**JMeter对各种类型接口的测试**

默认做接口测试前，已经给出明确的接口文档（如，<http://test.nnzhp.cn/wiki/index.php?doc-view-59>）；本地配好了JMeter 3.x的运行环境；

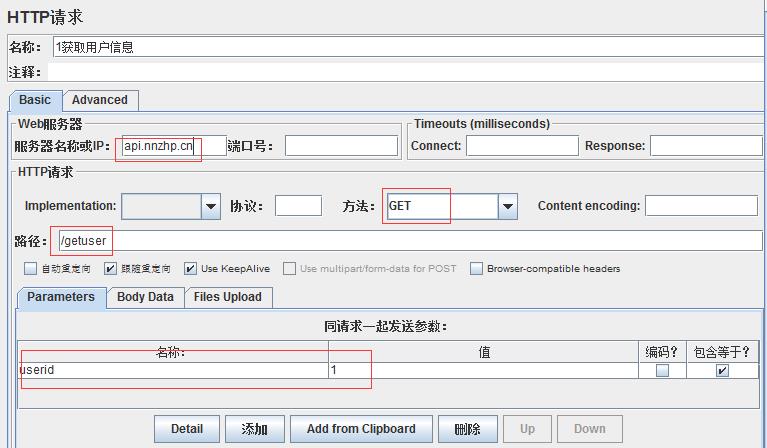
打开JMeter，添加一个线程组和该线程组的查看结果树。以下的几种接口请求我们都在这个线程组中添加和运行。

如果你的JMeter返回数据是乱码，解决办法是，在JMeter安装路径的bin目录下，打开文件jmeter.properties，把Sampleresult.default.encoding的值改为 utf-8 即可。

1. 普通的以key-value传参的get请求

e.g. 获取用户信息

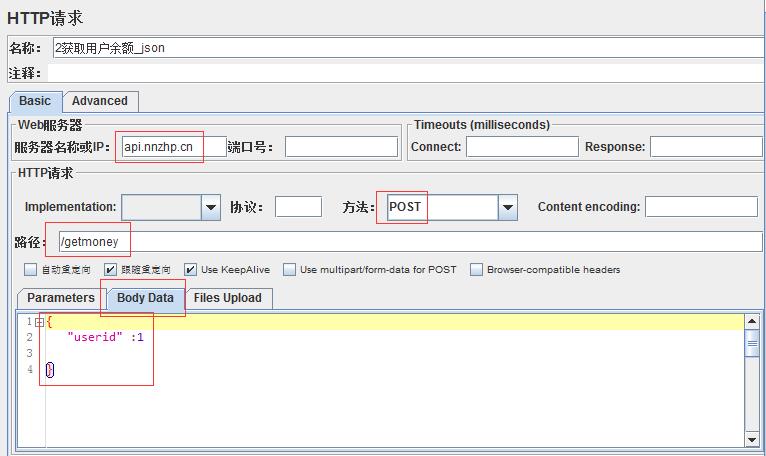
添加http请求；填写服务器域名或IP；方法选GET；填写路径；添加参数；运行并查看结果。



