

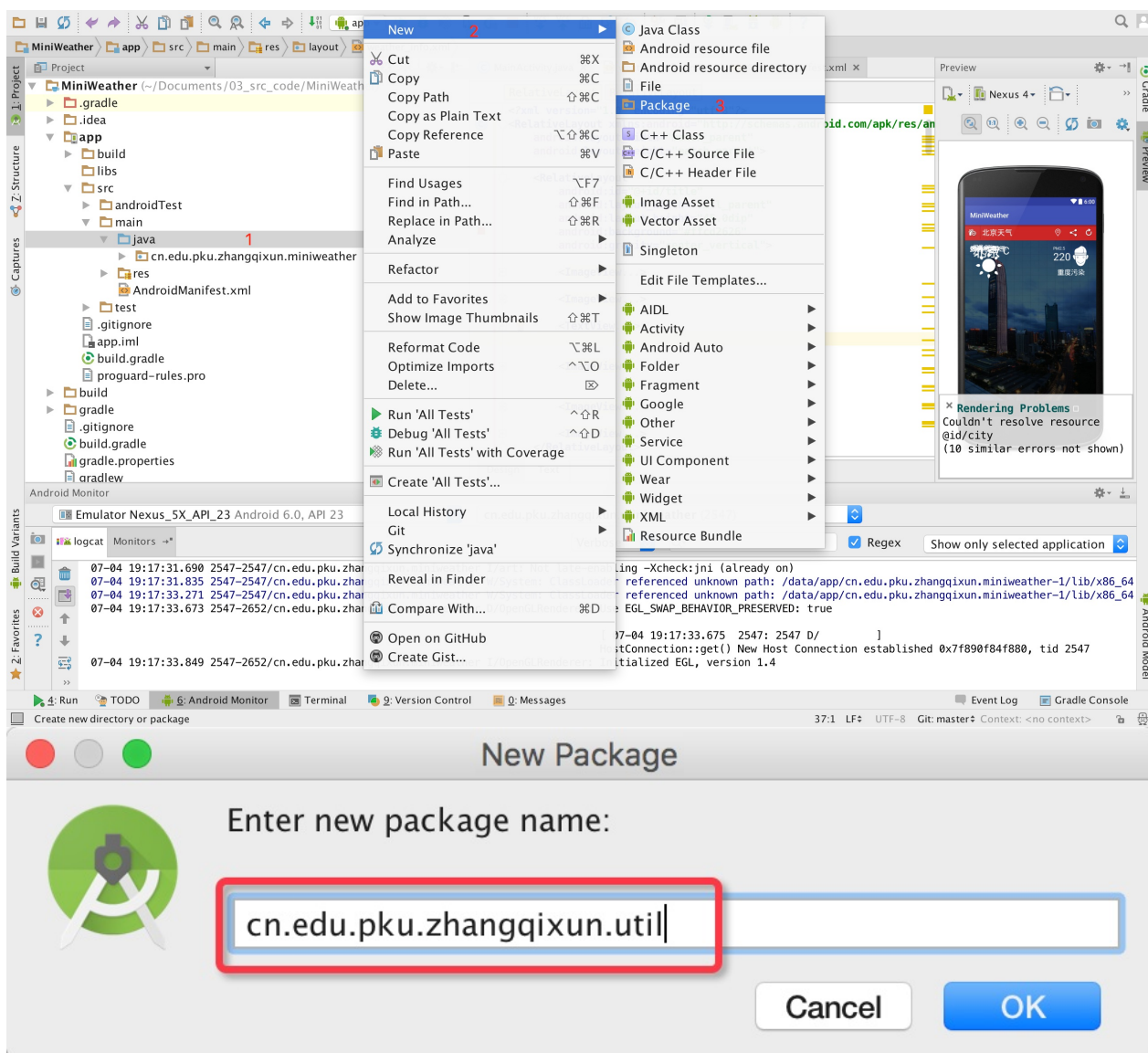
网络状态检查

应用程序在访问网络资源时，需要先检测网络状态，如果网络不可用，需要提示用户请打开网络，或检查网络连接状况。

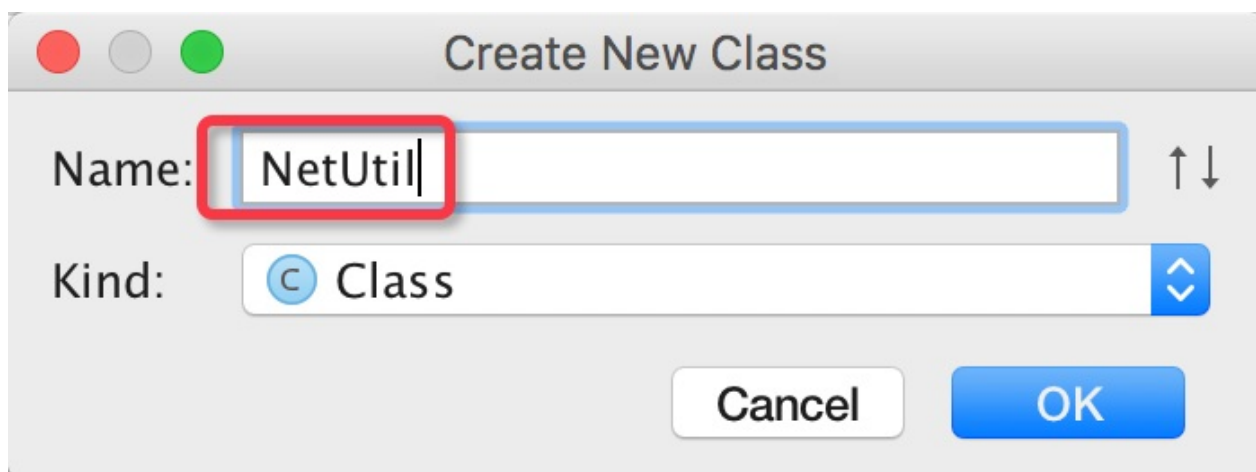
因此本次任务是制作一个工具类，用于检查网络状态。

主要步骤

1. 创建一个util包，存放公共的工具类



2. 在util文件夹中创建一个NetUtil类



3. 在NetUtil类中实现getNetworkState方法

```
package cn.edu.pku.zhangqixun.util;

import android.content.Context;
import android.net.ConnectivityManager;
import android.net.NetworkInfo;

/**
 * Created by zhangqixun on 16/7/4.
 */
public class NetUtil {
    public static final int NETWORK_NONE = 0;
    public static final int NETWORK_WIFI = 1;
    public static final int NETWORK_MOBILE = 2;

    public static int getNetworkState(Context context) {
        ConnectivityManager connManager = (ConnectivityManager) context
            .getSystemService(Context.CONNECTIVITY_SERVICE);

        NetworkInfo networkInfo = connManager.getActiveNetworkInfo();
        if (networkInfo == null) {
            return NETWORK_NONE;
        }

        int nType = networkInfo.getType();
        if (nType == ConnectivityManager.TYPE_MOBILE) {
            return NETWORK_MOBILE;
        } else if (nType == ConnectivityManager.TYPE_WIFI) {
            return NETWORK_WIFI;
        }
        return NETWORK_NONE;
    }
}
```

4.调用检测网络连接状态方法

在连接网络之前，可以直接通过以下语句检测网络是否连接。

```
NetUtil.getNetworkState(this) != NetUtil.NETWORK_NONE
```

例如，在Activity中增加以下代码，进行测试。

```
if (NetUtil.getNetworkState(this) != NetUtil.NETWORK_NONE) {
    Log.d("myWeather", "网络OK");
    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();
}
```

```
package cn.edu.pku.zhangqixun.miniweather;

import android.app.Activity;
import android.os.Bundle;
import android.util.Log;
import android.widget.Toast;

import cn.edu.pku.zhangqixun.util.NetUtil;

/**
 * Created by zhangqixun on 16/7/4.
 */
