Learn JMeter in 1 Day

By Krishna Rungta

Table Of Content

Chapter 9: Processor in JMeter: PreProcessor & PostProcessor

Chapter 10: Jmeter Distributed (Remote) Testing: Master Slave Configuration

Chapter 11: HTTP Proxy Server in JMeter: Record Example Script

Chapter 12: Best Practices for your Jmeter Tests

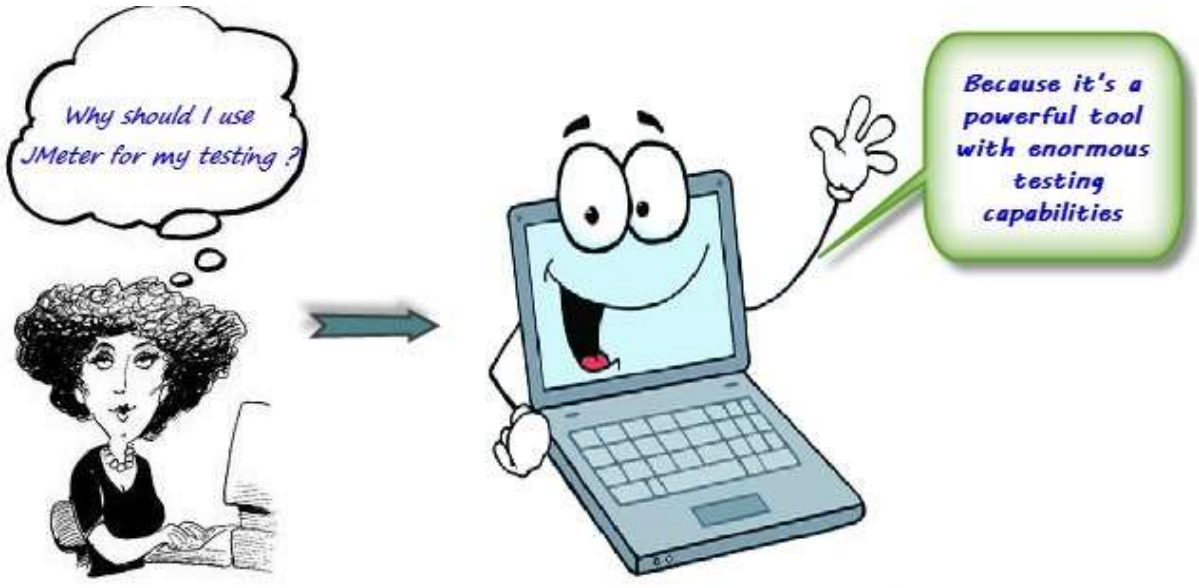
1. Limit the Number of Threads
2. Using a proxy server
3. Using variables
4. Reduce resource requirement
5. Check the JMeter logs
6. Erase the local path from CSV Data Set Config
7. Follow file naming convention

Chapter 1: What is JMeter? Introduction & Uses

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.

Why JMeter?



