

*What is JMeter? Why it is used?

- The Apache JMeter™ is pure Java open source software, which was first developed by Stefano Mazzocchi of the Apache Software Foundation, designed to load test functional behavior and measure performance.
- You can use JMeter to analyze and measure the performance of web application or a variety of services. Performance Testing means testing a web application against heavy load, multiple and concurrent user traffic.
- JMeter originally is used for testing Web Application or FTP application. Nowadays, it is used for a functional test, database server test etc.

* Advantages

- 1) Open source
- 2) Friendly GUI
- 3) Platform independent(pure 100% java application)
- 4) Full multithreading framework
- 5) Visualize Test Result
- 6) Easy installation
- 7) Support multi-protocols
- 8) Record & Playback – Record
- 9) Script Test(automated testing)

* steps to install Jmeter

- 1- install java jdk
- 2- download apache jmeter
- 3- extract all
- 4- go to bin folder and open jmeter

*Jmeter Elements-----

1) Thread Group

- Thread Groups is a collection of Threads.
- Each thread represents one user using the application under test. Basically, each Thread simulates one real user request to the server.

i) Number of Threads(users)

- how much user we want to create

ii) Ramp up period(seconds)

- How much time no of users would be taken for rampup

iii) Loop Count

- How many time we want to run this test

2) Sampler

- As we know already that JMeter supports testing HTTP, FTP, JDBC and many other protocols.
- We already know that Thread Groups simulate user request to the server
- The user request could be FTP Request, HTTP Request, JDBC Request...Etc.

i) HTTP Request

- *Protocol(http)- https *server name or IP- www.xyz.com port Number- 443(default for https) 80(http)

3) Listeners

- shows the results of the test execution.
 - They can show results in a different format such as a tree, table, graph or log file.
-

* How to record HTTP & HTTPS scripts in jmeter?

- open jmeter
 - right click on testplan
 - go to add>>Non-test element>>Http(s) script record
 - configure Target controller= Test plan Http(s) script record
 - go to firefox>>connection setting>>>manual proxy>>localhost(8888)
 - add certificate for handle secured sites
 - setting>>privacy>>certificate>>authorities>>import>>bin>>open>>check both boxes>>ok
 - click on start in jmeter
 - open FF browser and open required site
-

* Sampler HTTP Request(GET/POST/PUT/DELETE)

- 1) Right click on Test plan-->Add-->Threads(users)-->Thread group
 - 2) Right click on thread group-->Add-->sampler-->HTTP Request
 - 3) ADD protocol=https, server name= reqres.in request= get Path= api/users?page=2
 - 4) add listeners for report e.g- view result tree, and save and click on run
-

* for post request we need(upload new data or create new)

- Body data
- hTTP header manager
 - Add-->config element-->http header
- add-->Name=content-type value=application/json

* for PUT request(update existing data)

- Body data
-

* User defined variables-

- we can store url in some variable and use it in every request
 - rightclick on test plan-->Add-->Config-->user defined variable
 - server name= \${variable name}
-

*Logic controller-

- simple controller
- loop controller (how many times we want to repete this request loop)
- Random controller(taking any one random request)
- random order controller
- Throughput controller(build a distributed load test)-
 - 1) how to make our samplers run with defined percentage xecution (distribution the load)
 - 2) how to use throughput controller to achieve this

-->Threads(virtual users)=10
-->Web Application
home page-20% - 2 users
welcome page-30% - 3 users
Register page-50% -5 users

*What are Timers?

- Jmeter sends request without applying any delay between each sampler/request.
- If you perform load/stress testing on your server without any delay, it will be overloaded. Then it won't be able to give you realistic results and fail to simulate real world user traffic experience.
- Jmeter Timers are the solution to all these problems.
- Timer element can be added in a test plan to apply wait between each sampler/request.

1) Constant Timer

2) uniform random timer

i) Random delay max

ii) constant delay offset

- formula= $0.x * \text{random delay max} + \text{constant delay offset}$

x=0-9

e.g= $0.1 * 100+0=0-99$ mili sec
