



# ***SESI HANDS ON* PENGUJIAN PRESTASI MENGUNAKAN APACHE JMETER**

# SYLLABUS

1. **CREATING JMETER TEST PLAN**
  - ✓ C1 - Thread Group
  - ✓ C2 - Config Element
  - ✓ C3 - Controller
  - ✓ C4 - Listener
  
2. **RECORDING AND SCRIPTING**
  - ✓ R1 - Script Recording
  - ✓ R2 - JMeter Certificate
  
3. **VERIFYING SCRIPT**
  - ✓ V1 - Check Recorded Transaction
  - ✓ V2 - Run Recorded Transaction
  - ✓ V3 - View Result Tree
    - Regular Expression Extractor
    - Response Assertion
  - ✓ V4 - View Summary Report
  
4. **PERFORMING LOAD TEST**
  - ✓ P1 - CSV File
  - ✓ P2 - Timer (Think Time)
  - ✓ P3 - Smoke Test





01

## 1. CREATING JMETER TEST PLAN

- ✓ C1 - Thread Group
- ✓ C2 - Config Element
- ✓ C3 - Controller
- ✓ C4 - Listener

## 2. RECORDING AND SCRIPTING

- ✓ R1 - Script Recording
- ✓ R2 - JMeter Certificate

## 3. VERIFYING SCRIPT

- ✓ V1 - Record Transaction
- ✓ V2 - Running the Script
- ✓ V3 - View Result Tree
  - Regular Expression Extractor
  - Response Assertion
- ✓ V4 - View Summary Report

## 4. PERFORMING LOAD TEST

- ✓ P1 - CSV File
- ✓ P2 - Timer (Think Time)
- ✓ P3 - Smoke Test

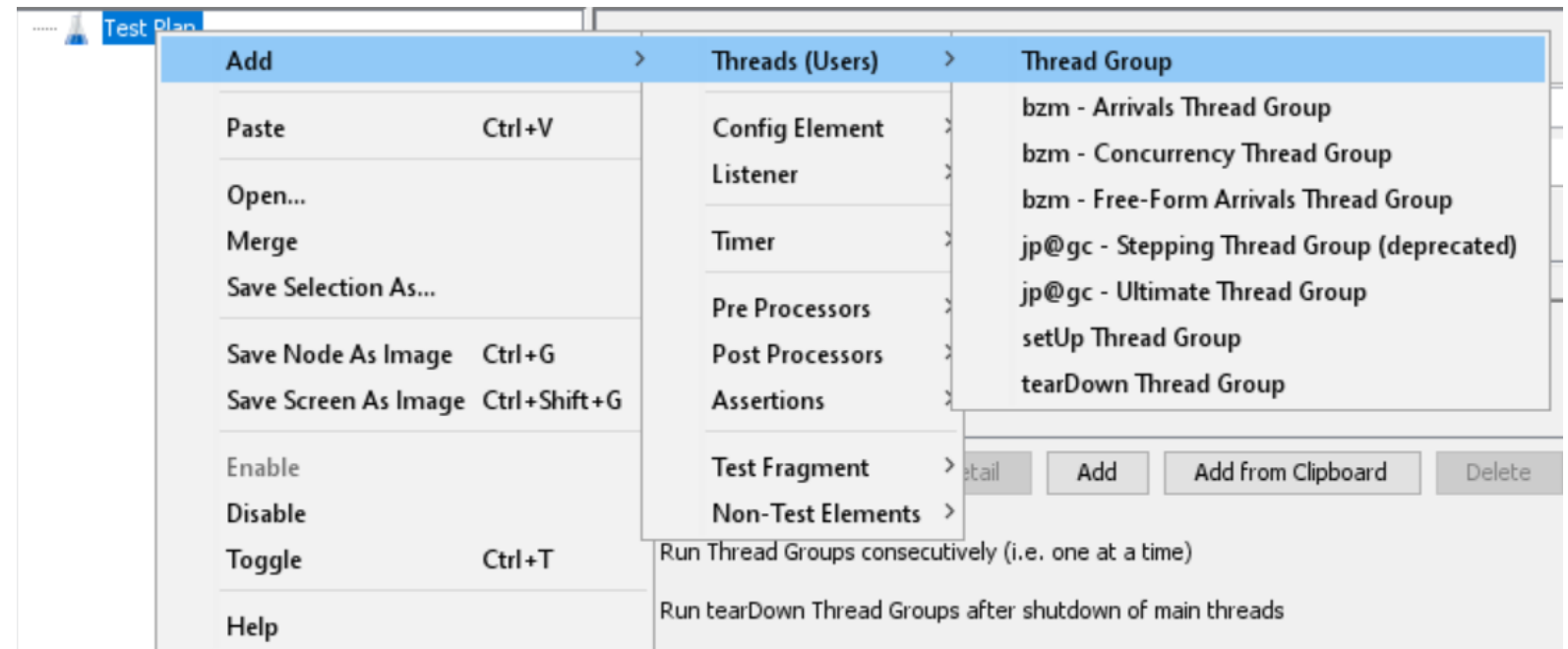
# CREATING TEST PLAN

## 1. Add Thread Group

Thread group element controls the number of threads JMeter will use to execute your test.

The controls for a thread group allow you to:

- ✓ Set the number of threads
- ✓ Set the ramp-up period
- ✓ Set the number of times to execute the test
- ✓ Scheduler



# CREATING TEST PLAN

## 1. Add Thread Group

Untuk *Recording*

Thread Group

Name:

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):

Ramp-Up Period (in seconds):

Loop Count:  Forever

Delay Thread creation until needed

Scheduler

Scheduler Configuration

⚠ If Loop Count is not -1 or Forever, duration will be  $\min(\text{Duration}, \text{Loop Count} * \text{iteration duration})$

Duration (seconds)

Startup delay (seconds)

Parameter that  
controlling number  
of thread  
(concurrent users)

# CREATING TEST PLAN

## 1. Add Thread Group

Untuk Pelaksanaan Ujian

Thread Group

Name:

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):


Ramp-Up Period (in seconds):

Loop Count:  Forever

Delay Thread creation until needed

Scheduler

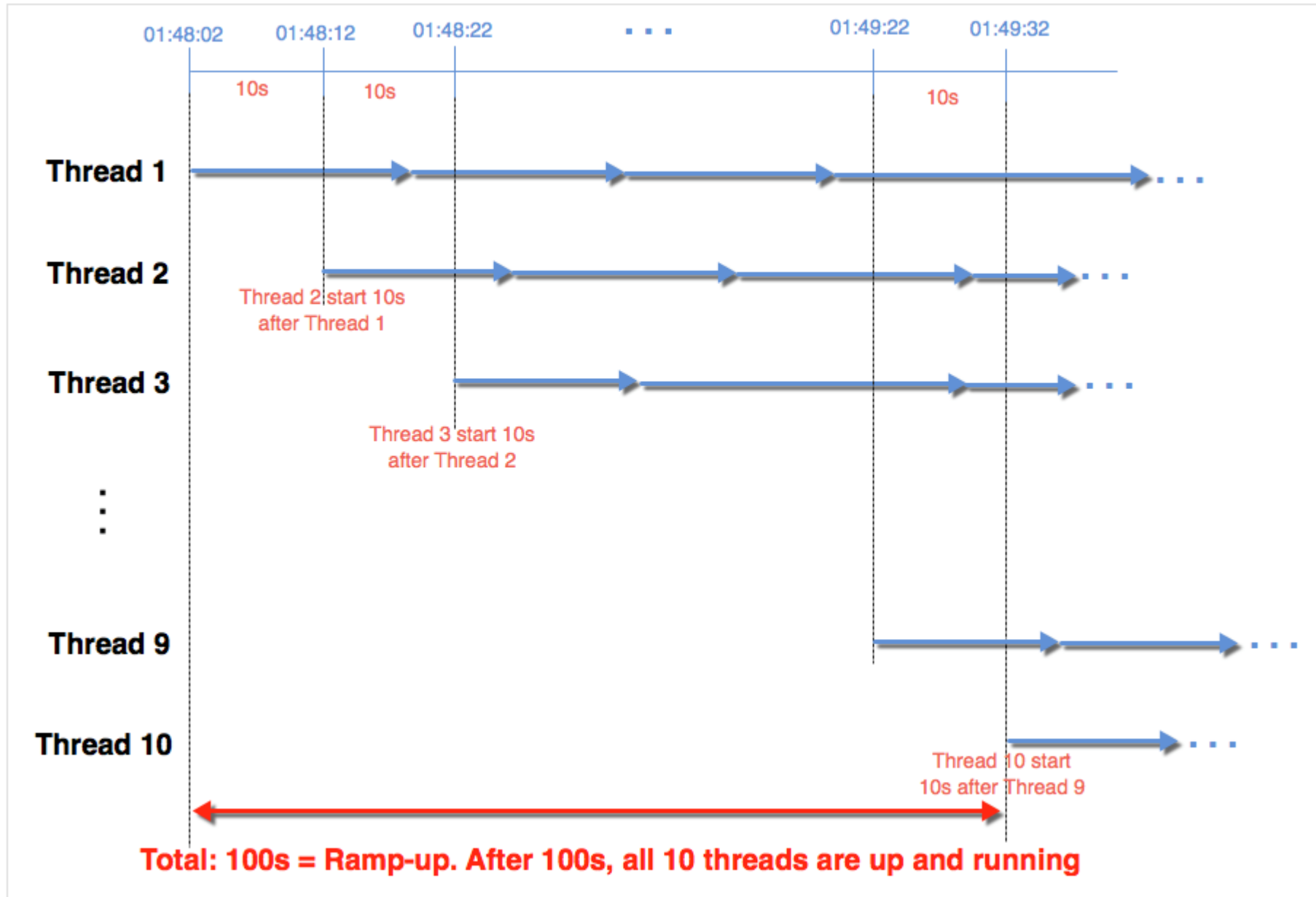
Scheduler Configuration

 If Loop Count is not -1 or Forever, duration will be  $\min(\text{Duration}, \text{Loop Count} * \text{iteration duration})$

Duration (seconds)

Startup delay (seconds)

# CREATING TEST PLAN



# CREATING TEST PLAN

## 2. Add Config Element

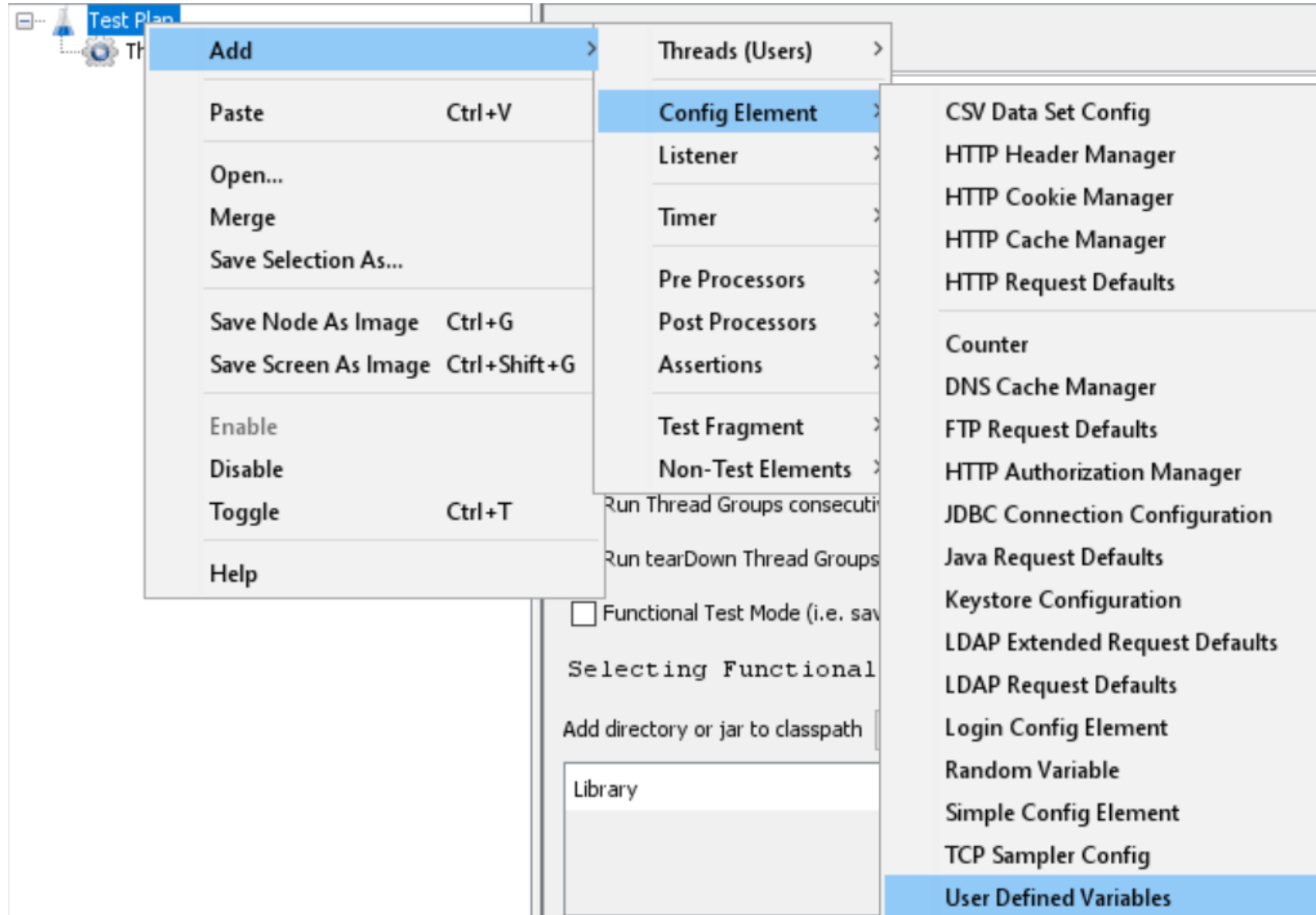
Config Element:

