学习网址: <http://www.cnblogs.com/QQParadise/articles/5020215.html>

学习网址http://mvnrepository.com/artifact/org.apache.httpcomponents/httpclient/4.5.2

http://www.yeetrack.com/?p=760

最近在使用Apache的httpclient的时候，maven引用了最新版本4.3,发现Idea提示DefaultHttpClient等常用的类已经不推荐使用了，之前在使用4.2.3版本的时候，还没有被deprecated。去看了下官方文档，确实不推荐使用了，[点击此处详情](http://hc.apache.org/httpcomponents-client-ga/httpclient/apidocs/org/apache/http/impl/client/DefaultHttpClient.html)。

* DefaultHttpClient —> CloseableHttpClient
* HttpResponse —> CloseableHttpResponse

官方给出了新api的样例，如下。

Get方法：

CloseableHttpClient httpclient = HttpClients.createDefault();

HttpGet httpGet = new HttpGet("http://targethost/homepage");

CloseableHttpResponse response1 = httpclient.execute(httpGet);

// The underlying HTTP connection is still held by the response object

// to allow the response content to be streamed directly from the network socket.

// In order to ensure correct deallocation of system resources

// the user MUST either fully consume the response content or abort request

// execution by calling CloseableHttpResponse#close().

//建立的http连接，仍旧被response1保持着，允许我们从网络socket中获取返回的数据

//为了释放资源，我们必须手动消耗掉response1或者取消连接（使用CloseableHttpResponse类的close方法）

try {

System.out.println(response1.getStatusLine());

HttpEntity entity1 = response1.getEntity();

// do something useful with the response body

// and ensure it is fully consumed

EntityUtils.consume(entity1);

} finally {

response1.close();

}

Post方法：

HttpPost httpPost = new HttpPost("http://targethost/login");

//拼接参数

List <NameValuePair> nvps = new ArrayList <NameValuePair>();

nvps.add(new BasicNameValuePair("username", "vip"));

nvps.add(new BasicNameValuePair("password", "secret"));

httpPost.setEntity(new UrlEncodedFormEntity(nvps));

CloseableHttpResponse response2 = httpclient.execute(httpPost);

try {

System.out.println(response2.getStatusLine());

HttpEntity entity2 = response2.getEntity();

// do something useful with the response body

// and ensure it is fully consumed

//消耗掉response

EntityUtils.consume(entity2);

} finally {

response2.close();

}

再往下看HttpClients的源码，具体的实现都在HttpClientBuilder的build方法中，有兴趣的可以去apache看源码。

/\*\*

\* Creates {@link CloseableHttpClient} instance with default

\* configuration.

\*/

public static CloseableHttpClient createDefault() {

return HttpClientBuilder.create().build();

}