

Software Engineering Task-10

Kavya Aggarwal

HU22CSEN0100288

What is J-Meter?

JMeter is a software that can perform load tests, performance oriented business (functional) tests, regression tests, etc., on different protocols or technologies.

Stefano Mazzocchi of the Apache Software Foundation was the original developer of JMeter. He wrote it primarily to test the performance of Apache JServ (now called Apache Tomcat project). Apache later redesigned JMeter to enhance the GUI and to add functional testing capabilities.

JMeter is a Java desktop application with a graphical interface that uses the Swing graphical API. It can therefore run on any environment / workstation that accepts a Java virtual machine, for example: Windows, Linux, Mac, etc.

The protocols supported by JMeter are:

- Web: HTTP, HTTPS sites 'web 1.0' web 2.0 (ajax, flex and flex-amf)
- Web Services: SOAP / XML-RPC
- Database via JDBC drivers
- Directory: LDAP
- Messaging Oriented service via JMS
- Service: POP3, IMAP, SMTP

- FTP Service

Types of Testing in JMeter

- Performance Test: This test sets the best possible performance expectation under a given configuration of infrastructure. It also highlights early in the testing process if any changes need to be made before the application goes into production.
- Load Test: This test is basically used for testing the system under the top load it was designed to operate under.
- Stress Test: This test is an attempt to break the system by overwhelming its resources.

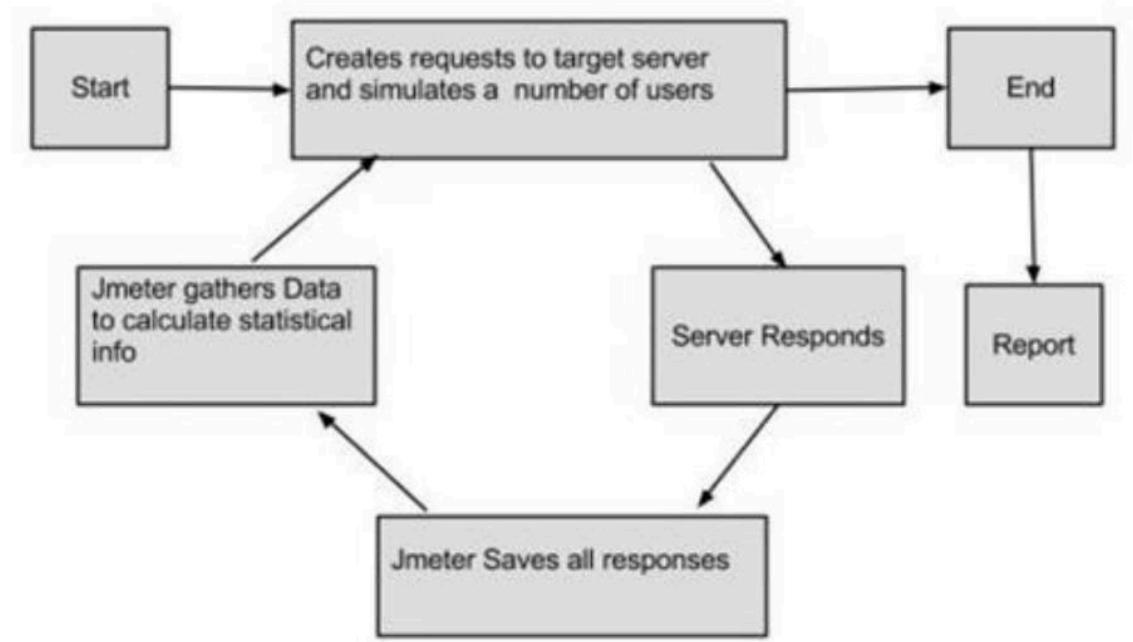
Features of JMeter

- Being an open-source software, it is freely available.
- It has a simple and intuitive GUI.
- JMeter can conduct load and performance tests for many different server types: Web - HTTP, HTTPS, SOAP, Database via JDBC, LDAP, JMS, Mail - POP3, etc.
- It is a platform-independent tool. On Linux/Unix, JMeter can be invoked by clicking on the JMeter shell script. On Windows, it can be invoked by starting the jmeter.bat file.
- It has full Swing and lightweight component support (precompiled JAR uses packages javax.swing.*).
- JMeter stores its test plans in XML format. It means you can generate a test plan using a text editor.

- Its full multithreading framework allows concurrent sampling by many threads and simultaneous sampling of different functions by separate thread groups.
- It is highly extensible.
- It can also be used to perform automated and functional testing of the applications.

How JMeter Works?

JMeter simulates a group of users sending requests to a target server, and returns statistics that show the performance/functionality of the target server/application via tables, graphs, etc.

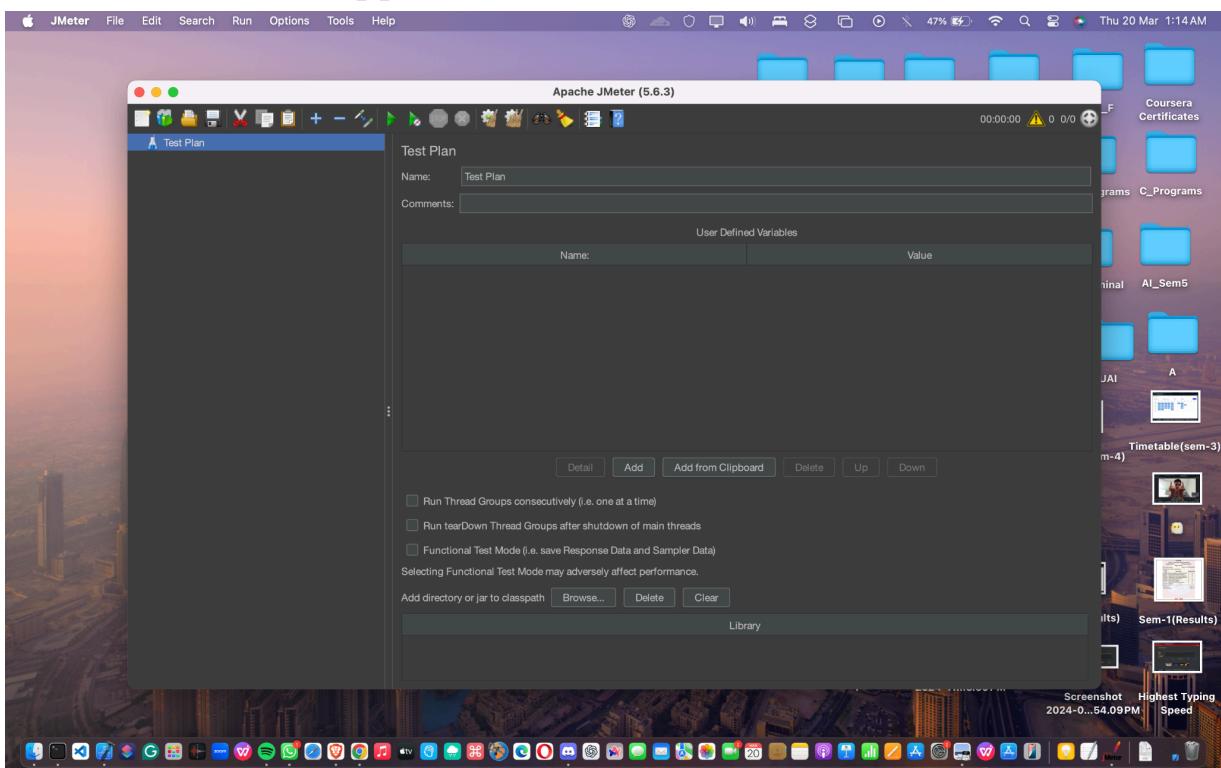


What is a Test Plan?

A Test Plan can be viewed as a container for running tests. It defines what to test and how to go about it. A complete test plan consists of one or more elements such as thread groups, logic controllers, sample-generating controllers, listeners, timers, assertions, and configuration elements. A test plan must have at least one thread group.

Writing a Test Plan?

Step 1: Start the JMeter Window Open the JMeter window by clicking /home/kundan/apache-jmeter-2.9/bin/jmeter.sh. The JMeter window will appear as shown below:



This is a plain and blank JMeter window without any additional elements added to it. It contains two nodes:

- **Test Plan node:** It is where the real test plan is kept.

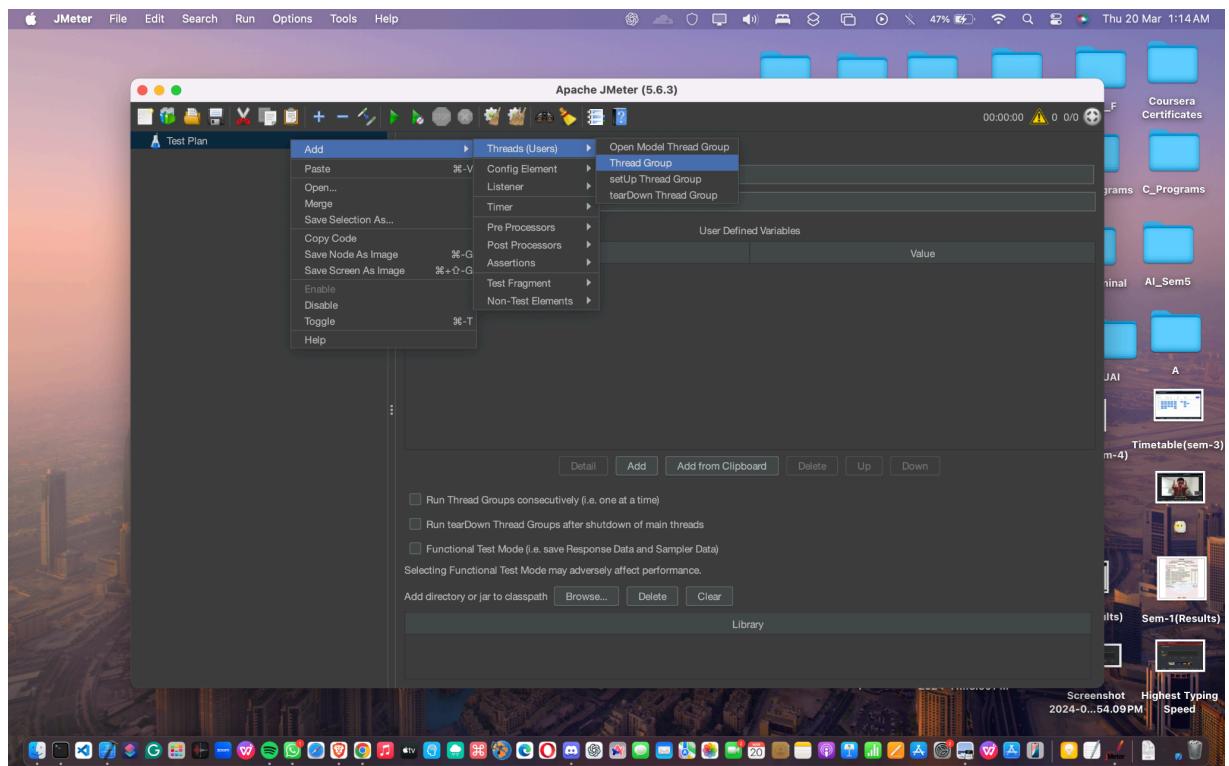
- **Workbench node:** It simply provides a place to temporarily store test elements while not in use, for copy/paste purposes. When you save your test plan, Workbench items are not saved with it.

