**JMETER**

To launch JMeter, launch "C:\Users\VJ\Documents\apache-jmeter-5.3\bin\ApacheJMeter.jar"

**Components of JMeter**

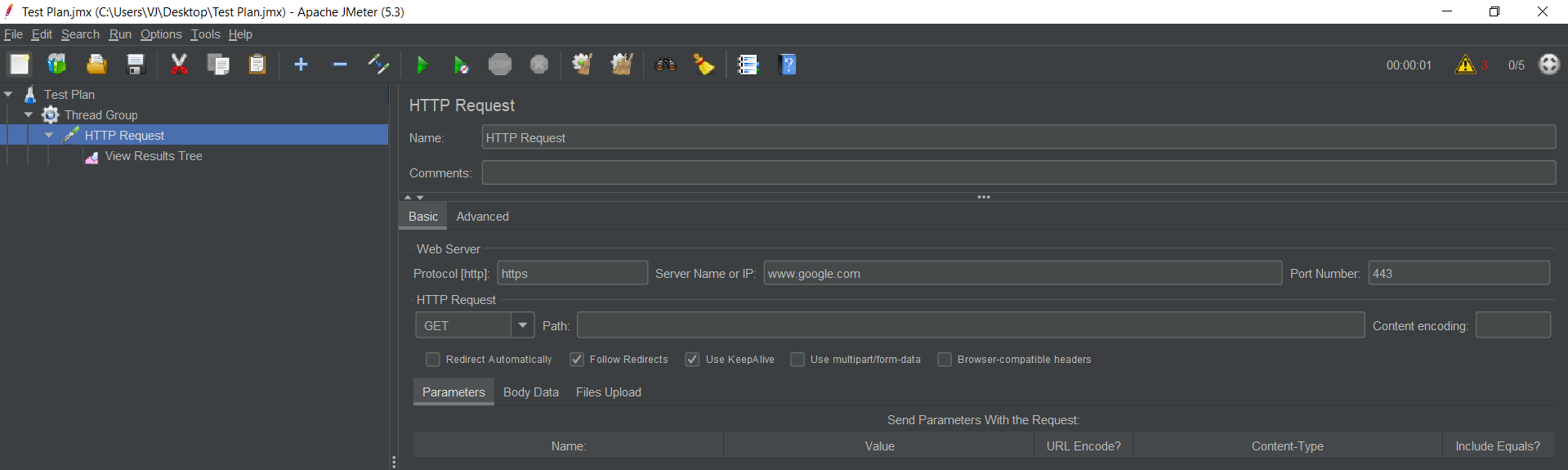
In Test Plan, we add Thread Group.

Thread Group – Here we define,

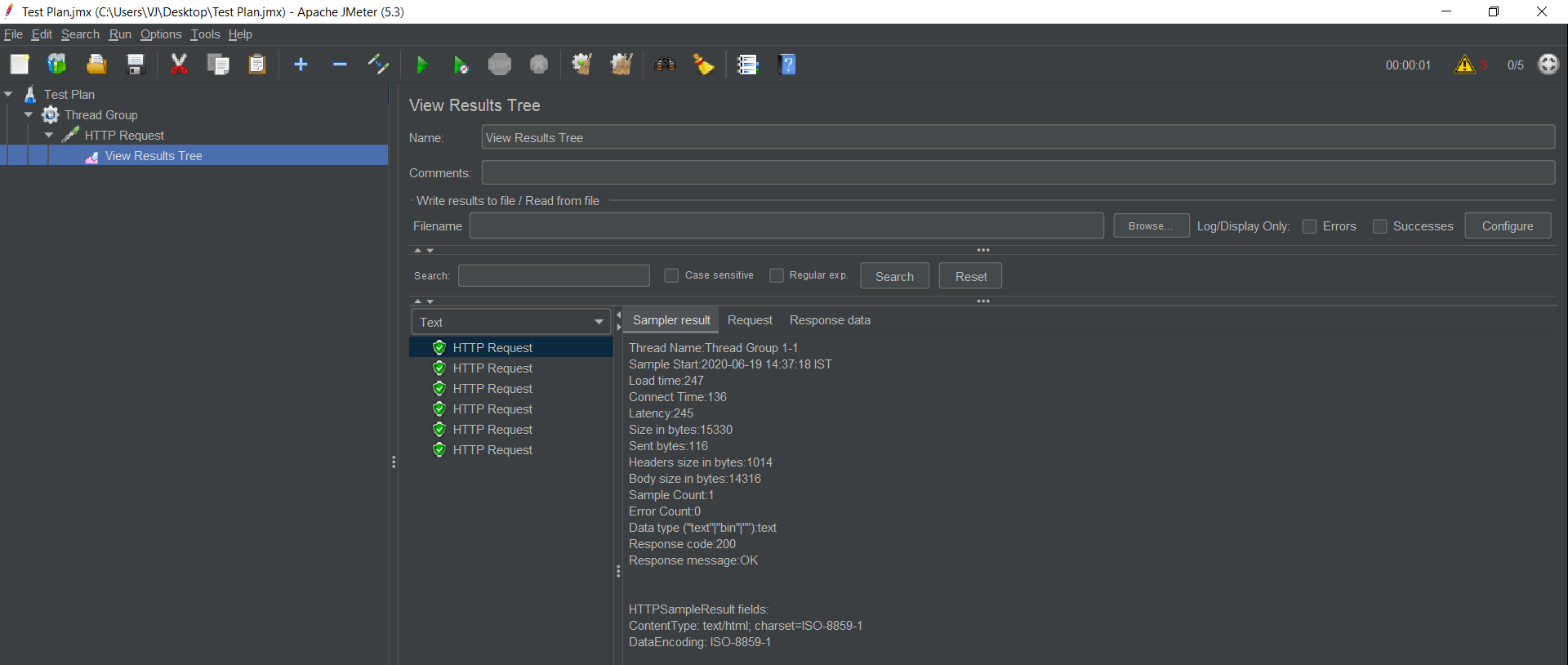
* + Number of Threads – Number of virtual users
  + Ramp up period – Total time allowed for complete execution of all threads.
  + Loop Count – How much time you want to execute the scenario

Note -: The ramp-up period tells JMeter how long to take to "ramp-up" to the full number of threads chosen. If 10 threads are used, and the ramp-up period is 100 seconds, then JMeter will take 100 seconds to get all 10 threads up and running. Each thread will start 10 (100/10) seconds after the previous thread was begun.

