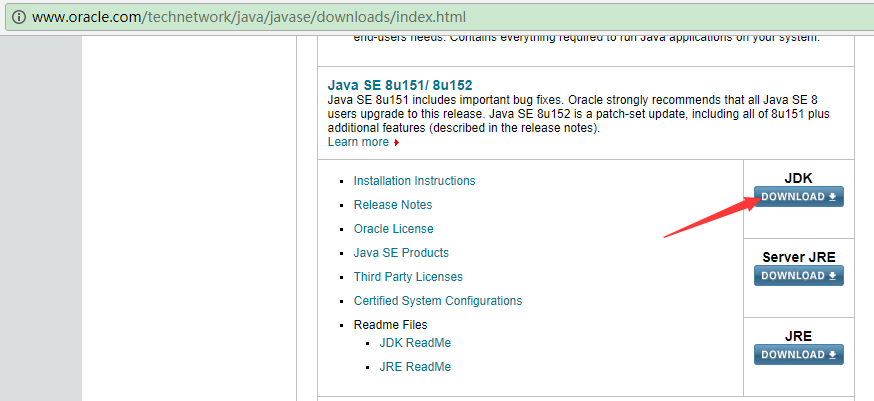
Apache JMeter是一款纯java编写负载功能测试和性能测试开源工具软件。相比Loadrunner而言，JMeter小巧轻便且免费，逐渐成为了主流的性能测试工具，是每个测试人员都必须要掌握的工具之一。

本文为JMeter性能测试完整入门篇，从Jmeter下载安装到编写一个完整性能测试脚本、最终执行性能测试并分析性能测试结果。

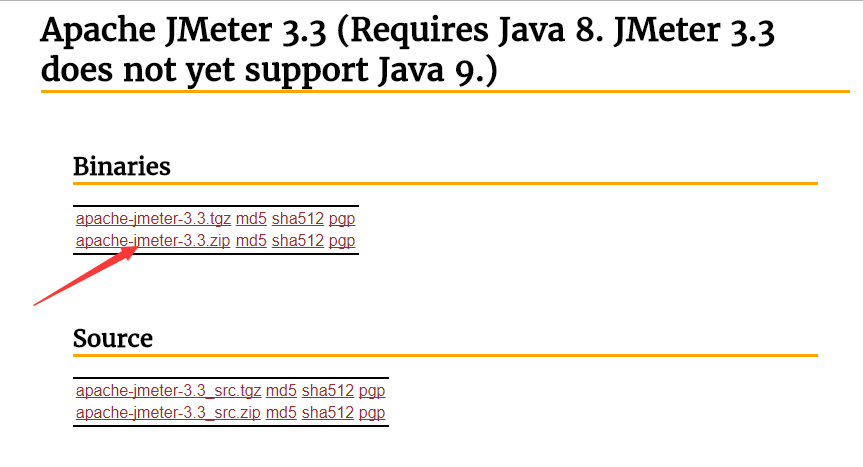
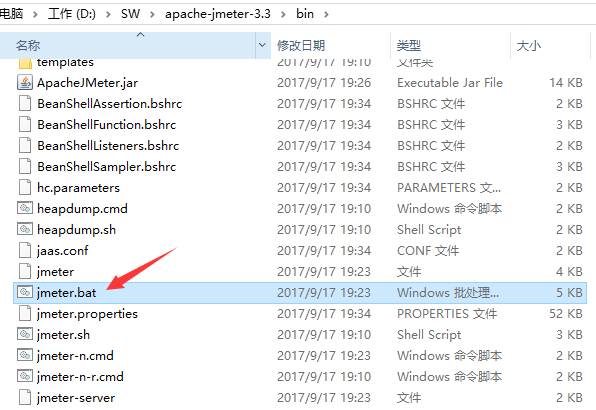
运行环境为Windows 10系统，JDK版本为1.8，JMeter版本为3.3。

# Jmeter安装

## JDK安装

由于Jmeter是基于java开发，首先需要下载安装JDK （目前JMeter只支持到Java 8，尚不支持 Java 9）   
1. 官网下载地址：<http://www.oracle.com/technetwork/java/javase/downloads/index.html>   
2. 选择Java SE 8u151/ 8u152，点击JDK下载   
  
