

定位功能

android 获取位置主要有 2 种方法：

1. 使用系统提供的原生 API：主要是系统的 android.location 中提供的类获取经纬度，并用 geocoder 或 geocoding api 根据经纬度解析成位置。
 2. 使用百度、高德等地图 SDK 获取位置
- 本文使用第 2 中，使用百度 SDK 来获取位置

实现步骤

1. 进入百度地图开放平台，网址：<http://lbsyun.baidu.com/>



2. 选择 开发文档-> Android 定位 SDK



3. 点击“申请密钥（AK）”或进入网址：
“<http://lbsyun.baidu.com/apiconsole/key?application=key>” 申请密钥

Android定位SDK

服务介绍

获取密钥>>

开发指南

类参考>>

更新日志

产品下载

常见问题

相关产品链接

产品简介

百度地图Android定位SDK是为Android移动端应用提供的一套简单易用的定位服务接口，专注于为广大开发者提供最好的综合定位服务。通过使用百度定位SDK，开发者可以轻松为应用程序实现智能、精准、高效的定位功能。

该套定位SDK**免费对外开放，接口使用无次数限制**。在使用前，**您需先申请密钥（AK）**才可使用。在您使用百度地图Android定位SDK之前，请先阅读**百度地图API使用条款**。

产品优势

百度地图Android定位SDK提供GPS、基站、Wi-Fi、地磁、蓝牙、传感器等多种定位方式，**适用于室内、室外多种定位场景**，具有出色的定位性能：定位精度高、覆盖率高、网络定位请求流量小、定位速度快。

目前百度已覆盖全国4000多家商场、交通枢纽、医院、场馆等，如百度未覆盖您的室内地图数据，您可以**联系我们**。
室内建筑商铺更新搬家等会直接影响到室内定位的准确度，目前百度会定期对室内建筑定位数据进行更新，如您有其他需求，可**联系我们**。

整体定位成功率高达99.6%
GPS：10M（取决于芯片）

全网络覆盖

4. 没有账号需要先注册

注册开发者

百度地图开放平台开发者注册

* 姓名：

请填写姓名

* 手机：

请填写手机号

获取验证码

验证码

* 邮箱：

请填写邮箱

☒

我已阅读并同意《百度地图开放平台服务条款》

提交

取消

友情提示：

此处填写的手机号、邮箱将被用于开放平台对您发送：1.配额预警通知，2.产品或服务升级等重要通知，为避免您错过重要通知导致服务受到影响，请谨慎填写此处信息。

5. 在应用列表中创建应用

百度地图开放平台

首页 功能与服务 解决方案 开发文档 反馈与帮助 服务升级 NEW! 控制台

我的应用

查看应用

创建应用

回收站

我的配额

配额管理

配额申请记录

应用列表

请输入AK 搜索

创建应用

回收站

每页显示30条

应用编号	应用名称	访问应用（AK）	应用类别	备注信息（双击更改）	应用配置
14658459	Weather	to32qmEI7rvaY2UVKvXdh5bmPg...	Android端		设置 删除
9646635	地图	ymnutZVdLQsuecfL7wdGGjb9TDn...	服务端		设置 删除

您当前创建了 2 个应用

< 1 >

6. 完善信息

➔ 创建应用

应用名称: ✔ 输入正确

应用类型:

启用服务:

☒ 云检索	☒ 正逆地理编码	☒ Android地图SDK
☒ Android定位SDK	☒ Android导航离线SDK	☒ Android导航SDK
☒ 静态图	☒ 全景静态图	☒ 坐标转换
☒ 鹰眼轨迹	☒ 全景URL API	☒ Android导航 HUD SDK
☒ 云逆地理编码	☒ 云地理编码	☒ 推荐上车点

*发布版SHA1: ! 请输入

开发版SHA1:

*包名:

安全码: 输入sha1和包名后自动生成

Android SDK安全码组成: SHA1+包名。(查看详细配置方法) 新申请的Mobile与Browser类型的ak不再支持云存储接口的访问, 如要使用云存储, 请申请Server类型ak。

提交

其中, 应用类型选择 Android SDK, SHA1 通过在 android studio 中的 Terminal, 输入

```
keytool -list -v -keystore ~/.android/debug.keystore -alias androiddebugkey
```