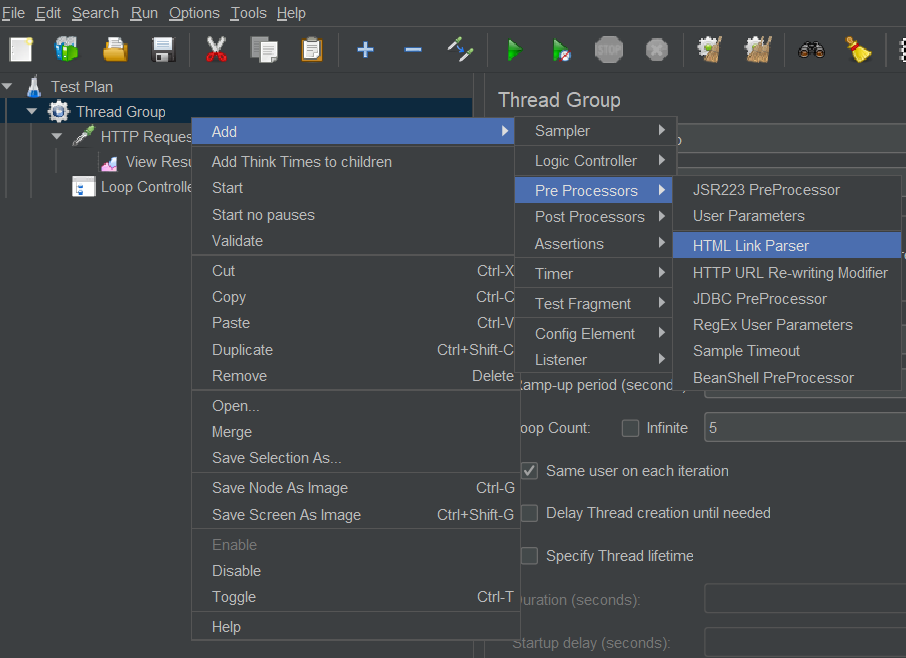
Sampler (HTTP Request) – It should be added under Thread Group



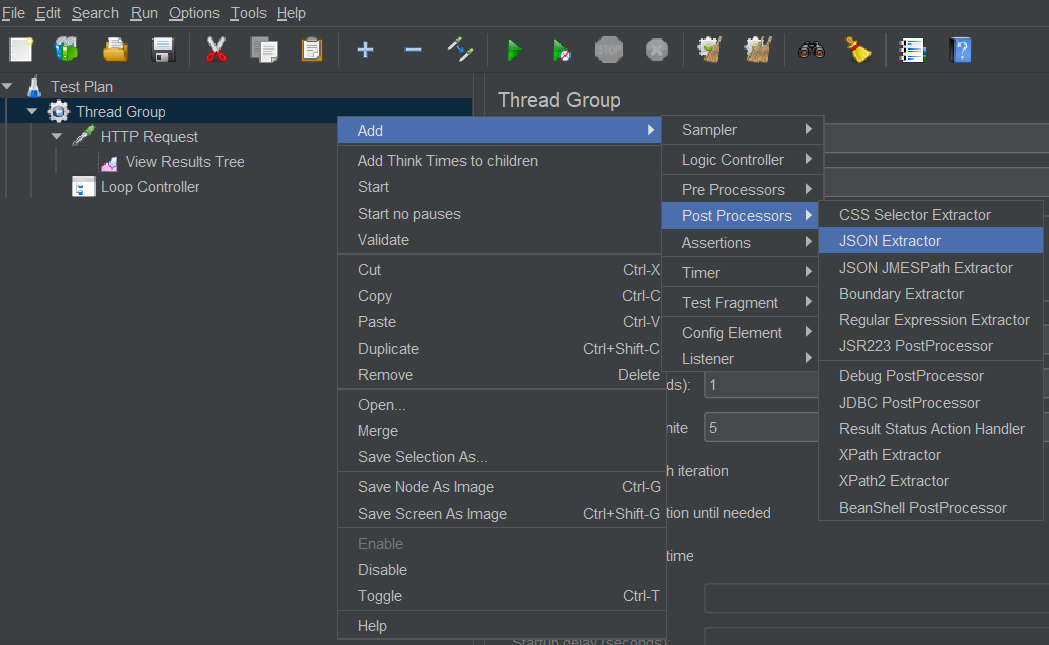
Listener (View Result Tree) – It is used to view the results. All the time shown here are always in milli seconds. In listener, we can also choose ‘view results in table’, ‘graph results’.



