



链滴

爬虫 --httpClient 学习

作者: [XiaoCoder](#)

原文链接: <https://ld246.com/article/1610375287069>

来源网站: 链滴

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)



起步例子

```
package HttpClient;

import org.apache.http.HttpEntity;
import org.apache.http.client.methods.CloseableHttpResponse;
import org.apache.http.client.methods.HttpGet;
import org.apache.http.impl.client.CloseableHttpClient;
import org.apache.http.impl.client.HttpClients;
import org.apache.http.util.EntityUtils;

import java.io.IOException;

/**
 * @_PackageName:PACKAGE_NAME
 * @_ClassName:XiaoMagicFirst
 * @_Description:
 * @_Author: 笑老二
 * @_data 2021/01/11 17:39
 */
public class XiaoMagicFirst {
    public static void main(String[] args) throws IOException {
        //1、打开浏览器
        CloseableHttpClient httpClient = HttpClients.createDefault();

        //2、输入网站
        HttpGet httpGet = new HttpGet("http://www.itcast.cn");

        //3、发起请求
        CloseableHttpResponse response = httpClient.execute(httpGet);
```

```

//4、解析获取数据
//判断状态码是否为200
if (response.getStatusLine().getStatusCode()==200){
    HttpEntity httpEntity = response.getEntity();
    String content = EntityUtils.toString(httpEntity, "utf8");

    System.out.println(content);
}
}
}
}

```

1、HttpGet

```

package HttpClient;

import org.apache.http.client.methods.CloseableHttpResponse;
import org.apache.http.client.methods.HttpGet;
import org.apache.http.impl.client.CloseableHttpClient;
import org.apache.http.impl.client.HttpClients;
import org.apache.http.util.EntityUtils;

import java.io.IOException;

/**
 * @_PackageName:PACKAGE_NAME
 * @_ClassName:HttpGetTest
 * @_Description:
 * @_Author: 笑老二
 * @_data 2021/01/11 18:02
 */
public class HttpGetTest {
    public static void main(String[] args) {
        //创建HttpClient对象
        CloseableHttpClient httpClient = HttpClients.createDefault();
        //创建HttpGet对象, 设置Url地址
        HttpGet httpGet = new HttpGet("http://www.itcast.cn");
        CloseableHttpResponse response = null;
        //使用HttpClient发起请求们获取Response
        try {
            response = httpClient.execute(httpGet);
            //解析
            if (response.getStatusLine().getStatusCode()==200){
                String content = EntityUtils.toString(response.getEntity(), "utf8");
                System.out.println(content.length());
            }

        } catch (IOException e) {
            e.printStackTrace();
        } finally {
            //关闭response
            try {
                response.close();
            } catch (IOException e) {

```



```

//使用HttpClient发起请求们获取Response
try {
    response = httpClient.execute(httpGet);
    //解析
    if (response.getStatusLine().getStatusCode()==200){
        String content = EntityUtils.toString(response.getEntity(), "utf8");
        System.out.println(content.length());
    }

} catch (IOException e) {
    e.printStackTrace();
}finally {
    //关闭response
    try {
        response.close();
    } catch (IOException e) {
        e.printStackTrace();
    }
    //关闭浏览器
    try {
        httpClient.close();
    } catch (IOException e) {
        e.printStackTrace();
    }
}
}
}
}

```

3、HttpPost

```

package HttpClient;

import org.apache.http.client.methods.CloseableHttpResponse;
import org.apache.http.client.methods.HttpGet;
import org.apache.http.client.methods.HttpPost;
import org.apache.http.impl.client.CloseableHttpClient;
import org.apache.http.impl.client.HttpClients;
import org.apache.http.util.EntityUtils;

import java.io.IOException;

/**
 * @_PackageName:PACKAGE_NAME
 * @_ClassName:HttpGetTest
 * @_Description:
 * @_Author: 笑老二
 * @_data 2021/01/11 18:02
 */
public class HttpPostTest {
    public static void main(String[] args) {
        //创建HttpClient对象
        CloseableHttpClient httpClient = HttpClients.createDefault();
    }
}

```

```

//创建HttpPost对象, 设置Url地址
HttpPost httpPost = new HttpPost("http://www.itcast.cn");

CloseableHttpResponse response = null;
//使用HttpClient发起请求们获取Response
try {
    response = httpClient.execute(httpPost);
    //解析
    if (response.getStatusLine().getStatusCode() == 200){
        String content = EntityUtils.toString(response.getEntity(), "utf8");
        System.out.println(content.length());
    }

} catch (IOException e) {
    e.printStackTrace();
}finally {
    //关闭response
    try {
        response.close();
    } catch (IOException e) {
        e.printStackTrace();
    }
    //关闭浏览器
    try {
        httpClient.close();
    } catch (IOException e) {
        e.printStackTrace();
    }
}
}
}
}