 2. 以Json串传参的post请求

e.g. 获取用户余额

添加http请求；填写服务器域名或IP；方法选POST；填写路径；在参数区域选Body Data标签；写入Json串参数，注意Json格式；运行并查看结果。



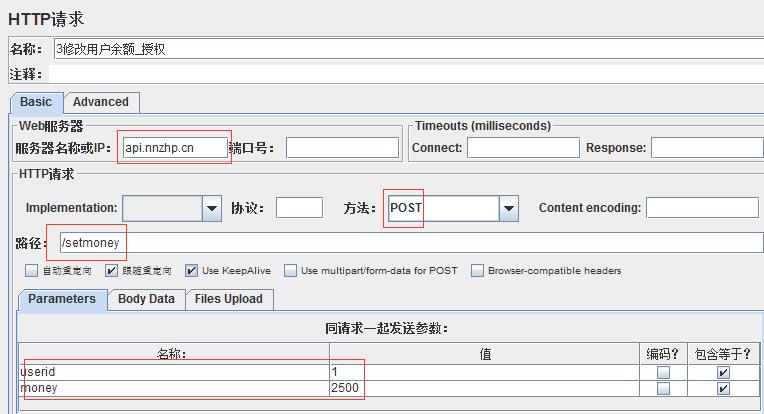
**\*注意：**

Json传参时，往往参数相对复杂，规模较大，很多时候是Json串再嵌套Json串，在写入body的时候如果哪里格式不对会影响传参，而JMeter本身是不能对Json进行格式校验的（Postman可以进行简单的Json校验），我们可以借助这个网站来校验你的Json串的格式，[www.bejson.com](http://www.bejson.com/) 再把确保格式正确的Json串填入body。

