**JMeter Timers**

Timers allow JMeter to delay between each request which a thread makes. A timer can solve the server overload problem.

Also, in real life visitors do not arrive at a website all at the same time, but at different time intervals. So Timer will help mimic the real-time behavior.

Purpose :

* To pause thread (users) for some time
* To add delay between threads
* To avoid over flooding the served and achieve real time behavior by pacing the load.

Type of Timer :

* Constant Timer
* Gaussian Random Timer
* Uniform Random Timer
* BeanShell Timer
* BSF Timer
* JSR223 Timer
* How to Use Constant Timer

Steps :

* Create new Test plan
* Add new Test group
* Add Samle Controller

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

* Add HTTP Request

A screenshot of a computer

Description automatically generated

This webpage is used:

Graphical user interface, text, application, website

Description automatically generated

A screenshot of a computer

Description automatically generated

* Add a Listener to display Result

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

* Run the Test : It will run with time 1 second without any delay

A screenshot of a computer

Description automatically generated

* Add the Timer – Constant Timer .

Constant timer delays each user request for the same amount of time.

A screenshot of a computer

Description automatically generated

Delay time : Set a time of delay in millisecond.

Here set 5000 milli second as delay

A screenshot of a computer

Description automatically generated

* Run the Test : After 5000millisecond it will execute

A screenshot of a computer

Description automatically generated

* Add some more HTTP Request and Constant Timer

Constant Timer A (delay :3000)is only for HTTP Request 1

Constant Timer B(delay :3000) is inside Test group so it is applicaple for HTTP Request 1&2

Constant Timer C (delay :3000)is inside Test plan so it is applicable for HTTP Request1,2&3

A picture containing text, screenshot, monitor, computer

Description automatically generated

* Run the Test

Request 1 is executed after delay of 9000

Request 2 is executed after delay of 6000

Request 3 is executed after delay of 3000

A screenshot of a computer

Description automatically generated

* Disable all timer
* Add a new Timer -Uniform Random Timer

Uniform random timer delays each user request for a random amount of time.

A picture containing text, screenshot, monitor, computer

Description automatically generated

Random Delay Maximum: Maximum random number of milliseconds to delay.

Constant Delay Offset (milliseconds): Additional value in milliseconds

A picture containing text, screenshot, monitor, computer

Description automatically generated

The total delay is the sum of the random value and the offset value.

A screenshot of a computer

Description automatically generated