

Introduction to Jmeter

- Pure Java open source software
- Designed to load test functional behaviour and measure performance
- Can use to analyze and measure the performance of web application or variety of services
- This was originally used for web application testing, now it is used for functional test ,database server test and API testing.

Advantages of Jmeter

- Open source license
- Friendly GUI
- Platform independent
- Full multithreading framework
- Visualize test result
- Easy installation
- Highly extensible
- Unlimited testing capabilities
- Support multi protocol

Installation

Prerequisite: Install Java

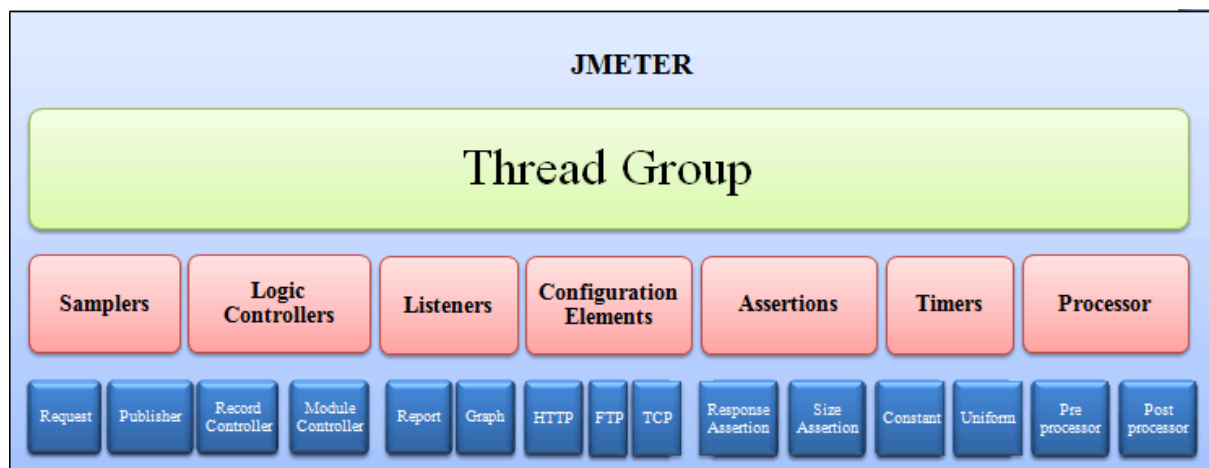
- Go to https://jmeter.apache.org/download_jmeter.cgi
- Download the zip file in binaries
- Extract the zip file and copy it to C or D drive
- Run the jmeter.bat file

Elements in Jmeter

The different components of JMeter are called Elements. Each element is designed for a specific purpose.

Under Thread group we can see:

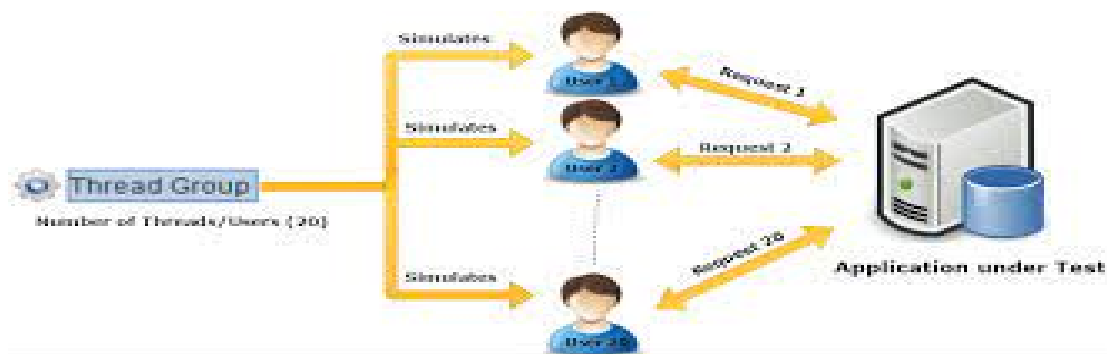
1. Samplers
2. Logic Controllers
3. Listeners
4. Configure Elements
5. Assertion
6. Timers
7. Processor