3.  请求时需要http权限验证

e.g. 修改用户余额

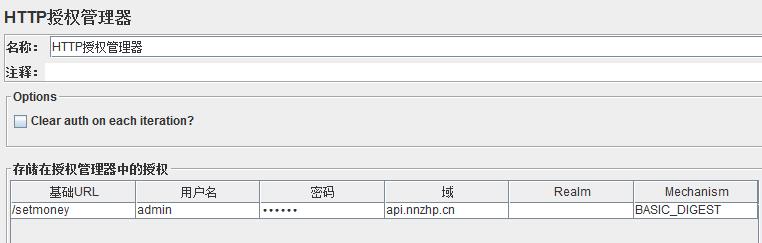
添加http请求；填写服务器域名或IP；方法选POST；填写路径；填写参数；



添加授权管理器，右击线程组->添加->配置元件->HTTP授权管理器；



打开HTTP授权管理器，在基础URL添加路径；填写用户名、密码（接口文档会给出）；添加域即请求的服务器域名或IP；其他默认；

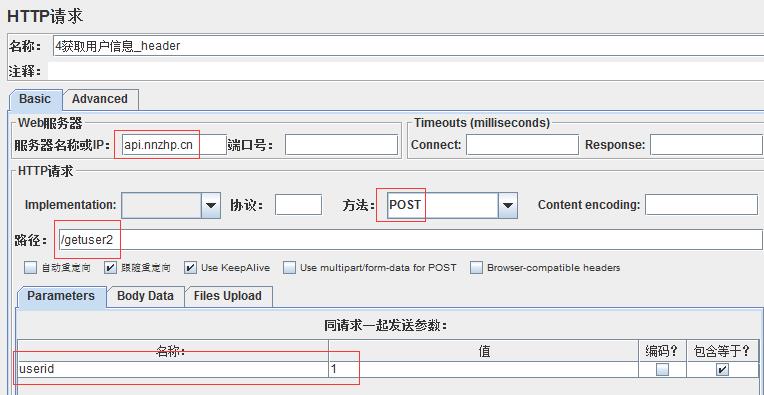


运行请求并查看结果。（该接口暂时有问题请求返回500，只要掌握需要权限验证的接口如何做就好）

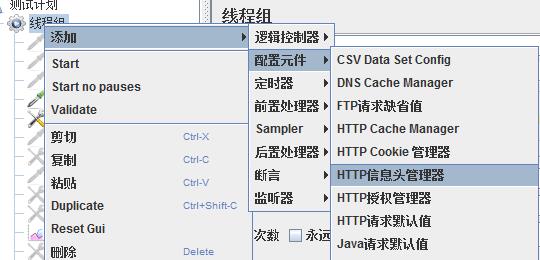
4.  请求时需要添加请求标头

e.g. 获取用户信息2

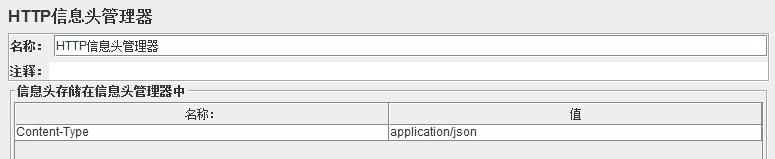
添加http请求；填写服务器域名或IP；方法选POST；填写路径；填写参数；



添加HTTP信息头管理器，右击线程组->添加->配置元件->HTTP信息头管理器；



打开HTTP信息头管理器，参照接口文档填写好header信息；

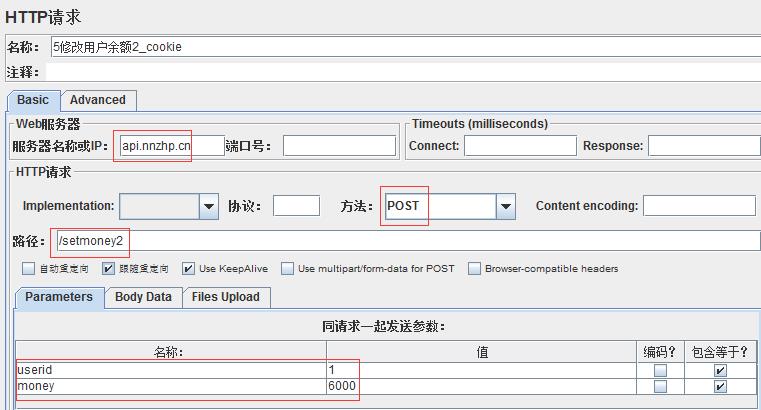


运行请求并查看结果。

5.  请求时需要添加Cookie

e.g. 修改用户余额2

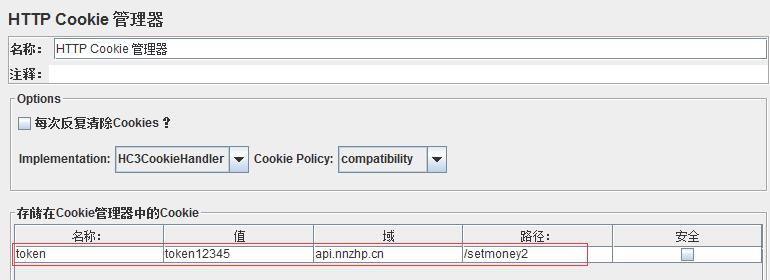
添加http请求；填写服务器域名或IP；方法选POST；填写路径；填写参数；



添加HTTP Cookie管理器，右击线程组->添加->配置元件->HTTP Cookie管理器；



打开Cookie管理器，参照接口文档填写好Cookie信息；

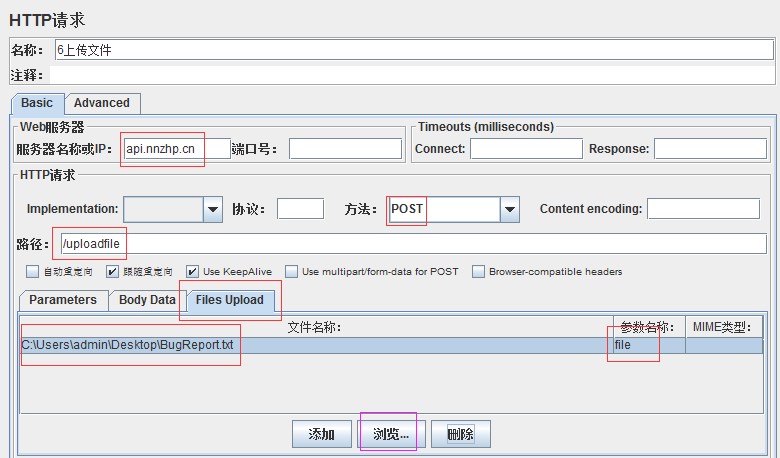


 发送请求并查看结果。

6.  请求通过某个接口上传文件

    e.g.上传文件

    添加http请求；填写服务器域名或IP；方法选POST；填写路径；在参数区域选Files Upload标签，添加文件全名（包含绝对路径，也可以点击浏览按钮添加），写入参数名；发送请求并查看结果。