- ✓ User Defined Variables
- ✓ HTTP Request Defaults
- ✓ HTTP Cookie Manager
- ✓ HTTP Cache Manager



# CREATING TEST PLAN

## 2. Add Config Element > User Defined Variables



# CREATING TEST PLAN

## 2. Add Config Element> User Defined Variables

### User Defined Variables

Name:

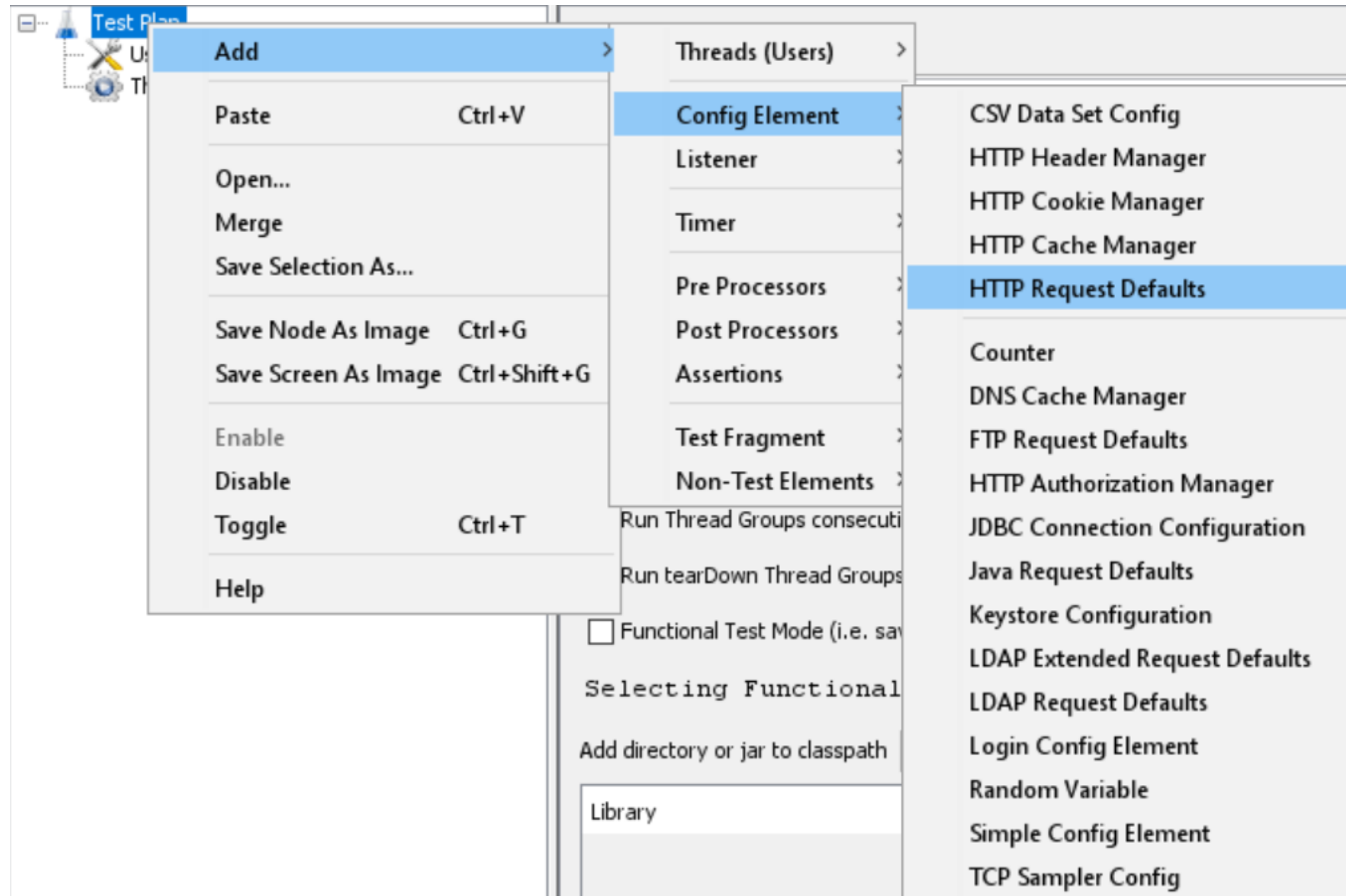
Comments:

#### User Defined Variables

Name:	Value	Description
host	10.85.226.243	
scheme	http	
path	/hesk/admin	

# CREATING TEST PLAN

## 2. Add Config Element > Http Request Default



# CREATING TEST PLAN

## 2. Add Config Element > Http Request Default

HTTP Request Defaults

Name: HTTP Request Defaults

Comments:

Basic Advanced

Web Server

Protocol [http]:  Server Name or IP:  Port Number:

HTTP Request

Path:  Content encoding:

Parameters Body Data

Input protocol, server name, path

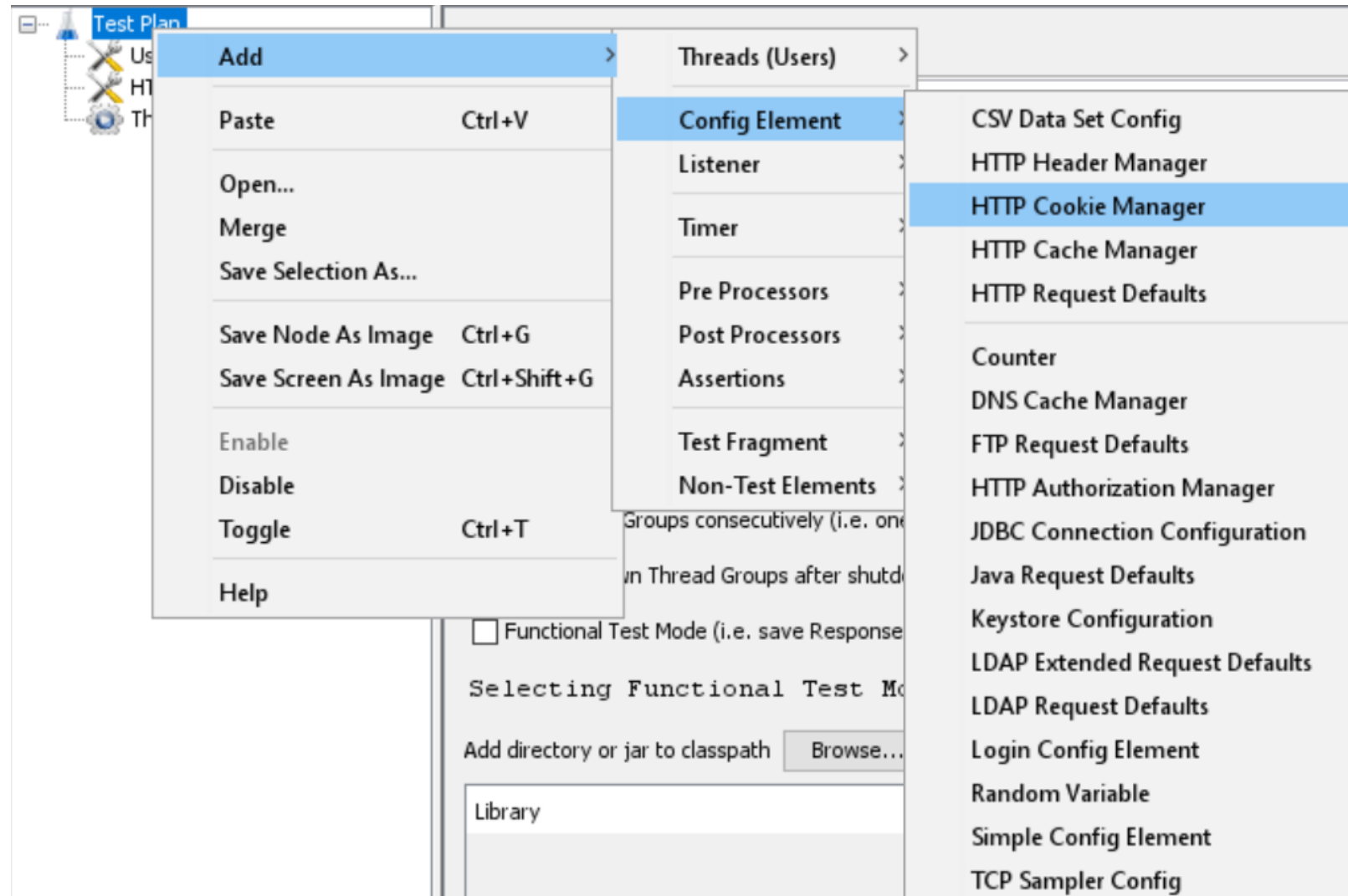
The main goals of using the HTTP Default Request is to:

- ✓ Avoid data duplication in tests.
- ✓ Make test scripts more (easily) maintainable.

