

1.接口测试

接口测试属于集成测试

接口测试是测试系统组件间接口的一种测试，主要用于测试系统与外部其他系统之间的接口，以及系统内部各个子模块之间的接口。（即接口测试分为两种）

测试的重点：

1. 检查接口参数传递的正确性
2. 接口功能实现的正确性
3. 输出结果的正确性
4. 对各种异常情况的容错处理的完整性和合理性。

针对软件接口的分类一般有如下几种情况：

1. 系统与系统之间的调用，如微信向用户提供统一的对外接口，程序员调用接口完成基于微信的小程序等；
2. 同一系统内部上层服务对下层服务的调用，如一个软件程序一般分为表示层，业务层和数据层，表示层调用业务层的接口来完成自己的工作，而业务层又会调用数据层的接口来实现相应的业务等。

其以保证系统的正确和稳定为核心，重要性主要体现为以下几个方面：

- (1) 能够**提早发现 bug**，符合质量控制前移的理念。
- (2) 接口测试**低成本高效益**，因为接口测试可以自动化并且是持续集成的。
- (3) 接口测试从用户的角度**对系统接口进行全面检测**。实际项目中，接口测试会覆盖一定程度的业务逻辑

接口设计风格

1. 基于soap接口架构的接口：基于webservice协议，基于XML传输数据的。（比较老）
2. 基于RPC接口架构的接口（大型项目）dubbo协议
Spring Cloud 微服务
基于工具：Jmeter（插件）
3. 最核心的主流：基于restful接口架构 http协议（用到最多）
基于工具：很多工具

2. Jmeter

2.1文件目录

jmeter依赖于java环境（jdk1.8以上）

- 1.安装jdk以及配置jdk的环境变量。
- 2.jmeter解压即可使用。

j

 backups	2022-05-16 17:33	文件夹	
 bin	2022-04-13 21:49	文件夹	
 docs	2022-02-23 19:15	文件夹	
 extras	2022-02-23 21:14	文件夹	
 lib	2022-03-21 21:14	文件夹	
 licenses	2022-02-23 19:15	文件夹	
 printable_docs	2022-02-23 19:15	文件夹	
 LICENSE	1980-02-01 0:00	文件	15 KB
 NOTICE	1980-02-01 0:00	文件	1 KB
 README.md	1980-02-01 0:00	Markdown File	10 KB

(1) bin: 核心可执行文件, 包含配置

jmeter.bat - 启动文件, 运行JMeter(默认在GUI模式)

jmeter-n.cmd - 使用一个JMX文件运行非GUI测试

jmeter-n-r.cmd - 使用一个JMX文件远程运行一个非GUI测试

jmeter-t.cmd - 在GUI模式使用一个JMX文件

jmeter-server.bat - 分布式压测使用的启动文件; 以服务器模式启动JMeter

jmeter.properties: 核心配置文件

在此文件下修改为中文: `language=zh_CH`

report-template: 报告模板

shut.down: 关闭jmeter

(2) docs: 文档

(3) extras: 插件拓展的包

(4) lib: 依赖的包, 常见的核心包、工具类

ext: 核心包

junit: 单元测试包

(5) printable_docs

一些离线文档

语言版本中英文切换: GUI页面的options-choose language

2.2启动

bin目录下:

ApacheJMeter.jar

名称

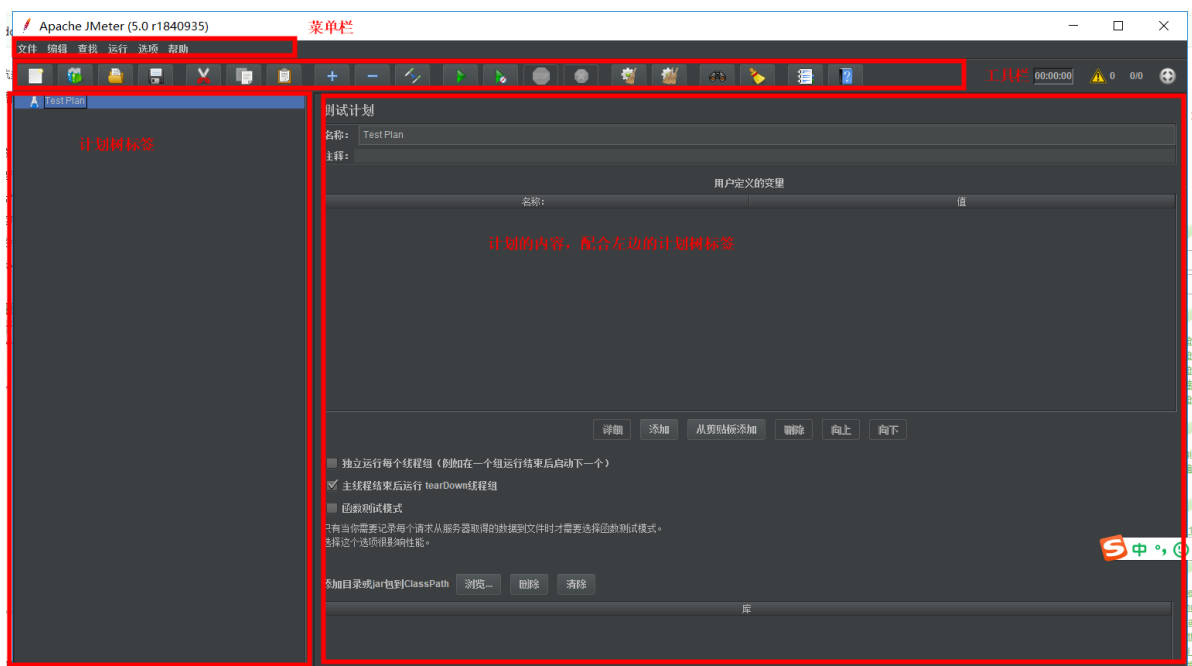
- examples
- report-template
- templates
- ApacheJMeter.jar
- BeanShellAssertion.bshrc
- BeanShellFunction.bshrc
- BeanShellListeners.bshrc
- BeanShellSampler.bshrc
- create-rmi-keystore.bat
- create-rmi-keystore.sh
- hc.parameters

2.3 软件界面

主界面

JMeter的主界面主要分为菜单栏、工具栏、树形标签栏和内容栏：

- 菜单栏：全部的功能的都包含在菜单栏中。
- 工具栏：工具栏中的按钮在菜单栏都可以找到，工具栏就相当于菜单栏常用功能的快捷按钮。
- 计划的树形标签栏：树形标签栏通常用来显示测试用例（计划）相关的标签。
- 内容栏：配合树形标签栏显示，树形标签中点击哪个标签，内容栏中就显示相应的内容和操作。



菜单栏

1. 文件



- **打开：打开一个JMX文件。**
- 模板：对常用的功能使用指导。主要有录制、JDBC测试、webserver测试等等，分为基本步骤和详细截图。如果点用户链接，则会链接到apache jmeter 网站查看详细的步骤和截图指导。
- 合并：会将多个JMX合并为一个。
- 保存测试计划：仅保存测试计划 工作台中添加的内容不会被保存。
- 保存测试计划为：将测试计划另存。
- **另存为：可以对工作台和测试计划或者测试例另存为JMX 注意另存为是点哪个位置，存的就是哪个内容。**
- 保存为测试片段：存为一个测试片段，只有线程组、测试计划可以保存为测试片段，工作台不能保存为一个测试片段。
- 还原，将现在的jmx还原为已经保存过的JMX

2. 编辑

- 保存节点为图片：将菜单的配置GUI保存为图片。
- 保存屏幕为图片：将整个jmeter界面保存为图片。
- 切换：类似于java中设置断点的意思。

3. 查找

- 搜索：搜索所有配置中匹配的项，匹配成功显示为红色。
- 重置搜索，清除搜索结果。

4. 运行

启动	Ctrl+R
不停顿开始	
停止	Ctrl+句点
关闭	Ctrl+逗号
远程启动	>
远程启动所有	Ctrl+Shift+R
远程停止	>
远程停止所有	Alt+X
远程关闭	>
远程关闭所有	Alt+Z
远程退出	>
远程退出所有	
清除	Ctrl+Shift+E
清除全部	Ctrl+E