**JMeter的参数化**

1. 指定参数值

这种方法常用来解决线程组中多个请求共用相同的数据，比如我们以上所有请求都有相同的IP地址，当地址改变时可以一改全改。

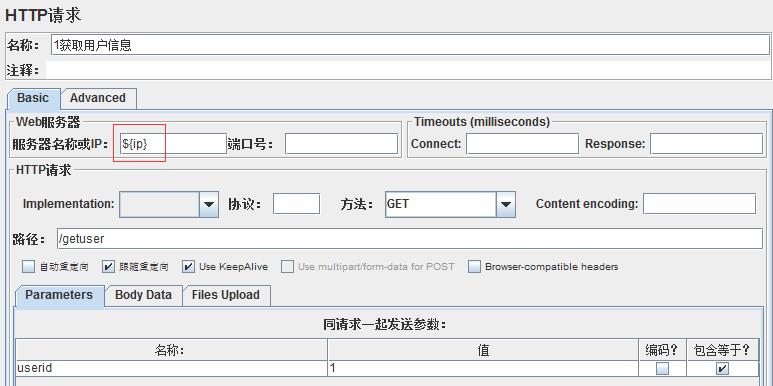
添加用户定义的变量，右击线程组->添加->配置元件->用户定义的变量；



打开用户定义的变量，添加变量，写好名称和值；



脚本中用到该值的所有地方都可以引用这个变量名了，引用变量的格式为 ${变量名}；



2. 借助函数生成参数

测试过程中，有时候数据不适合被指定，可选择借助函数来生成，以下是最常用的几种。

*      用函数生成随机数：（\_\_Random）

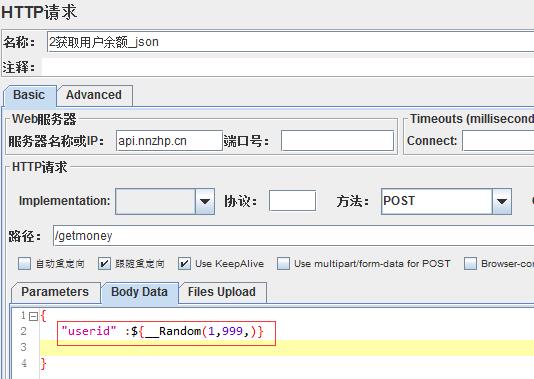
在Jmeter中点击 选项->函数助手对话框；



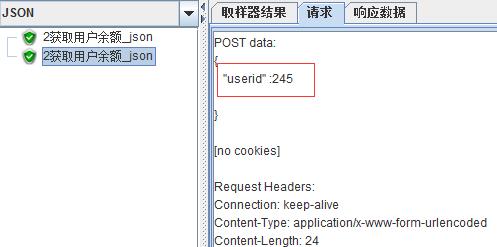
在弹出的对话框中选\_\_Random，填写随机数的范围，点击生成按钮，就看到你需要的字符串了；



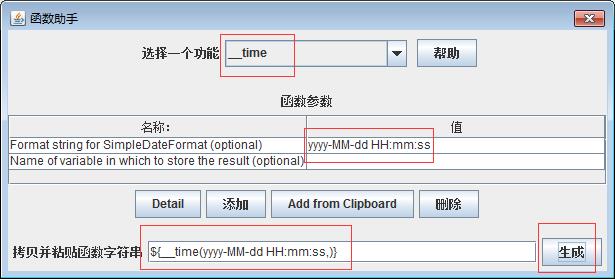
尝试在请求中使用生成的随机数，把刚拷贝的字符串粘贴到请求的参数里；



发送请求，查看请求中参数值的情况，参数生效了。



*     用函数生成当前时间：（\_\_time）

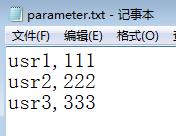


*     用函数生成通用的唯一识别码：（\_\_UUID）

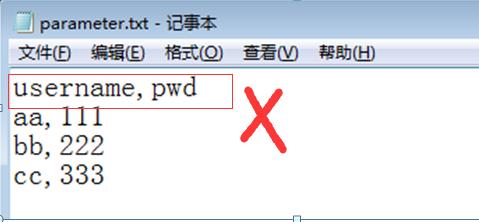


3. 文件传参

用于读取在文件中维护的参数，如参数文件内容如下；



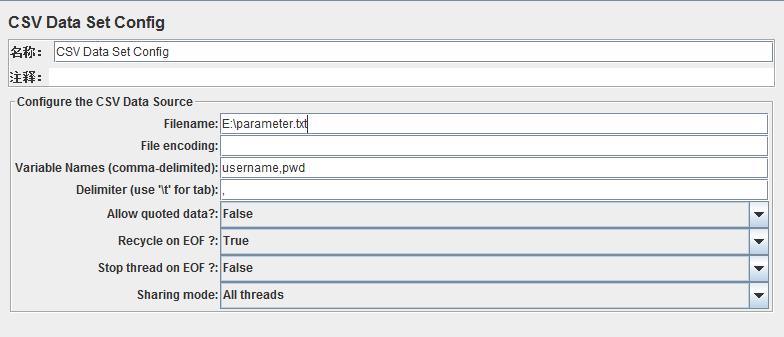
注意，Jmeter读取文件总的参数是从第一行就开始了的，千万不要犯下面的错误。



在Jmeter中右击线程组->添加->配置元件->CSV Data Set Config；



打开设置窗口，写入参数文件的全名（含绝对路径），变量名，Delimiter就用默认的逗号，表示在准备参数文件时参数表的列之间用逗号隔开；



在请求中引用参数时应该是酱紫的：${username}  ${pwd}，用它们来顶替对应的参数值。

4. 关联传参

在线程组内，当你想用一个请求的响应结果作为另一个请求的入参时就需要用到关联。

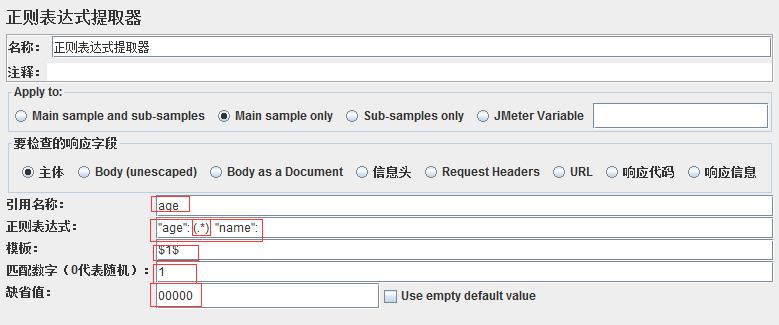
* 用正则表达式关联

思路是先从某个请求的响应数据中提取你需要的值，在把这个值在另一个请求中入参，操作如下：

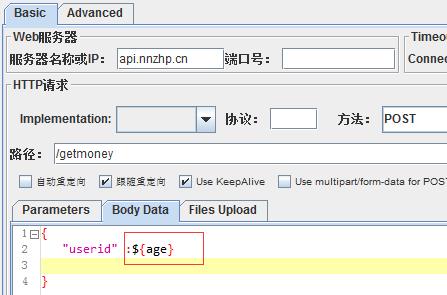
在需要被提取响应数据的请求下添加正则表达式提取器，右击添加->后置处理器->正则表达式提取器；



在正则表达式提取器中填写 引用名称，不说也知道用途；正则表达式，最简单的办法是在响应数据中把要提取的目标值左右若干字符包含目标值的一行拷贝出来，粘贴到正则表达式框内，在把目标值用一对小括号替换，括号里用添上合适的匹配符本例如(.\*)； 注意在响应数据中拷贝时尽量避开需要转义的字符如果不能避开，请转义；模板通常固定填$1$；匹配数字1代表区匹配的第一个值；缺省值是在匹配失败时取得的值，通常用来检查匹配是否成功；