Step 2: Add/Remove Elements

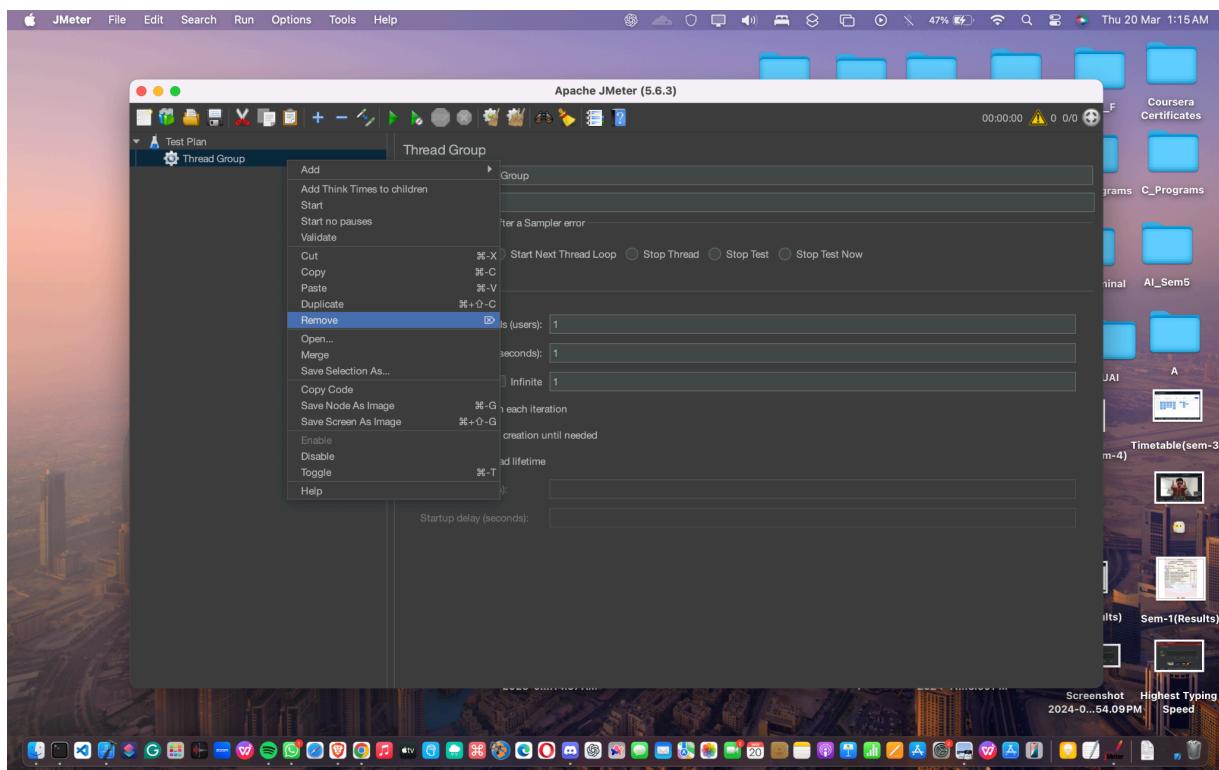
- Elements (which will be discussed in the next chapter Test Plan Elements) can be added to a test plan by right-clicking on the Test Plan node and choosing a new element from the "add" list.

Alternatively, you can load an element from a file and add it by choosing the "merge" or "open" option.

For example, let us add a Thread Group element to a Test Plan as shown below:



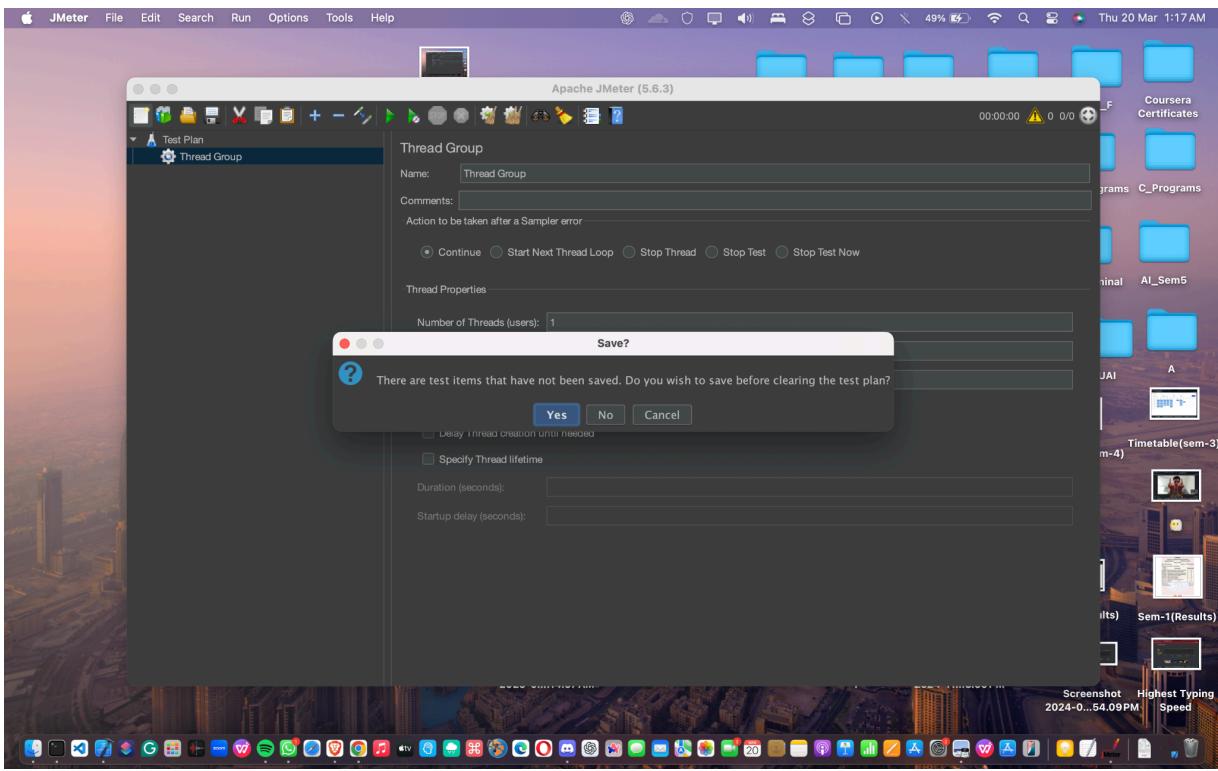
To remove an element, make sure the element is selected, rightclick on the element, and choose the "remove" option.



Step 3: Load and Save the Elements To load an element from file:

- Right-click on the existing tree element to which you want to add the loaded element.
- Select Merge.
- Choose the file where you saved the elements.
- JMeter will merge the elements into the tree.

By default, JMeter does not save the element, you need to explicitly save it.



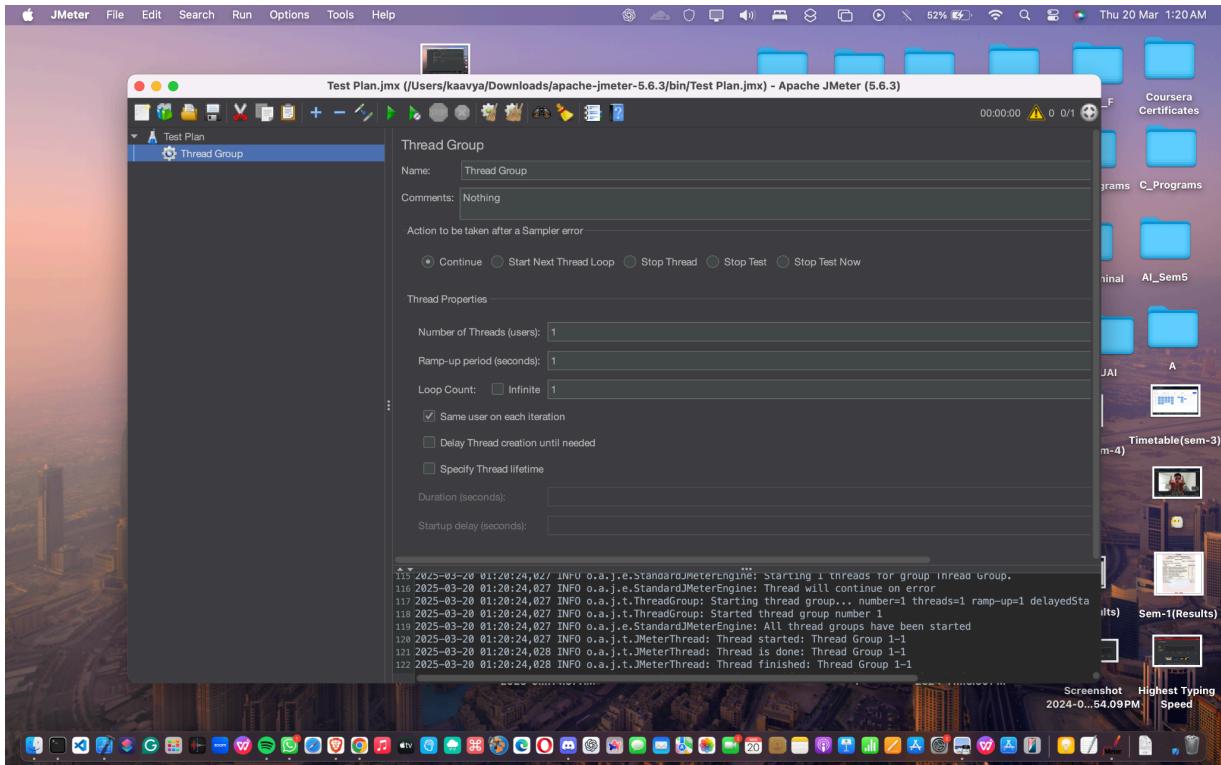
JMeter 15 To save a tree element:

- Right-click on the element.
- Choose the Save Selection As ... option.