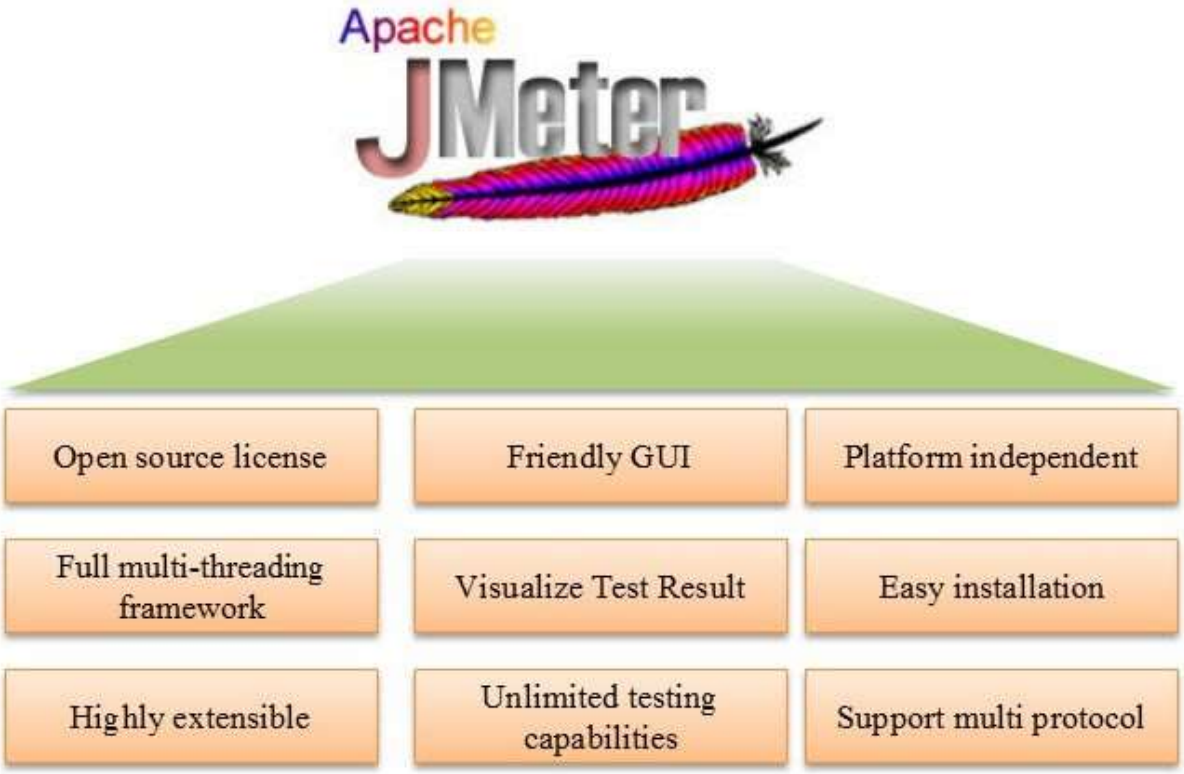
Have you ever **tested** a web server to know how efficiently it works? How many concurrent users can a web server handle?

Let say that one day, your boss asks you to do **performance testing** of www.google.com for 100 users. What would you do?

It's not feasible to arrange 100 people with PC and internet access simultaneously accessing google.com Think of the infrastructure requirement when you test for 10000 users (a small number for a site like google). Hence you need a software tool like JMeter that will simulate real-user behaviors and performance/load test your site.

JMeter Advantages

JMeter advantages are described as figure below