在提示输入密钥口令时直接回车, 不需要输入, 就会得到 SHA1

```
MD5:  
SHA1:  
SHA256:  
签名算  
版本:
```

7. 复制 AK

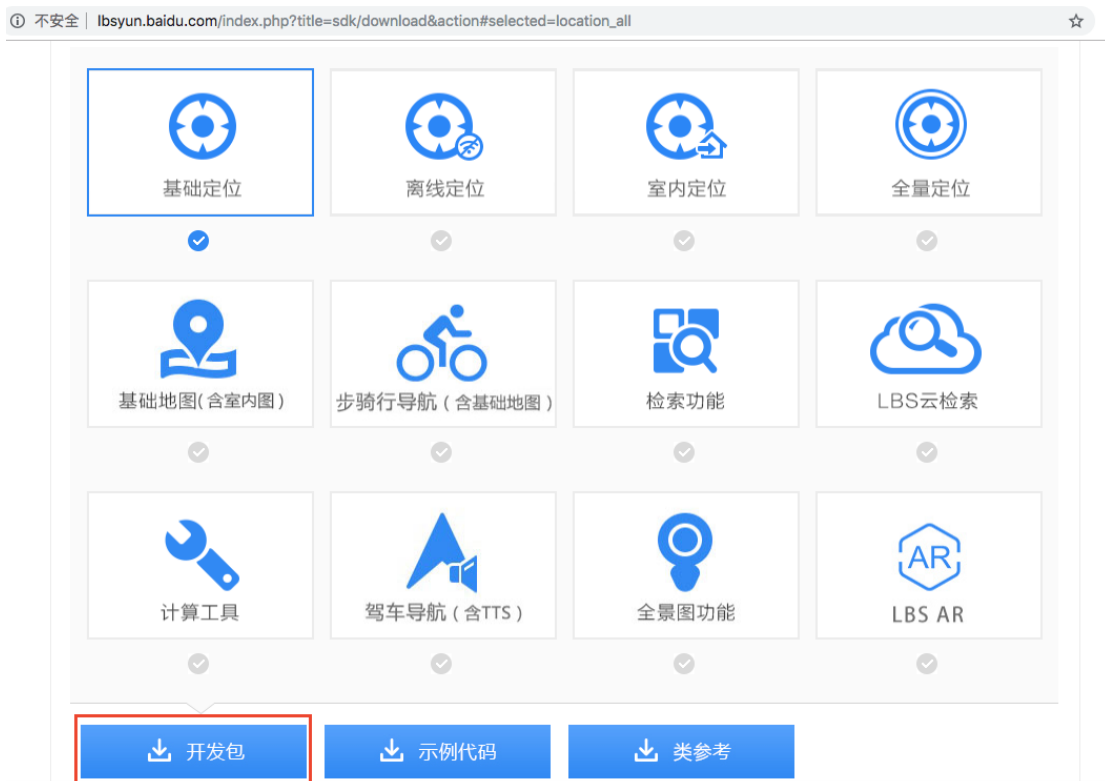
创建应用后即可获得密钥

应用编号	应用名称	访问应用 (AK)	应用类别	备注信息 (双击更改)	应用配置
14658459	Weather		Android端		设置 删除
9646635	地图		服务端		设置 删除

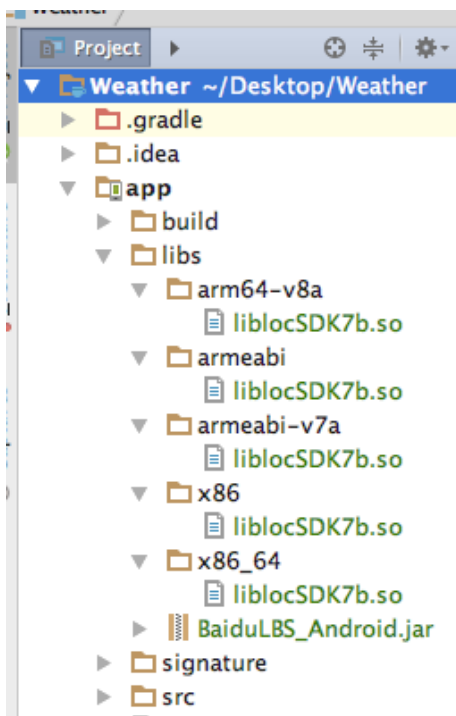
8. 下载开发包

http://lbsyun.baidu.com/index.php?title=sdk/download&action#selected=location_all