Thread Group

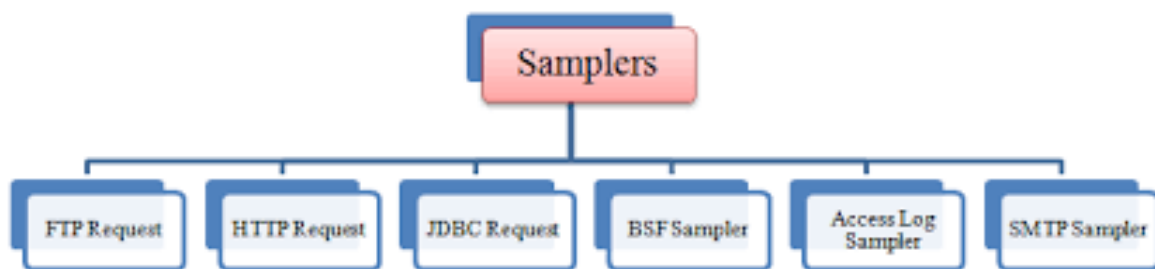
- Thread group is a collection of threads.
- Each thread represents one user using the application under test
- Basically, each thread simulates one real user request to the server
- The controls for a thread group allow you to set the number of threads for each group

Example: If you set the number of threads as 100; LMeter will create and simulate 100 user requests to the server under test



Samplers

- Samplers are different type of requests send by thread group
- The user request could be FTP request, HTTP request, JDBC request..etc.



Listeners

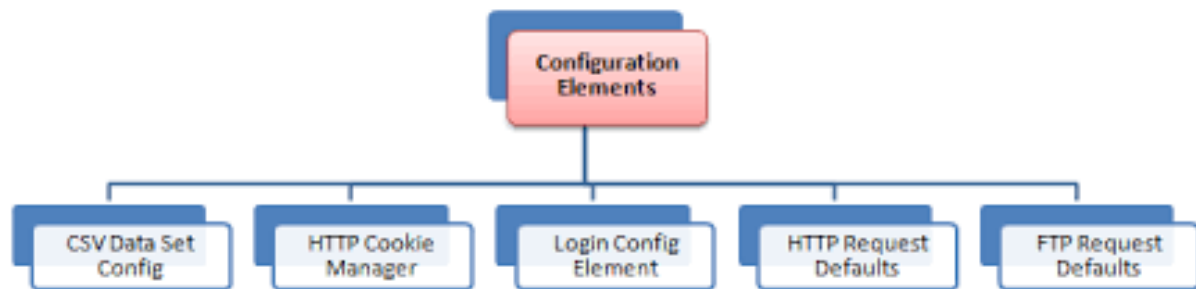
- Listeners shows the results of the test execution
- They can show results in a different format such as a tree, table, graph or log file
- **Graph result** listeners display the server response times on a graph
- **View result** tree show results of the user request in basic HTML format
- **Table result** show summary of a test result in a table format
- **Log** show summary of a test results in the text file



