    NSLog(@"request: %@", request);
    NSLog(@"response: %@", response);

    if (response.pois.count > 0)
    {
        //...
        // 清空标注
        [_mapView removeAnnotations:_annotations];
        [_annotations removeAllObjects];
    }
}
```

## 为POI点添加Annotation



艾尚比萨宅急送  
望京

方恒假日酒店大宴会厅  
阜通东大街6号院3号楼方恒假日酒店2层

锦绣餐厅(北京方恒假日酒店北)  
望京阜通东大街6号院3号楼北京方恒假日酒店6层

盛世唐宫(方恒购物中心)  
望京阜通东大街6号院4号楼F3层

火宴山(方恒购物中心)  
望京阜通东大街6号院4号楼F4层

食尚坊美食汇  
阜通东大街6号方恒购物中心负1层(物美生活超市望...)

泰油能餐吧



艾尚比萨宅急送  
望京

方恒假日酒店大宴会厅  
阜通东大街6号院3号楼方恒假日酒店2层

锦绣餐厅(北京方恒假日酒店北)  
望京阜通东大街6号院3号楼北京方恒假日酒店6层

盛世唐宫(方恒购物中心)  
望京阜通东大街6号院4号楼F3层

火宴山(方恒购物中心)  
望京阜通东大街6号院4号楼F4层

食尚坊美食汇  
阜通东大街6号方恒购物中心负1层(物美生活超市望...)

泰油能餐吧



艾尚比萨宅急送  
望京

方恒假日酒店大宴会厅  
阜通东大街6号院3号楼方恒假日酒店2层

锦绣餐厅(北京方恒假日酒店北)  
望京阜通东大街6号院3号楼北京方恒假日酒店6层

盛世唐宫(方恒购物中心)  
望京阜通东大街6号院4号楼F3层

火宴山(方恒购物中心)  
望京阜通东大街6号院4号楼F4层

食尚坊美食汇  
阜通东大街6号方恒购物中心负1层(物美生活超市望...)

泰油能餐吧

# 作业

- 1 选中annotation时，把选中的annotation移到mapView中间
- 2 点击搜索结果之后，不要重复添加annotation

# coming soon...

自定义AnnotationView





# Thanks!

高德LBS开发者论坛 <http://lbsbbs.amap.com/>

本课程源码下载 [https://github.com/hadesh/HelloAmap/tree/lesson\\_02](https://github.com/hadesh/HelloAmap/tree/lesson_02)