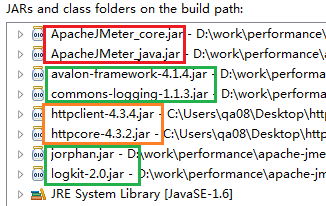
一、创建工程、引包

1、创建JAVA工程

2、引入Jmeter中lib\ext基础包：ApacheJMeter\_java.jar、ApacheJMeter\_core.jar

3、引入Jmeter日志包：jorphan.jar，logkit-2.0.jar，commons-logging-1.1.1.jar，avalon-framework-4.1.4.jar

4、引入httpclient-4.3.4.jar（是基于httpcore的一套客户端）、httpcore-4.3.2.jar（是一套http协议实现包）



Jmeter用到的4个类：

|  |  |
| --- | --- |
| [Arguments](http://jmeter.apache.org/api/org/apache/jmeter/config/Arguments.html) | [**getDefaultParameters**](http://jmeter.apache.org/api/org/apache/jmeter/protocol/java/sampler/JavaSamplerClient.html#getDefaultParameters())() 用于获取界面的参数 |
| [SampleResult](http://jmeter.apache.org/api/org/apache/jmeter/samplers/SampleResult.html) | [**runTest**](http://jmeter.apache.org/api/org/apache/jmeter/protocol/java/sampler/JavaSamplerClient.html#runTest(org.apache.jmeter.protocol.java.sampler.JavaSamplerContext))([JavaSamplerContext](http://jmeter.apache.org/api/org/apache/jmeter/protocol/java/sampler/JavaSamplerContext.html) context) 类似于LR的Action |
| void | [**setupTest**](http://jmeter.apache.org/api/org/apache/jmeter/protocol/java/sampler/JavaSamplerClient.html#setupTest(org.apache.jmeter.protocol.java.sampler.JavaSamplerContext))([JavaSamplerContext](http://jmeter.apache.org/api/org/apache/jmeter/protocol/java/sampler/JavaSamplerContext.html) context) 初始化方法，类似于LR的init和Junit中的setUp() |
| void | [**teardownTest**](http://jmeter.apache.org/api/org/apache/jmeter/protocol/java/sampler/JavaSamplerClient.html#teardownTest(org.apache.jmeter.protocol.java.sampler.JavaSamplerContext))([JavaSamplerContext](http://jmeter.apache.org/api/org/apache/jmeter/protocol/java/sampler/JavaSamplerContext.html) context) 类似于LR的end和Junit中的tearDown() |

执行的先后顺序为：**[getDefaultParameters](http://jmeter.apache.org/api/org/apache/jmeter/protocol/java/sampler/JavaSamplerClient.html" \l "getDefaultParameters()" \t "_blank)**() --> **[setupTest](http://jmeter.apache.org/api/org/apache/jmeter/protocol/java/sampler/JavaSamplerClient.html" \l "setupTest(org.apache.jmeter.protocol.java.sampler.JavaSamplerContext)" \t "_blank)**([JavaSamplerContext](http://jmeter.apache.org/api/org/apache/jmeter/protocol/java/sampler/JavaSamplerContext.html" \o "class in org.apache.jmeter.protocol.java.sampler" \t "_blank) context) --> **[runTest](http://jmeter.apache.org/api/org/apache/jmeter/protocol/java/sampler/JavaSamplerClient.html" \l "runTest(org.apache.jmeter.protocol.java.sampler.JavaSamplerContext)" \t "_blank)**([JavaSamplerContext](http://jmeter.apache.org/api/org/apache/jmeter/protocol/java/sampler/JavaSamplerContext.html" \o "class in org.apache.jmeter.protocol.java.sampler" \t "_blank) context) --> **[teardownTest](http://jmeter.apache.org/api/org/apache/jmeter/protocol/java/sampler/JavaSamplerClient.html" \l "teardownTest(org.apache.jmeter.protocol.java.sampler.JavaSamplerContext)" \t "_blank)**([JavaSamplerContext](http://jmeter.apache.org/api/org/apache/jmeter/protocol/java/sampler/JavaSamplerContext.html" \o "class in org.apache.jmeter.protocol.java.sampler" \t "_blank) context)