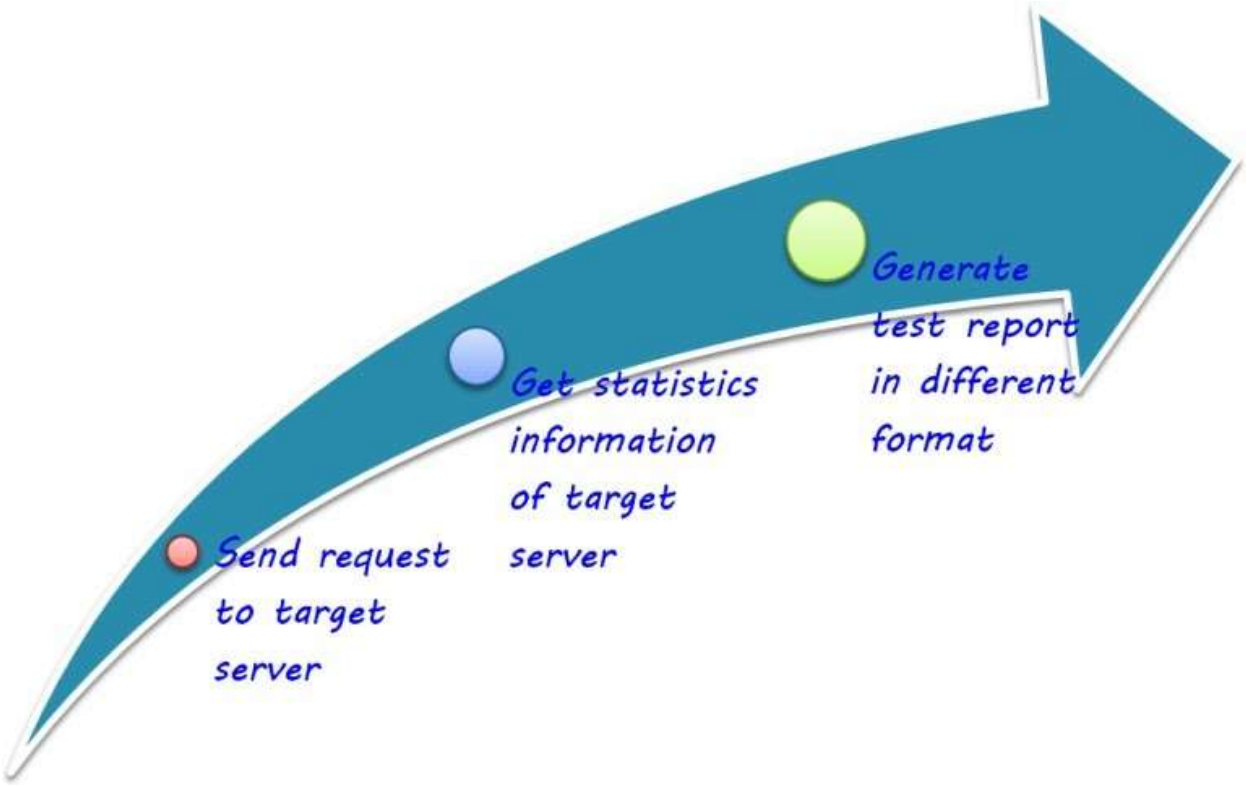
- **Open source license**: JMeter is totally free, allows developer use the source code for the development
- **Friendly GUI**: JMeter is extremely easy to use and doesn't take time to get familiar with it
- **Platform independent**: JMeter is 100% pure Java desktop application. So it can run on multiple platforms
- **Full multithreading framework**. JMeter allows concurrent and simultaneous sampling of different functions by a separate thread group
- **Visualize Test Result:** Test result can be displayed in a different format such as chart, table, tree and log file
- **Easy installation**: You just copy and run the *.bat file to run JMeter. No installation needed.