```

4、HttpPostParamTest

```

package HttpClient;

import org.apache.http.NameValuePair;
import org.apache.http.client.entity.UrlEncodedFormEntity;
import org.apache.http.client.methods.CloseableHttpResponse;
import org.apache.http.client.methods.HttpPost;
import org.apache.http.impl.client.CloseableHttpClient;
import org.apache.http.impl.client.HttpClients;
import org.apache.http.message.BasicNameValuePair;
import org.apache.http.util.EntityUtils;

import java.io.IOException;
import java.util.ArrayList;
import java.util.List;

/**
 * @_PackageName:PACKAGE_NAME
 * @_ClassName:HttpGetTest

```

```

* @_Description:
* @_Author: 笑老二
* @_data 2021/01/11 18:02
*/
public class HttpPostParamTest {
    public static void main(String[] args) throws Exception {
        //创建HttpClient对象
        CloseableHttpClient httpClient = HttpClients.createDefault();
        //创建HttpPost对象, 设置Url地址
        HttpPost httpPost = new HttpPost("http://yun.itheima.com/search");

        //声明List集合, 封装表单中的参数
        List<NameValuePair> params = new ArrayList<NameValuePair>();
        //设置请求地址: http://yun.itheima.com/search?keys=Java
        params.add(new BasicNameValuePair("keys", "Java"));

        //创建表单的Entity对象 params:是表单数据, 第二个是: 编码格式
        UrlEncodedFormEntity formEntity = new UrlEncodedFormEntity(params, "utf8");

        //设置表单的Entity对象到Post请求中
        httpPost.setEntity(formEntity);

        CloseableHttpResponse response = null;
        //使用HttpClient发起请求们获取Response
        try {
            response = httpClient.execute(httpPost);
            //解析
            if (response.getStatusLine().getStatusCode() == 200){
                String content = EntityUtils.toString(response.getEntity(), "utf8");
                System.out.println(content.length());
            }
        } catch (IOException e) {
            e.printStackTrace();
        } finally {
            //关闭response
            try {
                response.close();
            } catch (IOException e) {
                e.printStackTrace();
            }
            //关闭浏览器
            try {
                httpClient.close();
            } catch (IOException e) {
                e.printStackTrace();
            }
        }
    }
}

```

5、HttpConfig

```
package HttpClient;

import org.apache.http.client.config.RequestConfig;
import org.apache.http.client.methods.CloseableHttpResponse;
import org.apache.http.client.methods.HttpGet;
import org.apache.http.impl.client.CloseableHttpClient;
import org.apache.http.impl.client.HttpClients;
import org.apache.http.util.EntityUtils;

import java.io.IOException;

/**
 * @_PackageName:PACKAGE_NAME
 * @_ClassName:HttpGetTest
 * @_Description:
 * @_Author: 笑老二
 * @_data 2021/01/11 18:02
 */
public class HttpConfigTest {
    public static void main(String[] args) {
        //创建HttpClient对象
        CloseableHttpClient httpClient = HttpClients.createDefault();
        //创建HttpGet对象, 设置Url地址
        HttpGet httpGet = new HttpGet("http://www.itcast.cn");

        //配置请求信息
        RequestConfig config = RequestConfig.custom().setConnectTimeout(500)//设置创建的
        接最长时间, 毫秒
            .setConnectionRequestTimeout(1000)//设置获取连接最长时间, 单位毫秒
            .setSocketTimeout(10 * 1000) //设置数据传输的最长时间
            .build();

        //给请求设置请求信息
        httpGet.setConfig(config);

        CloseableHttpResponse response = null;
        //使用HttpClient发起请求们获取Response
        try {
            response = httpClient.execute(httpGet);
            //解析
            if (response.getStatusLine().getStatusCode() == 200) {
                String content = EntityUtils.toString(response.getEntity(), "utf8");
                System.out.println(content.length());
            }
        } catch (IOException e) {
            e.printStackTrace();
        } finally {
            //关闭response
            try {
                response.close();
            } catch (IOException e) {
```



```

//不是每次创建新的HttpClient, 而是从连接池中进行获取
CloseableHttpClient httpClient = HttpClients.custom().setConnectionManager(cm).build()

HttpGet httpGet = new HttpGet("http://www.itcast.cn");

CloseableHttpResponse response = null;

try {
    response = httpClient.execute(httpGet);
    if (response.getStatusLine().getStatusCode()==200){
        String content = EntityUtils.toString(response.getEntity(), "utf8");
        System.out.println(content.length());
    }
} catch (IOException e) {
    e.printStackTrace();
}finally {
    if (response!=null){
        try {
            response.close();
        } catch (IOException e) {
            e.printStackTrace();
        }
    }
}
//这里不能关闭httpClient, 连接池进行管理
//httpClient.close();
}
}
}
}
}

```