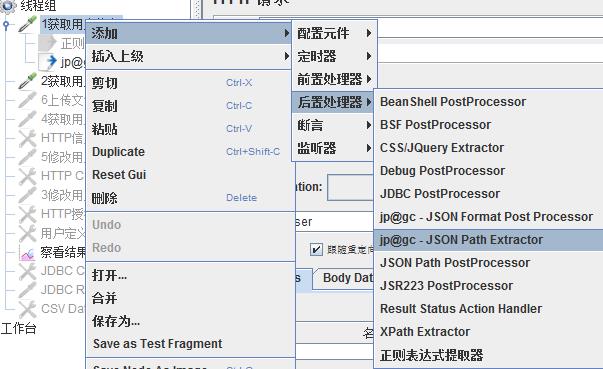
用${age}入参给另一个请求，查看结果传参成功。



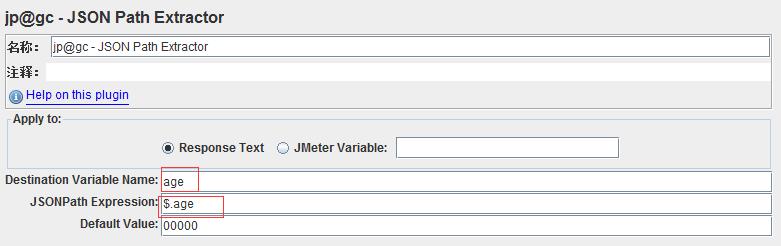


* 用Json Path Extractor 插件关联

只针对Json格式的Response 取值，右击请求添加->后置处理器->JSON Path Extractor；



在JSON Path Extractor中填入引用名和JSONPath Expression；

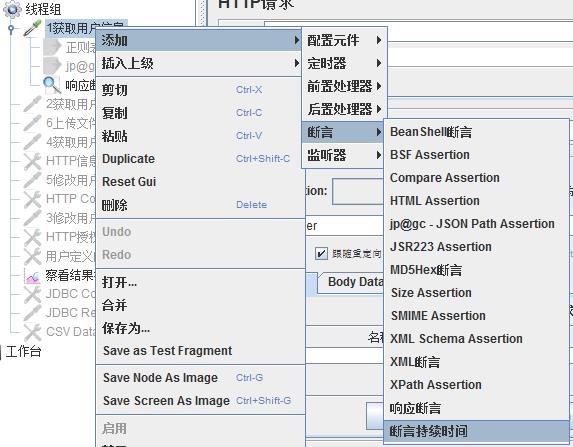


在另一个请求中入参，也和上面的结果一样转参成功。

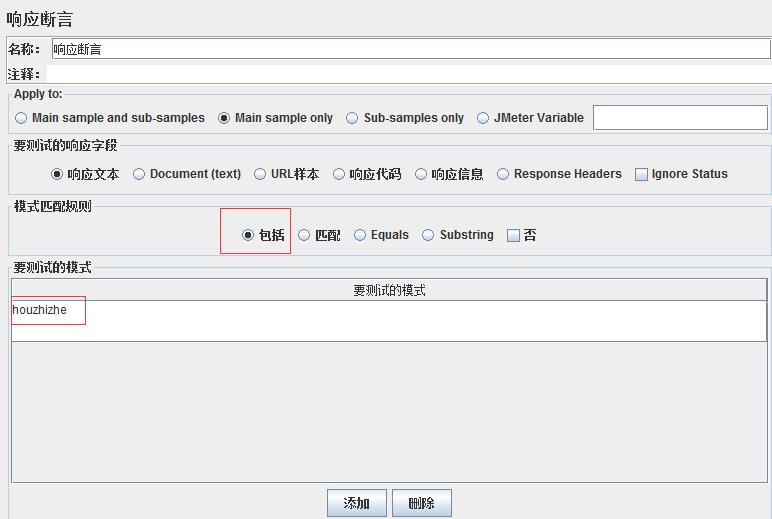
注意，JSON Path Extractor这个插件只能在Jmeter3.x以上的版本安装。