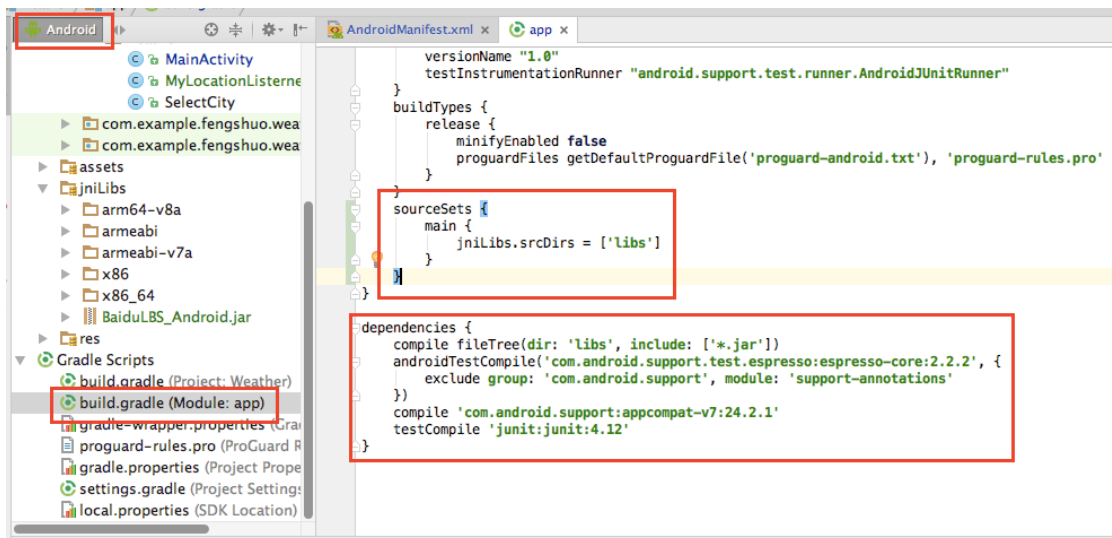
下载开发包



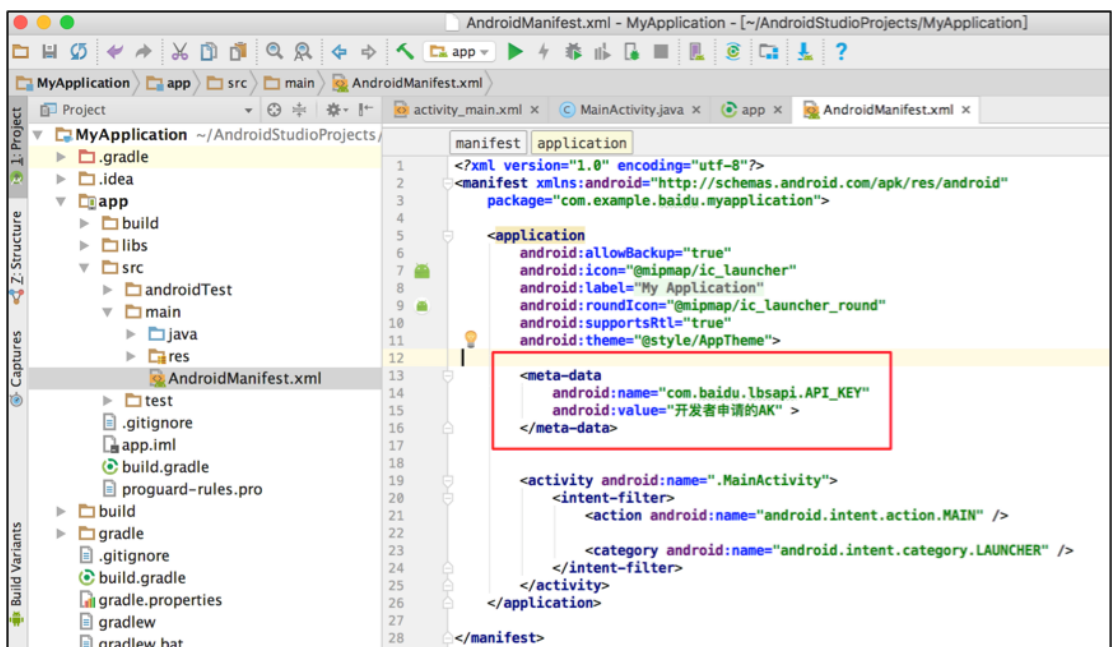
9. 将下载好的 SDK 放置工程相应位置



10. 配置 build.gradle 文件



11. 添加 AK



12. 添加定位权限

使用定位 SDK，需在 Application 标签中声明 service 组件，每个 App 拥有自己单独的定位 service，代码如下：

```

<service
    android:name="com.baidu.location.f"
    android:enabled="true"
    android:process=":remote">
</service>