- **Highly Extensible**: You can write your own tests. JMeter also supports visualization plugins allow you to extend your testing
- **Multiple testing strategy**: JMeter supports many testing strategies such as Load Testing, Distributed Testing, and Functional Testing.
- **Simulation**: JMeter can simulate multiple users with concurrent threads, create a heavy load against web application under test **Support multi-**
- **protocol**: JMeter does not only support web application testing but also evaluate database server performance. All basic protocols such as HTTP, JDBC, LDAP, SOAP, JMS, and FTP are supported by JMeter
- **Record & Playback** - **Record** the user activity on the browser and simulate them in a web application using JMeter
- **Script Test**: Jmeter can be integrated with Bean Shell & Selenium for automated testing.

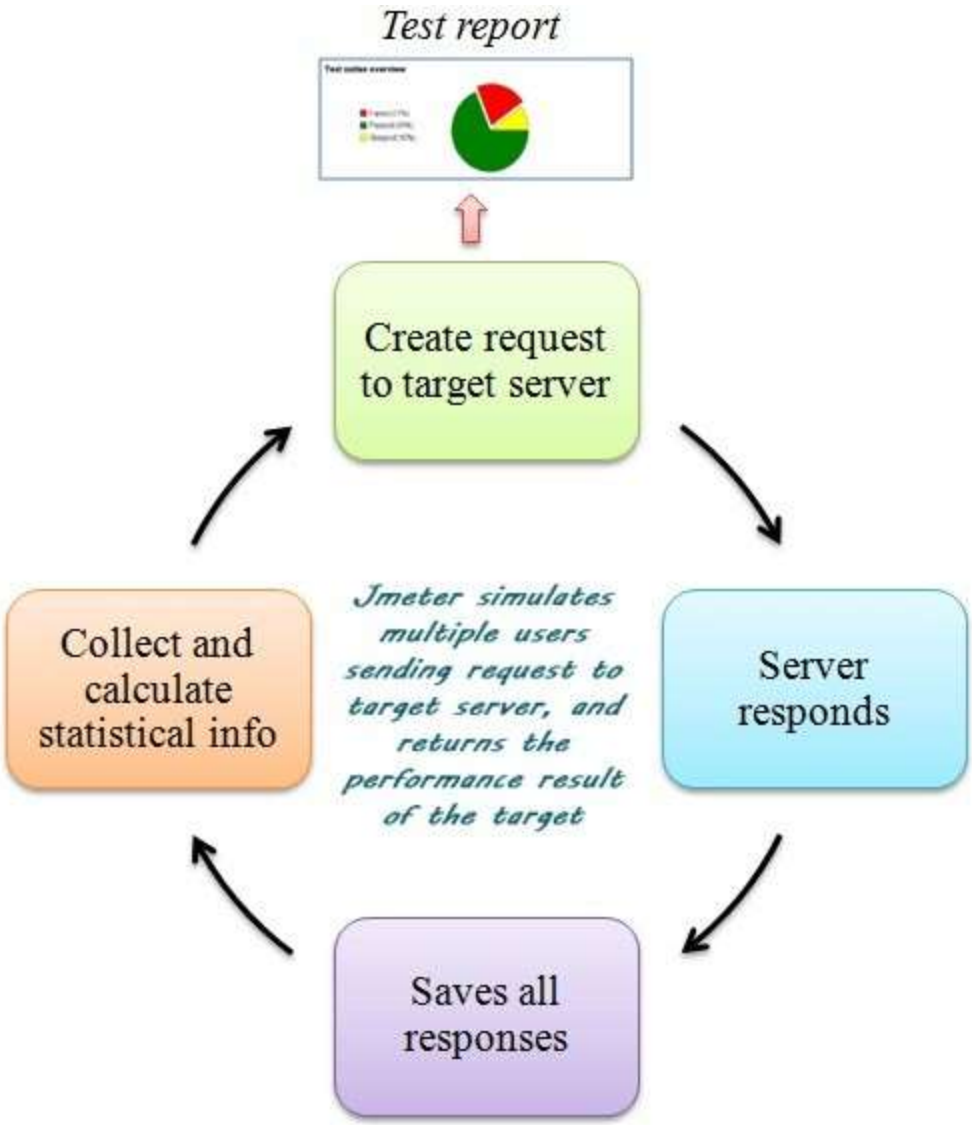
How does JMeter work?

The basic workflow of JMeter as shown in the figures below

JMeter simulates a group of users sending requests to a target server, and return statistics information of target server through graphical diagrams



The completed workflow of JMeter as shown in the figure below



Jmeter Version History

Version	Description
1	first official release
2.13	Java 6+
3	Java 7+
3.1	Java 7+
3.2	Java 8+
3.3	Java 8
4	Java 8 / 9

Chapter 2: How to Download & Install Apache JMeter in easy steps

Operating system Support for JMeter

JMeter is a **pure Java** application and should run correctly on any system that has a compatible Java implementation.

Here is the list of an operating system compatible with JMeter

- Linux
- Windows
- Mac OS
- Ubuntu

Steps to Install JMeter

Step 1) Install Java

Because JMeter is pure Java desktop application, it requires a fully compliant JVM 6 or higher. You can download and install the latest version of Java SE Development Kit. Download Java Platform (JDK)

After installation is finished, you can use the following procedure to check whether Java JDK is installed successfully in your system

- In Window/Linux, go to **Terminal**
- Enter command **java -version**

If the Java runtime environment is installed successfully, you will see the output as the figure below

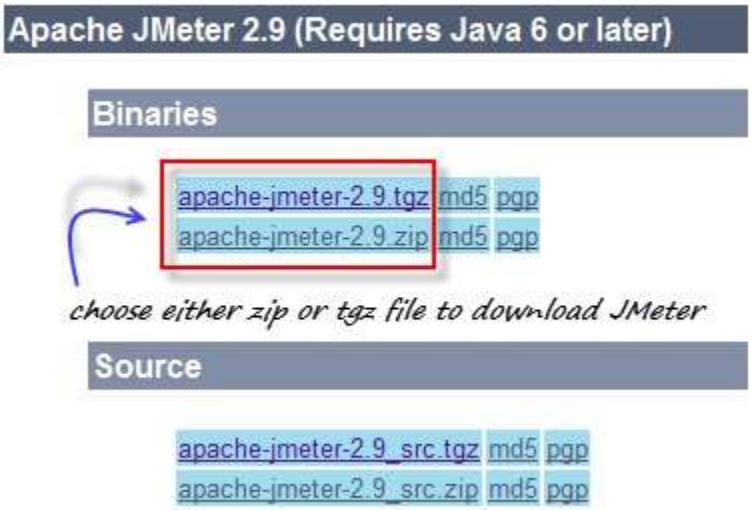
If nothing displays, please re-install Java SE runtime environment

Please see the link for details instructions
https://www.guru99.com/install-java.html

Step 2) Download Jmeter

As of this writing, the latest version of JMeter is **Apache JMeter 4.2**. You can download it here But this tutorial demos installation of version 2.9, the install process remains the same.