**VARIABLE IN JMETER**  
**`${variable}`**

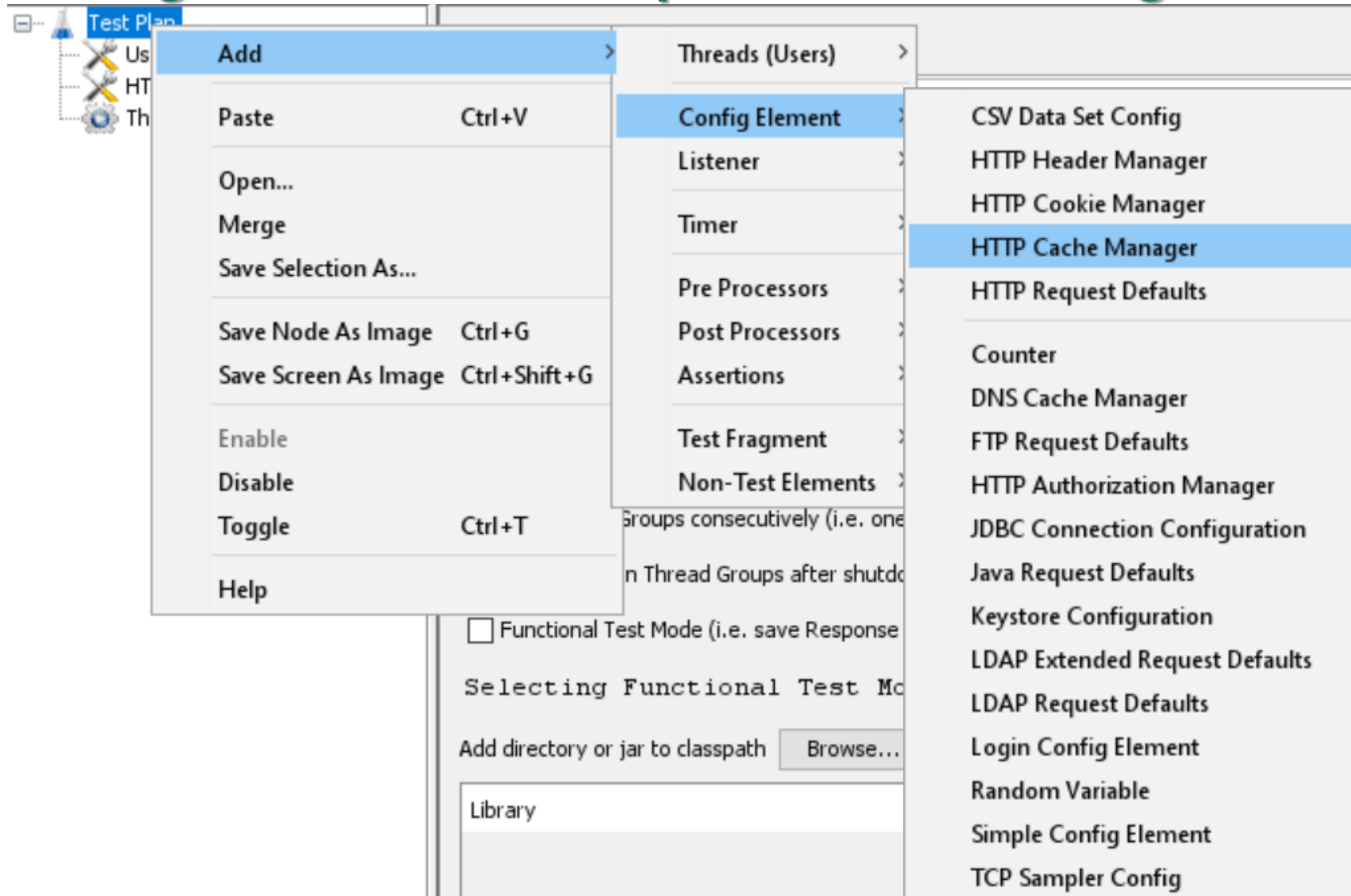
# CREATING TEST PLAN

## 2. Adding Config Element> Http Cookie Manager



# CREATING TEST PLAN

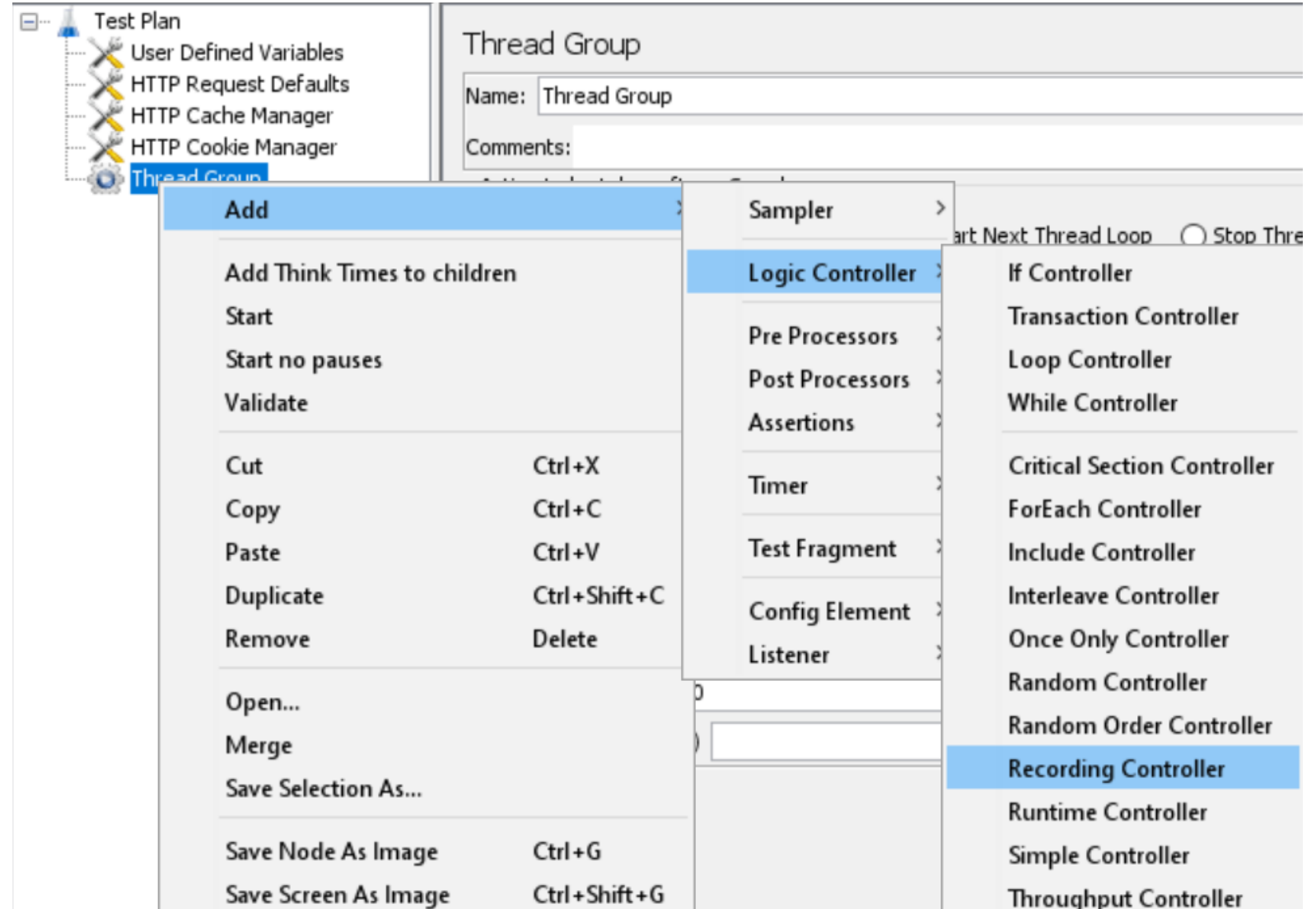
## 2. Add Config Element > Http Cache Manager



# CREATING TEST PLAN

## 3. Add Controller

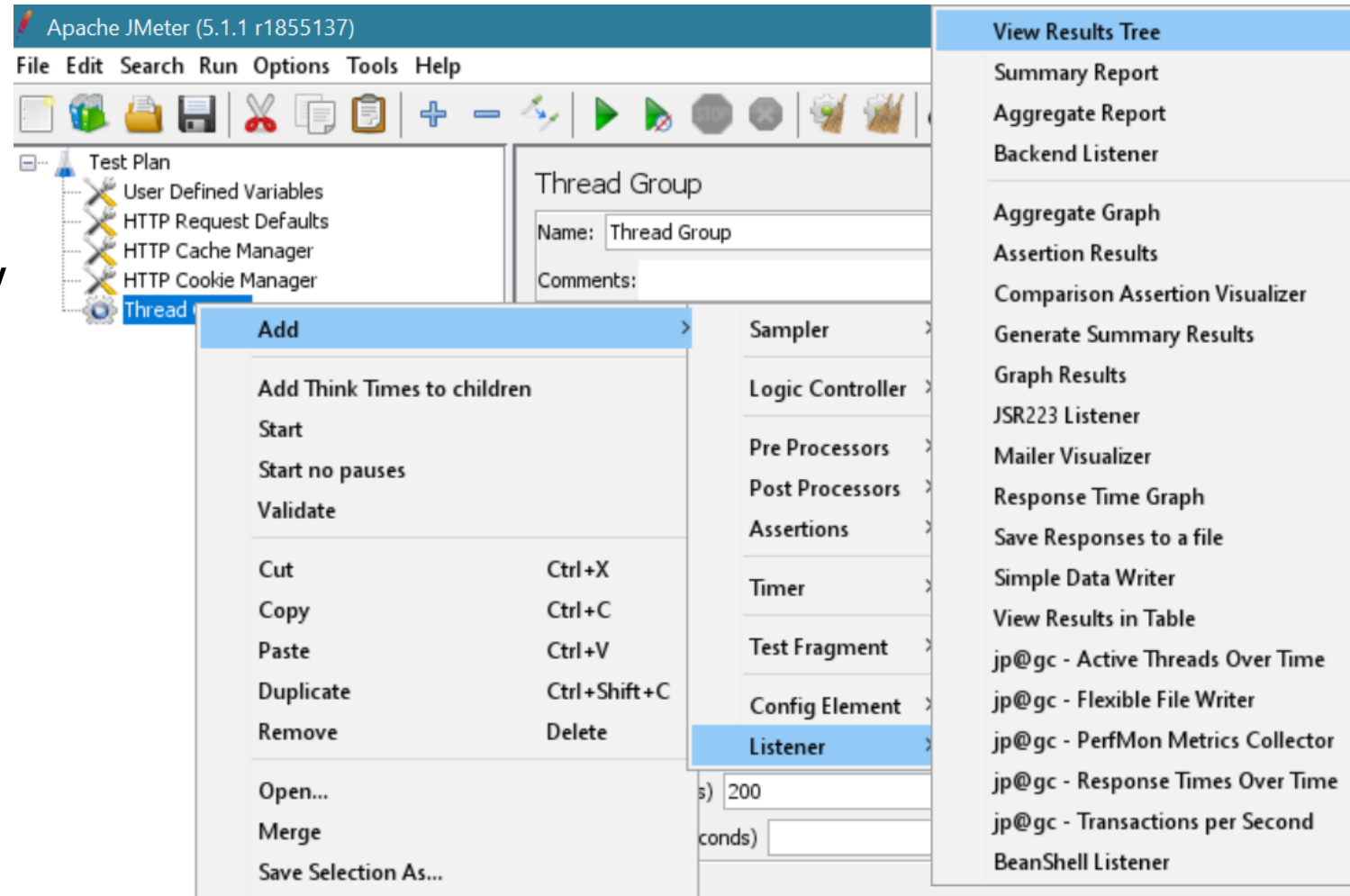
- JMeter can record your testing steps.
- Recording controller is a placeholder to store these recording steps.



# CREATING TEST PLAN

## 4. Adding Listener

- A listener is a component that shows the results of the samples.
  - ✓ View Results Tree - to view the contents of a response from any given sampler
  - ✓ Summary Report - shows values about the measurement JMeter has done while calling the same page in tabular format

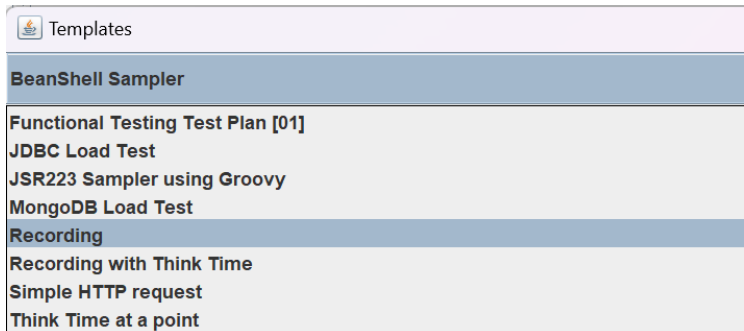


The screenshot shows the Apache JMeter interface. The main window displays a Test Plan with a Thread Group selected. The 'Add' context menu is open, showing options like 'Add Think Times to children', 'Start', 'Start no pauses', 'Validate', 'Cut', 'Copy', 'Paste', 'Duplicate', 'Remove', 'Open...', 'Merge', and 'Save Selection As...'. The 'Listener' option is highlighted, and a list of listeners is displayed on the right. The listeners include: View Results Tree, Summary Report, Aggregate Report, Backend Listener, Aggregate Graph, Assertion Results, Comparison Assertion Visualizer, Generate Summary Results, Graph Results, JSR223 Listener, Mailer Visualizer, Response Time Graph, Save Responses to a file, Simple Data Writer, View Results in Table, jp@gc - Active Threads Over Time, jp@gc - Flexible File Writer, jp@gc - PerfMon Metrics Collector, jp@gc - Response Times Over Time, jp@gc - Transactions per Second, and BeanShell Listener.

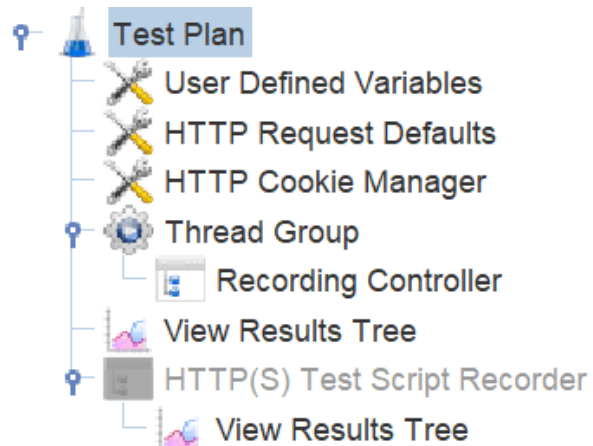


# USING TEMPLAT

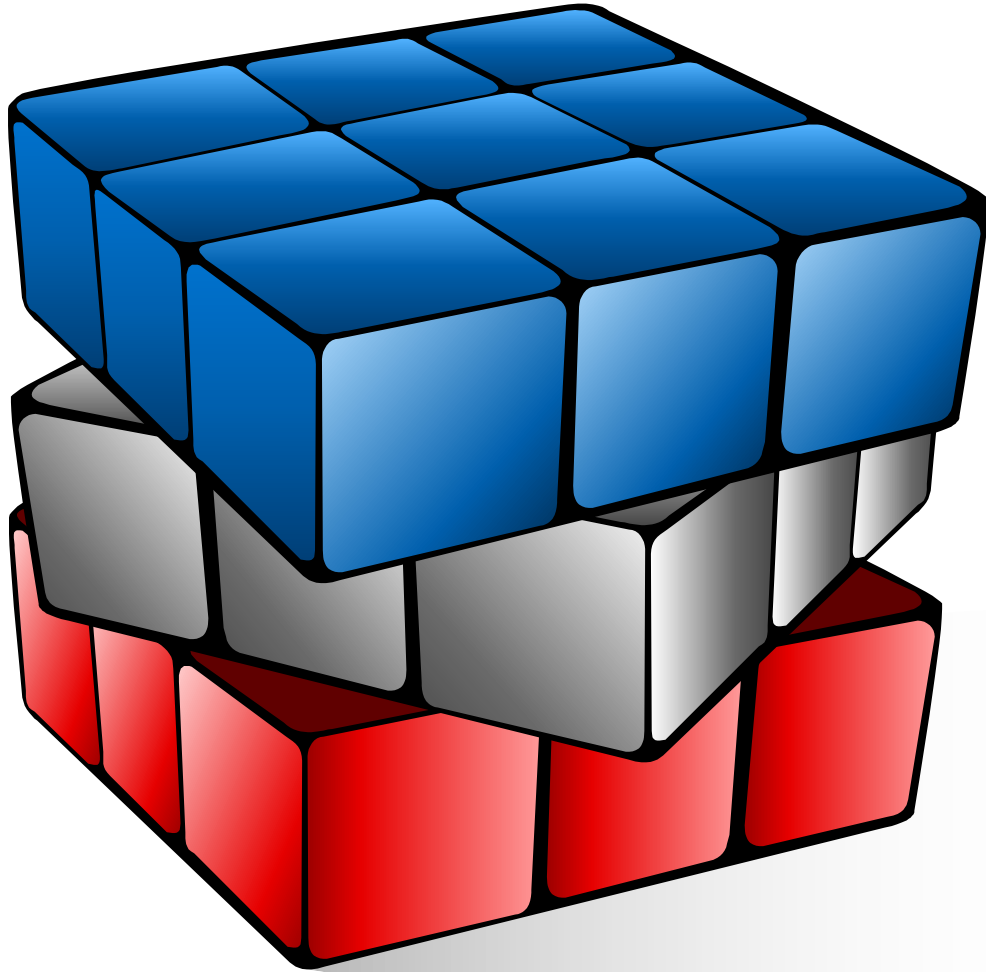
- Klik File -> Template ->
- Pilih *Recording*



- Klik *Create*, seterusnya *Create*



- Tambah dan kemaskini komponen JMeter yang diperlukan.



**LET'S DO IT**



# 02

## 1. CREATING JMETER TEST PLAN

- ✓ C1 - Thread Group
- ✓ C2 - Config Element
- ✓ C3 - Controller
- ✓ C4 - Listener

## 2. RECORDING AND SCRIPTING

- ✓ R1 - Script Recording
- ✓ R2 - JMeter Certificate (https recording)

## 3. VERIFYING SCRIPT

- ✓ V1 - Check Recorded Transaction
- ✓ V2 - Run Recorded Transaction
- ✓ V3 - View Result Tree
  - Regular Expression Extractor
  - Response Assertion
- ✓ V4 - View Summary Report

## 4. PERFORMING LOAD TEST

- ✓ P1 - CSV File
- ✓ P2 - Timer (Think Time)
- ✓ P3 - Smoke Test

# RECORDING

## 1. Script Recording

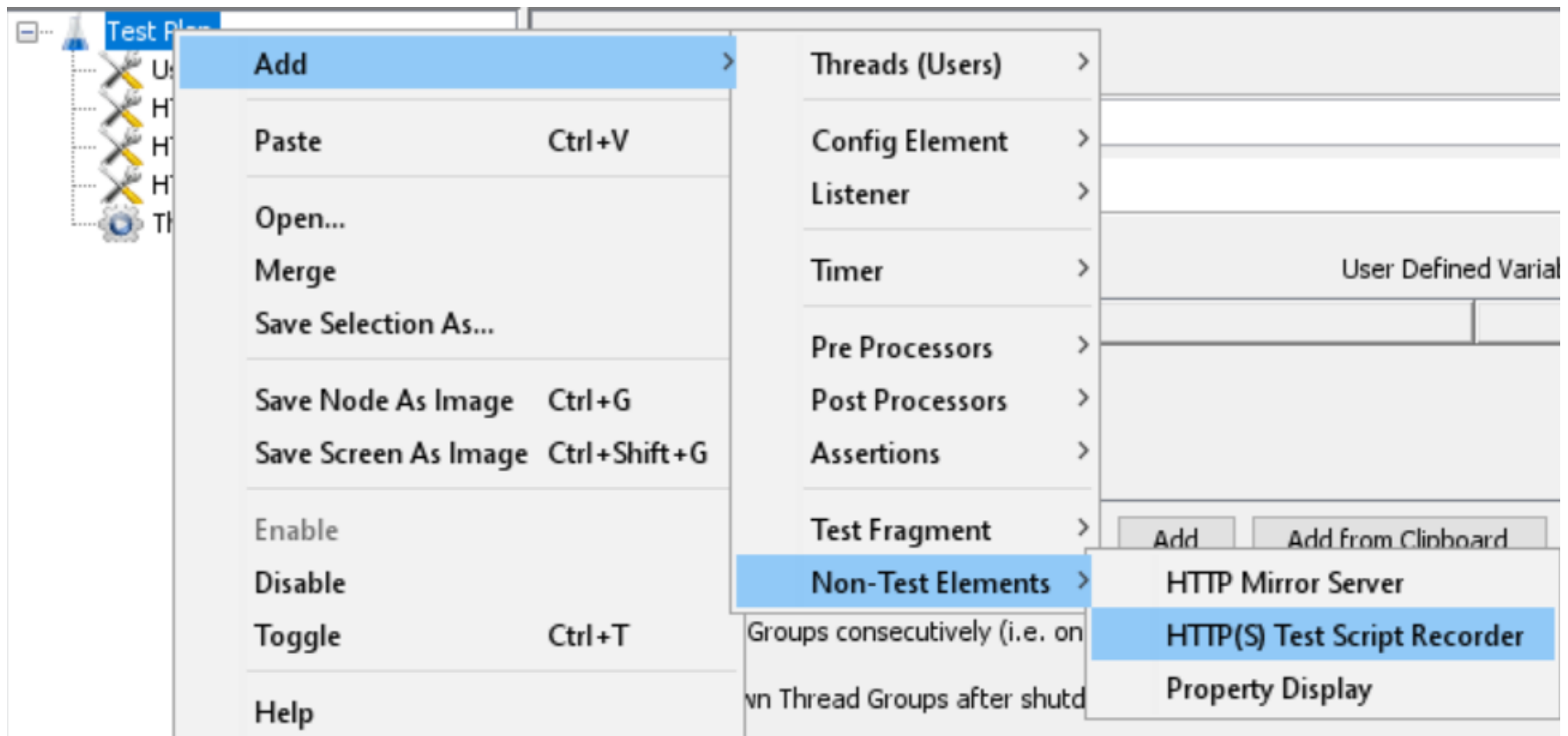
### Steps of Script Recording

- ✓ Add Non-Test Element HTTP(s) Test Script Recorder
  - ✓ Certificate Check
  - ✓ Enter the port number which is not in use. (8888)
  - ✓ Apply proxy on the browser you want to record and use the same port number
  - ✓ Start the JMeter Proxy Server
  - ✓ Record the Actions > Transaction Control

# RECORDING

## 1. Add HTTP(S) Test Script Recorder

- Can be used to record all the requests which your web or mobile application is making to server.
- Capture the HTTP(s) requests.



# RECORDING

## 2. Add Port Number

a) Port - Ensure the port number is not being used and as same as stated in browser.

b) Test Plan Creation:

- Target Controller
- Grouping
- Transaction name

The screenshot shows the JMeter configuration interface for recording. The 'State' section at the top has 'Start', 'Stop', and 'Restart' buttons. Below it, the 'Global Settings' section shows 'Port: 8888' and 'HTTPS Domains:'. The 'Test Plan Creation' tab is active, showing 'Test plan content' with 'Target Controller' set to 'hesk > createuser > Recording Controller-createuser' and 'Grouping' set to 'Put each group in a new transaction controller'. There are checkboxes for 'Capture HTTP Headers', 'Add Assertions', and 'Regex matching'. The 'HTTP Sampler settings' section includes 'Transaction name' (Landingpage), 'Naming scheme' (Transaction name), 'Counter start value', 'Create new transaction after request (ms)', 'Recording's default encoding' (UTF-8), and checkboxes for 'Retrieve All Embedded Resources', 'Use KeepAlive', 'Redirect Automatically', and 'Follow Redirects'. The 'GraphQL HTTP Sampler settings' section has a checkbox for 'Detect GraphQL Request'.

State

Start Stop Restart

Global Settings

Port: 8888 HTTPS Domains:

Test Plan Creation Requests Filtering

Test plan content

Target Controller: hesk > createuser > Recording Controller-createuser

Grouping: Put each group in a new transaction controller

Capture HTTP Headers  Add Assertions  Regex matching

HTTP Sampler settings

Transaction name Landingpage

Naming scheme Transaction name

Counter start value

Create new transaction after request (ms):

Recording's default encoding UTF-8

Retrieve All Embedded Resources  Redirect Automatically  Follow Redirects

Use KeepAlive

Type:

GraphQL HTTP Sampler settings

Detect GraphQL Request

# RECORDING

## 3. Proxy Setting in Browser



- Firefox  
-General -> Network Setting -> setting -> Manual proxy configuration

Connection Settings

Configure **Proxy** Access to the Internet

- No **proxy**
- Auto-detect **proxy** settings for this network
- Use system **proxy** settings

**Manual proxy** configuration

HTTP **Proxy** localhost Port 8888

Also use this **proxy** for HTTPS

HTTPS **Proxy** localhost Port 8888

SOCKS Host Port 0

SOCKS v4  SOCKS v5

OK

Cancel

Port number must be same with HTTP(S) Test Script Recorder

# RECORDING

## 3. Proxy Setting in Browser

- Chrome  
-Setting -> Proxy-> Setting ->  
Use a proxy server: **ON**  
Proxy IP address: <http://localhost>  
Port: **8888**



### Edit proxy server

Use a proxy server

On

Proxy IP address	Port
<input type="text" value="http://localhost"/>	<input type="text" value="8888"/>

Port number must be same with HTTP(S) Test Script Recorder

Use the proxy server except for addresses that start with the following entries.  
Use semicolons (;) to separate entries.

Don't use the proxy server for local (intranet) addresses

Save

Cancel



# RECORDING

## 5. Start JMeter Proxy Server

- Click "Start" to initial the recording

The screenshot shows the JMeter Proxy Server configuration window. At the top, there is a 'State' section with three buttons: 'Start' (highlighted with a blue callout 'Click Here'), 'Stop', and 'Restart'. Below this is the 'Global Settings' section, where the 'Port' is set to '8888' (highlighted with a yellow box) and 'HTTPS Domains' is empty. The 'Test Plan Creation' tab is active, showing 'Test plan content' with a 'Target Controller' dropdown set to 'hesk > createuser > Recording Controller-createuser' (highlighted with a yellow box and a blue callout 'Choose the right one!'). Other options include 'Grouping' set to 'Put each group in a new transaction controller', and checkboxes for 'Capture HTTP Headers' (checked), 'Add Assertions' (unchecked), and 'Regex matching' (checked). The 'HTTP Sampler settings' section includes 'Transaction name' (Landingpage), 'Naming scheme' (Transaction name), 'Counter start value' (empty), 'Create new transaction after request (ms):' (empty), 'Recording's default encoding' (UTF-8), and checkboxes for 'Retrieve All Embedded Resources' (unchecked), 'Use KeepAlive' (checked), 'Redirect Automatically' (unchecked), and 'Follow Redirects' (checked). The 'Type:' dropdown is empty. The 'GraphQL HTTP Sampler settings' section has a checked checkbox for 'Detect GraphQL Request'.

# RECORDING

## 6. Record the Actions

- Transaction Control
  - ✓ Use to name the transaction
  - ✓ Input the name before proceeding to next transaction

Global Settings

Port:  HTTPS Domains:

Test Plan Creation **Requests Filtering**

Test plan content

Target Controller:

Grouping:   Capture HTTP Headers  Add Assertions  Regex matching

HTTP Sampler settings

Transaction name

Create new transaction  Retrieve All Embedded Resources  Redirect Automatically  Use KeepAlive

HTTP Sampler settings

Transaction name

Create new transaction after request (ms):

Type:

# PERFORMING LOAD TEST

## 2. JMeter Certificate

Pastikan Jmeter  
certificate masih  
valid



Chrome:

Setting > Manage Certificate > Trusted Root Certification Authorities > Import >

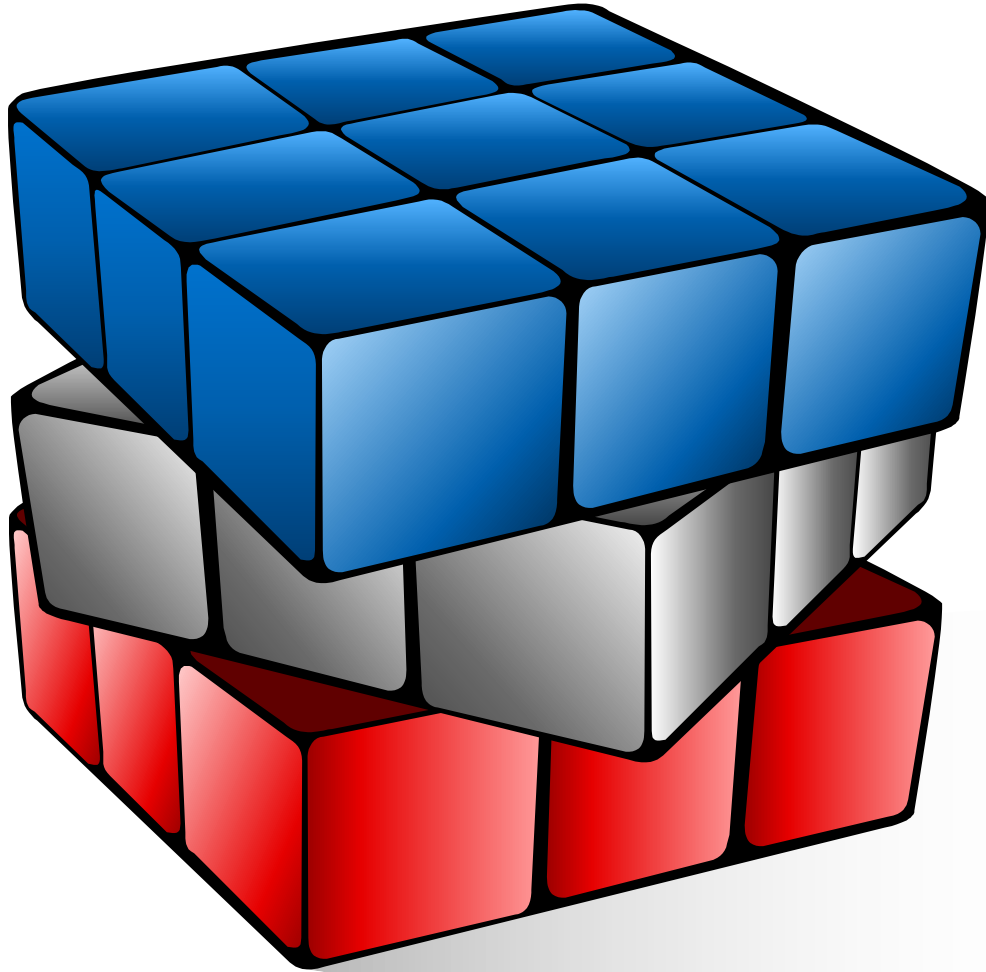
browse file > ..\apache-jmeter-5.1\bin



Firefox:

Option > Certificate > View Certificate > Authorities > Import

`../bin/ApacheJMeterTemporaryRootCA`



**LET'S DO IT**

1

Transaction Name/Prefix	Activities Create User
Landingpage	Type url <b>your_ip_address/hesk/admin</b>
Login	Fill up login credential <b>administrator, administrator</b>
Clickusers	Click tab <b>users</b>
	Fill up user info <ul style="list-style-type: none"><li>• Fill up Profile Information<ul style="list-style-type: none"><li>✓ Real name: <b>pengguna01</b></li><li>✓ Email: <b>pengguna01@gmail.com</b></li><li>✓ Username: <b>pengguna01</b></li><li>✓ Password: <b>password</b></li><li>✓ Confirm password: <b>password</b></li></ul></li><li>• Permission<ul style="list-style-type: none"><li>✓ Click : <b>Staff</b></li><li>✓ Choose : Can run reports (all)</li></ul></li></ul>
Createuser	Click <b>Create User</b>
Logout	Click <b>logout</b>

2

Transaction Name/Prefix	Activities Create Ticket
Landingpage	Type url <a href="#">your_ip_address/hesk/admin</a>
Login	Fill up login credential <a href="#">username01, password</a>
Newticket	Click tab <a href="#">+New Ticket</a> Fill up the form: Name: Email: Priority: Subject: Message: Untick:
Createticket	Click <a href="#">Submit Ticket</a>
Gohome	Click <a href="#">Home</a>
Logout	Click <a href="#">Logout</a>



# 03

## 1. CREATING JMETER TEST PLAN

- ✓ C1 - Thread Group
- ✓ C2 - Config Element
- ✓ C3 - Controller
- ✓ C4 - Listener

## 2. RECORDING AND SCRIPTING

- ✓ R1 - Script Recording
- ✓ R2 - JMeter Certificate

## 3. VERIFYING SCRIPT

- ✓ V1 - Check Recorded Transaction
- ✓ V2 - Run Recorded Transaction
- ✓ V3 - View Result Tree
  - Regular Expression Extractor
  - Response Assertion
- ✓ V4 - View Summary Report

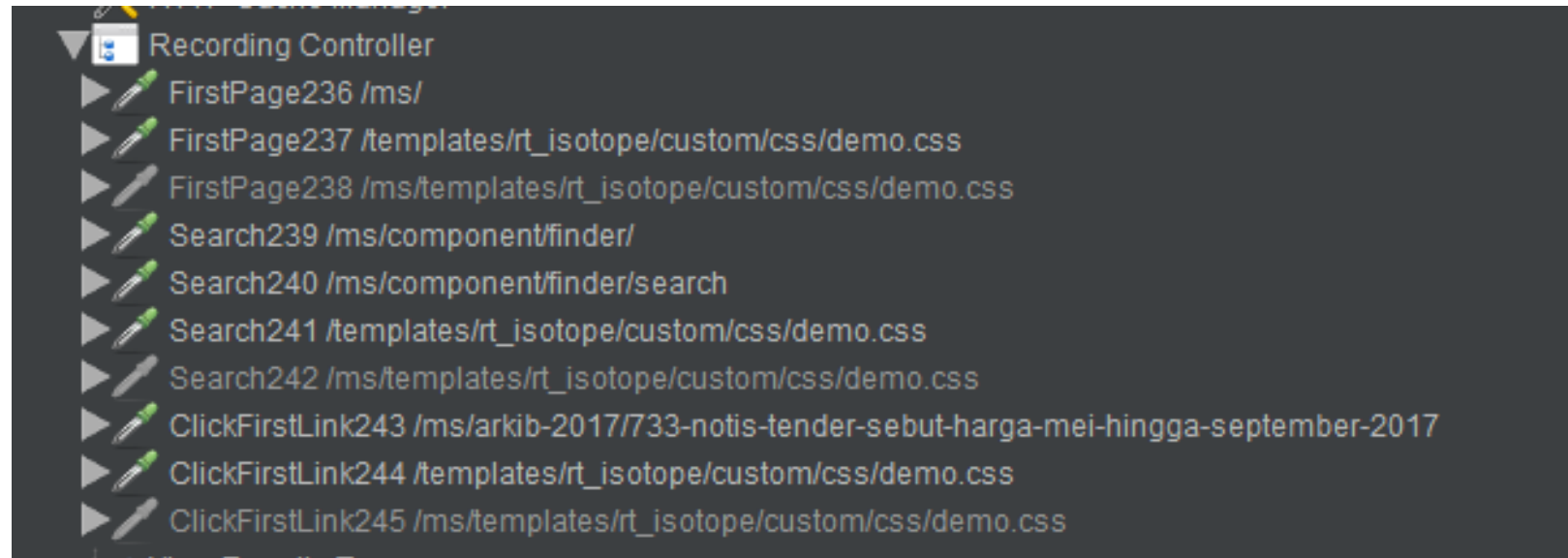
## 4. PERFORMING LOAD TEST

- ✓ P1 - CSV File
- ✓ P2 - Timer (Think Time)
- ✓ P3 - Smoke Test

# VERIFYING RESULT

## 1. Check Recorded Transactions

- All the transactions will be recorded under “Recording Controller”
- Can remove unnecessary component such as \*.css, \*.png, \*.txt and so on.

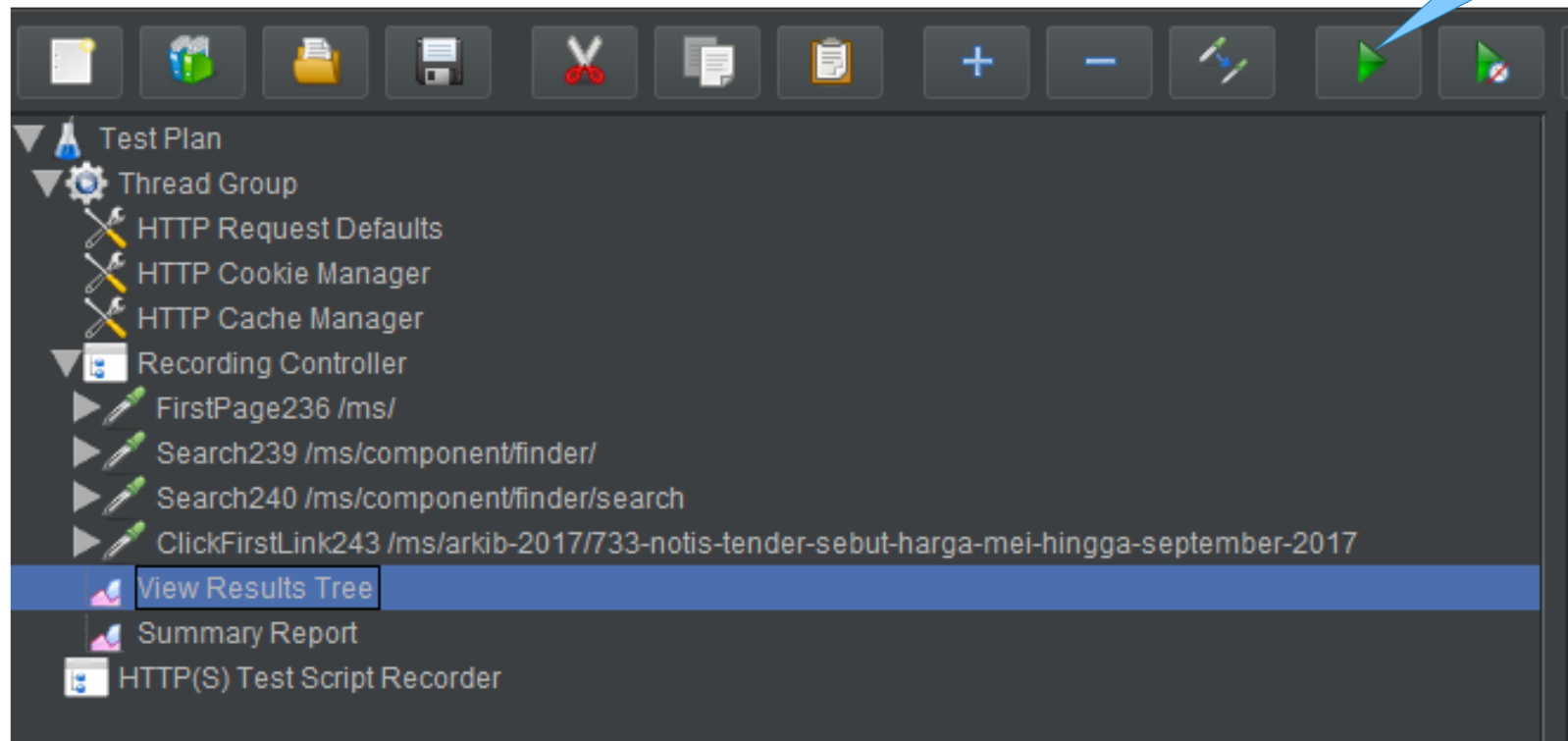




# VERIFYING RESULT

## 2. Run Recorded Transactions

Click Start to run the script



# VERIFYING RESULT

## 3. View Result Tree

- The View Results Tree shows:
  - ✓ A tree of all sample responses
  - ✓ Response code
  - ✓ Assertion result
  - ✓ Time it took to get this response

No error found!

The screenshot displays the 'View Results Tree' and 'Sampler result' panels of a web testing tool. The 'View Results Tree' panel on the left shows a list of test samples, each with a green checkmark icon indicating a successful result. The samples include 'FirstPage236 /ms/' and 'Search239 /ms/component/finder/'. The 'Sampler result' panel on the right shows the details for the selected 'FirstPage236 /ms/' sample. It includes the following information:

- Thread Name: Thread Group 1-1
- Sample Start: 2018-11-28 00:49:45 SGT
- Load time: 6356
- Connect Time: 18
- Latency: 6326
- Size in bytes: 19465
- Sent bytes: 329
- Headers size in bytes: 525
- Body size in bytes: 18940
- Sample Count: 1
- Error Count: 0
- Data type ("text"|"bin"|"): text
- Response code: 200
- Response message: OK

Below this information, the 'HTTPSampleResult fields' are listed:

- ContentType: text/html; charset=utf-8
- DataEncoding: utf-8

# VERIFYING RESULT

## If error found?

- Check error code (401, 404..) in Response Body, Sampler Result
- Check if token used (variable need to be declared)
- Authentication issue – user account

# VERIFYING RESULT

CONTOH

- Add Regular Expression Extractor

- Check if there any token used  
(variable need to be declared)
- Replace the token value with **(.+?)**
- Variable in JMeter

**`${variable}`**



# VERIFYING RESULT

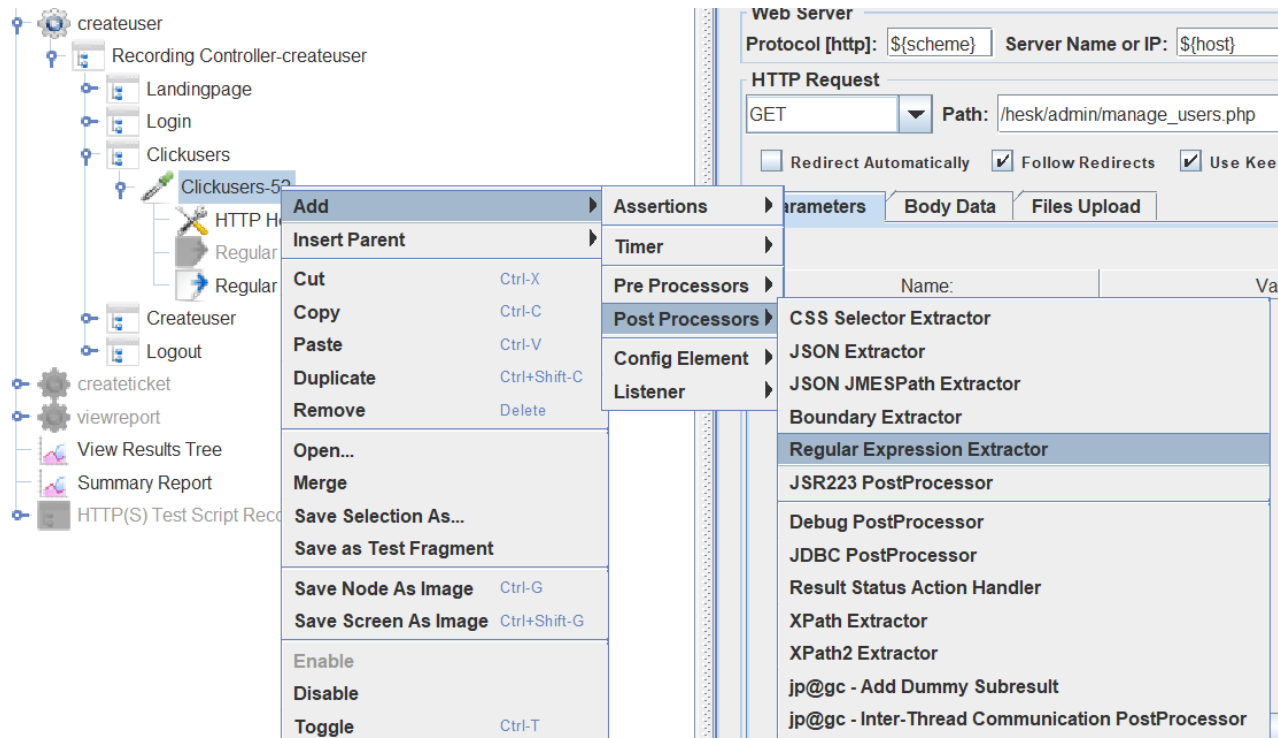
- Salin format token dan gantikan nilai token dengan (.+?)

Gantikan  
dengan (.+?)

- token=09cbd1e9f349581da0955f1bdbde10880bed7163">
- token=(.+?)">

# VERIFYING RESULT

- Tambah Regular Expression Extractor pada http request di mana token dijumpai



The screenshot shows the 'Regular Expression Extractor' configuration dialog box. The 'Name' field is 'Regular Expression Extractor'. The 'Apply to:' section has 'Main sample only' selected. The 'Field to check' section has 'Body' selected. The 'Name of created variable:' field contains 'token'. The 'Regular Expression:' field contains 'token=(.+?)">'. The 'Template (\$1\$ where i is capturing group number, starts at 1)' field contains '\$1\$'. The 'Match No. (0 for Random):' field contains '1'. The 'Default Value:' field is empty. A yellow box highlights the 'Name of created variable:', 'Regular Expression:', and 'Match No.' fields. A blue speech bubble points to the 'Name of created variable:' field with the text 'Isikan maklumat token'.

Name:	Regular Expression Extractor
Comments:	
Apply to:	<input type="radio"/> Main sample and sub-samples <input checked="" type="radio"/> Main sample only <input type="radio"/> Sub-samples <input type="radio"/> JMeter Variable Name to use
Field to check	<input checked="" type="radio"/> Body <input type="radio"/> Body (unescaped) <input type="radio"/> Body as a Document <input type="radio"/> Response Headers <input type="radio"/> Request Headers <input type="radio"/> URL
Name of created variable:	token
Regular Expression:	token=(.+?)">
Template (\$1\$ where i is capturing group number, starts at 1)	\$1\$
Match No. (0 for Random):	1
Default Value:	<input type="text"/> <input type="checkbox"/> Use empty default value

# VERIFYING RESULT

- Gantikan nilai token pada http request yang mengandung ralat invalid error dengan variable `#{token}`

HTTP Request

Name: Createuser-54

Comments: Detected the start of a redirect chain

Basic Advanced

Web Server

Protocol [http]: `#{scheme}` Server Name or IP: `#{host}`

HTTP Request

POST Path: /hesk/admin/manage\_users.php

Redirect Automatically  Follow Redirects  Use KeepAlive  Use multipart/form-data  Browser-com

Parameters Body Data Files Upload

Send Parameters With the

Name:	Value
notify_customer_new	1
notify_customer_reply	1
show_suggested	1
reload_time	30
secmin	sec
notify_new_unassigned	1
notify_new_my	1
notify_reply_unassigned	1
notify_reply_my	1
notify_assigned	1
notify_note	1
notify_pm	1
a	new
token	<code>#{token}</code>



# VERIFYING RESULT

## Response Assertion, TIPS!

- Go to View Result Tree
- Choose HTML
- Click Response Data
- Choose Unique Word..

**View Results Tree**

Name: View Results Tree

Comments:

Write results to file / Read from file

Filename   Log/Display Only:  Error

Search:   Case sensitive  Regular exp.

**HTML**

- ✓ Landingpage-44
- ✓ Landingpage
- ✓ Login-45
- ✓ Login-46
- ✓ Login-48
- ✓ Login
- ✓ Clickusers-52
- ✓ Clickusers
- ✓ Createuser-54
- ✓ Createuser-56
- ✓ Createuser
- ✓ Logout-59
- ✓ Logout

**Response data**

Response Body

Find

[Home](#) [Users](#) [Categories](#) [Canned](#) [Knowledgebase](#) [Reports](#) [Tools](#) [Settings](#) [Profile](#) [Mail\(0\)](#) [Logout](#)

**Success:** New user pengguna1001 with password password has been successfully added

**Manage users [2]**

Name	Email	Username	Administrator	Rating
mieza	<a href="mailto:mieza@mail.com">mieza@mail.com</a>	mieza	NO	
nama pengguna 01	<a href="mailto:pengguna01@mail.com">pengguna01@mail.com</a>	pengguna01	NO	



# VERIFYING RESULT

## Response Assertion

Response Assertion

Name:

Comments:

Apply to:

Main sample and sub-samples  Main sample only  Sub-samples only  JMeter Variable Name to use

Field to Test

Text Response  Response Code  Response Message  Response Headers

Request Headers  URL Sampled  Document (text)  Ignore Status

Request Data

Pattern Matching Rules

Contains  Matches  Equals  Substring  Not  Or

Patterns to Test

Patterns to Test	
1	Pendaftar Akaun Baru SPA9 Berjaya

# VERIFYING RESULT

## Response Assertion

**Response Assertion**

Name:

Comments:

Apply to:

Main sample and sub-samples  Main sample only  Sub-samples only  JMeter Variable Name to use

Field to Test

Text Response  Response Code  Response Message  Response Headers

Request Headers  URL Sampled  Document (text)  Ignore Status

Request Data

Pattern Matching Rules

Contains  Matches  Equals  Substring  Not  Or

Patterns to Test

Patterns to Test

1	has been successfully added
---	-----------------------------

# VERIFYING RESULT

## 3. View Summary Report

- The summary report creates a table row for each differently named request in your test
- JMeter takes into account the total time over which the requests have been generated

**Summary Report**

Name:

Comments:

Write results to file / Read from file

Filename   Log/Display Only:  Errors  Successes

Label	# Samples	Average	Min	Max	Std. Dev.	Error %	Throughput	Received KB/sec	Sent KB/sec	Avg. Bytes
Landingpage-44	1	20	20	20	0.00	0.00%	50.0/sec	262.35	17.48	5373.0
Landingpage	1	20	20	20	0.00	0.00%	31.9/min	2.79	0.19	5373.0
Login-45	1	102	102	102	0.00	0.00%	9.8/sec	1325.92	12.22	138490.0
Login-46	1	33	33	33	0.00	0.00%	30.3/sec	4075.14	15.83	137707.0
Login-48	1	15	15	15	0.00	0.00%	66.7/sec	780.14	25.98	11983.0
Login	1	150	150	150	0.00	0.00%	11.3/min	52.82	0.41	288180.0
Clickusers-52	1	116	116	116	0.00	0.00%	8.6/sec	13595.85	4.64	1614969.0
Clickusers	1	116	116	116	0.00	0.00%	24.8/min	650.63	0.22	1614969.0
Createuser-54	1	137	137	137	0.00	0.00%	7.3/sec	11529.40	14.46	1617437.0
Createuser-56	1	79	79	79	0.00	0.00%	12.7/sec	19983.42	6.84	1616579.0
Createuser	1	216	216	216	0.00	0.00%	19.7/min	1037.86	0.83	3234016.0
Logout-59	1	16	16	16	0.00	0.00%	62.5/sec	322.02	36.74	5276.0
Logout	1	16	16	16	0.00	0.00%	44.6/min	3.83	0.44	5276.0
TOTAL	13	79	15	216	62.97	0.00%	55.6/min	717.14	0.88	791971.4

# LOOP CONTROL

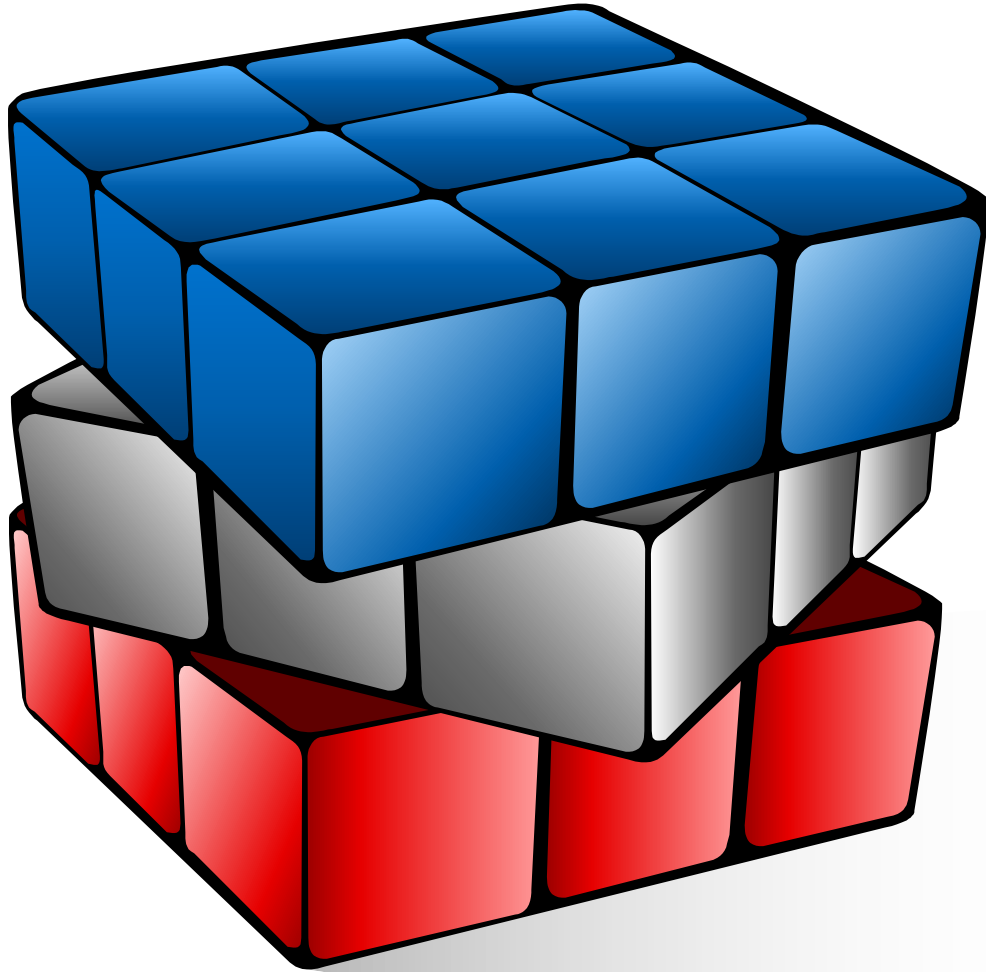
Setiap pengguna akan menjana 3 tiket setiap seorang

The screenshot shows the JMeter configuration interface. On the left, a tree view shows the project structure under 'hesk4.0'. A context menu is open over the 'Login' element, with 'Add' > 'Logic Controller' > 'Loop Controller' selected. The 'Transaction Controller' configuration panel is visible on the right, showing 'Name: Login' and 'Generate parent sample' checkbox.

Transaction Controller
Name: Login
Comments:
<input type="checkbox"/> Generate parent sample

The screenshot shows the JMeter project tree with the 'Loop Controller' added to the 'Login' element. The configuration panel for the 'Loop Controller' is shown on the right, with 'Name: Loop Controller' and 'Loop Count: 3'.

Loop Controller
Name: Loop Controller
Comments:
Loop Count: <input type="checkbox"/> Infinite 3



**LET'S DO IT**

## 1. CREATING JMETER TEST PLAN

- ✓ C1 - Thread Group
- ✓ C2 - Config Element
- ✓ C3 - Controller
- ✓ C4 - Listener

## 2. RECORDING AND SCRIPTING

- ✓ R1 - Script Recording
- ✓ R2 - JMeter Certificate

## 3. VERIFYING SCRIPT

- ✓ V1 - Check Recorded Transaction
- ✓ V2 - Run Recorded Transaction
- ✓ V3 - View Result Tree
  - Regular Expression Extractor
  - Response Assertion
- ✓ V4 - View Summary Report

## 4. PERFORMING LOAD TEST

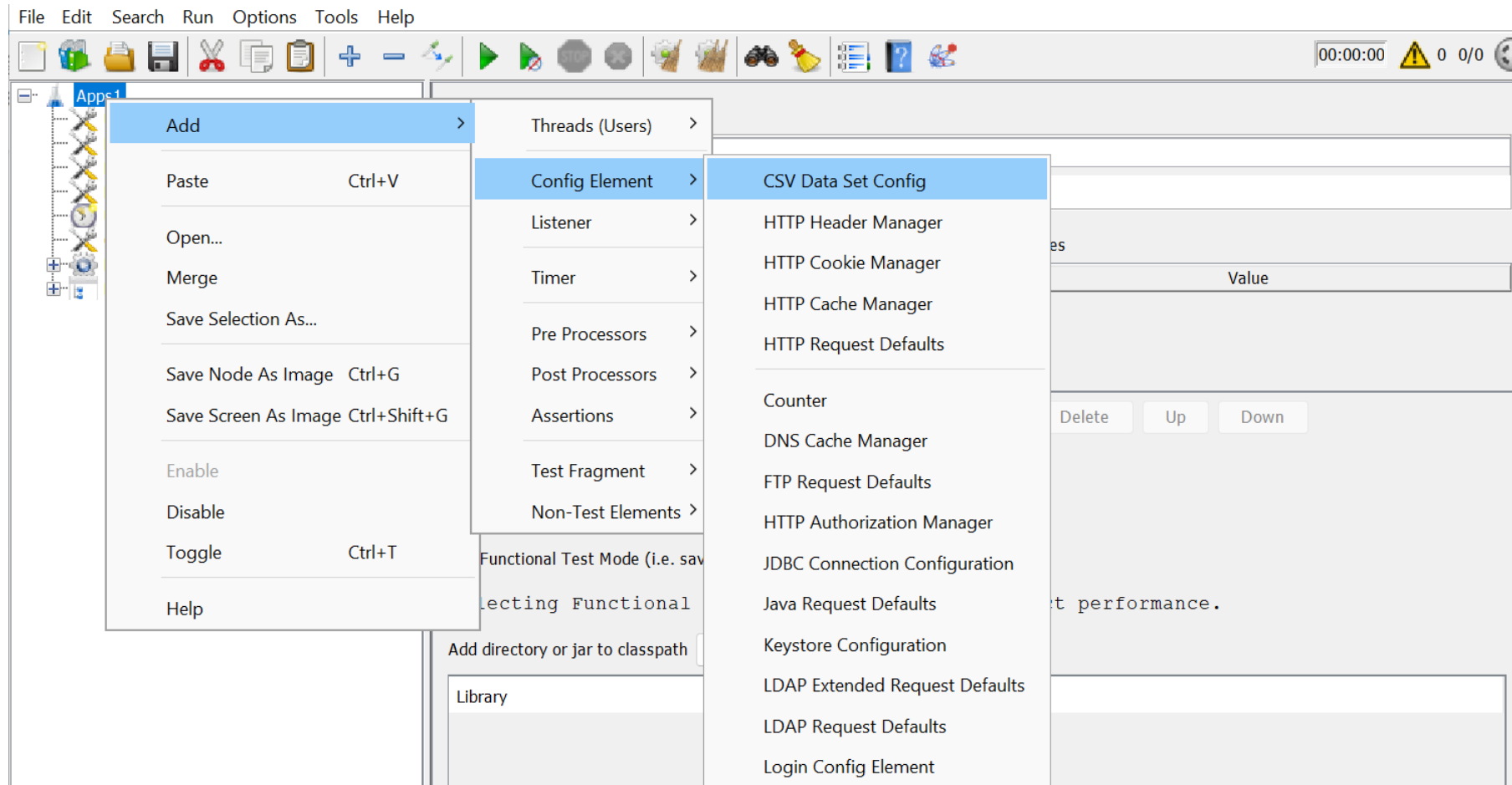
- ✓ P1 - CSV File
- ✓ P2 - Timer (Think Time)
- ✓ P3 - Smoke Test



# PERFORM LOAD TEST

## 1. Add CSV File

- This element that allows you to use external data sets in a CSV format.



# PERFORMING LOAD TEST

## Add CSV File

### CSV Data Set Config

Name: CSV Data Set Config

Comments:

- Configure the CSV Data Source

Filename	C:/Testplanjmeter/project2021/MyMInD/loginurusetia.csv	Browse...
File encoding		▼
Variable Names (comma-delimited)	nokp,password	
Ignore first line (only used if Variable Names is not empty)	True	▼
Delimiter (use '\t' for tab)	,	
Allow quoted data?	False	▼
Recycle on EOF ?	False	▼
Stop thread on EOF ?	True	▼
Sharing mode	All threads	

# PERFORMING LOAD TEST

## Add CSV File

**HTTP Request**

Name: Login133 /index.php/ms/log-masuk/admin-portal

Comments: Detected the start of a redirect chain

Basic Advanced

Web Server

Protocol [http]: http Server Name or IP: Port Number:

HTTP Request

Method: POST Path: /index.php/ms/log-masuk/admin-portal Content encoding: utf-8

Redirect Automatically  Follow Redirects  Use KeepAlive  Use multipart/form-data  Browser-compatible headers

Parameters Body Data Files Upload

Send Parameters With the Request:

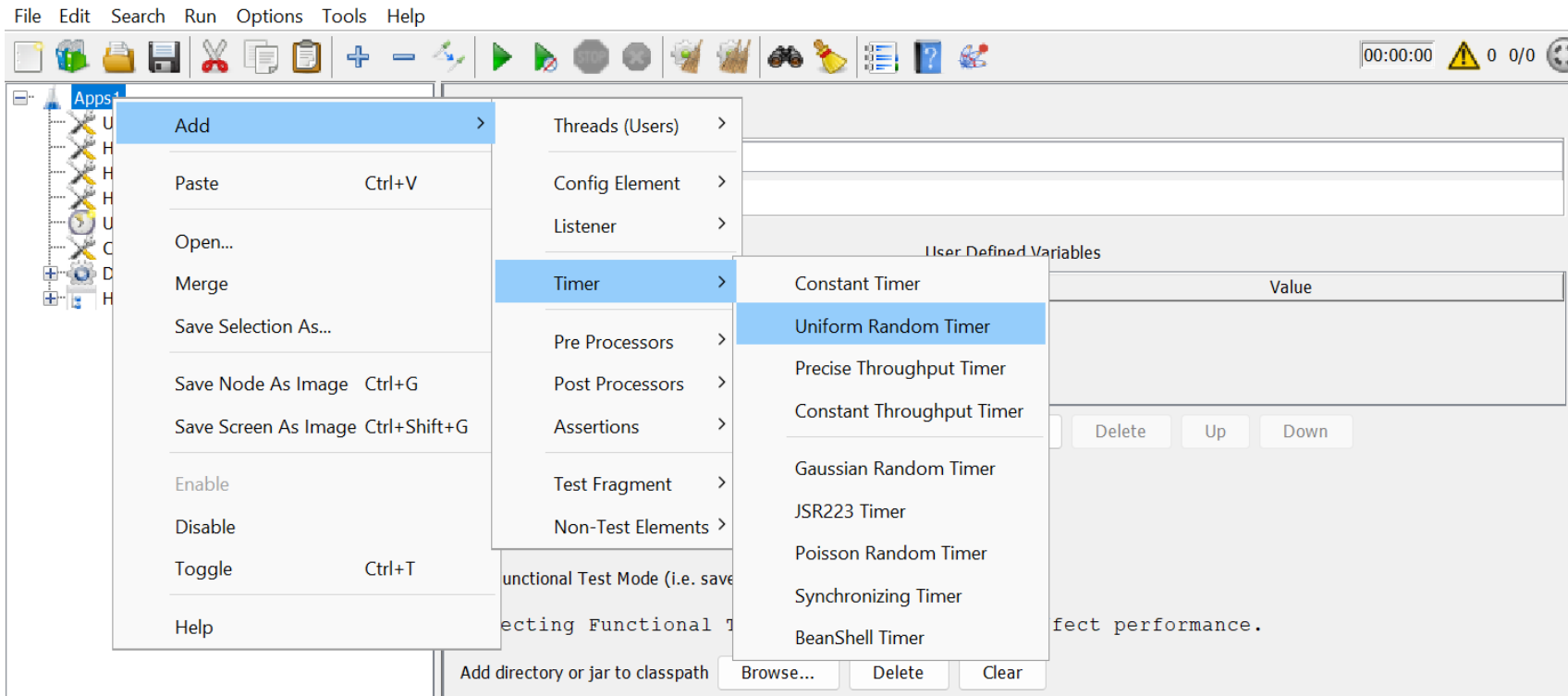
Name:	Value	URL Encode?	Content-Type	Include Equals?
username	\${loginid}	<input type="checkbox"/>	text/plain	<input checked="" type="checkbox"/>
password	\${password}	<input type="checkbox"/>	text/plain	<input checked="" type="checkbox"/>
Submit		<input type="checkbox"/>	text/plain	<input checked="" type="checkbox"/>
option	com_users	<input type="checkbox"/>	text/plain	<input checked="" type="checkbox"/>
task	user.login	<input type="checkbox"/>	text/plain	<input checked="" type="checkbox"/>
...	...	<input checked="" type="checkbox"/>	text/plain	<input checked="" type="checkbox"/>