常用的方法：

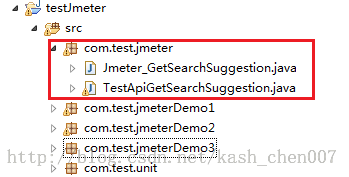
①、addArgument("name", "value")    定义参数

②、sampleStart()    定义事务的开始，类似于LR的lr\_start\_transaction，和LR一样事务间不要放无关代码

③、sampleEnd()    定义事务的结束，类似于LR的lr\_end\_transaction

④、setSuccessful(true、false)    设置运行结果的成功或失败，Jmeter统计成功失败的次数，在聚合报告中能够体现。

二、用例



1、TestApiGetSearchSuggestion.java，该代码实现对Api接口的请求，对返回的数据进行判断，并写文件。

2、Jmeter\_GetSearchSuggestion.java，为本节介绍的重点

**[java]** [view plaincopyprint?](http://blog.csdn.net/kash_chen007/article/details/37690411)

1. package com.test.jmeter;
3. import java.io.IOException;
5. import org.apache.http.client.ClientProtocolException;
6. import org.apache.jmeter.config.Arguments;
7. import org.apache.jmeter.protocol.java.sampler.AbstractJavaSamplerClient;
8. import org.apache.jmeter.protocol.java.sampler.JavaSamplerContext;
9. import org.apache.jmeter.samplers.SampleResult;
11. public class Jmeter\_GetSearchSuggestion extends AbstractJavaSamplerClient {
13. private static String label = "Jmeter\_GetSearchSuggestion ";    //定义label名称，显示在jmeter的结果窗口
14. private String url;
15. private String data;
17. public void setupTest(){
18. //定义测试初始值，setupTest只在测试开始前使用
19. System.out.println("setupTest");
20. }
22. @Override
23. public SampleResult runTest(JavaSamplerContext arg0) {
25. url = arg0.getParameter("url");
26. data = arg0.getParameter("data");
27. SampleResult sr;
28. sr = new SampleResult();
29. sr.setSampleLabel(label);
31. TestApiGetSearchSuggestion t = new TestApiGetSearchSuggestion();
32. sr.sampleStart();
33. try {
34. //调用被压测接口的方法
35. t.PostJson(url, data);
36. sr.setSuccessful(true);
37. } catch (ClientProtocolException e) {
38. sr.setSuccessful(false);
39. e.printStackTrace();
40. } catch (IOException e) {
41. sr.setSuccessful(false);
42. e.printStackTrace();
43. }
45. sr.sampleEnd(); // jmeter 结束统计响应时间标记
46. return sr;
47. }
49. public void teardownTest(JavaSamplerContext arg0){
50. super.teardownTest(arg0);
51. }
53. public Arguments getDefaultParameters(){
54. //参数定义，显示在前台，也可以不定义
55. Arguments params = new Arguments();
56. params.addArgument("url", "http://gapp.test.com/merchandise/GetSearchSuggestion");
57. params.addArgument("data", "data={\"token\": \"aaaaaaaaaa\",\"body\": {\"keywords\": \"蓝月亮\"}}");
58. return params;
59. }
61. }

package com.test.jmeter;

import java.io.IOException;

import org.apache.http.client.ClientProtocolException;

import org.apache.jmeter.config.Arguments;

import org.apache.jmeter.protocol.java.sampler.AbstractJavaSamplerClient;

import org.apache.jmeter.protocol.java.sampler.JavaSamplerContext;

import org.apache.jmeter.samplers.SampleResult;

public class Jmeter\_GetSearchSuggestion extends AbstractJavaSamplerClient {

private static String label = "Jmeter\_GetSearchSuggestion "; //定义label名称，显示在jmeter的结果窗口

private String url;

private String data;

public void setupTest(){

//定义测试初始值，setupTest只在测试开始前使用

System.out.println("setupTest");

}

@Override

public SampleResult runTest(JavaSamplerContext arg0) {

url = arg0.getParameter("url");

data = arg0.getParameter("data");

SampleResult sr;

sr = new SampleResult();

sr.setSampleLabel(label);

TestApiGetSearchSuggestion t = new TestApiGetSearchSuggestion();

sr.sampleStart();

try {

//调用被压测接口的方法

t.PostJson(url, data);

sr.setSuccessful(true);

} catch (ClientProtocolException e) {

sr.setSuccessful(false);

e.printStackTrace();

} catch (IOException e) {

sr.setSuccessful(false);

e.printStackTrace();

}

sr.sampleEnd(); // jmeter 结束统计响应时间标记

return sr;

}

public void teardownTest(JavaSamplerContext arg0){

super.teardownTest(arg0);

}

public Arguments getDefaultParameters(){

//参数定义，显示在前台，也可以不定义

Arguments params = new Arguments();

params.addArgument("url", "http://gapp.test.com/merchandise/GetSearchSuggestion");

params.addArgument("data", "data={\"token\": \"aaaaaaaaaa\",\"body\": {\"keywords\": \"蓝月亮\"}}");

return params;

}

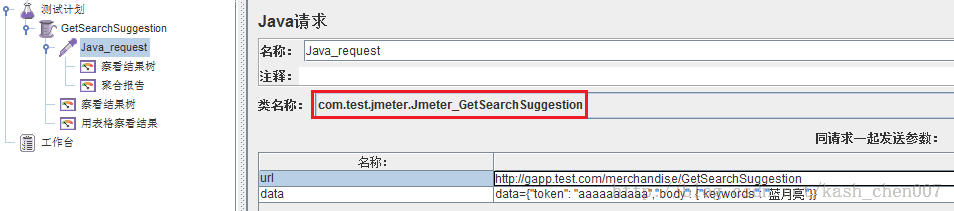
}

3、main函数测试成功后，打成jar包，放到%Jmeter\_Home%\lib\ext目录中即可

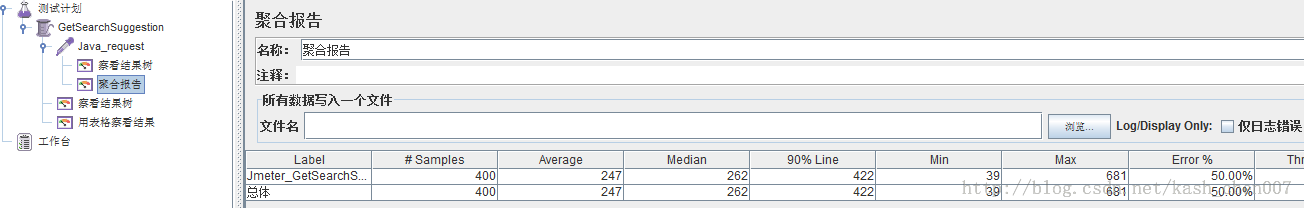
三、运行用例

1、运行%Jmeter\_Home%\bin\jmeter.bat

2、添加线程组，Java请求、查看结果树、聚合报告



3、执行后的结果为



Tips:

快捷键

Ctrl+R运行用例、Ctrl+E清除运行结果