Configure

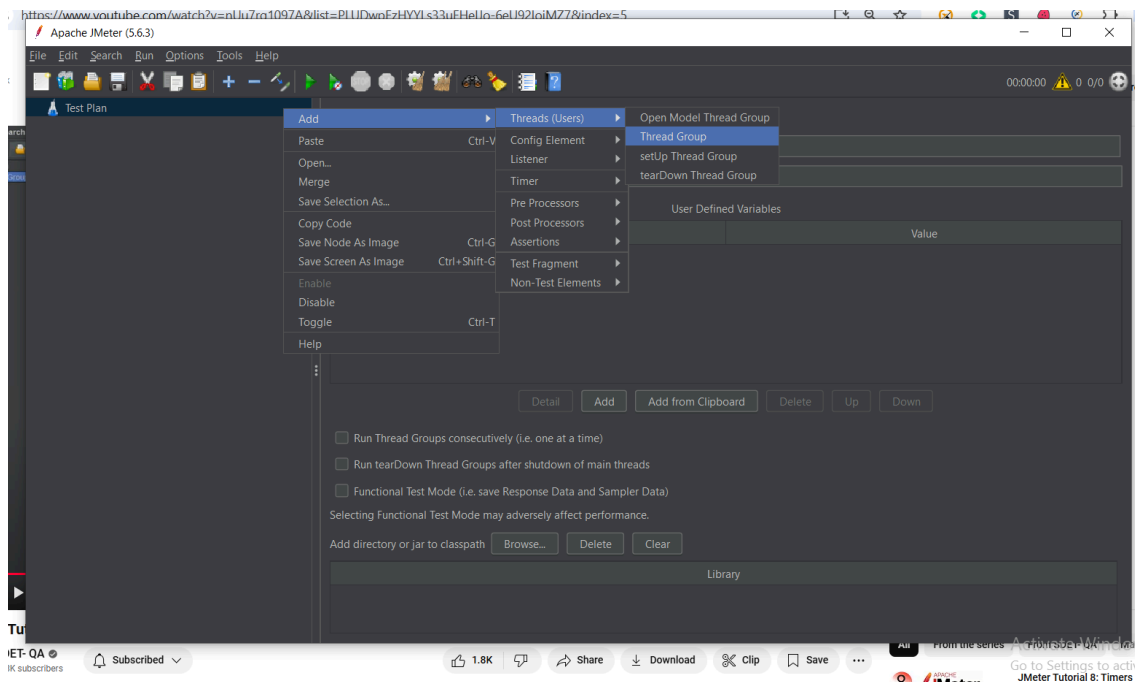
- Set up defaults and variables for later use by samplers

Commonly used configuration elements in Jmeter;