Preprocessors 🡪

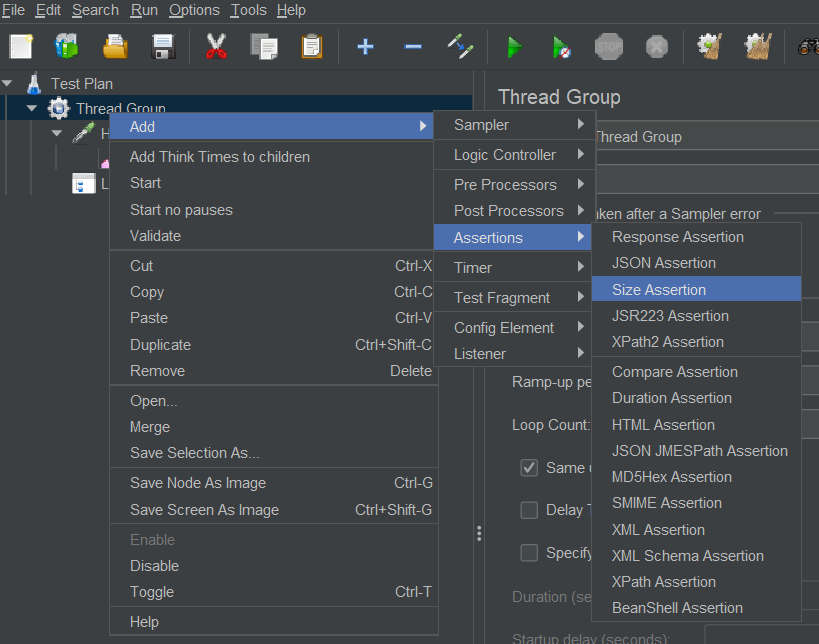


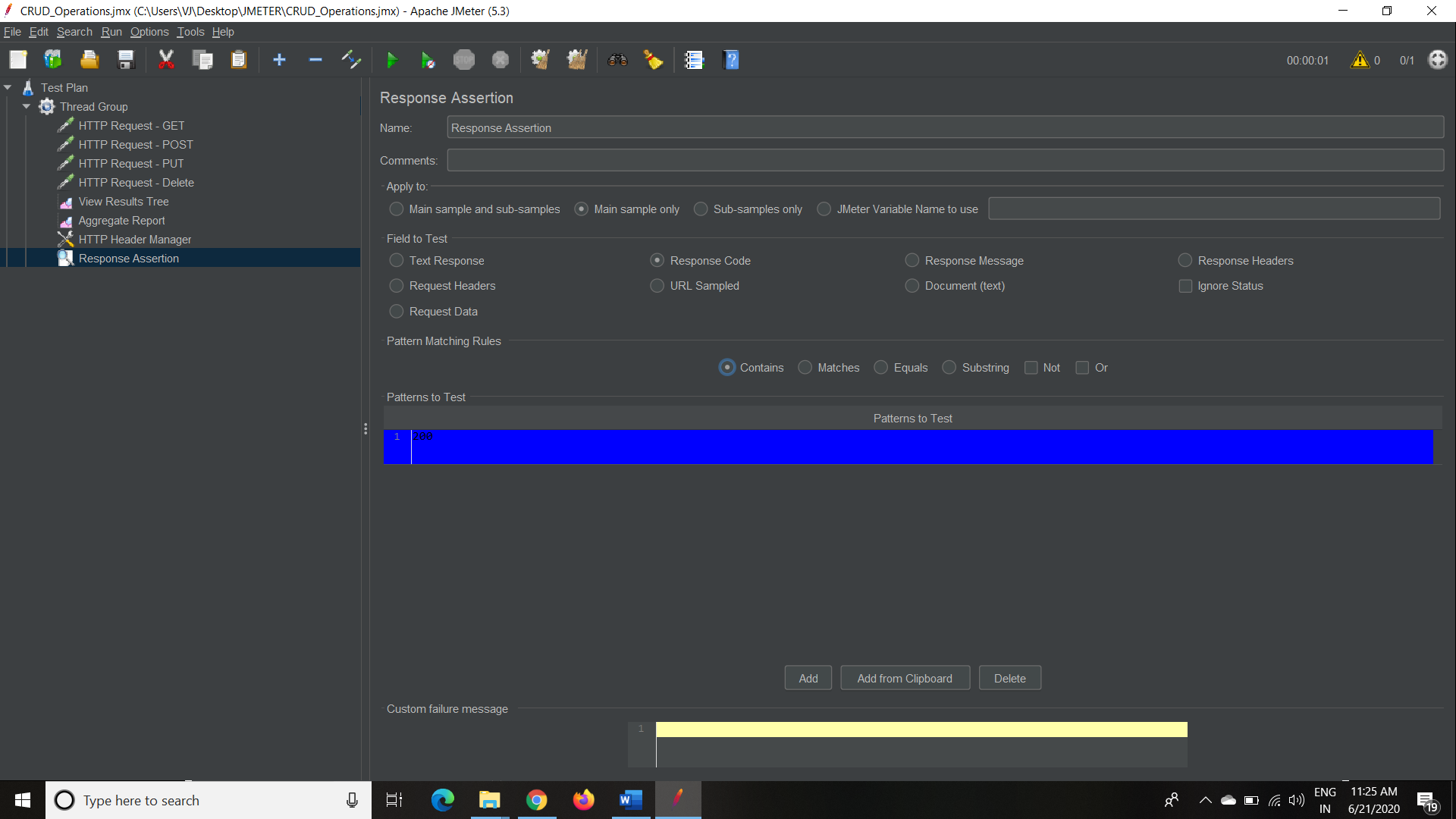
Post processors 🡪



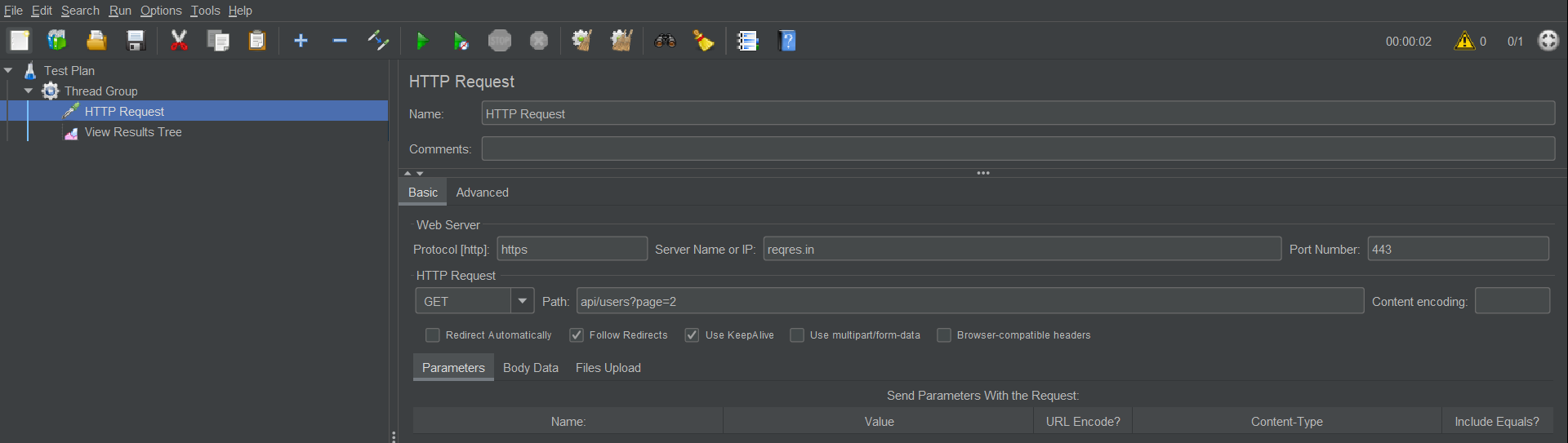
Assertions 🡪

Response Assertion – It is used to assert the response of request.

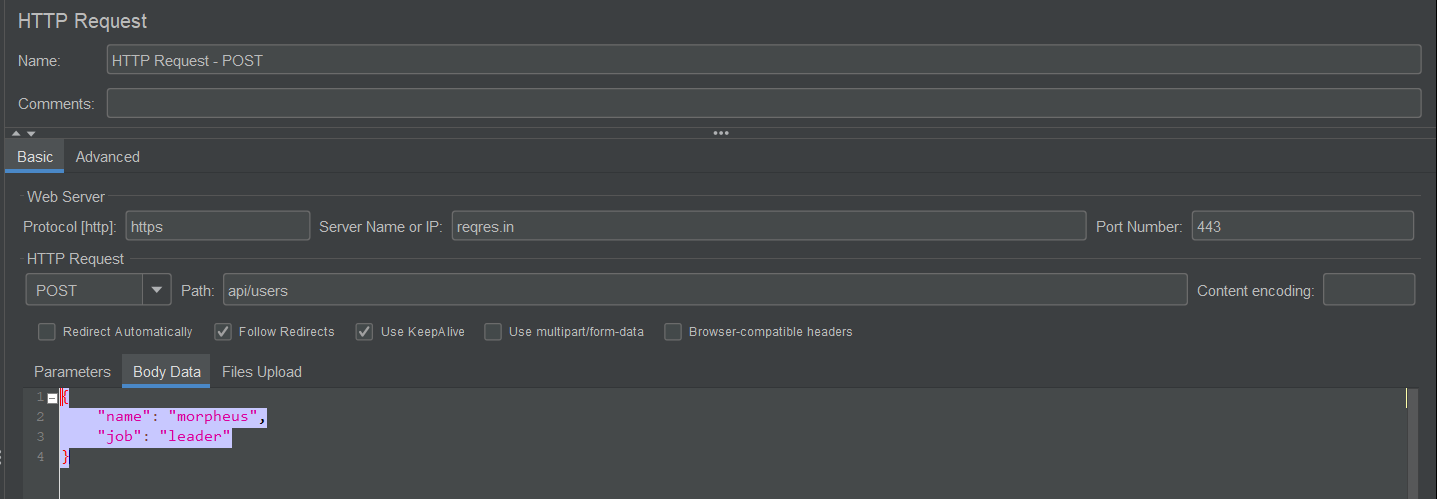




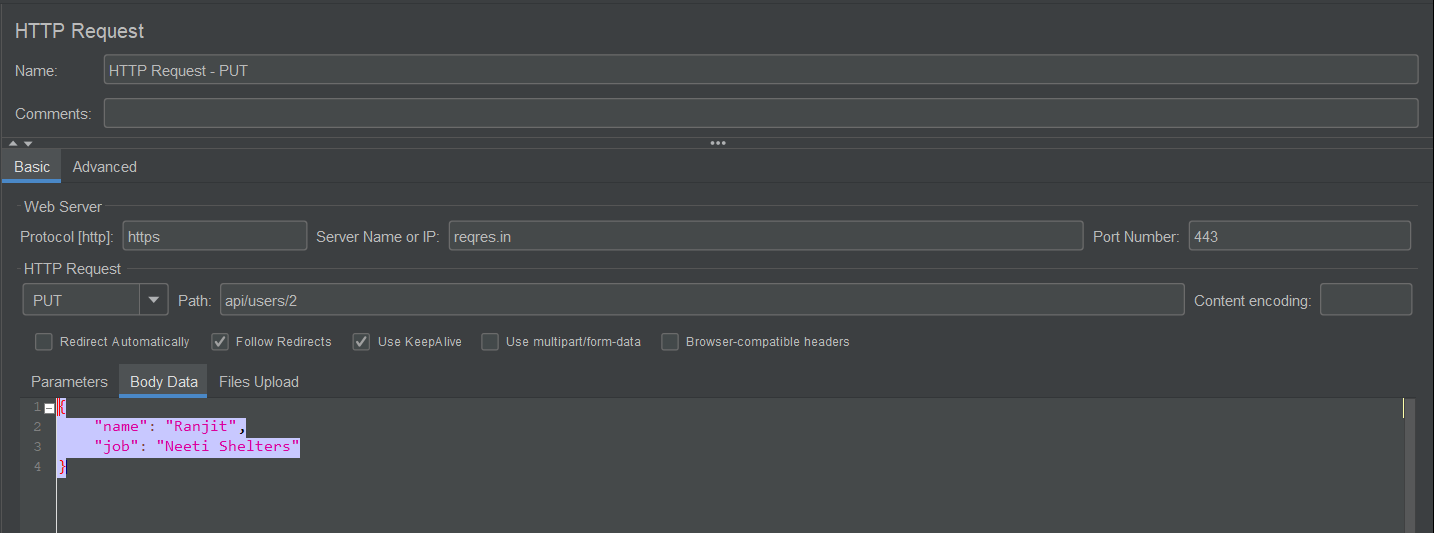
GET Request Example



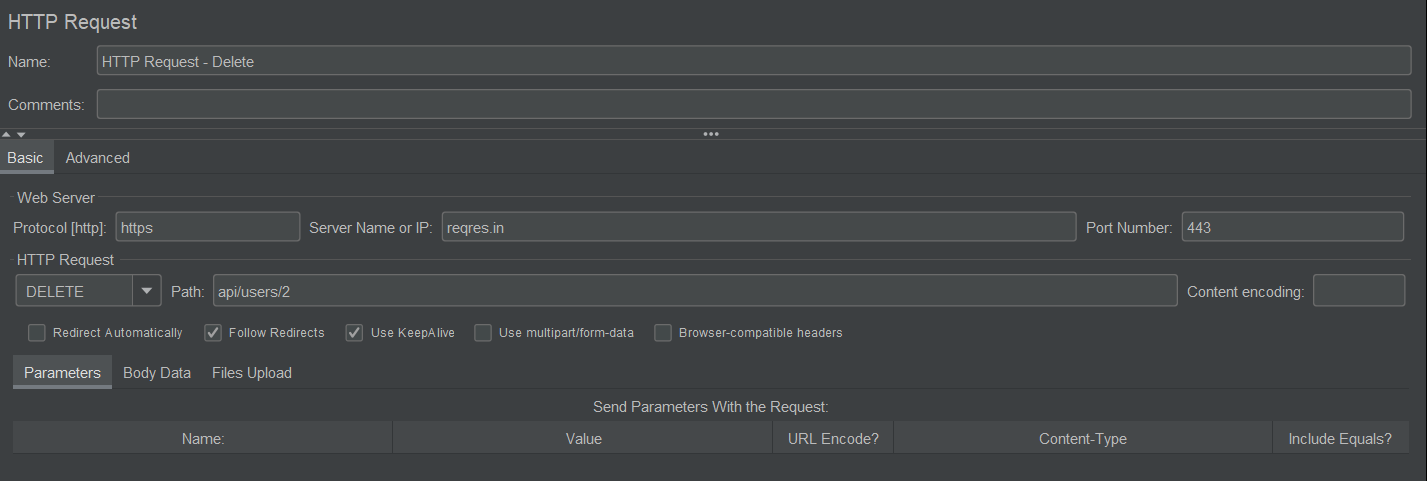
POST Example



PUT Example



DELETE Request –



Project is at - "C:\Users\VJ\Desktop\JMETER\CRUD\_Operations.jmx"

* Port number for HTTPs is 443 and for HTTP is 80.

**Terminologies 🡪**

Performance Testing – It is defined as a type of software testing that is conducted to ensure software applications that will perform well under their expected workload.