**Jmeter添加断言检查结果**

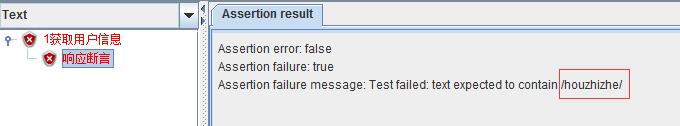
右击请求添加->断言->响应断言；

****

添加并写入希望在响应结果中看到的内容；



发送请求，查看结果；

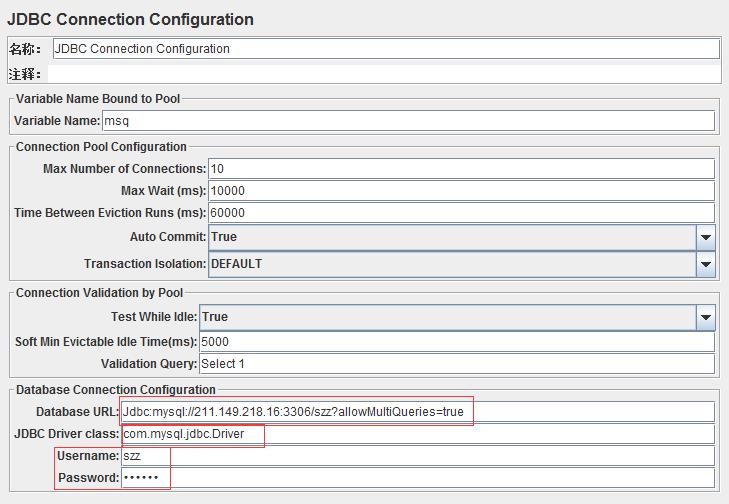


**Jmeter操作Mysql**

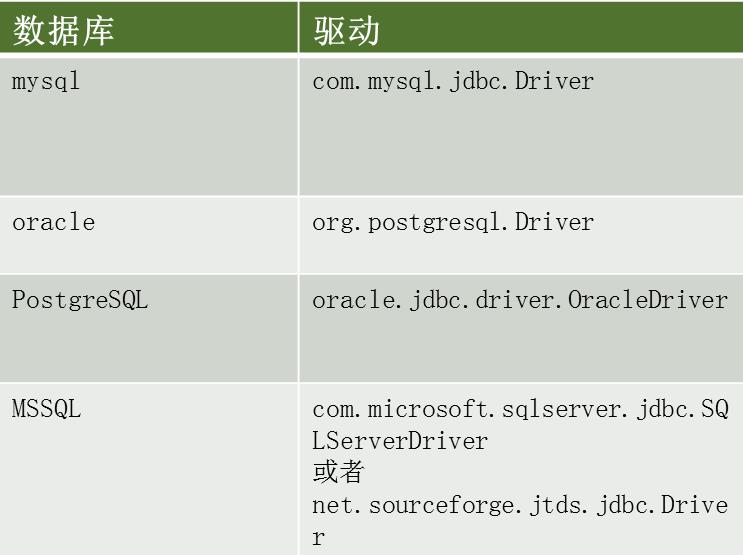
测试计划添加.jar包 mysql-connector-java-5.1.7-bin.jar用于使Jmeter可以读取Mysql；



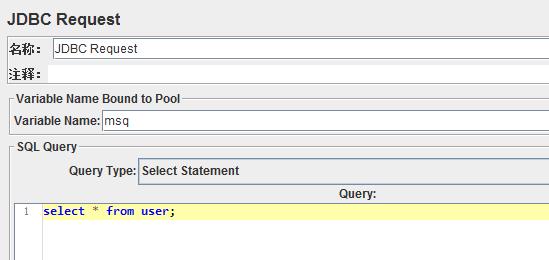
 线程组添加 JDBC Connection Configuration （添加->配置元件-> JDBC Connection Configuration）用于配置Jmeter和Mysql的连接；



JDBC Driver class是固定的，Jmeter操作各类数据的Driver参考下图；



线程组添加JDBC Request（添加->Sampler-> JDBC Request） 用于写sql来操作Mysql；



发送请求，成功访问Msql。

