**7.1.3 HTTP请求方式HttpURLConnection**

本节学习Android提供的请求方式之一HttpURLConnection，还有HttpClient。前者一旦请求复杂起来，非常麻烦；后者Java抓包时经常会用到。HttpURLConnection在4.4版本被替换为OkHttp。

实际开发中使用别人封装好的第三方网络请求框架，比如：Volley，android-async-http，loopj。因为网络操作涉及到异步以及多线程，自己写的话很麻烦。

1.HttpURLConnection的介绍

多用途，轻量级的HTTP客户端，使用它来进行Http操作可以适用于大多数的应用程序。虽然HttpURLConnection的API简单，但是我们也更容易去扩展和使用。继承自URLConnection，抽象类。通过调用openCollection()方法获取对象实例，默认是gzip压缩的。

2.HttpURLConnection的使用步骤

1）创建一个URL对象：

URL url = new URL(https://www.baidu.com);

2）调用URL对象的openConnection()来获取HttpURLConnecton对象实例：

HttpURLConnection conn = (HttpURLConnection)url.openConnection();

3）设置HTTP请求使用的方法：GET或者POST，或者其它：

conn.setRequestMethod(“GET”);

4）设置连接超时，读取超时的毫秒数，以及服务器希望得到的消息头

conn.setConnectionTimeout(6\*1000);

conn.setReadTimeout(6\*1000);

5）调用getInputStream()方法获得服务器返回的输入流，然后进行读取。

InputStream in = conn.getInputStream();

6）最后调用disconnect()方法关掉HTTP

conn.disconnect();

除了上面这些，还有对响应码进行的判断，比如if(conn.getResponseCode() != 200)，还有一些处理。有时候并不需要要传递参数，而是直接访问一个页面，可以直接用：final InputStream in = new URL(“url”).openStream();，然后直接读流，不过这个方法适合直接访问页面，底层实现也是return openConnection().getInputStream()，而且不能设置请求头等。

3.HttpURLConnection使用示例

对GET和POST请求写两个不同的示例，用conn.getInputStream()获取到的是一个流，所以需要写一个类将流转化为二进制数组。

工具类：

1）HttpURLConnection发送GET请求代码示例

public class StreamTool{

public static byte[] read(InputStream inStream) throws Exception{

ByteArrayOutputStream out Stream = new ByteArrayOutStream();

byte[] buffer = new byte[1024];

int len = 0;

while((len = inStream.read(buffer)) != -1)

{

outStream.write(buffer, 0, len);

}

inStream.close();

return outStream.toByteArray();

}

}

布局：activity\_main.xml

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:orientation="vertical">

<TextView

android:id = “@+id/txtMenu”

android:layout\_width = “match\_parent”

android:orientation = “vertcal”/>

<ImageView

android:id = “@+id/imgPic”

android:layout\_width = “match\_parent”

android:layout\_height = “match\_parent”

android:visibility = “gone”/>

<ScrollView

android:id = “@+id/scroll”

android:layout\_width = “match\_parent”

android:layout\_height = “match\_parent”

android:visibility = “gone”>

<TextView

android:id = “@+id/txtshow”

android:layout\_width = “wrap\_content”

android:layout\_height = “wrap\_content”/>

</ScrollView>

<WebView

android:id = “@+id/webView”

android:layout\_width = “match\_parent”

android:layout\_height = “match\_parent”/>

</LinearLayout>

获取数据类：