- **启动：启动运行测试计划**
- 不停顿开始: 无停顿启动运行测试计划 1，可以忽略定时器 2，再启动时运行更快
- 远程启动/停止：指定一个远程agent运行/停止测试计划。
- 远程全部启动/停止: 让所有远程agent运行/停止测试。
- **停止: 停止执行测试计划。**
- **关闭: 关闭测试计划。**
- 远程关闭：关闭一个指定远程agent。
- 远程关闭所有: 关闭所有远程agent。
- 远程退出：指定一个远程agent退出执行。
- 远程退出所有: 所有远程agent退出执行。
- 清除：清除选择菜单的执行结果。
- 清除全部: 清除所有菜单的执行结果。

5. 选项

外观	>
日志查看	
日志级别	>
SSL管理器	Ctrl+M
选择语言	>
全部折叠	Ctrl+减号
全部展开	Ctrl+Shift+减号
放大	
缩小	
运行前自动保存	

- **外观：jmeter界面样式。**
- **日志查看器，选中后可以在右下方查看运行日志。**
- SSL管理器: 导入外置的SSL管理器，用于更好的管理证书, JMeter代理服务器不支持记录SSL(https)。
- 选择语言：选择界面的语言，目前支持中文、英文、法语、德语等等。中文版很多翻译不全，可以直接使用英文版的。
- 展开所有菜单。
- 折叠所有菜单

6. 帮助

- **What's this node?:** 当鼠标放在某个菜单的时候显示其含义
- Enable debug: 开启调试。
- Disable debug: 取消调试。
- Create a heap dump: 创建堆转储。这是创建当JVM崩溃的堆转储。这个文件可以用堆分析工具（如JHAT），以确定根本原因进行分析。

组件

1. **测试计划:** Jmeter的起点和容器

2. **线程组:** 代表一定的虚拟用户

3. **取样器:** 发送请求的最小单元

取样器是用来模拟用户操作，向服务器发送请求以及接受服务器的响应数据的一类元件，其中HTTP请求取样器比较常用

4. **逻辑控制器:** 控制组件的执行顺序

逻辑控制器可以控制取样器(samplers)的执行顺序。由此可知，控制器需要和取样器一起使用，否则控制器就没有什么意义了。放在控制器下面的所有的取样器都会当做一个整体，执行时也会一起被执行。

5. **前置处理器:** 在请求之前的操作

6. **后置处理器:** 在请求之后的操作

7. **断言:** 判断请求是否成功

8. **定时器:** 是否延迟或者间隔发送请求

9. **配置元件:** 取样器的配置信息

10. **监听器:** 复制收集测试结果

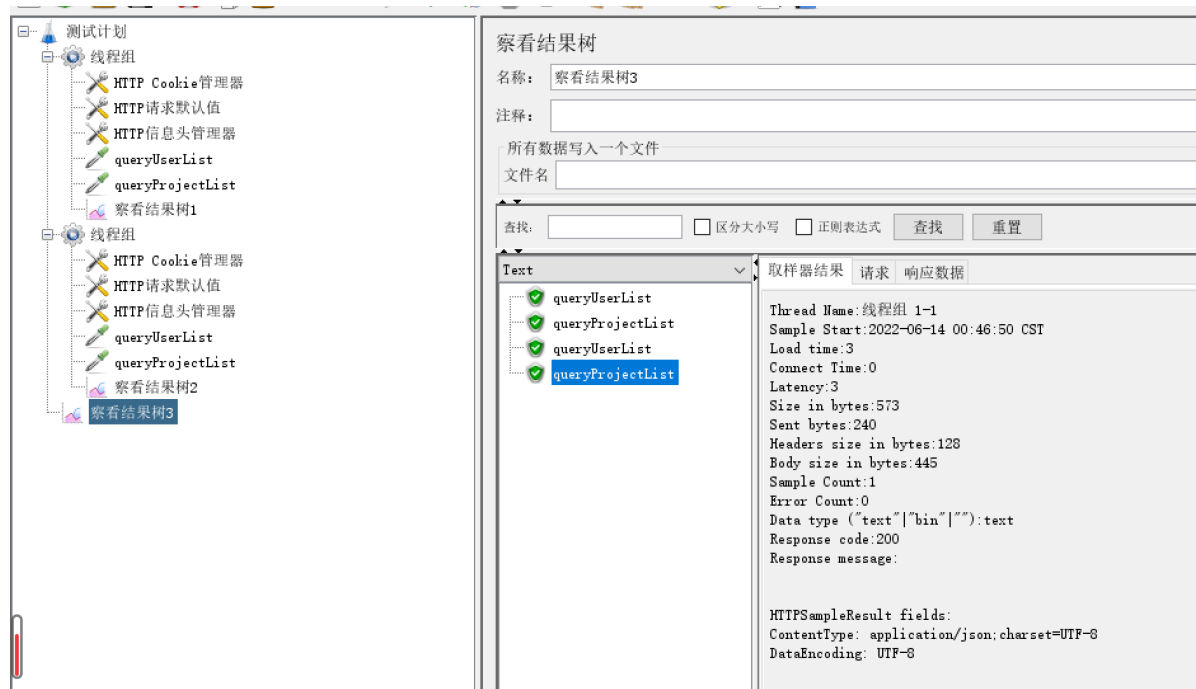
组件作用域:

察看结果树可以查看:

1.兄弟(同级)组件

2.兄弟组件下的子组件

在上图中，结果数3可以看到结果数1和2的结果，结果树1只能看到自己的结果（同级的取样器的结果）



执行顺序：测试计划》线程组》配置元件》前置处理器》定时器》逻辑控制器》取样器》后置处理器》断言》监听器（查看结果树）

所有的请求都是在取样器中发送的

2.4 接口测试流程与方法

- 1、找到接口文档，或者浏览器、fiddler抓包，找到接口请求地址，请求头，请求参数，
- 2、在测试计划中新建线程组
- 3、在线程组中去添加http请求
- 4、http请求中填写抓包中的请求地址，协议，路径，端口号，请求方法，请求参数

如果没有在body里面填写请求参数，就可以不用添加content-type这个请求头。这个的作用就是请求格式的解析

- 5、保存一个.jmx文件，这个文件就是我们刚刚所有操作保存为文档，ctrl+s或是鼠标点都可

下次可以在软件重启的时候去加载之前的脚本（右键打开最近）

- 6、去添加请求头，并在里面添加请求头，（content-type或token）
- 7、去线程组的监听添加查看结果树，它可以看到运行的结果



断言设置:

什么是断言：断言就是去把预期结果和实际结果做比较。

通常采用响应断言

jmeter断言判断的过程：

jmeter的响应的状态码是否大于等于400，如果是，则状态码不通过，则不判断断言。

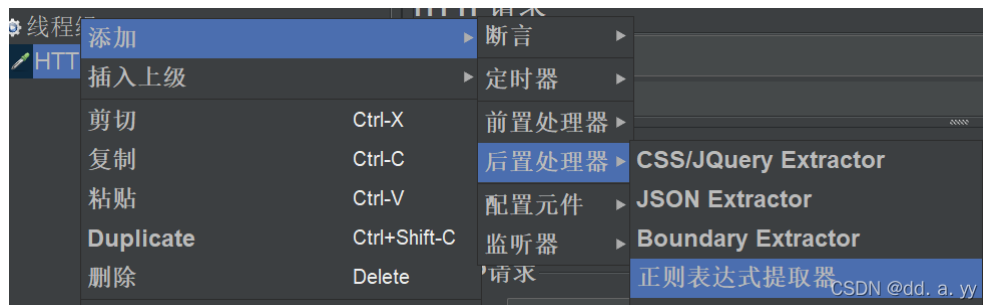
如果状态码通过则判断断言

关联设置:

什么是关联：上面的接口返回值作为下面接口的参数，这个参数是动态变化的，需要每次运行的时候动态获取参数的传递，比如token

如何设置关联

- 1、在需要获取token的接口中，添加后置处理器中的正则表达式提取器



- 2、在响应数据中抓取token的值，填入 (.*) ,这个值就是变量的值。旁边的，是其周围的匹配数据。然后填写模板

- 3、在需要引用的这个动态值的接口的请求头去添加，数据为 \${引用名称}



3.接口测试操作中可能遇到的问题

信息头问题 <https://www.cnblogs.com/yuan1225/p/11840345.html>

参数格式问题 <https://blog.csdn.net/romon19/article/details/92124853>

JMeter测试增删改查例子 <https://blog.csdn.net/sulia1234567890/article/details/123353045>

添加cookie方法 <https://blog.csdn.net/LinAndCurry/article/details/112861356>