3. 安装下载的JDK   
4. 配置系统环境变量

## JMeter安装

* 官网下载地址：<http://jmeter.apache.org/download_jmeter.cgi>
* 下载最新JMeter 3.3版本：apache-jmeter-3.3.zip   
  
* 下载完成后解压zip包
* 启动JMeter   
  双击JMeter解压路径（apache-jmeter-3.3\bin）bin下面的jmeter.bat即可   
  

# 测试实例

我们选取最常见的百度搜索接口：

## 接口地址

<http://www.baidu.com/s?ie=utf-8&wd=jmeter>性能测试

## 请求参数

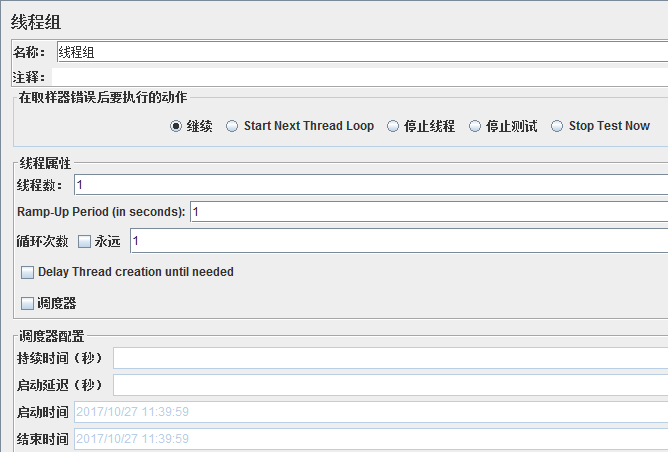
ie：编码方式，默认为utf-8   
 wd: 搜索词

## 返回结果

搜索结果，我们可以通过校验结果中是否含有搜索词wd来判断本次请求成功或失败。

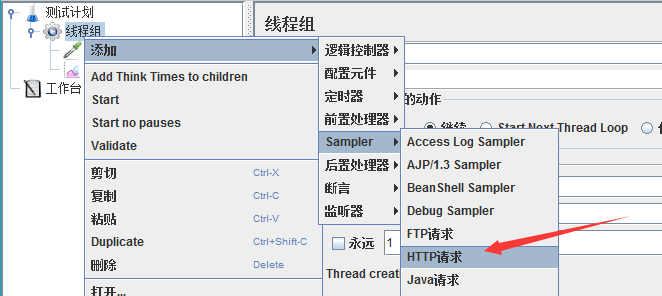
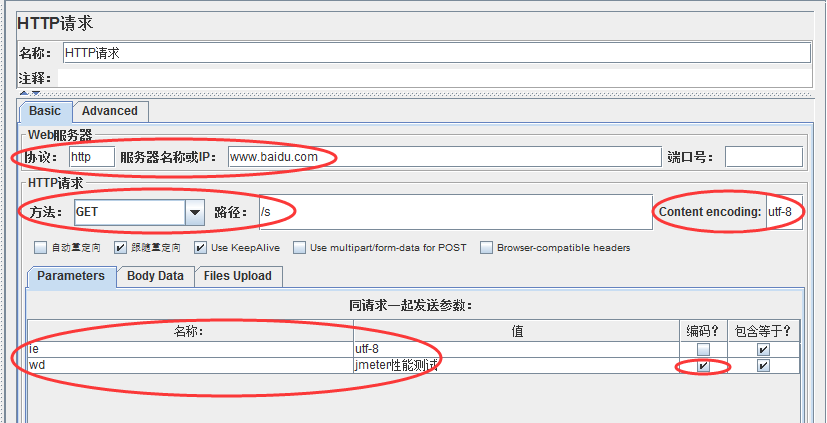
# JMeter脚本编写

## 添加线程组

右键点击“测试计划” -> “添加” -> “Threads(Users)” -> “线程组”   
   
这里可以配置线程组名称，线程数，准备时长（Ramp-Up Period(in seconds)）循环次数，调度器等参数：   
   
线程组参数详解：   
1. 线程数：虚拟用户数。一个虚拟用户占用一个进程或线程。设置多少虚拟用户数在这里也就是设置多少个线程数。   
2. Ramp-Up Period(in seconds)准备时长：设置的虚拟用户数需要多长时间全部启动。如果线程数为10，准备时长为2，那么需要2秒钟启动10个线程，也就是每秒钟启动5个线程。   
3. 循环次数：每个线程发送请求的次数。如果线程数为10，循环次数为100，那么每个线程发送100次请求。总请求数为10\*100=1000 。如果勾选了“永远”，那么所有线程会一直发送请求，一到选择停止运行脚本。   
4. Delay Thread creation until needed：直到需要时延迟线程的创建。   
5. 调度器：设置线程组启动的开始时间和结束时间(配置调度器时，需要勾选循环次数为永远)   
持续时间（秒）：测试持续时间，会覆盖结束时间   
启动延迟（秒）：测试延迟启动时间，会覆盖启动时间   
启动时间：测试启动时间，启动延迟会覆盖它。当启动时间已过，手动只需测试时当前时间也会覆盖它。   
结束时间：测试结束时间，持续时间会覆盖它。

因为接口调试需要，我们暂时均使用默认设置，待后面真正执行性能测试时再回来配置。

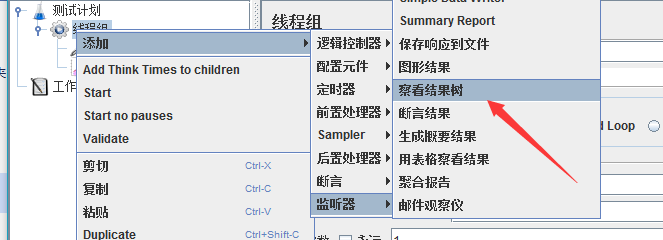
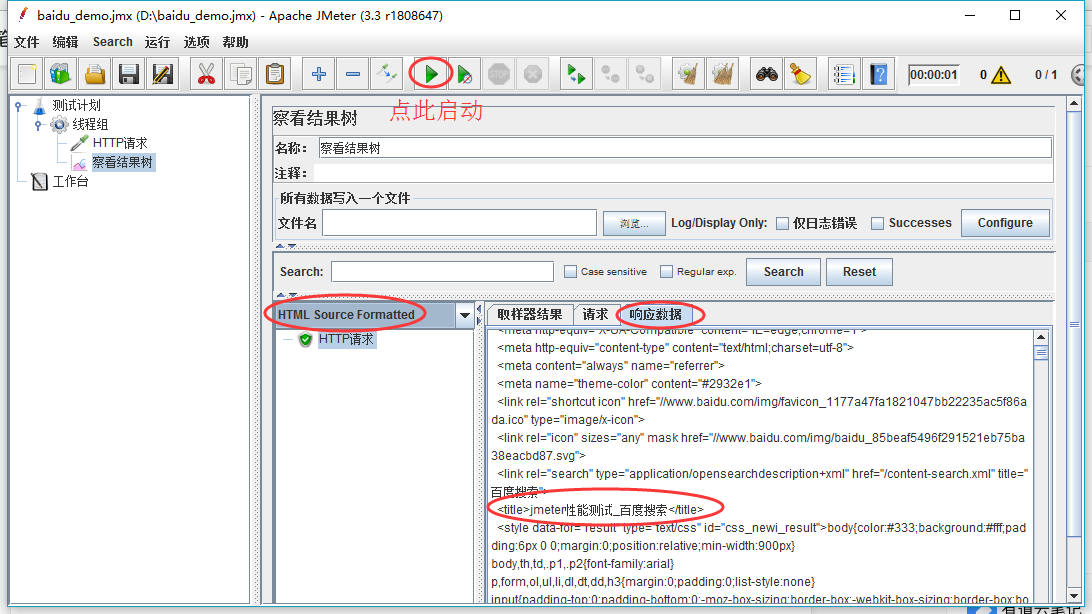
## 添加HTTP请求

右键点击“线程组” -> “添加” -> “Sampler” -> “HTTP请求”   
   
对于我们的接口<http://www.baidu.com/s?ie=utf-8&wd=jmeter>性能测试，可以参考下图填写：   
   
Http请求主要参数详解：

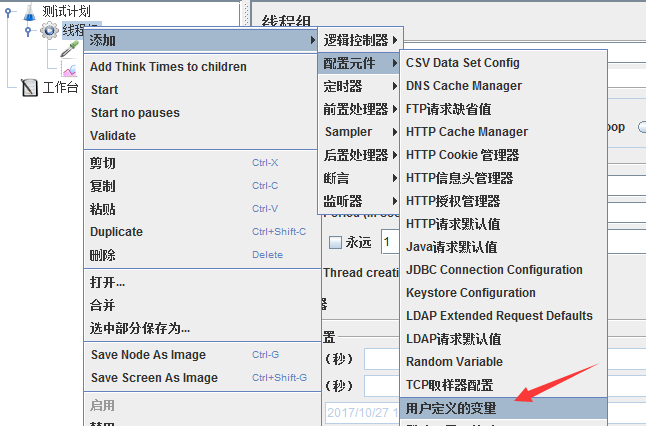
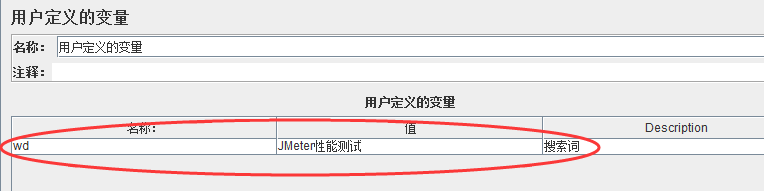
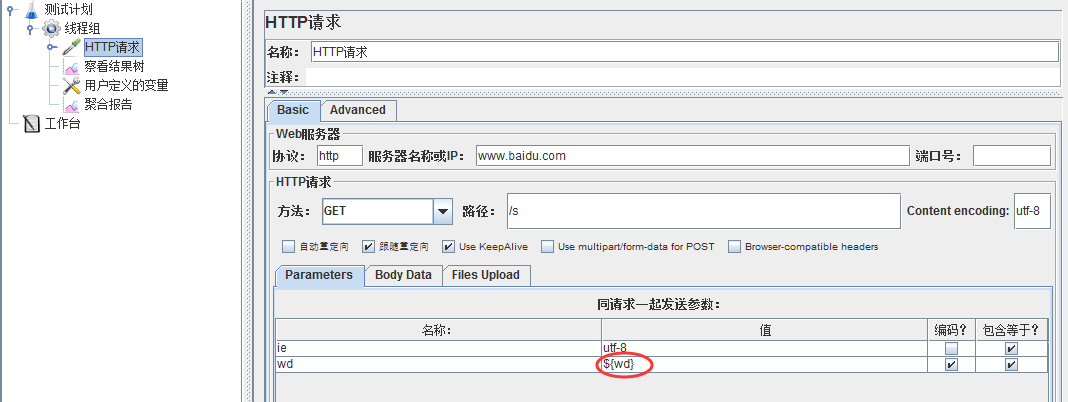
Web服务器   
协议：向目标服务器发送HTTP请求协议，可以是HTTP或HTTPS，默认为HTTP   
服务器名称或IP ：HTTP请求发送的目标服务器名称或IP   
端口号：目标服务器的端口号，默认值为80   
2.Http请求   
方法：发送HTTP请求的方法，可用方法包括GET、POST、HEAD、PUT、OPTIONS、TRACE、DELETE等。   
路径：目标URL路径（URL中去掉服务器地址、端口及参数后剩余部分）   
Content encoding ：编码方式，默认为ISO-8859-1编码，这里配置为utf-8

同请求一起发送参数   
在请求中发送的URL参数，用户可以将URL中所有参数设置在本表中，表中每行为一个参数（对应URL中的 name=value），注意参数传入中文时需要勾选“编码”

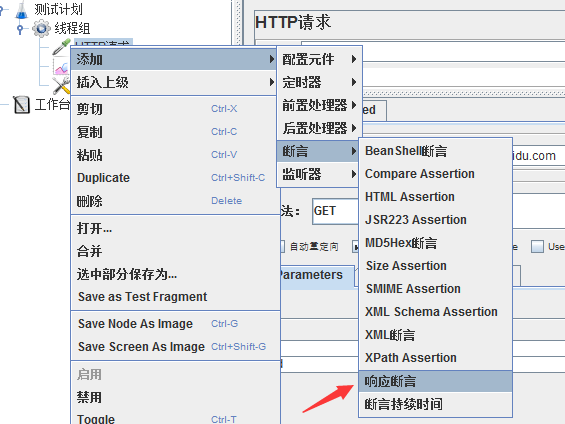
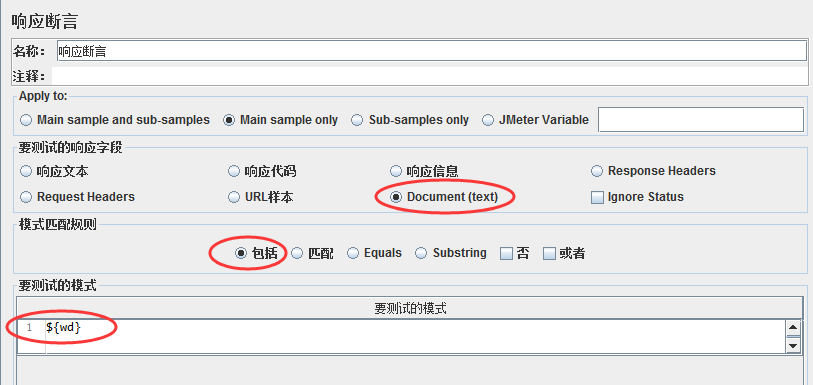
## 添加察看结果树

右键点击“线程组” -> “添加” -> “监听器” -> “察看结果树”   
   
这时，我们运行Http请求，修改响应数据格式为“HTML Source Formatted”，可以看到本次搜索返回结果页面标题为”jmeter性能测试\_百度搜索“。   


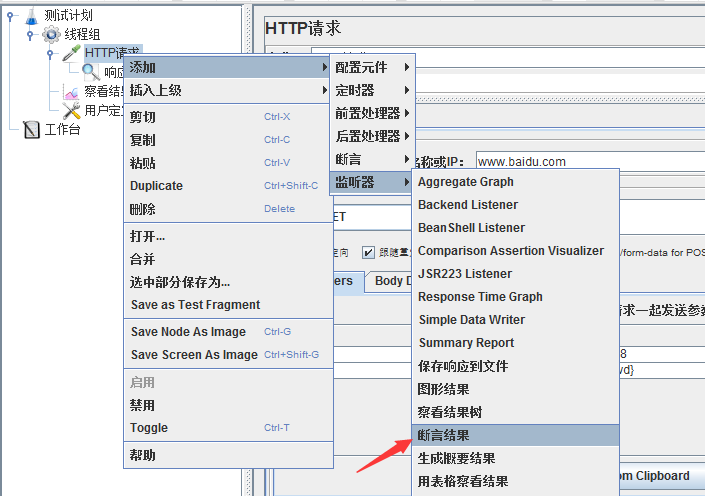
## 添加用户自定义变量

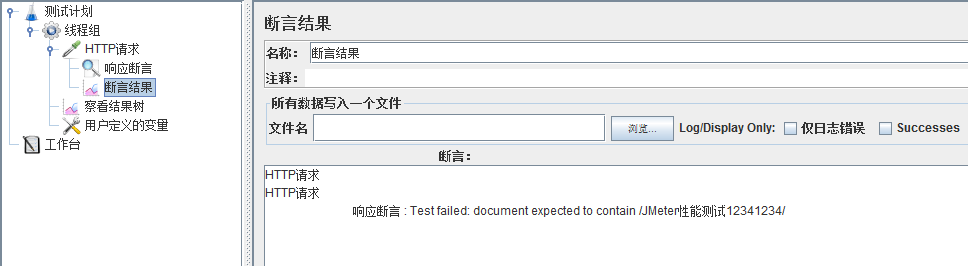
我们可以添加用户自定义变量用以Http请求参数化，右键点击“线程组” -> “添加” -> “配置元件” -> “用户定义的变量”：   
   
新增一个参数wd，存放搜索词：   
   
并在Http请求中使用该参数，格式为：${wd}   


## 添加断言

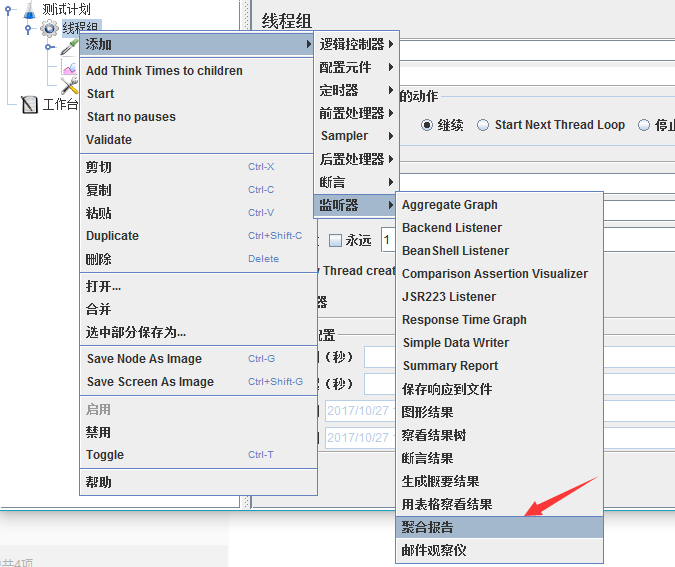
右键点击“HTTP请求” -> “添加”-> “断言” -> “响应断言”   
   
我们校验返回的文本中是否包含搜索词，添加参数${wd}到要测试的模式中：   


## 添加断言结果

右键点击“HTTP请求” -> “添加”-> “监听器” -> “断言结果”   


