Jmeter Praticals

step1) launch the JMeter IDE

> This PC > Local Disk (D:) > JmeterSoft apache-jmeter-5.6.2 > bin

Name

examples

Date modified

09-10-2023 18:57

09-10-2023 18:57

report-template

templates

ApacheJMeter

09-10-2023 18:57

right click on the jar file 322-> open with JAVA.

step2) create the Testplan (Performence Testing Project) and save it

File ---> new ---> name :: PTestProj1

step3) Add users (thread group) the Project

--->save---> ---

Right click on the Project ----> add ---> threads(users) ----> Thread Group

Thread Group

Thread Group

Name:

ThreadGroup1

Comments:

Action to be taken after a Sampler error

Continue Start Next Thread Loop Stop Thread Stop Test Stop Test Now

Thread Properties

Number of Threads (users):

10

users count

Ramp-up period (seconds): 2

The gap b/w two users requests

Loop Count:

✔Infinite

(make the users repeating the reuests)

step3) add sampler object (HttpRequest)

Right click on Threadgroup1---> add ---> sampler---> HttpRequest

APTestPlanProj01

ThreadGroup1

HTTP Request1

HTTP Request

Name:

HTTP Request1 (name)

Comments:

Basic Advanced

Server Name or IP: www.nareshit.com (Base Url)

-Web Server

optional for http/https

Protocol [http]:

HTTP Request

GET

Path: /new-batches

Redirect Automatically✔ Follow Redirects

Parameters Body Data Files Upload

(request path)

Use KeepAlive Use multipart/form-data Browser-compatible headers

Name:

Save the Httprequest1 (Right click on HttpRequest1) ---> save as --->.....

step5) Add the Listeners to ThreadGroup1

Add

Sampler

HTTP

Add Think Times to children

Logic Controller ▸

Start

Pre Processors

Start no pauses

Post Processors

›ampler error

Validate

Assertions

Next Thread Loop Stop Thread

Cut

Ctrl-X

Timer

Copy

Ctrl-C

Test Fragment

Paste

Ctrl-V

Config Element

►

10

Duplicate

Remove

Open...

Merge

Save Selection As...

Copy Code

Save Node As Image

Save Screen As Image

Enable

Disable

Toggle

Help

Ctrl+Shift-C

Listener

Delete amp-up period (second

Ctrl-G

op Count:

✔Infini

✔Same user on each it Delay Thread creatio

Ctrl+Shift-G Specify Thread lifetir

uration (seconds):

Summary Report

Aggregate Report Backend Listener

Aggregate Graph

Assertion Results

Comparison Assertion Visualizer Generate Summary Results

Graph Results

JSR223 Listener

Ctrl-T

artup delay (seconds):

Mailer Visualizer

-

View Results Tree

save the Listener (right on listener --->save as ---> .....

step6) Run the Project (Start the Performence Test)

Response Time Graph

Save Responses to a file

Simple Data Writer

View Results in Table

(menu bar ----> run button /play button)

step7) Open the Listener and observe the report

Thread Name

Start Time Label 09:40:30.701 ThreadGroup1... HTTP Request1 09:40:30.900 ThreadGroup1... HTTP Request1 09:40:30.104 ThreadGroup1... HTTP Request1

Sample Time(ms)

Status

3551

3370

4206

09:40:30.308 ThreadGroup1... HTTP Request1

4048

09:40:30.510 ThreadGroup1... HTTP Request1

3925

09:40:30.096 ThreadGroup1... HTTP Request1 09:40:31.303 ThreadGroup1... HTTP Request1

4989

3916

09:40:31.114 ThreadGroup1... HTTP Request1

4120

09:40:31.510 ThreadGroup1... HTTP Request1

3818

09:40:31.708 ThreadGroup1... HTTP Request1

4122

09:40:34.253 ThreadGroup1... HTTP Request1

3168

09:40:34.357 ThreadGroup1... HTTP Request1

3206

09:40:34.312 ThreadGroup1... HTTP Request1

3370

09:40:34.436 ThreadGroup1... HTTP Request1

3372

09:40:34.270 ThreadGroup1... HTTP Request1

3654

09:40:35.085 ThreadGroup1. HTTP Request1

3234

Ramp-up and startup spike

Type message here...

You might used "ramp-up" or similar approaches to avoid a spike at the test start. F configure Thread Group to have 100 threads, and set Ramp-up Period to e (or to To

...

then all the threads would start at the same time, and it would produce an unwanted spike or the way. On top of that, if you set Ramp-up Period too high, it might result in "too few" threads being available

at the very beginning to achieve the required load.