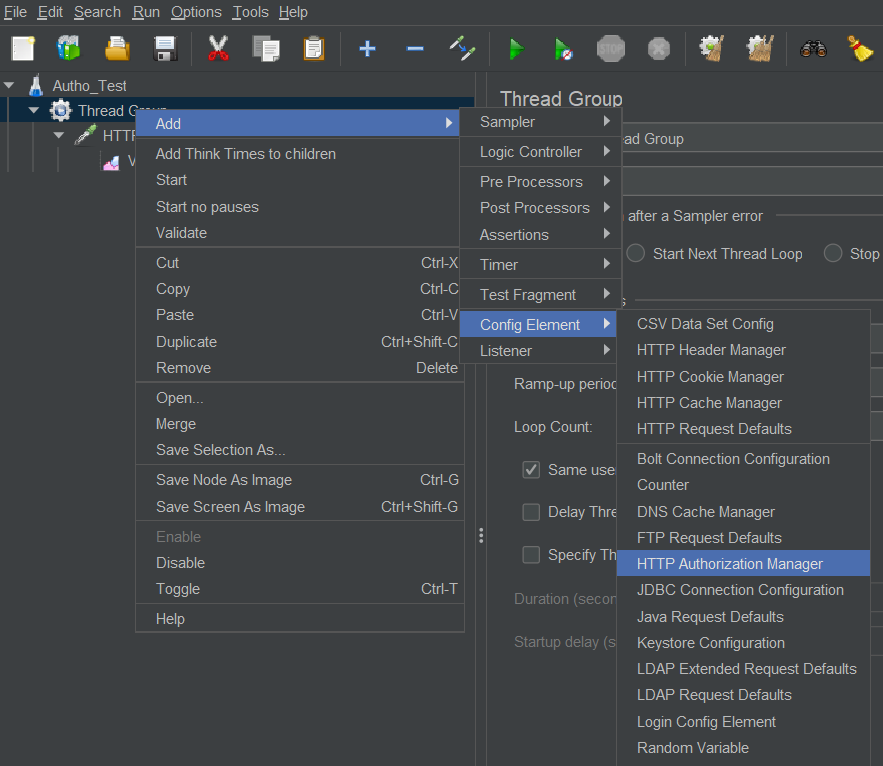
Load Testing - A type of Performance Testing used to evaluate the behavior of a system or component when the load on the system (via users and transaction) progressively increases up to and including peak levels.

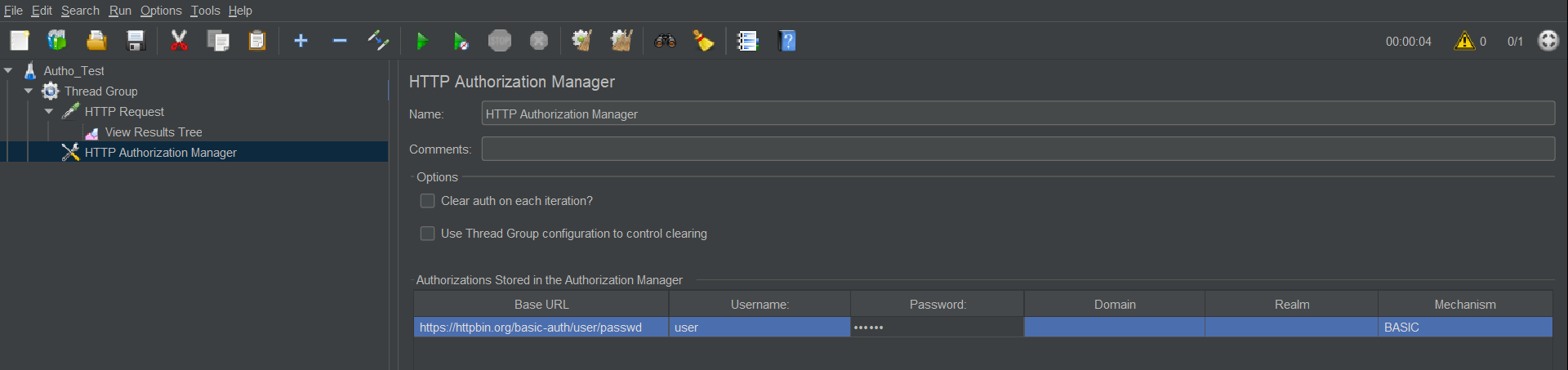
Stress Testing - A type of Performance Testing used to evaluate the behavior of a system or component when subjected to load beyond the anticipated workload or by reducing the resources the system can use, such as CPU or memory. It is also known as Endurance Testing. It is used to find a point where software / hardware breaks. Here also we check whether systems displays effective error message when the system is under stress.

**Authorization in JMeter 🡪**

To add Authorization, right click on Thread Group -> Add -> Config Element -> HTTP Authorization Manager.

In Base URL, provide the complete URL

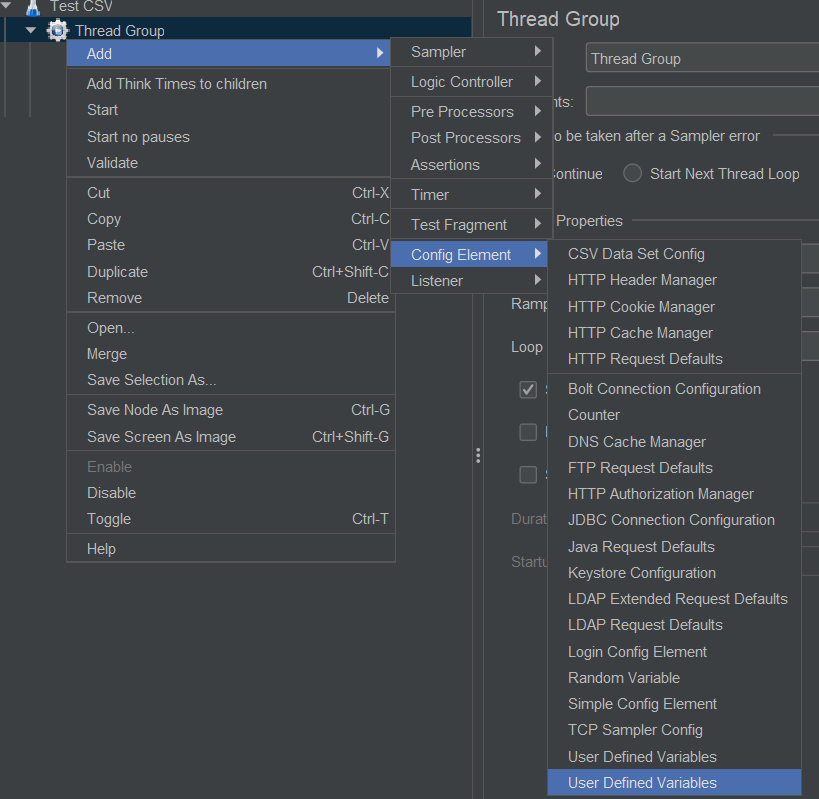




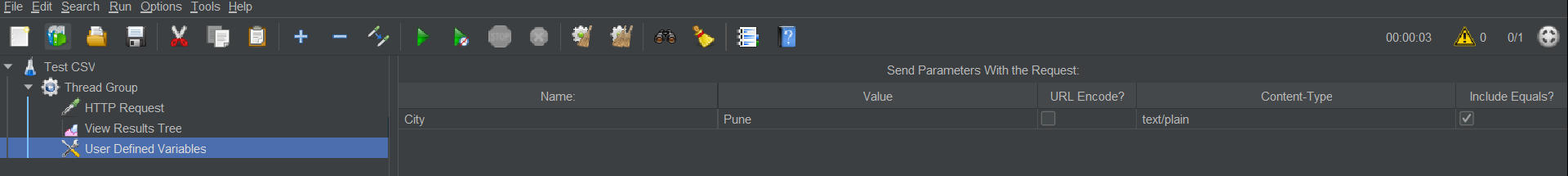
**Parameterization in JMeter 🡪**

To create any variable, right click on Test Plan / Thread Group -> Add -> Config Element -> User Defined Variables.

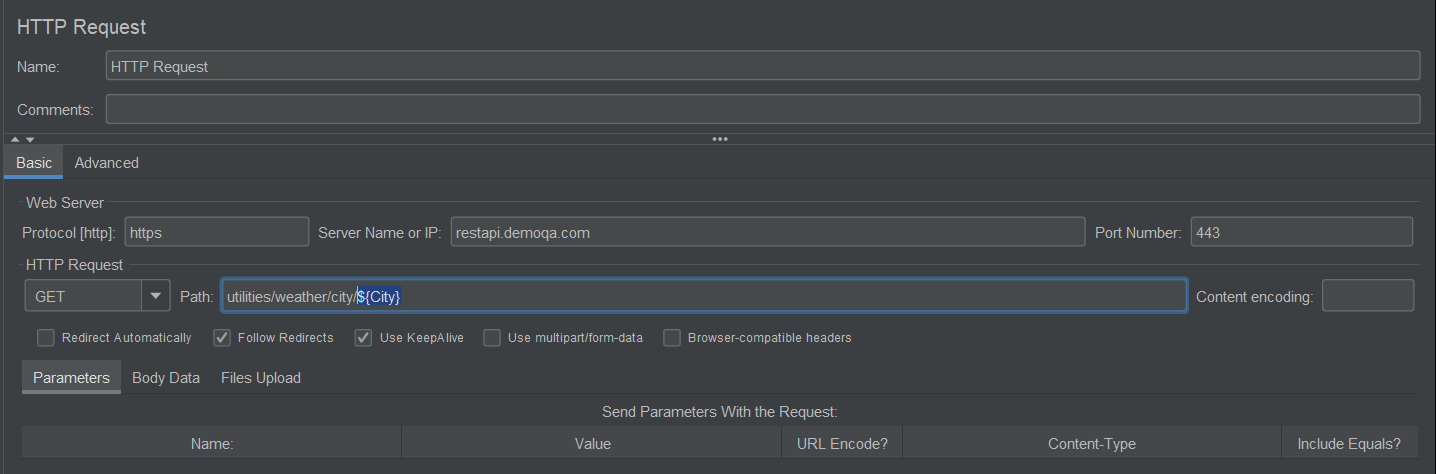
Note :- If variable is created under Test Plan, then it will be applicable to all the Thread Groups under it. But if variable is created under Thread Group then it is applicable to that specific Thread Group and will not be accessible to other Thread Groups of same Test Plan.



In this section, create a parameter (say City and provide value Pune).

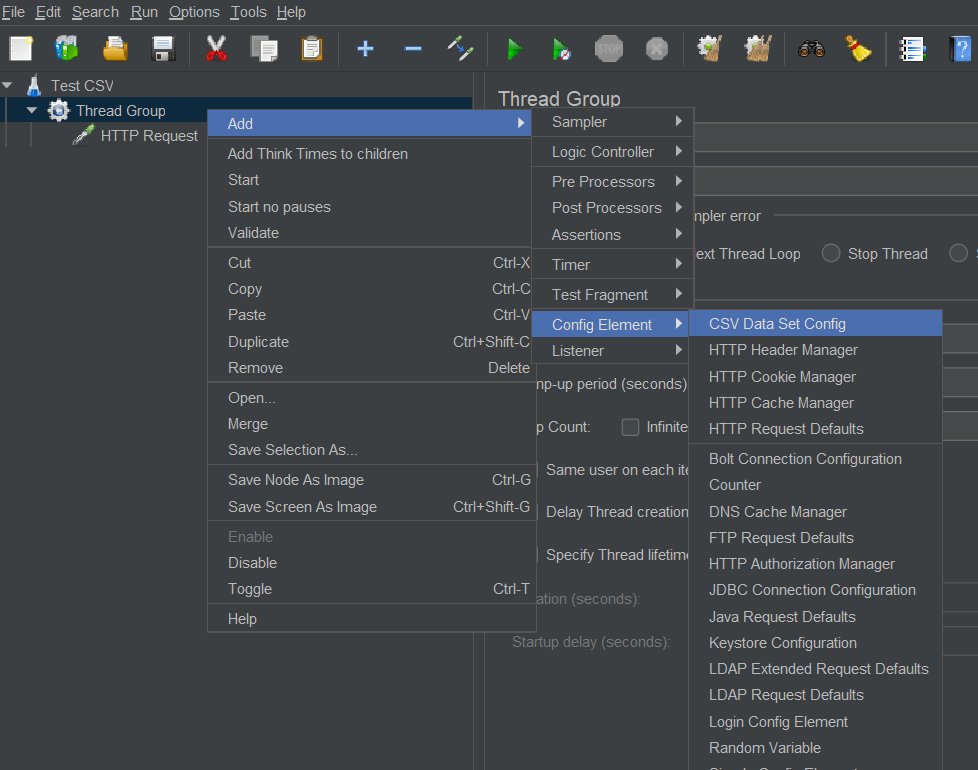


Now provide this parameter in HTTP request. Now move to HTTP request and change Pune with ${City}

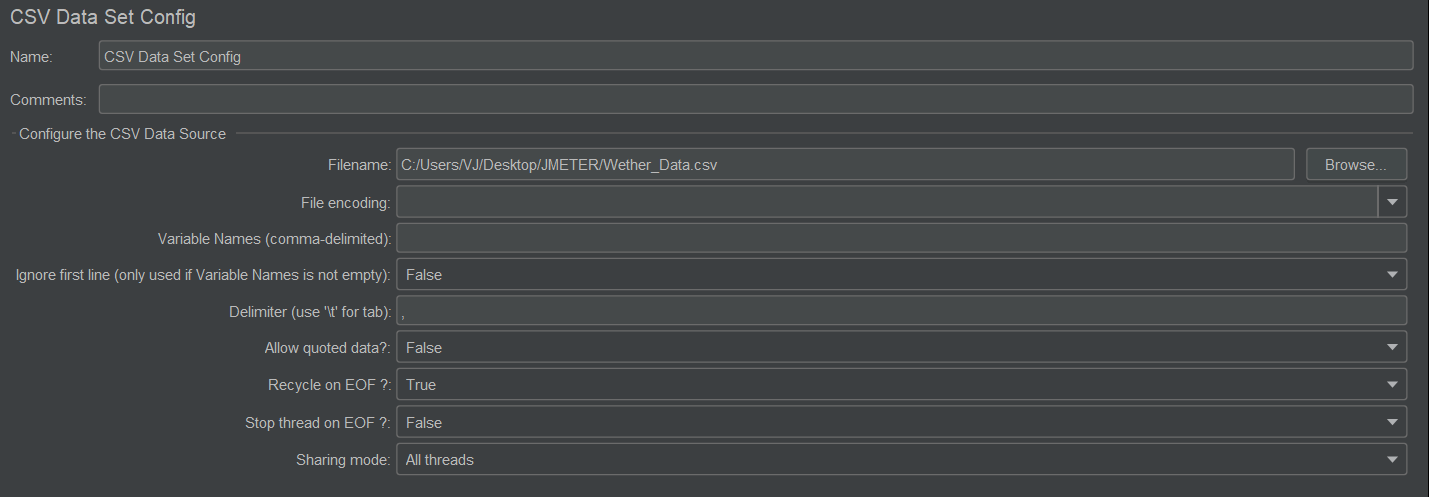


**Reading data from CSV**

To add CSV Config data element, right click on Thread Group -> Add -> Config Element -> CSV Data Set Config



Provide the file name

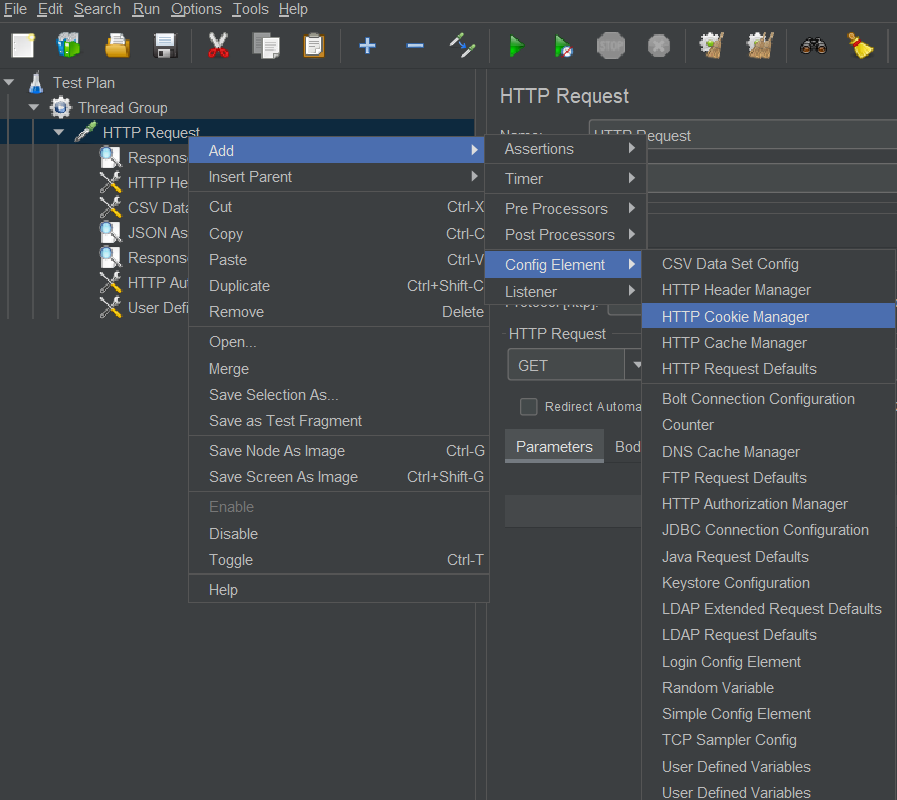


Modify request same as above to fetch the data.

Note -: If you have 5 set of data in CSV file then you must provide at least 5 ‘Number of Threads’ in ‘Thread Group’. If you provide more than 5 number of threads then some data sets will be executed again.

**Cookie Manager**

To add cookie manager, right click on Thread Group -> Add -> Config Element -> HTTP Cookie Manager



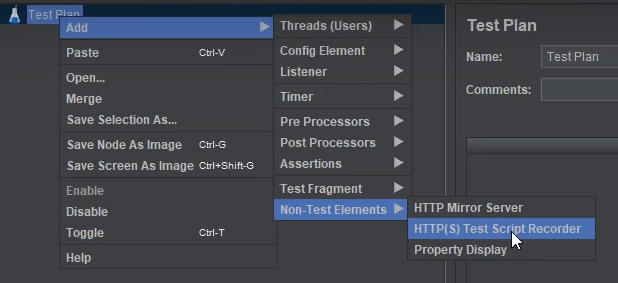
* By using cookie manager, we can bypass the authentication.

**Use of Test Script Recorder 🡪**

The purpose of test script recorder is to record a scenario (say login, search an item, add to cart and logout) and capture all the APIs using Proxy. So that next time you can just run this test script instead of manually running the scenario again.

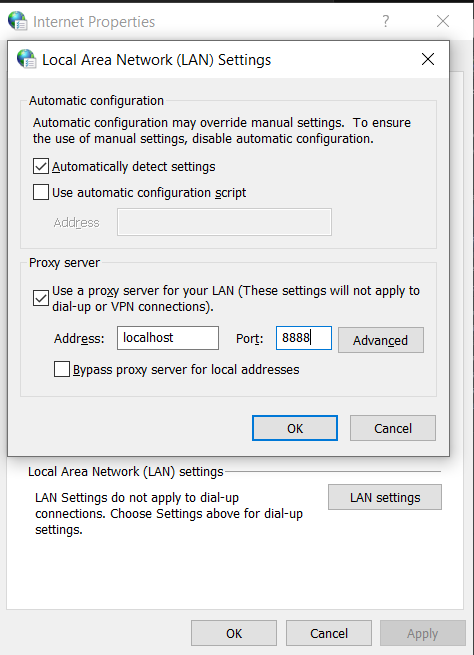
replay it by varying the number of threads, loop count etc as a part of Load Test.

