Jmeter Performance Testing

Winter 2016 – Author: Thao Le

Contents

[1. Week 2: Browser Recording, WorkBench, Running Your First Test, Listener, Reading Test Result 4](#_Toc467784699)

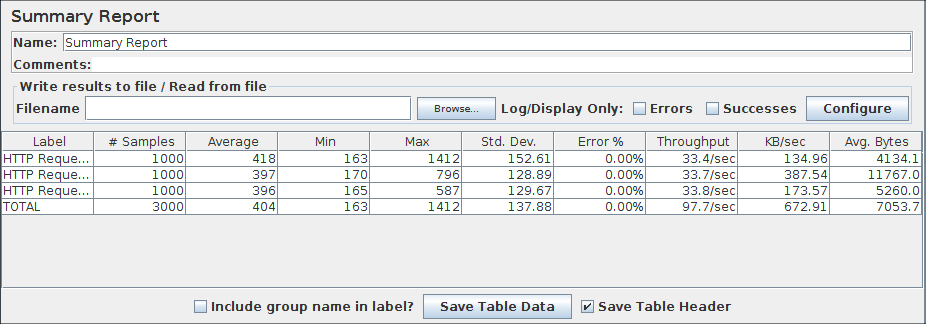
[2. Contact Us 6](#_Toc467784700)

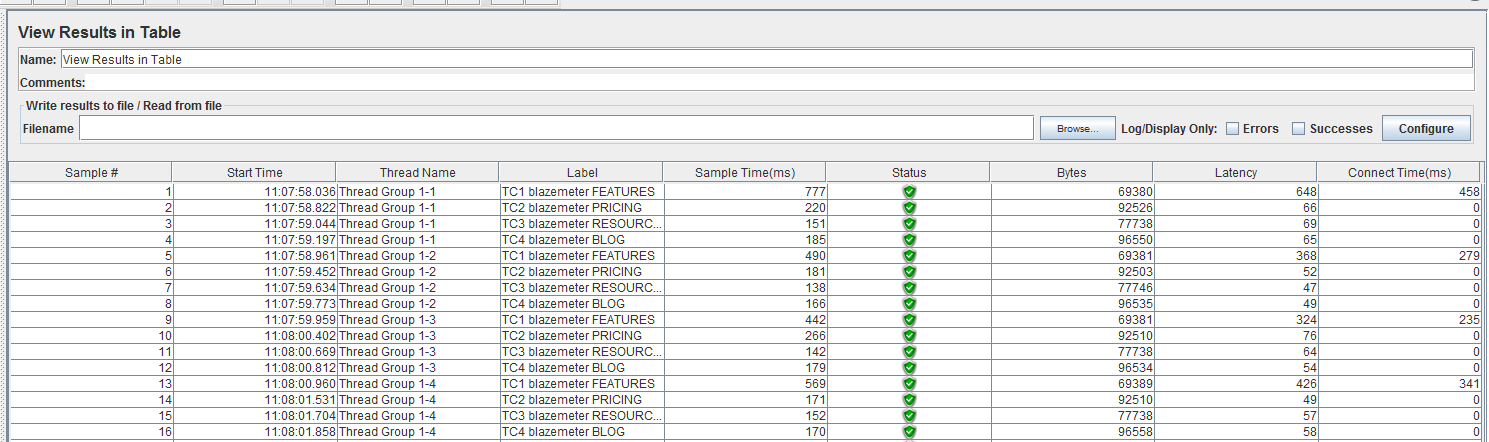
# Week 2: Browser Recording, WorkBench, Running Your First Test, Listener, Reading Test Result

