# GET

## 普通的GET请求

@Test

**public** **void** testBaidu() **throws** Exception, IOException {

String url="http://study-perf.qa.netease.com/fgadmin/address/list";

HttpEntity entity=**null**;

//1、创建一个httpclient对象

CloseableHttpClient httpClient=HttpClients.*createDefault*();

//2、创建get请求对象

HttpGet get=**new** HttpGet(url);

//3、执行请求，得到响应

CloseableHttpResponse response=httpClient.execute(get);

StatusLine status=response.getStatusLine();

System.***out***.println(status);

**if**(status!=**null**) {

//4、得到响应的实体

entity=response.getEntity();

System.***out***.println(EntityUtils.*toString*(entity,"UTF-8"));

}

// 5、释放资源

EntityUtils.*consume*(entity);

// 6、关闭连接

response.close();

httpClient.close();

## 如果GET参数带有特殊字符，需要进行转换

http://127.0.0.1:8080/supermarket/analysis/lookupprice?goodsCode={"pId":"123456"}

String para=URLEncoder.*encode*("{\"pid\",\"123456\"}","UTF-8");

String url="http://127.0.0.1:8080/supermarket/analysis/lookupprice?goodsCode="+para;

# POST

POST与GET区别，在于有请求体

比GET请求多2步，设置请求体，设置Content-Type

## 请求体类型为x-www-form-urlencoded

@Test

**public** **void** testLoginMiniBlog() **throws** Exception {

String loginurl="http://study-miniblog-new.qa.netease.com/ajax/user/login";

CloseableHttpClient httpClient=HttpClients.*createDefault*();

HttpPost post =**new** HttpPost(loginurl);

//设置请求体

HttpEntity user=**new** StringEntity("username=lihuanzhen&password=123456");

// StringEntity user1=new StringEntity("");

post.setHeader("Content-Type","application/x-www-form-urlencoded;");

post.setEntity(user);

CloseableHttpResponse response=httpClient.execute(post);

HttpEntity responseEntity=response.getEntity();

String result=EntityUtils.*toString*(responseEntity,"UTF-8");

System.***out***.println(result);

EntityUtils.*consume*(responseEntity);

response.close();

httpClient.close();

}

## 请求体类型为x-www-form-urlencoded

@Test

**public** **void** testMiniLogin() **throws** Exception {

String loginurl="http://study-miniblog-new.qa.netease.com/ajax/user/login";

CloseableHttpClient httpClient=HttpClients.*createDefault*();

HttpPost post =**new** HttpPost(loginurl);

//设置请求体

List<NameValuePair> pairs=**new** ArrayList<NameValuePair>();

NameValuePair u\_name=**new** BasicNameValuePair("username", "lihuanzhen");

NameValuePair u\_pwd=**new** BasicNameValuePair("password", "123456");

pairs.add(u\_name);

pairs.add(u\_pwd);

HttpEntity user=**new** UrlEncodedFormEntity(pairs);

// post.setHeader("Content-Type","application/x-www-form-urlencoded;");

post.setEntity(user);

CloseableHttpResponse response=httpClient.execute(post);

HttpEntity responseEntity=response.getEntity();

String result=EntityUtils.toString(responseEntity,"UTF-8");

System.out.println(result);

EntityUtils.consume(responseEntity);

response.close();

httpClient.close();

}

## 请求体类型为application/json

@Test

**public** **void** testLoginQG() **throws** Exception {

String loginurl="http://study-perf.qa.netease.com/common/fgadmin/login";

CloseableHttpClient httpClient=HttpClients.*createDefault*();

HttpPost post =**new** HttpPost(loginurl);

//设置请求体

HttpEntity user=**new** StringEntity("{\"phoneArea\":\"86\",\"phoneNumber\":\"20000000000\",\"password\":\"netease123\"}");

// StringEntity user1=new StringEntity("");

post.setHeader("Content-Type","application/json");

post.setEntity(user);

CloseableHttpResponse response=httpClient.execute(post);

HttpEntity responseEntity=response.getEntity();

String result=EntityUtils.*toString*(responseEntity,"UTF-8");

System.***out***.println(result);

EntityUtils.*consume*(responseEntity);

response.close();

httpClient.close();

}

# 掌握GET与POST简单的封装

**public** **static** String doGet(String url) **throws** Exception {

CloseableHttpClient client=HttpClients.*createDefault*();

//get请求参数在url中

HttpGet get=**new** HttpGet(url);

CloseableHttpResponse response =client.execute(get);

HttpEntity result\_entity=response.getEntity();

String result=EntityUtils.*toString*(result\_entity,"utf-8");

EntityUtils.*consume*(result\_entity);

response.close();

client.close();

**return** result;

}

**public** **static** String doPost(String url, String body) **throws** Exception {

CloseableHttpClient client=HttpClients.*createDefault*();

HttpPost post=**new** HttpPost(url);

post.setHeader("content-type","application/json");

HttpEntity request\_entity=**new** StringEntity(body);

post.setEntity(request\_entity);

CloseableHttpResponse response =client.execute(post);

HttpEntity result\_entity=response.getEntity();

String result=EntityUtils.*toString*(result\_entity,"utf-8");

EntityUtils.*consume*(result\_entity);

response.close();

client.close();

**return** result;

}

# 实现Cookie的传递

RequestConfig gConfig =

RequestConfig.*custom*().setCookieSpec(CookieSpecs.***STANDARD***).build();

CookieStore cookie=**new** BasicCookieStore();

CloseableHttpClient client=HttpClients.*custom*().

setDefaultRequestConfig(gConfig).setDefaultCookieStore(cookie).build();

# JSON使用

1、结果解析

JSONObject jsonResult=JSONObject.*fromObject*(result);

// 验证请求的message

*assertEquals*(jsonResult.getString("message"), "success");

// 验证请求的code

*assertEquals*(jsonResult.getInt("code"), 200);

2、构建请求体类型为JSON文件

JSONObject user=**new** JSONObject();

user.element("phoneArea", "86");

user.element("phoneNumber", "20000000000");

user.element("password", "netease123");