* To create a new recorder,

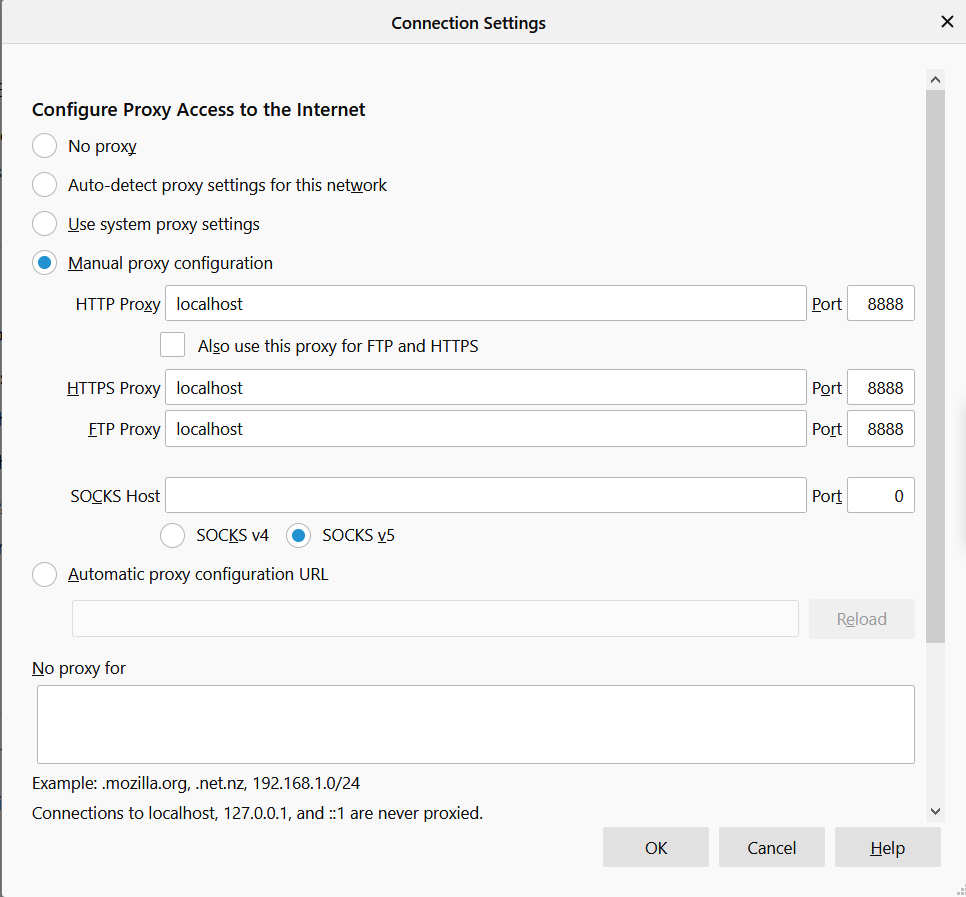


Configuring Internet Options,

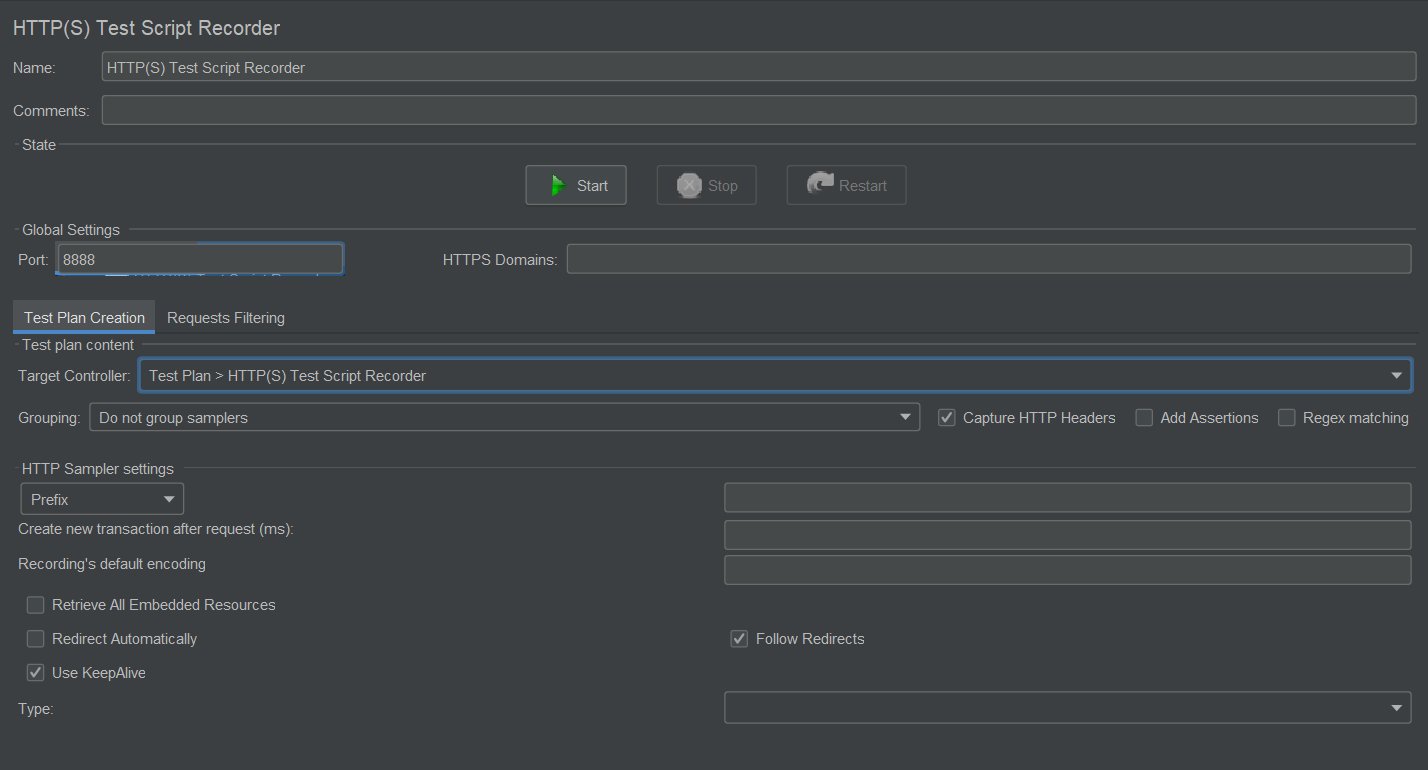
* Go to Internet Options on your system, Connections -> LAN Setting -> select proxy server -> Address as ‘localhost’ and Port as ‘8888’.



* This is global setting, we can configure proxy for only FireFox browser instead of above option which affects to complete system.
* From FF hamburger menu, select Options -> Network Setting (at the bottom) ->



* Now the start proxy from JMeter and go to <http://classic.crmpro.com/>
* At this point we can access HTTP but not HTTPs (as certificate is not added).
* To ignore some specific type of files being recorded, got to ‘URL Patterns to Exclude’ -> ‘Add Suggested Exclude’ and enter extensions (like jpg, css, gif) separated by pipe |
* In order to record any HTTPs connection, we need to import JMeter certificate to FF browser (Options -> Privacy & Security -> View Certificate) which is created by default by Jmeter at Jmeter Directory -> bin
* In JMeter, change ‘Target Controller’ to ‘Test Plan > HTTPs Test Script Recorder’ and click on Start.



**What you have learnt / done in JMeter ?**

* Configuration for Load Test or Stress Test
* Test Plan
  + Thread Group
    - Sampler (HTTP Request)
      * Config Element (HTTP Header Manager)
      * Config Element (HTTP Cookie Manager)
      * Config Element (CSV Data Set Config)
      * Config Element (HTTP Authorization Manager)
      * Config Element (User Defined Variables) - Parameterization
      * Assertion (JSON Assertion)
      * Assertion (Response Assertion)
      * Listener (View Result Tree)
  + Test Script Recorder