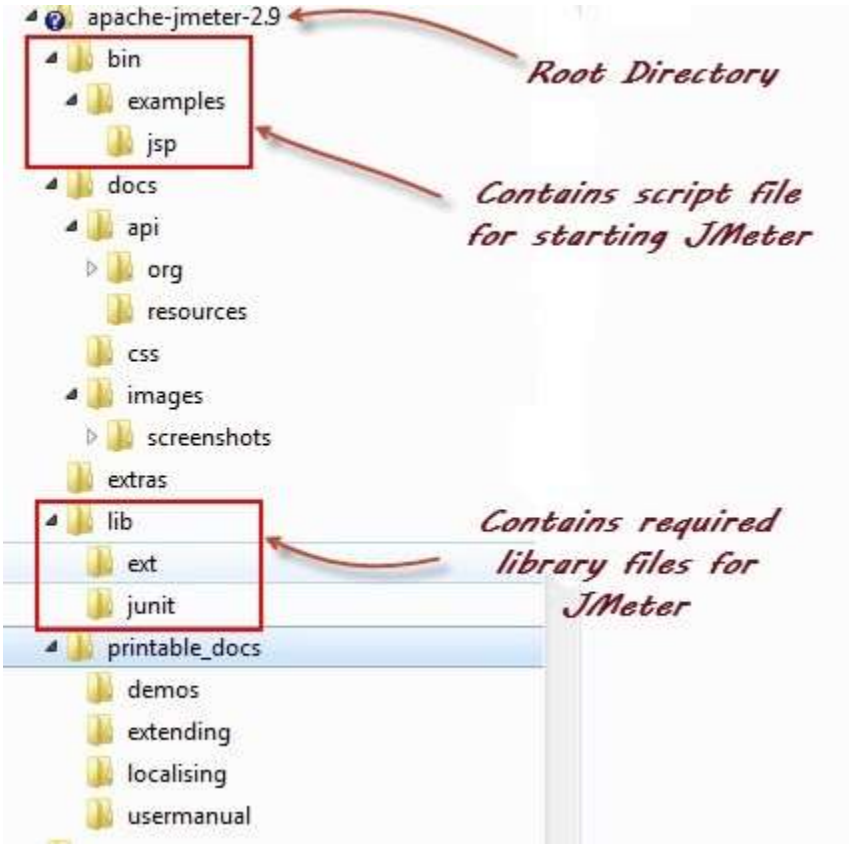
Choose the Binaries file (either zip or tgz) to download as shown in the figure below



Step 3) Installation

Installation of JMeter is extremely easy and simple. You simply unzip the zip/tar file into the directory where you want JMeter to be installed. There is no tedious installation screen to deal with! Simply unzip and you are done!

Once the unzipping is done installation directory structure should look like as figure below



Given below is the description of the JMeter directories and its importance JMeter directory contains many files and directory

- /**bin**: Contains JMeter script file for starting JMeter
- /**docs**: JMeter documentation files
- **/extras**: ant related extra files
- /**lib**/: Contains the required Java library for JMeter
- **/lib/ext**: contains the core jar files for JMeter and the protocols
- **/lib/junit**: Junit library used for JMeter
- /**printable_docs**:

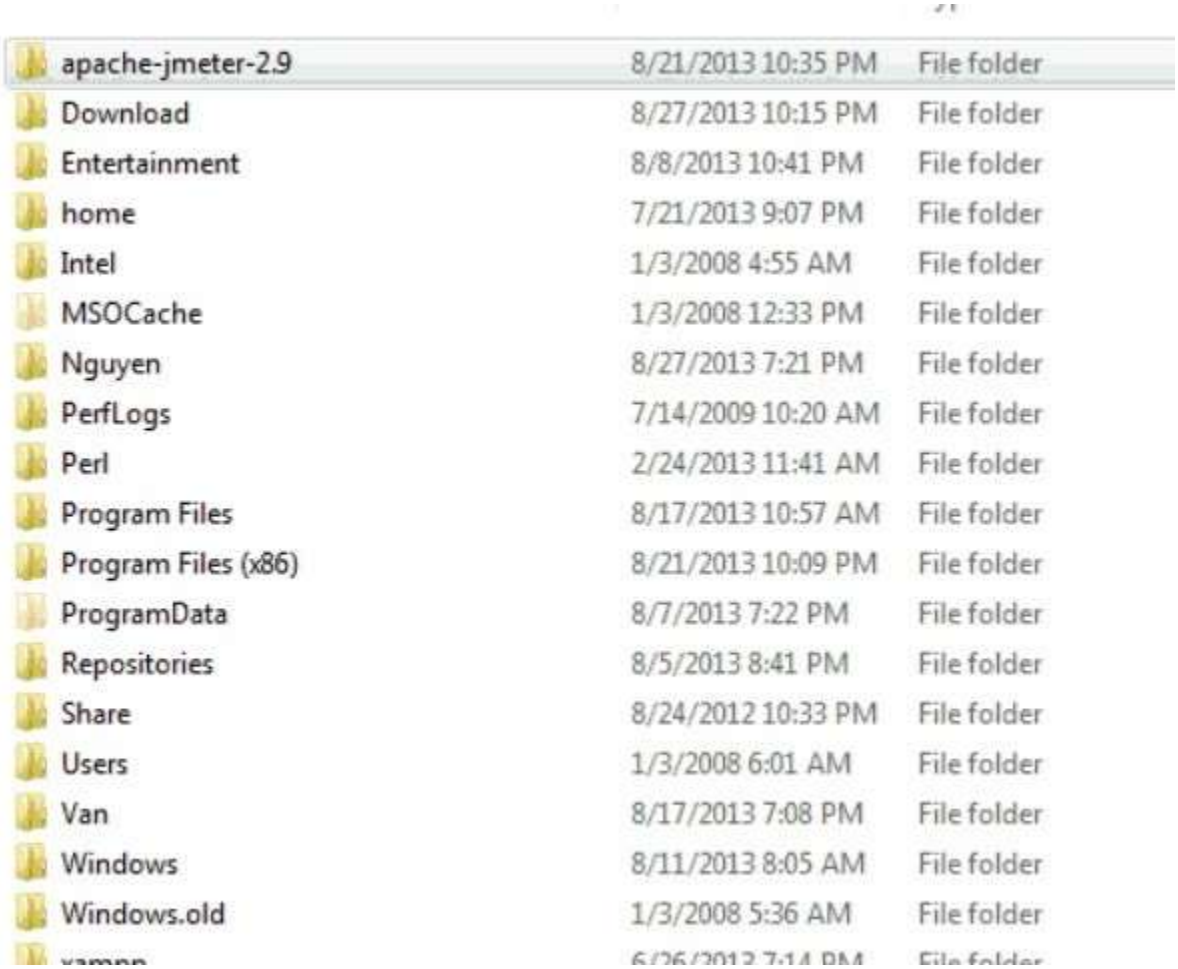
Step 4) Launch JMeter

You can start JMeter in 3 modes

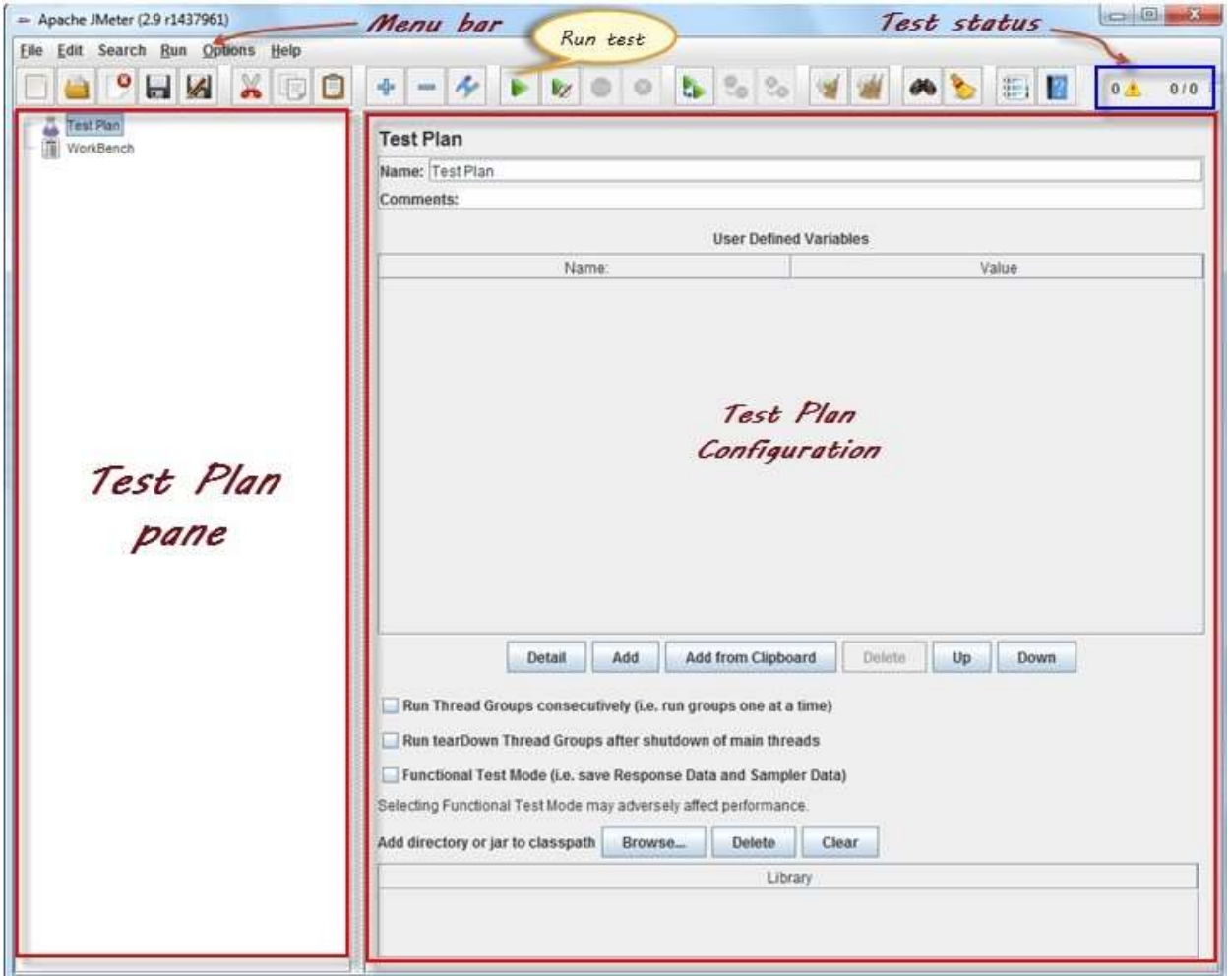
- GUI Mode
- Server Mode
- Command Line Mode

Start JMeter in GUI Mode

If you are using Window, just run the file **/bin/jmeter.bat** to start JMeter in GUI mode as shown below



The following figure annotates the various components in the JMeter GUI