```

除添加 service 组件外，使用定位 SDK 还需添加如下权限：

```

<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION"/>
<uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION"/>

<uses-permission android:name="android.permission.ACCESS_WIFI_STATE"/>
<uses-permission android:name="android.permission.CHANGE_WIFI_STATE"/>
<uses-permission android:name="android.permission.READ_PHONE_STATE"/>
<uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE"/>
<uses-permission android:name="android.permission.MOUNT_UNMOUNT_FILESYSTEMS"/>

```

13. 初始化 LocationClient 类

```

public class MainActivity extends Activity implements View.OnClickListener{

    private static final int UPDATE_TODAY_WEATHER=1;
    private static final int DB=2;

    private ImageView mUpdateBtn,mTitleShare,mTitleLocation;

    private ImageView mCitySelect;

    private TextView cityTv,timeTv,humidityTv,weekTv,pmDataTv,pmQualityTv,
        temperatureTv,climateTv,windTv,city_name_Tv,nowTv;
    private ImageView weatherImg,pmImg;

    private ProgressBar mUpdateProgress;

    private LocationManager locationManager;

    public LocationClient mLocationClient=null;
    private MyLocationListener myListener=new MyLocationListener();
}

```

在 onCreate()中添加

```

mLocationClient=new LocationClient(getApplicationContext());
//声明LocationClient类
mLocationClient.registerLocationListener(myListener);
//注册监听函数

```

14. 配置定位 sdk 参数

```

private void initLocation(){
    LocationClientOption option = new LocationClientOption();
    option.setLocationMode(LocationClientOption.LocationMode.High_Accuracy);
    //可选，默认高精度，设置定位模式，高精度，低功耗，仅设备
    option.setCoorType("bd09ll");
    //可选，默认gcj02，设置返回的定位结果坐标系
    int span = 1000;
    option.setScanSpan(0);
    //可选，默认0，即仅定位一次，设置发起定位请求的间隔需要大于等于1000ms才是有效的
    option.setIsNeedAddress(true);
    //可选，设置是否需要地址信息，默认不需要
    option.setOpenGps(true);
    //可选，默认false，设置是否使用gps
    option.setLocationNotify(true);
    //可选，默认false，设置是否当gps有效时按照1S1次频率输出GPS结果
    option.setIsNeedLocationDescribe(true);
    //可选，默认false，设置是否需要位置语义化结果，
    // 可以在BDLocation.getLocationDescribe里得到，结果类似于“在北京天安门附近”
    option.setIsNeedLocationPoiList(true);
    //可选，默认false，设置是否需要POI结果，可以在BDLocation.getPoiList里得到
    option.setIgnoreKillProcess(false);
    //可选，默认true，定位SDK内部是一个SERVICE，并放到了独立进程，
    // 设置是否在stop的时候杀死这个进程，默认不杀死
    option.SetIgnoreCacheException(false);
    //可选，默认false，设置是否收集CRASH信息，默认收集
    option.setEnableSimulateGps(false);
    //可选，默认false，设置是否需要过滤gps仿真结果，默认需要
    mLocationClient.setLocOption(option);
}

```

并调用

```

mLocationClient=new LocationClient(getApplicationContext());
//声明LocationClient类
mLocationClient.registerLocationListener(myListener);
//注册监听函数
initLocation();
}

```

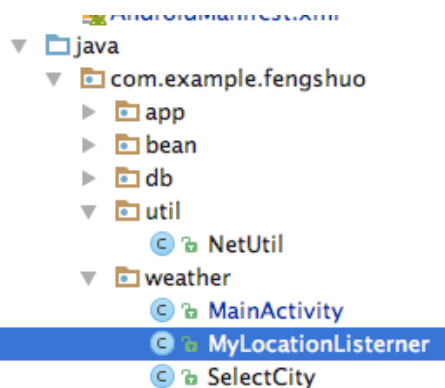
```

private void initLocation(){
    LocationClientOption option = new LocationClientOption();
    option.setLocationMode(LocationClientOption.LocationMode.High_Accuracy);
    //可选，默认高精度，设置定位模式，高精度，低功耗，仅设备
    option.setCoorType("bd09ll");
    //可选，默认gcj02，设置返回的定位结果坐标系
    int span = 1000;
    option.setScanSpan(0);
    //可选，默认0，即仅定位一次，设置发起定位请求的间隔需要大于等于1000ms才是有效的
    option.setIsNeedAddress(true);
    //可选，设置是否需要地址信息，默认不需要
    option.setOpenGps(true);
    //可选，默认false，设置是否使用gps
    option.setLocationNotify(true);
    //可选，默认false，设置是否当gps有效时按照1S1次频率输出GPS结果
    option.setIsNeedLocationDescribe(true);
    //可选，默认false，设置是否需要位置语义化结果，
    // 可以在BDLocation.getLocationDescribe里得到，结果类似于“在北京天安门附近”
    option.setIsNeedLocationPoiList(true);
    //可选，默认false，设置是否需要POI结果，可以在BDLocation.getPoiList里得到
    option.setIgnoreKillProcess(false);
    //可选，默认true，定位SDK内部是一个SERVICE，并放到了独立进程，
    // 设置是否在stop的时候杀死这个进程，默认不杀死
    option.SetIgnoreCacheException(false);
    //可选，默认false，设置是否收集CRASH信息，默认收集
    option.setEnableSimulateGps(false);
    //可选，默认false，设置是否需要过滤gps仿真结果，默认需要
    mLocationClient.setLocOption(option);
}

```

15. 实现 BDAbstractLocationListener 接口

新建 MyLocationListener



获取城市编码

```

public class MyLocationListener extends BDAbstractLocationListener {
    public String recity;
    public String cityCode;
    @Override
    public void onReceiveLocation(BDLocation location){
        //此处的BDLocation为定位结果信息类，通过它的各种get方法可获取定位相关的全部结果
        //以下只列举部分获取地址相关的结果信息
        //更多结果信息获取说明，请参照类参考中BDLocation类中的说明

        String addr = location.getAddrStr();    //获取详细地址信息
        String country = location.getCountry();    //获取国家
        String province = location.getProvince();    //获取省份
        String city = location.getCity();    //获取城市
        String district = location.getDistrict();    //获取区县
        String street = location.getStreet();    //获取街道信息
        Log.d("location_city",city);
        recity=city.replace("市","");

        List<City> mCityList;
        MyApplication myApplication;
        myApplication= MyApplication.getInstance();

        mCityList=myApplication.getCityList();

        for (City cityl:mCityList){
            if (cityl.getCity().equals(recity)){
                cityCode=cityl.getNumber();
                Log.d("location_code",cityCode);
            }
        }
    }
}

```

16. 监听定位按钮

```

private ImageView mTitleLocation;

```

```

if (view.getId()==R.id.title_location){
    setUpdateProgress();
    if (mLocationClient.isStarted()){
        mLocationClient.stop();
    }
    mLocationClient.start();

    final Handler BDHandler=new Handler(){
        public void handleMessage(Message msg){
            switch (msg.what){
                case DB:
                    if (msg.obj != null) {
                        if (NetUtil.getNetworkState(MainActivity.this) != NetUtil.NETWORK_NONE) {
                            Log.d("myWeather", "网络ok");
                            queryWeatherCode(myListener.cityCode);
                            // Toast.makeText(MainActivity.this,"网络ok! ",Toast.LENGTH_LONG).show();
                        } else {
                            Log.d("myWeather", "网络挂了");
                            Toast.makeText(MainActivity.this, "网络挂了! ", Toast.LENGTH_LONG).show();
                        }
                    }
                    myListener.cityCode=null;
                    break;
                default:
                    break;
            }
        }
    };
}
}
};

```

```

new Thread(new Runnable() {
    @Override
    public void run() {
        try {
            while (myListener.cityCode==null){
                Thread.sleep(2000);
            }
            Message msg=new Message();
            msg.what=DB;
            msg.obj=myListener.cityCode;
            BDHandler.sendMessage(msg);
        }catch (Exception e){
            e.printStackTrace();
        }
    }
}).start();

```

17. 运行一下