JMeter will save the element selected, plus all the child elements beneath it. By default, JMeter doesn't save the elements, you need to explicitly save it as mentioned earlier.

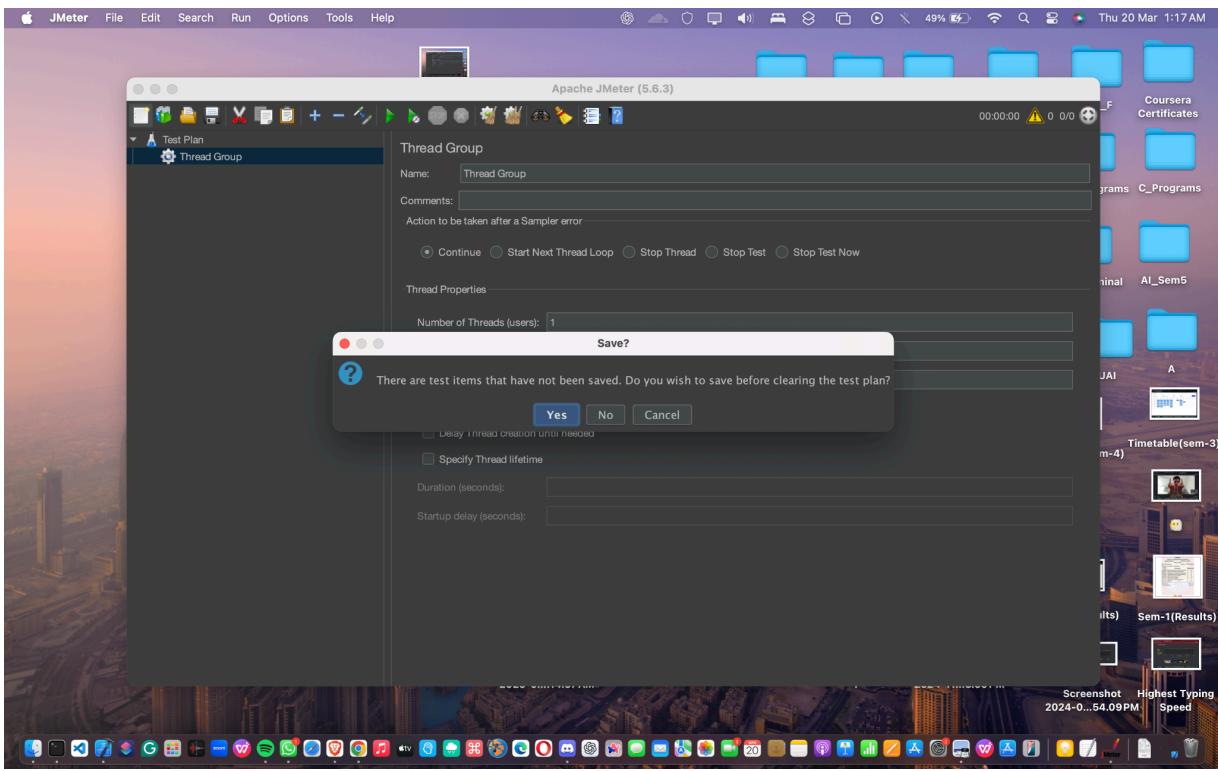
Step 4: Configure the Tree Elements

Any element in the Test Plan can be configured using the controls present in JMeter's righthand side frame. These controls allow you to configure the behavior of that particular test element. For example, the Thread Group can be configured for a number of users, ramp up periods, etc., as shown below:



Step 5: Save the Test Plan

You can save an entire Test Plan by using either Save or "Save Test Plan As ..." from the File menu.



Step 6: Run the Test Plan

You can run the Test Plan by clicking Start (Control + r) from the Run menu item. When JMeter starts running, it shows a small green box at the right-hand end of the section just under the menu bar.

The numbers to the left of the green box are the number of active threads / total number of threads. These only apply to a locally run test; they do not include any threads started on remote systems when using client-server mode.

Step 7: Stop the Test Plan

You can stop your test in two ways:

- Using Stop (Control + '.'). It stops the threads immediately if possible.
- Using Shutdown (Control + ','). It requests the threads to stop at the end of any current work.