How to run JMeter in Non-GUI Mode

Start JMeter in Server Mode

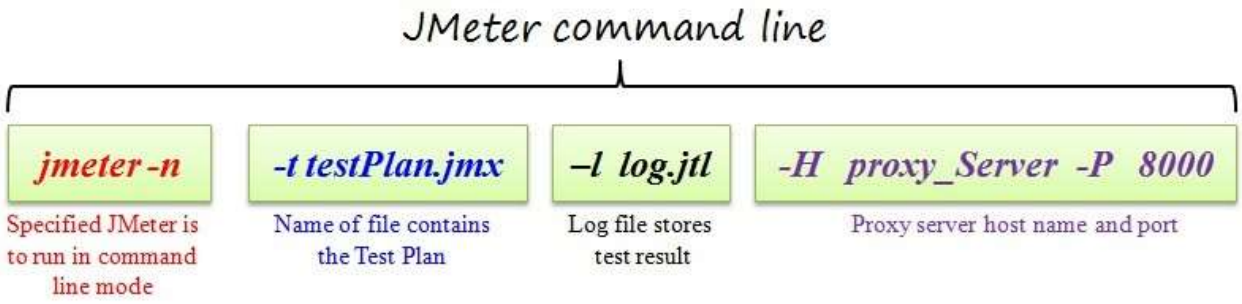
Server mode is used for **distributed** testing. This Testing works as **a client-server** model. In this model, JMeter runs on a server computer in **server** mode. On a client computer, JMeter runs in **GUI** mode.