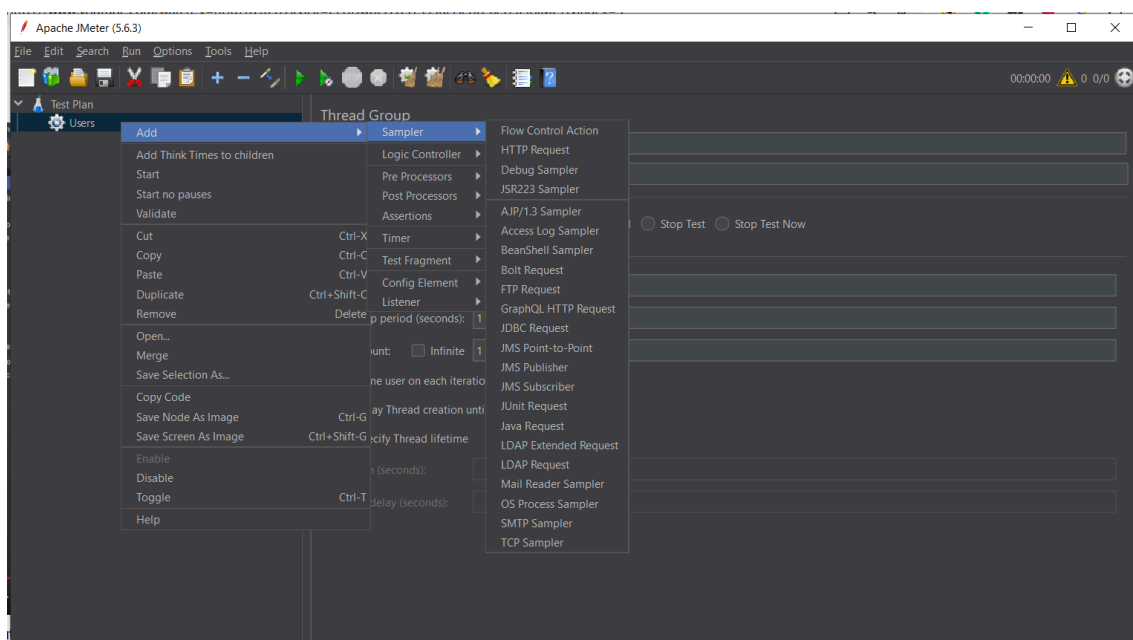


Create First Jmeter Test

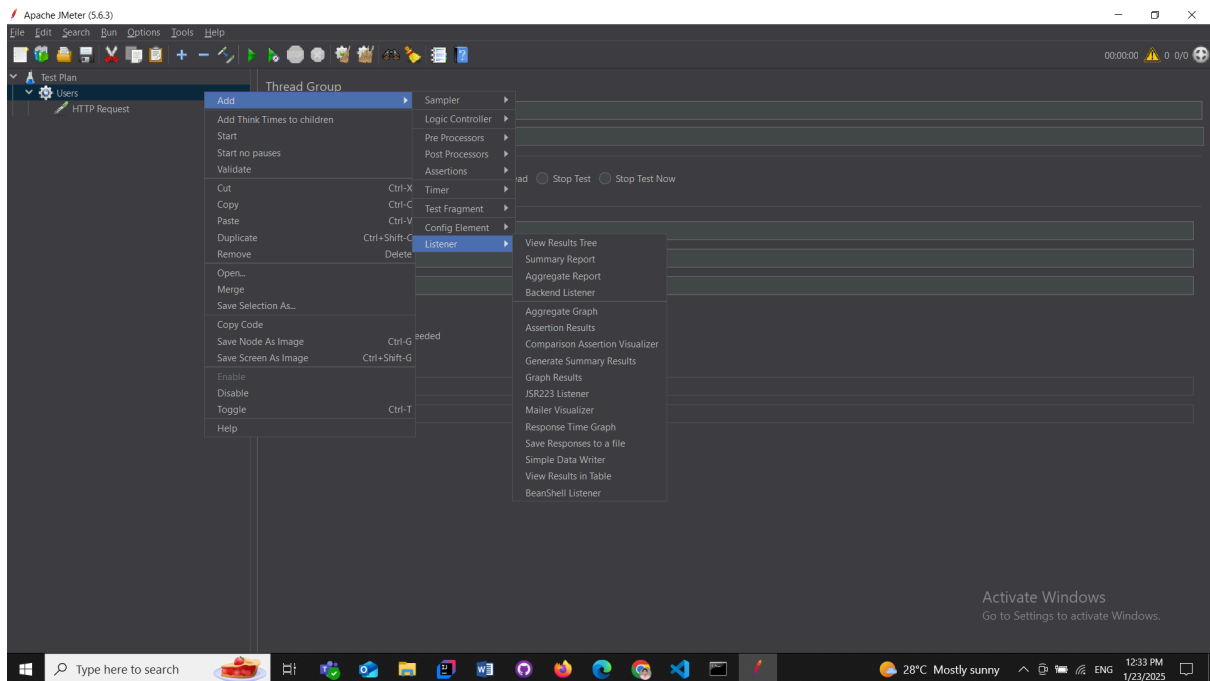
1. Start Jmeter
2. Create a TestPlan
3. Create a Thread Group (Users)



4. Add a Sampler (HTTP)



5. Add Listeners
6. Run Test Plan
7. Save Test Plan



Assertions in JMeter

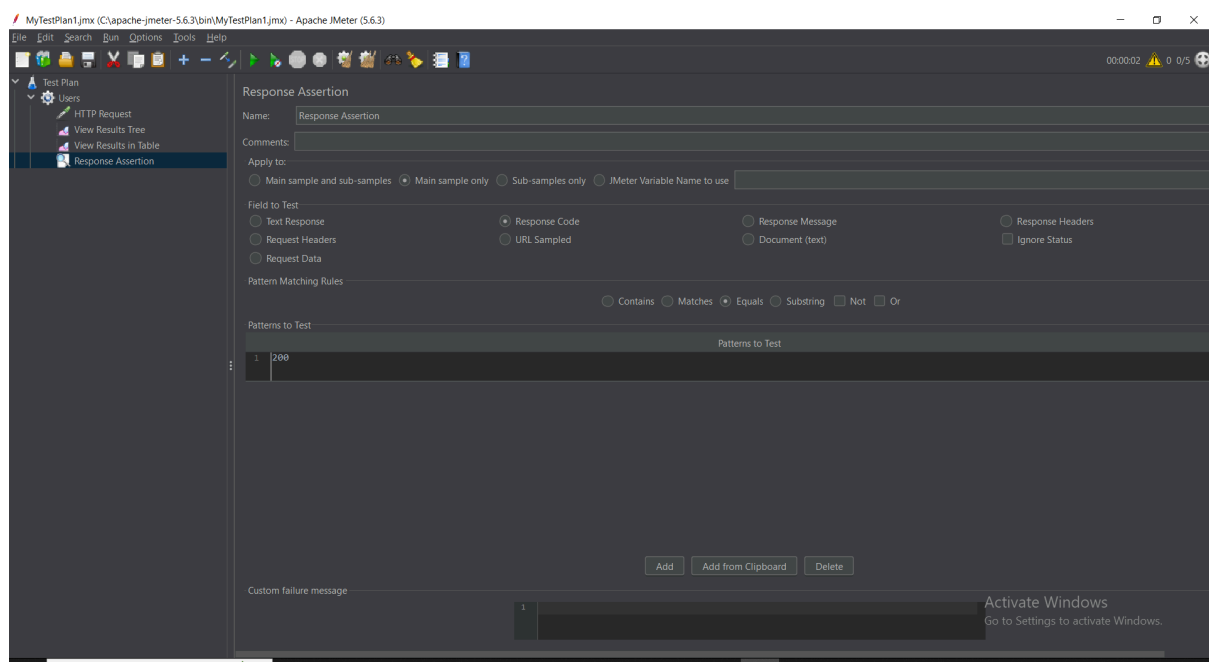
1. Response Assertion:
 - Checks for specific text, status code, or response time.
 - Most commonly used assertion.
2. JSON Assertion:
 - Validates JSON responses using JSONPath expressions.
3. Size Assertion:
 - Validates the size of the response (e.g., number of bytes).
4. Duration Assertion:
 - Validates that the response time is within a specified limit.
5. XPath Assertion:
 - Validates XML responses using XPath expressions.
6. BeanShell Assertion:
 - Allows custom validation using Java or BeanShell scripting.

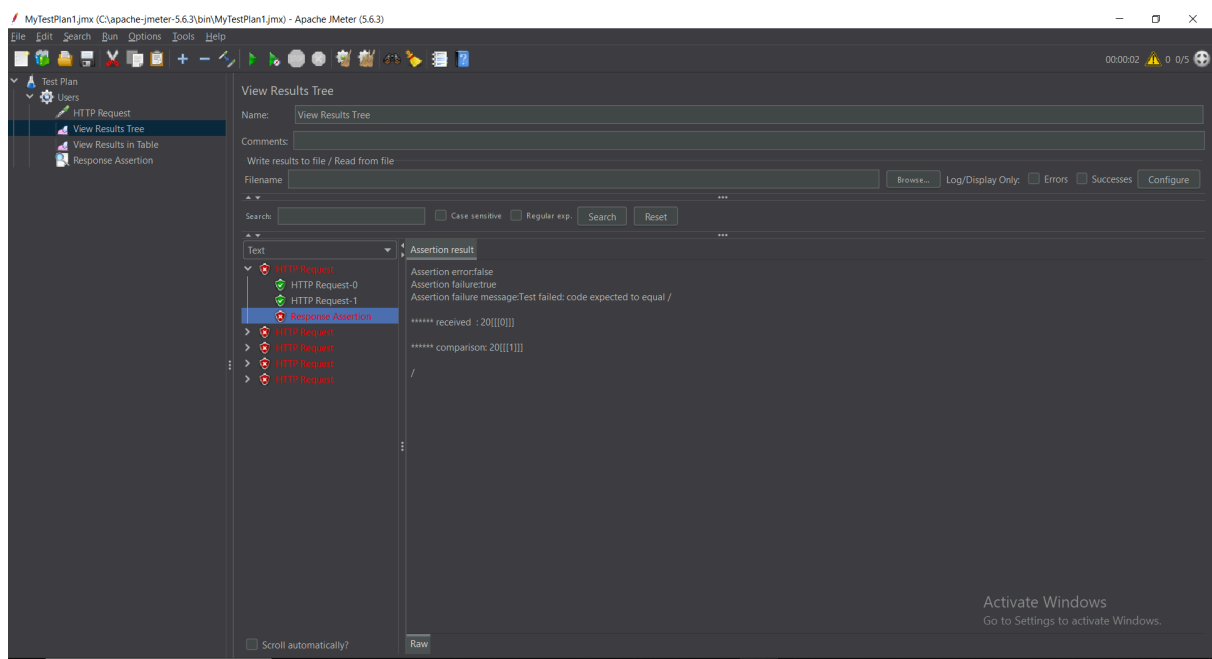
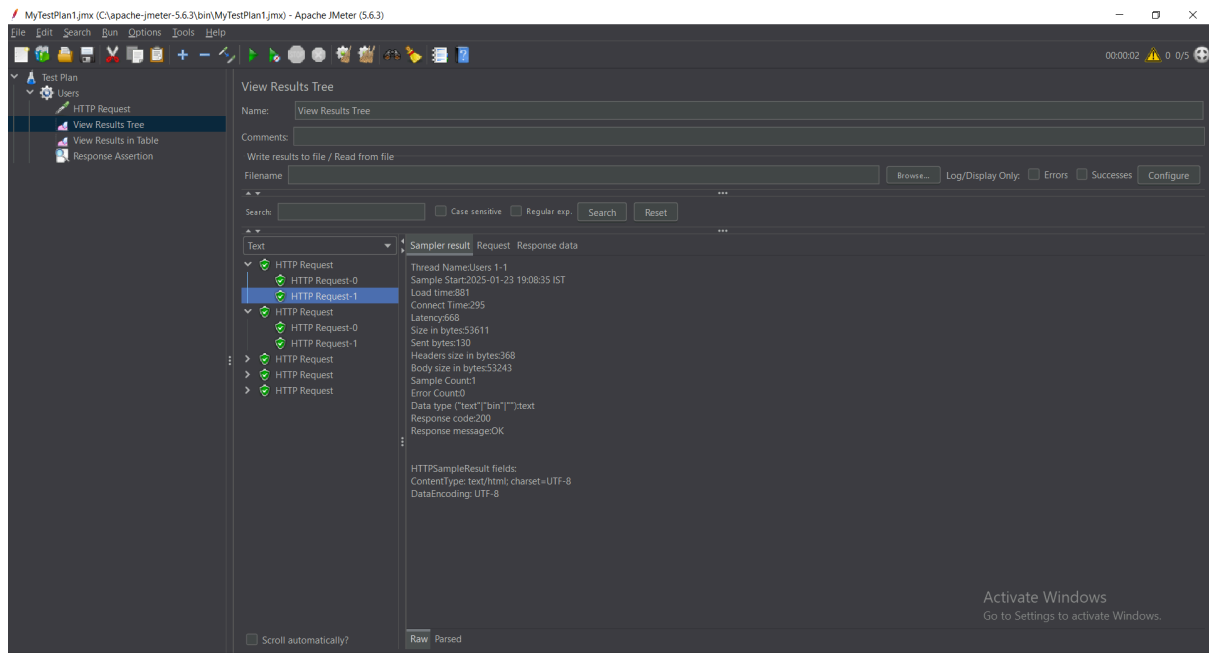
How to Add an Assertion

1. Select a Sampler (e.g., HTTP Request).
2. Right-click on the Sampler → Add → Assertions → Choose the type of assertion (e.g., Response Assertion).

Add a Response Assertion:

- Right-click on the Sampler → Add → Assertions → Response Assertion.
- Check for specific text or status codes in the response.