public class MainActivity extends Activity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.weather_info);

        if (NetUtil.getNetworkState(this) != NetUtil.NETWORK_NONE) {
            Log.d("myWeather", "网络OK");
            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();
        }
    }
}
```

5. 需要增加访问网络状态的权限

```
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="cn.edu.pku.zhanggixun.miniweather">
    <uses-permission android:name="android.permission.ACCESS_NETWORK_STATE"/>
    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="MiniWeather"
        android:supportsRtl="true"
        android:theme="@style/AppTheme">

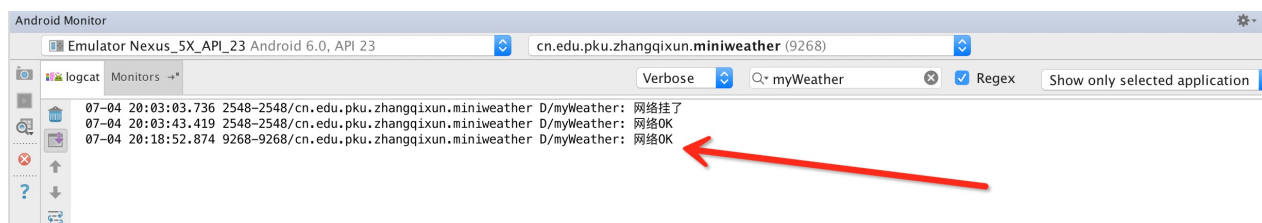
        <activity android:name=".MainActivity">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>

    </application>
</manifest>
```

6. 运行结果



4. 网络状态检查



附录：NetUtil.java源码

```
package cn.edu.pku.zhangqixun.util;

import android.content.Context;
import android.net.ConnectivityManager;
import android.net.NetworkInfo;

/**
 * Created by zhangqixun on 16/7/4.
 */
public class NetUtil {
    public static final int NETWORK_NONE = 0;
    public static final int NETWORK_WIFI = 1;
    public static final int NETWORK_MOBILE = 2;

    public static int getNetworkState(Context context) {
        ConnectivityManager connManager = (ConnectivityManager)
context
                .getSystemService(Context.CONNECTIVITY_SERVICE);

        NetworkInfo networkInfo = connManager.getActiveNetworkIn
fo();
        if (networkInfo == null) {
            return NETWORK_NONE;
        }

        int nType = networkInfo.getType();
        if (nType == ConnectivityManager.TYPE_MOBILE) {
            return NETWORK_MOBILE;
        } else if (nType == ConnectivityManager.TYPE_WIFI) {
            return NETWORK_WIFI;
        }
        return NETWORK_NONE;
    }
}
```