Detail Add Add from Clipboard Delete Up Down

# PERFORMING LOAD TEST

## Timer

- Timers are used to define the time period that you want to wait between requests
- By default, a JMeter thread sends requests without pausing between each request
- It is recommend that to specify a delay by adding one of the available timers to the Thread Group



**Uniform Random Timer**

Name:

Comments:

- Thread Delay Properties

Random Delay Maximum (in milliseconds):

Constant Delay Offset (in milliseconds):

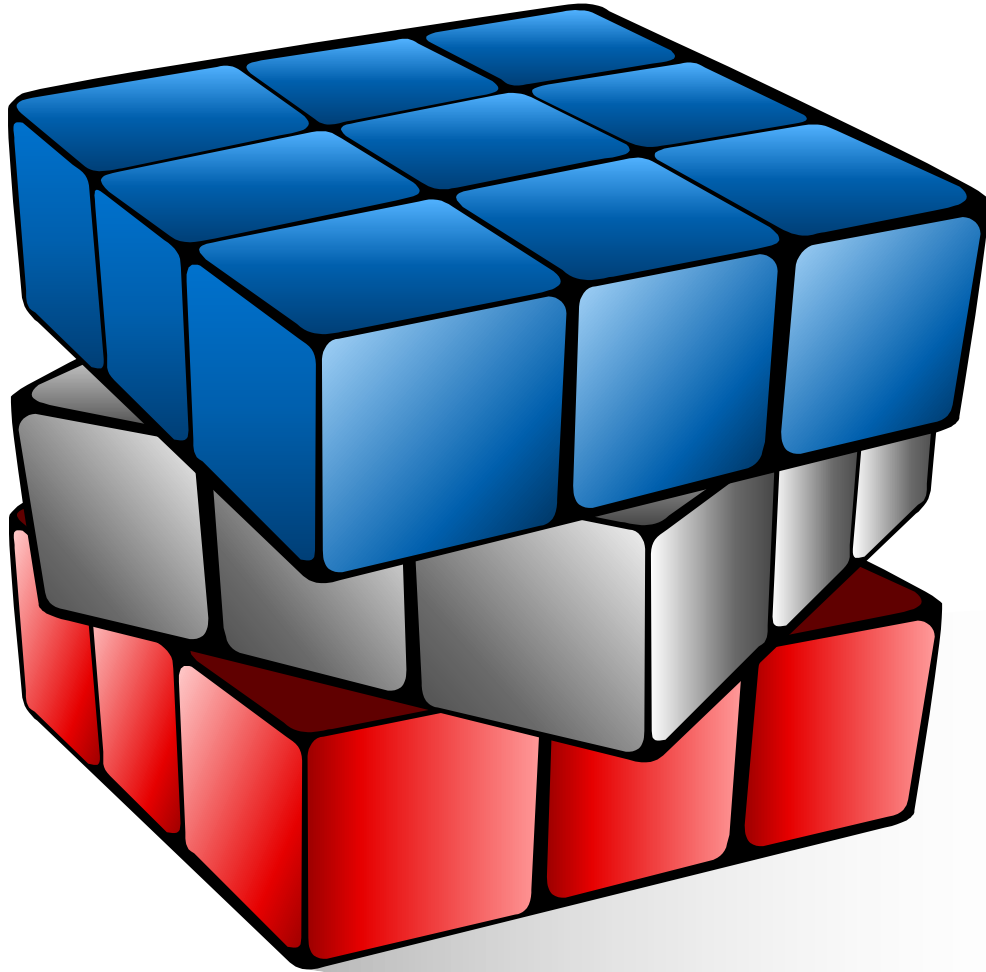
# PERFORMING LOAD TEST

## Perform Smoke Test

### > Add JMeter Plugin

Test plan > Add > Listener > jp@gc – Active Threads Over Time

Monitor thread using jp@gc – Active Threads Over Time



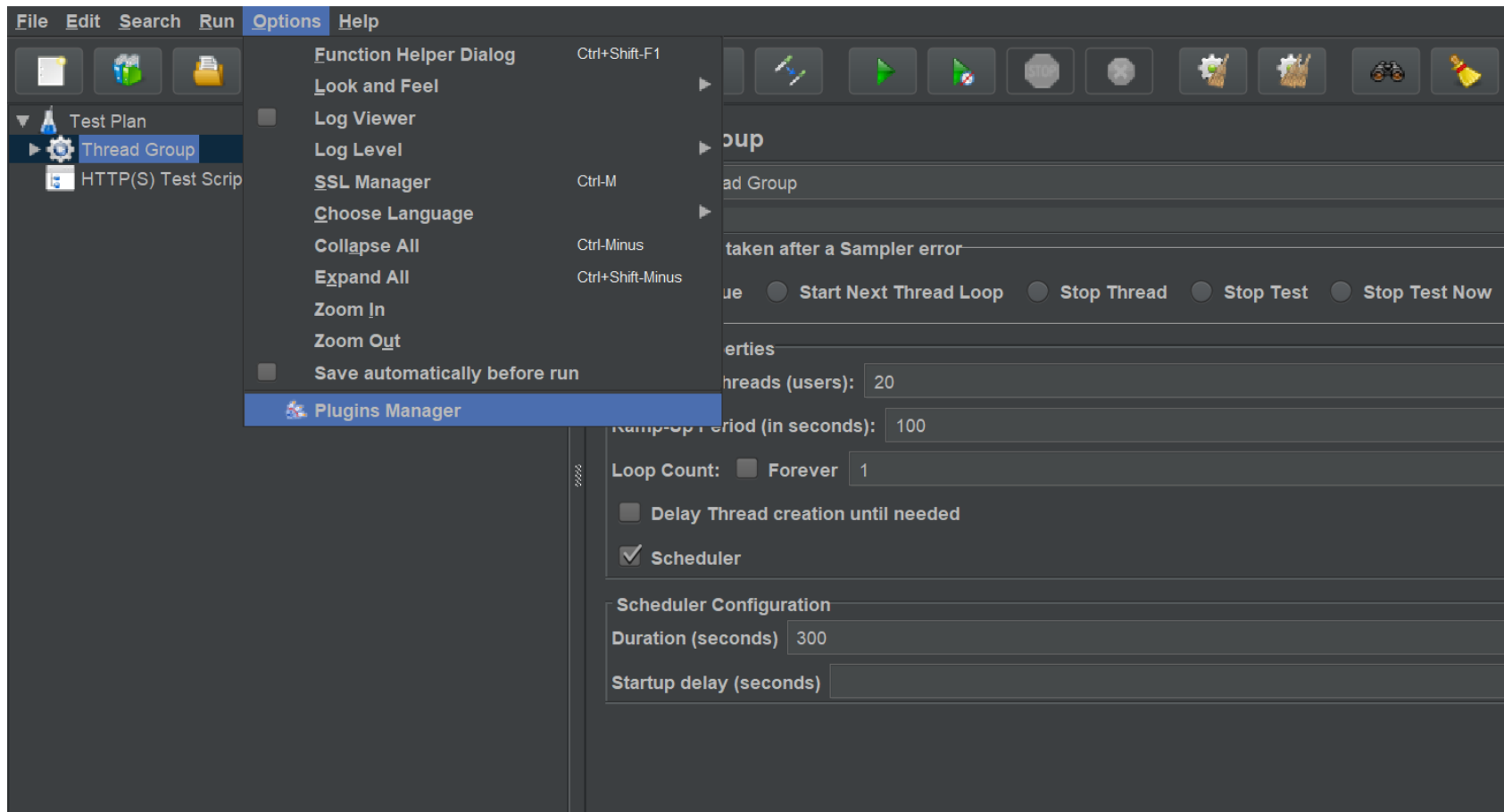
LET'S DO IT



**SELAMAT MENCUBA!!**

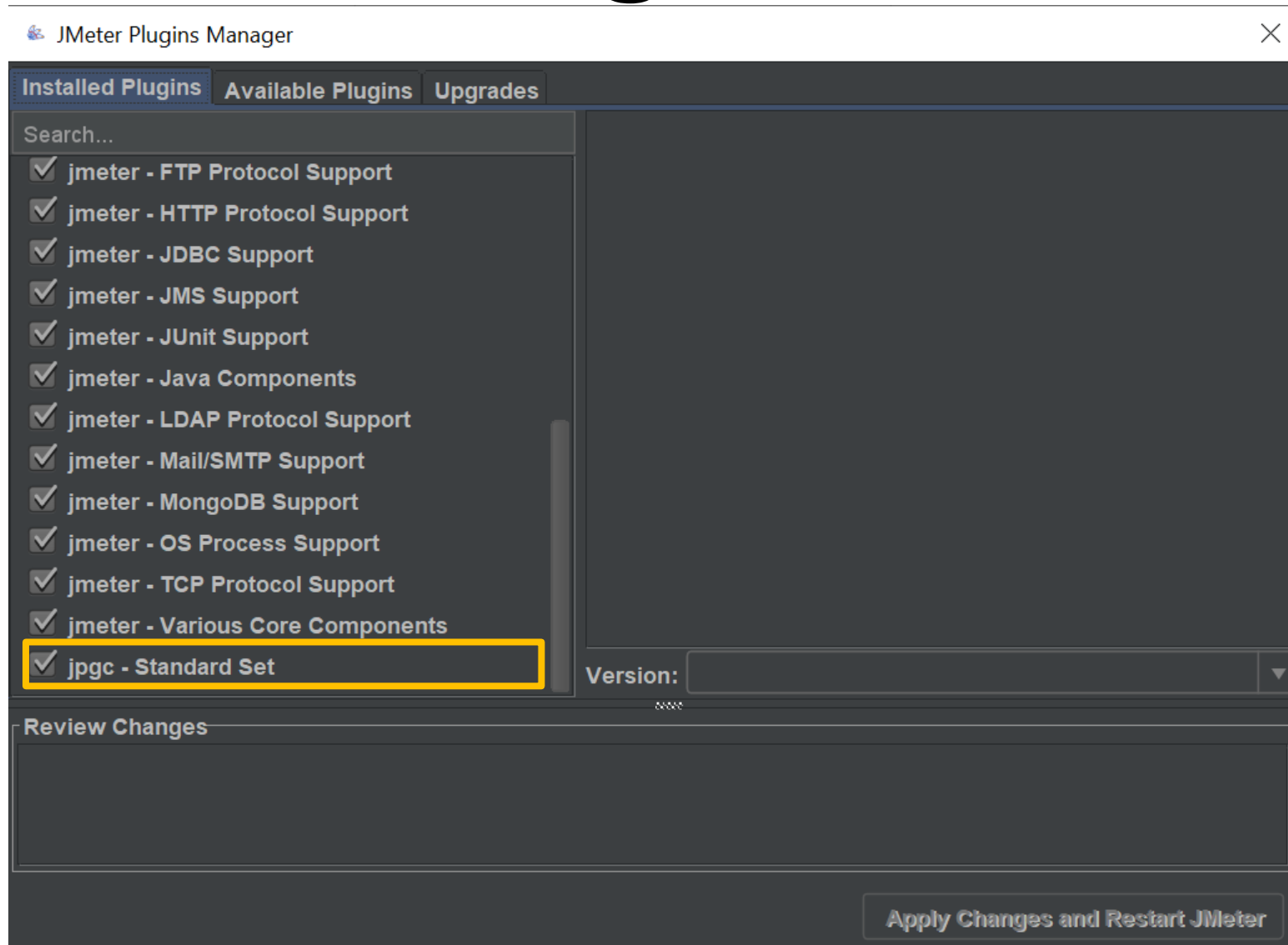
# JMeter Plugins Installation Guide

Download [plugins-manager.jar](#) and put it into `lib/ext` directory, then restart JMeter.





# JMeter Plugins Installation Guide



# JMeter Plugins Installation Guide

Available Plugin > select jpgc-Standard Set > Apply Changes and Restart JMeter