To start the server mode, you run the bat file bin**\jmeter-server.bat**

as below figure

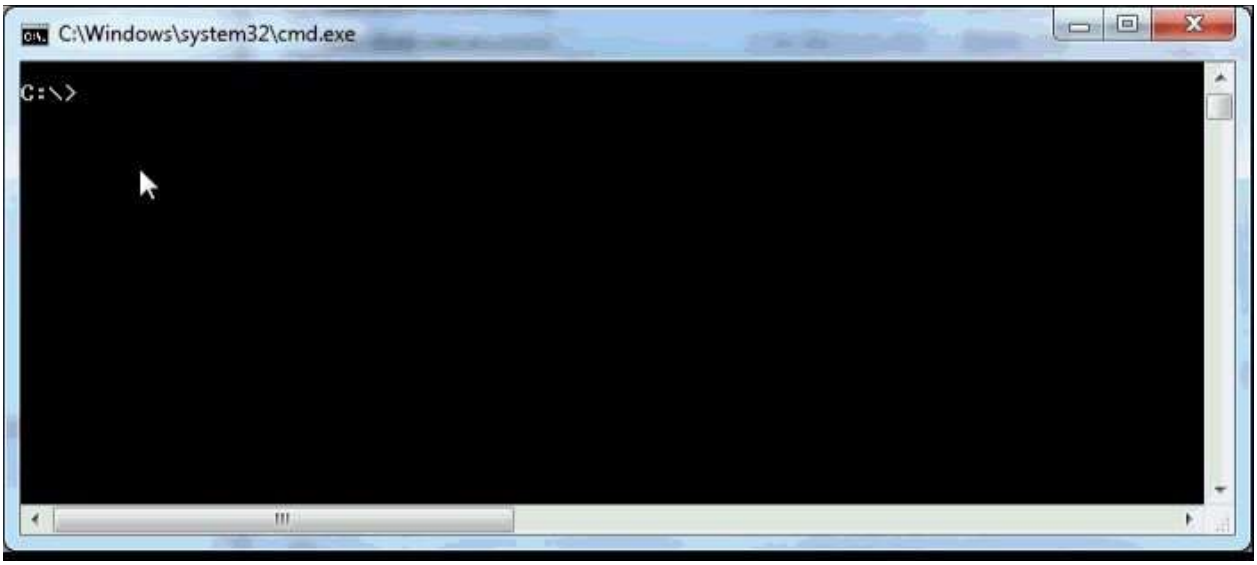
Start JMeter in command line mode

JMeter in GUI mode consumes much computer memory. For saving the resource, you may choose to run JMeter without the GUI. To do so, use the following command options



This is a command line example

```
$jmeter -n -t testPlan.jmx - l log.jtl -H 127.0.0.1 -P 8000
```



Additional Packages

Based on your requirement, you will need one or more optional packages listed below.

- ***Java Compiler***

Java Compiler allows developers to build JMeter source code and other JMeter plugins

- ***SAX XML parser***

SAX is the Simple API for XML, originally a Java-only API. You can use SAX XML parser as an alternative to XML parser in JMeter

- ***Email Support***

JMeter has extensive Email capabilities. It can send email based on test results and has a POP3(S)/IMAP(S) sampler. It also has an SMTP sampler.

- ***JDBC driver***

If you want to test database server, you have to install JDBC driver

Use JMeter in Linux

- Using JMeter in Linux is the same as in Window; you simply run the following shell script.
- Run the script file **jmeter** (This file has no extension)- run JMeter (in GUI mode by default).
- Run the script file **jmeter-server** - start JMeter in server mode (calls JMeter script with appropriate parameters)
- **jmeter.sh** - very basic JMeter script with no JVM options specified.
- **mirror-server.sh** - runs the JMeter Mirror Server in non-GUI mode
- **shutdown.sh** - Run the Shutdown client to stop a non-GUI instance gracefully
- **stoptest.sh** - Run the Shutdown client to stop a non-GUI instance abruptly