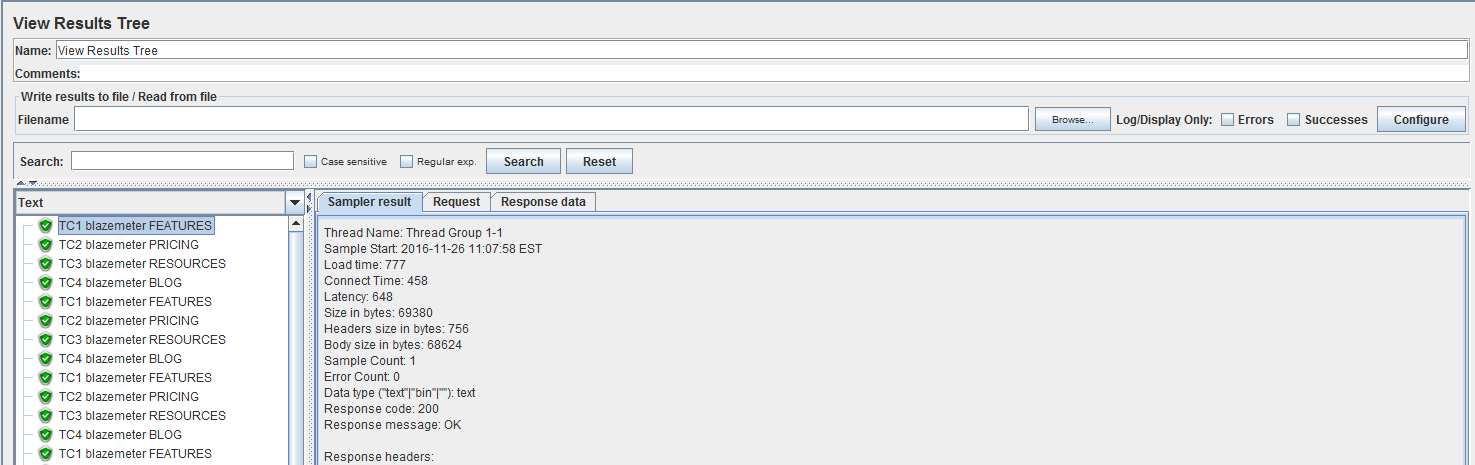
* Download and install Firefox from https://www.mozilla.org/en-US/firefox/new/?scene=2
* WorkBench: is a temporary workspace in Jmeter to store test elements
* Browser Set-Up For Recording:  Following Jmeter\_Recording\_Step\_by\_Step.pdf
* HTTP Test Script Recorder: Following Jmeter\_Recording\_Step\_by\_Step.pdf
* Including or Excluding URL Patterns (Optional)
  + In the HTTP(S) Script Recorder, you may add URL Patterns, written as regular expressions, to include or exclude when you record. This can be useful to either include only the types of content you want to request (e.g. \*.html, \*.php, etc) or to exclude the types of content you do not want to request (e.g. \*.jpg, \*.png, \*.js, etc).
  + To add a URL Pattern, click the "Add" button under the URL Patterns to Include or URL Patterns to Exclude section, then click on the top of the white area in the section. You should now be able to type in a pattern. Repeat the process to add more patterns.
  + Example: URL Patterns for webpages
    - .\*\.html
    - .\*\.php
    - .\*\.htm
  + Example: URL Patterns for images
    - .\*\.png
    - .\*\.jpg
    - .\*\.gif
* Using Controller To Organize
  + Simple Controller: The Simple Logic Controller lets you organize your Samplers and other Logic Controllers.
  + Random Controller: The Random Logic Controller will pick up randomly one sub-controllers and samplers.
  + Random Order Controller: The Random Order Controller is much like a Simple Controller in that it will execute each child element at most once, but the order of execution of the nodes will be random.
  + Loop Controller: JMeter will loop through test elements below it a certain number of times, in addition to the loop value you specified for the Thread Group.
  + If Controller: allows the user to control the condition to run test elements below it (its children).
* Listener: Summary Report, View Result In Table, View Result Tree
  + Summary Report: is a table row for each differently named request in test plan. This report includes: number of samples; Average response time; Min and Max response time; Standard Deviation of the sample elapsed time; Percent of requests with errors; the Throughput which is measured in requests per second; Kb/sec - The throughput measured in Kilobytes per second; Avg. Bytes - average size of the sample response in bytes.
  + View Result In Table shows a tree of all sample responses, allowing you to view the response for each sample. It shows the detail of sampler such as Request URL, request Header, response code and Response page.

(\*View Results Tree MUST NOT BE USED during load test as it consumes a lot of resources (memory and CPU). Use it only during Test Plan debugging and Validation.)

* How To Read Your Test Result
  + **Label** - The label of the sample. If "Include group name in label?" is selected, then the name of the thread group is added as a prefix. This allows identical labels from different thread groups to be collated separately if required.
  + **# Samples** - The number of samples with the same label
  + **Average** - The average elapsed time of a set of results
  + **Min** - The lowest elapsed time for the samples with the same label
  + **Max** - The longest elapsed time for the samples with the same label
  + **Std. Dev**. - the Standard Deviation of the sample elapsed time
  + **Error** % - Percent of requests with errors
  + **Throughput** - the Throughput is measured in requests per second/minute/hour. The time unit is chosen so that the displayed rate is at least 1.0. When the throughput is saved to a CSV file, it is expressed in requests/second, i.e. 30.0 requests/minute is saved as 0.5.
  + **Kb/sec** - The throughput measured in Kilobytes per second
  + **Avg. Bytes** - average size of the sample response in bytes
  + **Latency** is a difference between time when request was sent and time when response has started to be received.
  + **Response time** (= Sample time = Load time = Elapsed time) is a difference between time when request was sent and time when response has been fully received.

[](http://jmeter.apache.org/images/screenshots/summary_report.png)





* Running Your First Recording Test Plan
  + Recording and running a Performance Test for https://www.clinique.ca website: