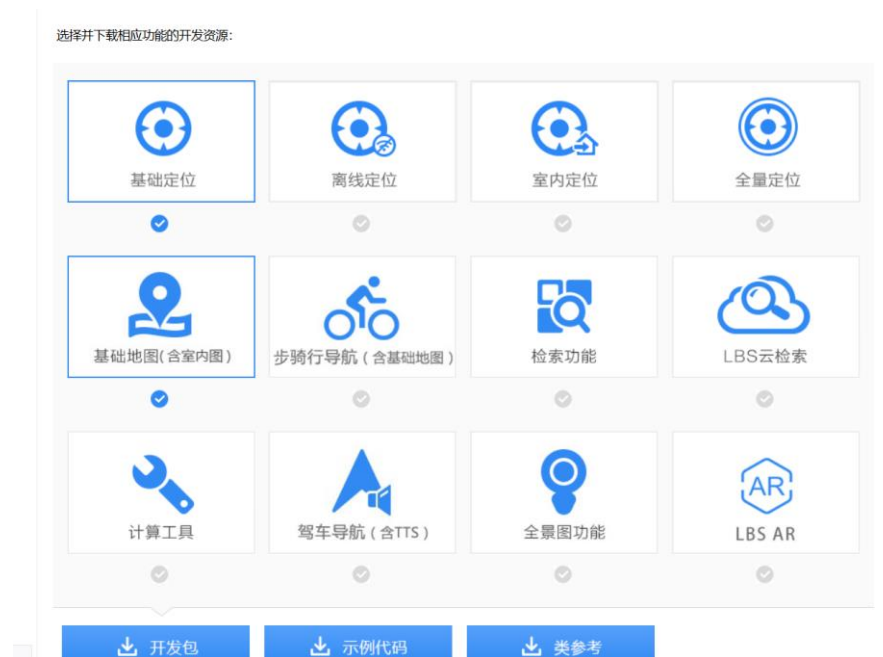
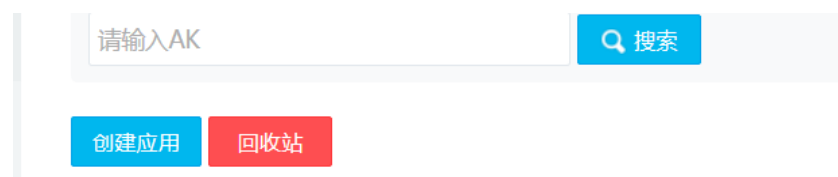


首先，登录百度地图开放平台，获取 Android 定位 SDK，在百度 LBS 平台中，选择基础定位和基础地图（地图可选择安装，方便扩展），然后选择开发包。

LSB 平台地址：http://lbsyun.baidu.com/index.php?title=sdk/download&action#selected=location_all。



然后在控制台界面中，选择创建应用，按照要求填写选项。



其中 Android Studio 中 SHA1 的获取方法请看这里：<https://blog.csdn.net/woaisunyizheng>

[/article/details/68942651](#)。

下载完后的安装包的内容需放入到指定文件夹中，同时在 AndroidManifest.xml 中配置权限、service 和 AK 信息，具体操作看这里：<http://lbsyun.baidu.com/index.php?title=android-loc-sdk/guide/create-project/android-studio>。

以上步骤配置好后，开始结合百度官网开发文档，实现定位查询天气功能。
首先在 MainActivity 中，增加以下变量

```
private List<City> mCityList;//用于存放获取的数据库中的城市信息
private String mLocCityCode;//定位获取的所在城市代码
private String cityName;//定位获取城市名称
private ImageView mtitleLocation;//监听点击定位使用
public LocationClient mLocationClient;
private MyLocationListener myLocationListener;
```

，在 onCreate 部分添加以下代码

```
mtitleLocation=(ImageView)findViewById(R.id.title_location);
mtitleLocation.setOnClickListener(this);
```

```
mLocationClient=new LocationClient( context: this);
myLocationListener=new MyLocationListener();
mLocationClient.registerLocationListener(myLocationListener);
checkPermission();
```

在 MainActivity 中，需要定义或重构以下方法，并创建 MyLocationListener 类。

```
private void checkPermission() {
    List<String> permissionList = new ArrayList<>();
    if (ContextCompat.checkSelfPermission( context: MainActivity.this, Manifest.permission.ACCESS_FINE_LOCATION) != PackageManager.PERMISSION_GRANTED) {
        permissionList.add(Manifest.permission.ACCESS_FINE_LOCATION);
    }
    if (ContextCompat.checkSelfPermission( context: MainActivity.this, Manifest.permission.READ_PHONE_STATE) != PackageManager.PERMISSION_GRANTED) {
        permissionList.add(Manifest.permission.READ_PHONE_STATE);
    }
    if (ContextCompat.checkSelfPermission( context: MainActivity.this, Manifest.permission.WRITE_EXTERNAL_STORAGE) != PackageManager.PERMISSION_GRANTED) {
        permissionList.add(Manifest.permission.WRITE_EXTERNAL_STORAGE);
    }
    if (!permissionList.isEmpty()) {
        String [] permissions = permissionList.toArray(new String[permissionList.size()]);
        ActivityCompat.requestPermissions( activity: MainActivity.this, permissions, requestCode: 1);
    } else {
        requestLocation();
    }
}
```

```

private void requestLocation() {
    initLocation();
    mLocationClient.start();
}

private void initLocation() {
    LocationClientOption option=new LocationClientOption();
    option.setIsNeedAddress(true);
    option.setOpenGps(true);
    option.setLocationMode(LocationClientOption.LocationMode.High_Accuracy);
    option.setCoorType("bd09ll");
    option.setScanSpan(1000);
    mLocationClient.setLocOption(option);
}

```

```

@Override
protected void onDestroy(){
    super.onDestroy();
    mLocationClient.stop();
}

public class MyLocationListener implements BDLocationListener {
    @Override
    public void onReceiveLocation(BDLocation location){
        cityName=location.getCity();
        Log.d( tag: "Locate",cityName);
    }
}

```

最后，在 onClick 部分的监听按钮判断部分，添加如下部分代码

```

if(view.getId()==R.id.title_location){
    if (NetUtil.getNetworkState( context: this) != NetUtil.NETWORK_NONE) {//监测网络状态
        MyApplication myapp=(MyApplication)getApplication();
        mCityList=myapp.getCityList();
        for(City city:mCityList){
            String locateCityName=cityName.toString();
            if(city.getCity().equals(locateCityName.substring(0,locateCityName.length()-1))){
                mLocCityCode=city.getNumber();
                break;
            }
        }
    }
    //Log.d("myWeather123",mLocCityCode);
    queryWeatherCode(mLocCityCode);
}

```

即可实现根据定位城市获取相应城市天气信息。