Precise Throughput Timer schedules executions in a random way, so it can be used to generate constant load, and it is recommended to set both Ramp-up Period and Delay to e.

step8) Add different types of Listeners to see the report in different formats

try with following Listeners

View Results in Table

View Results Tree

Summary Report

Aggregate Graph

Aggregate Report

Timeers in JMeter

=>The Timers in Jmeters are useful to provide gap between requests (indirectly scheduling the requests)

The avaiable timers are

Constant Timer

Uniform Random Timer

Precise Throughput Timer Constant Throughput Timer Gaussian Random Timer

JSR223 Timer

Poisson Random Timer

Synchronizing Timer

BeanShell Timer

note:; we can apply timers at request level and also at threadgroup level

The timer apllied at thread group level will be applicable for all the requests of that threadgroup

step1) create Project

step2) add thread group having users count as 1

step3) add the 3 requests pointing to same url

Send Parameters With the Request:

Value

URL Encode?

step4) add Constant timer at thread group level having timer value :; 5000 Ms

right click thread group ---> timer ---> constant timer--->

To work with uniform Random Timer Uniform Random Timer

Constant Timer

Name:

Uniform Random Timer

Name:

Constant Timer1

Comments:

Comments:

Thread Delay (in milliseconds): 5000

step5) add "viewResultsInTable" Listener

step6) Run the application

and see the Lister Report

Filename

Sample #

Start Time

Thread Name

Label

1

10:32:10.663 Thread Group1 1-1

2 10:32:17.290 Thread Group1 1-1

HTTP Request1 HTTP Request2

3

10:32:23.495 Thread Group1 1-1

HTTP Request3

Thread Delay Properties

Random Delay Maximum (in milliseconds): 10000.0

Constant Delay Offset (in milliseconds): 1000

While creating sampler for the java web application url that created locally in our (Testing the Local web

application) Eclipse IDE, we need to use the following details

HTTP Request

Name:

HTTP Request3-1

Comments:

Basic Advanced

Web Server

Protocol [http]: http

HTTP Request

GET (MODE)

Server Name or IP: localhost

(protocol)

(hostname)

Path: /JMeterWebProj01/index.jsp (url)

Redirect Automatically✔ Follow Redirects✔ Use KeepAlive

Parameters Body Data Files Upload

FAQs

======

Name:

Use multipart/form-data

Port Number: 3131

(server port)

Content encoding:

Browser-compatible headers

Send Parameters With the Request:

Value

Q) What is the need of Performance Testing?

Q) Explain difference flavours of the Performance Testing?

a) Load Test b) Spike TEst c) StressTest d) Endurance Test and etc..

Q) List out different Performance Testing tools?

Q) List out various terminologies related 9meter Projectreation

a) TEst plan b) Thread Group c) Sampler d) Listener e) Timer

Q) List out various samplers?

Q) List out various Listeners?

URL Encode?

Content-Type

Include Equals?

Q) List out Various Timers?

Q) How Exactly Performance Test is useful in real Projects?

Q) What is ramp up period while creating Threads?

Q) What are the different options that are there to graph reports in Jmeter

Testing the web application url that need to pass request param values

Create the HttpRequest as shown below

HTTP Request

Name:

HTTP Request1

Comments:

Basic Advanced

Web Server

Protocol [http]:

Server Name or IP: www.google.com

HTTP Request

GET

Path: /search

Redirect Automatically✔ Follow Redirects✔ Use KeepAlive Use multipart/form-data

Parameters Body Data Files Upload

Name:

q

Port Number:

Content encoding:

Browser-compatible headers

Send Parameters With the Request:

Value

URL Encode?

Content-Type

Include Equals?

nataraj+sir

text/plain

Generating Graph Result using the Listener support

==========

=======

===

step1) Add Aggregate Grpah to the Thread Group1 Aggregate Graph

Name:

Aggregate Graph1

Comments:

Write results to file / Read from file

Filename

Browse... Log/Display Only: Errors

Successes

C

Label HTTP Requ... TOTAL

Samples

Average

Median

90% Line 95% Line 99% Line

Min

Maximum

200

662

559

1251

1357

1936

166

200

662

559

1251

1357

1936

166

1975 1975

Error % Throughput Received K... 0.00% 11.3/sec 0.00% 11.3/sec

977.72 977.72

Settings Graph

Display Graph

✔Median

✓ 90% Line

Size: 10

Style: Normal

Column settings

Columns to display:✔ Average

Value font: Sans Serif

✔ Column label selection:

use this option to generate the report

Milliseconds

2,000

1,500

1,000-

500

0

Save Graph

Save Table Data

Save Ta

✓ Max

Foregrou

✔95% Line

✔99% Line ✓ Min ✓ Draw outlines bar?✔ Show number grouping? ✔ Value labels vertical?

Summary report

1,357

Apply filter

Case sensitive ✔