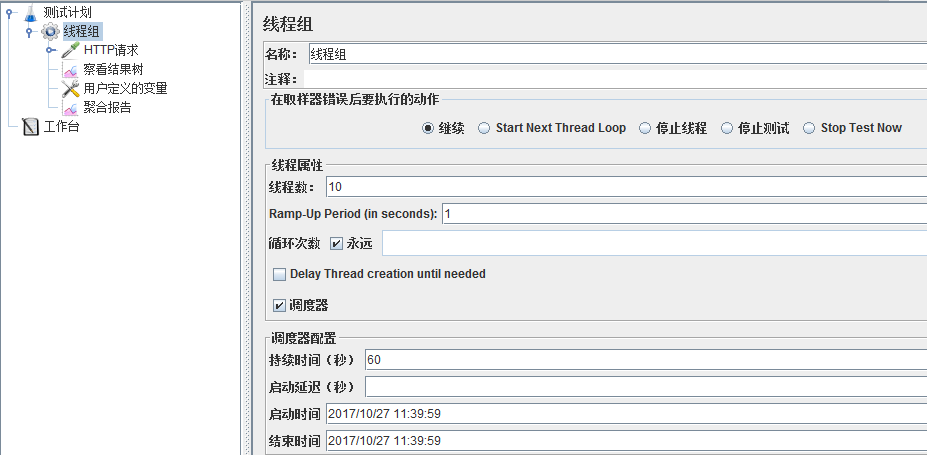
这时，我们再运行一次就可以看到断言结果成功或失败了   


## 添加聚合报告

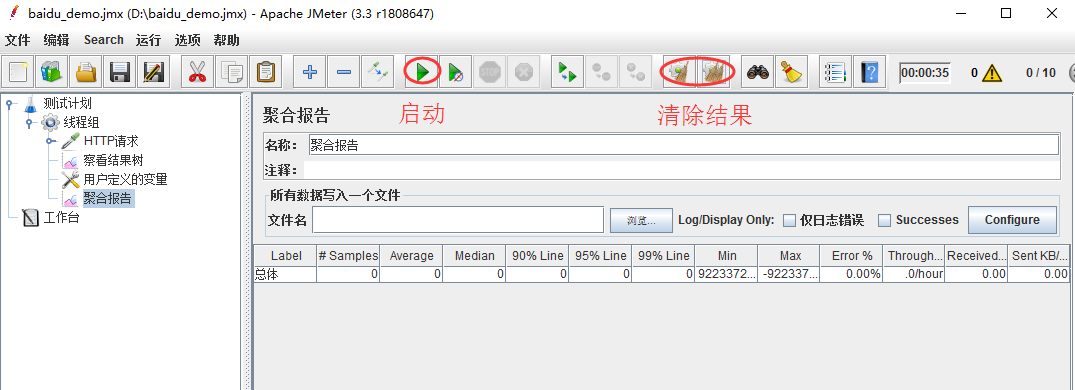
右键点击“线程组” -> “添加” -> “监听器” -> “聚合报告”，用以存放性能测试报告   
   
这样，我们就完成了一个完整Http接口的JMeter性能测试脚本编写。

# 执行性能测试

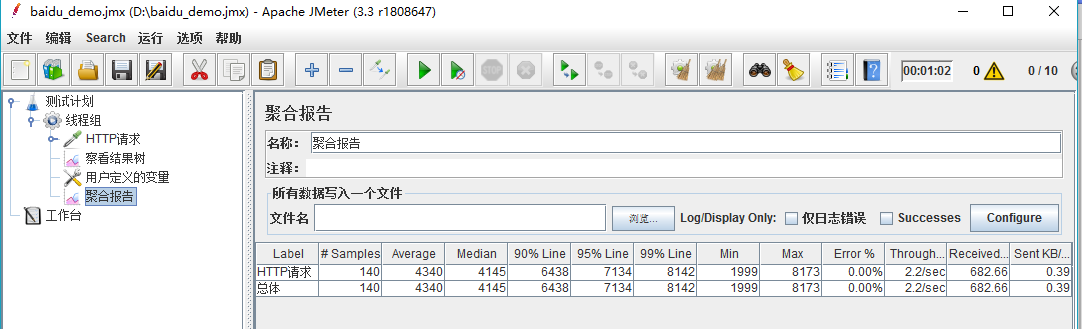
## 配置线程组

点击线程组，配置本次性能测试相关参数：线程数，循环次数，持续时间等，这里我们配置并发用户数为10，持续时间为60s   


## 执行测试

点击绿色小箭头按钮即可启动测试，测试之前需要点击小扫把按钮清除之前的调试结果。   


# 分析测试报告

待性能测试执行完成后，打开聚合报告可以看到：   


聚合报告参数详解：   
1. Label：每个 JMeter 的 element（例如 HTTP Request）都有一个 Name 属性，这里显示的就是 Name 属性的值   
2. #Samples：请求数——表示这次测试中一共发出了多少个请求，如果模拟10个用户，每个用户迭代10次，那么这里显示100   
3. Average：平均响应时间——默认情况下是单个 Request 的平均响应时间，当使用了 Transaction Controller 时，以Transaction 为单位显示平均响应时间   
4. Median：中位数，也就是 50％ 用户的响应时间   
5. 90% Line：90％ 用户的响应时间   
6. Min：最小响应时间   
7. Max：最大响应时间   
8. Error%：错误率——错误请求数/请求总数   
9. Throughput：吞吐量——默认情况下表示每秒完成的请求数（Request per Second），当使用了 Transaction Controller 时，也可以表示类似 LoadRunner 的 Transaction per Second 数   
10. KB/Sec：每秒从服务器端接收到的数据量，相当于LoadRunner中的Throughput/Sec

一般而言，性能测试中我们需要重点关注的数据有： #Samples 请求数，Average 平均响应时间，Min 最小响应时间，Max 最大响应时间，Error% 错误率及Throughput 吞吐量。