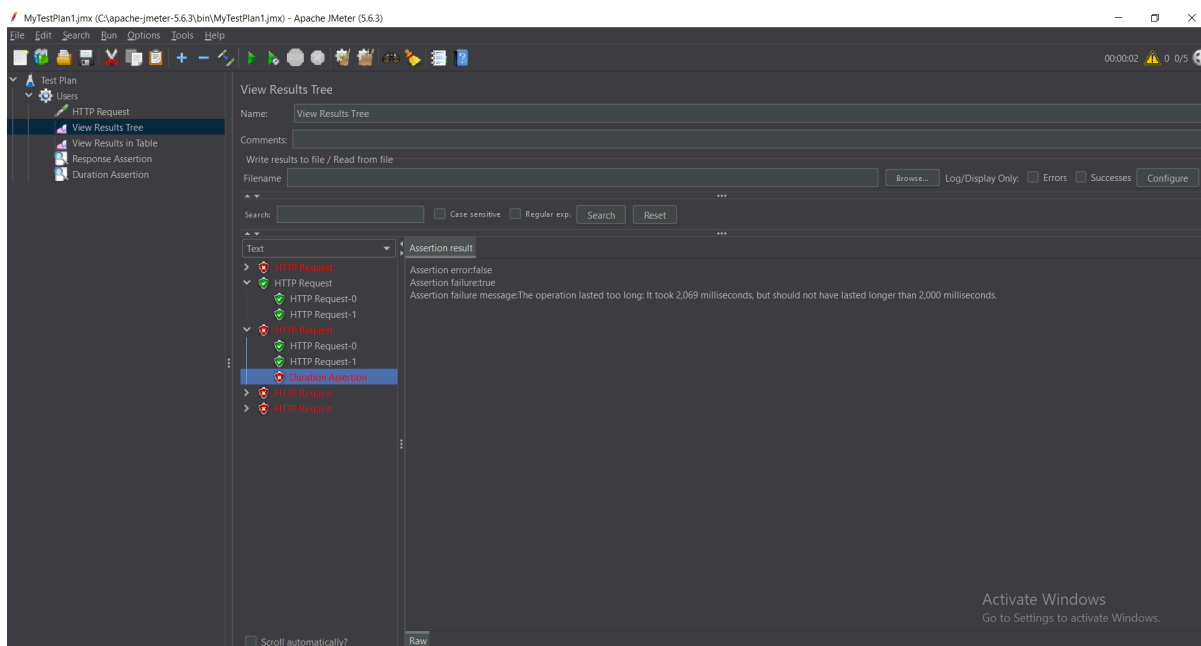
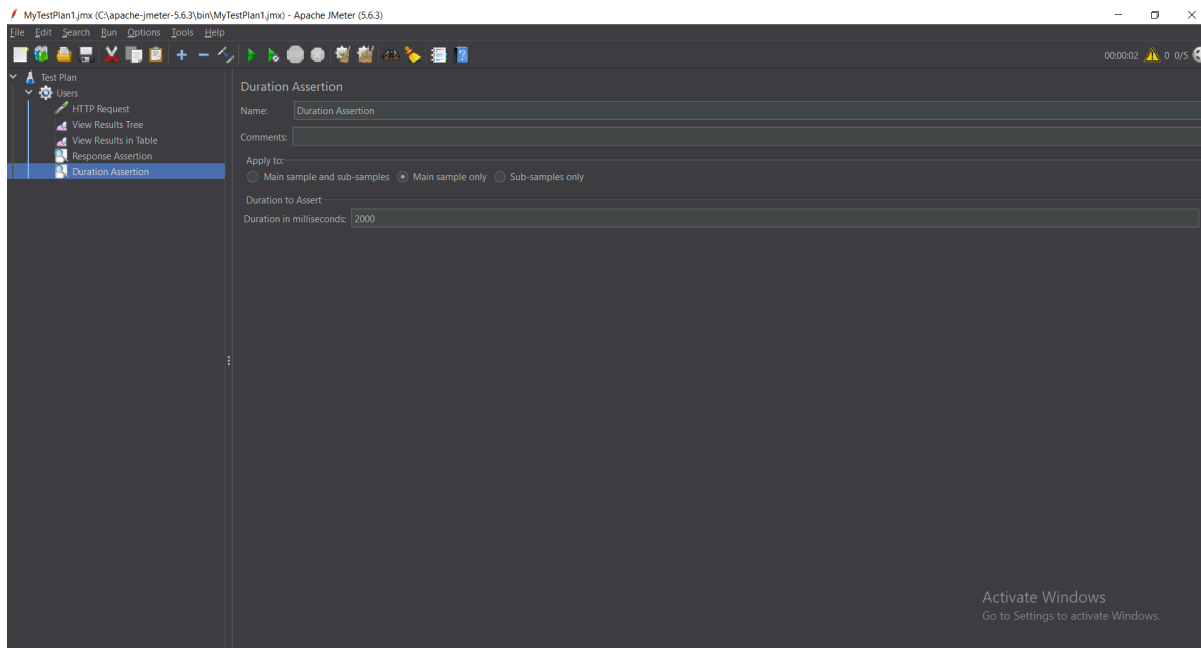




Validate Response Time

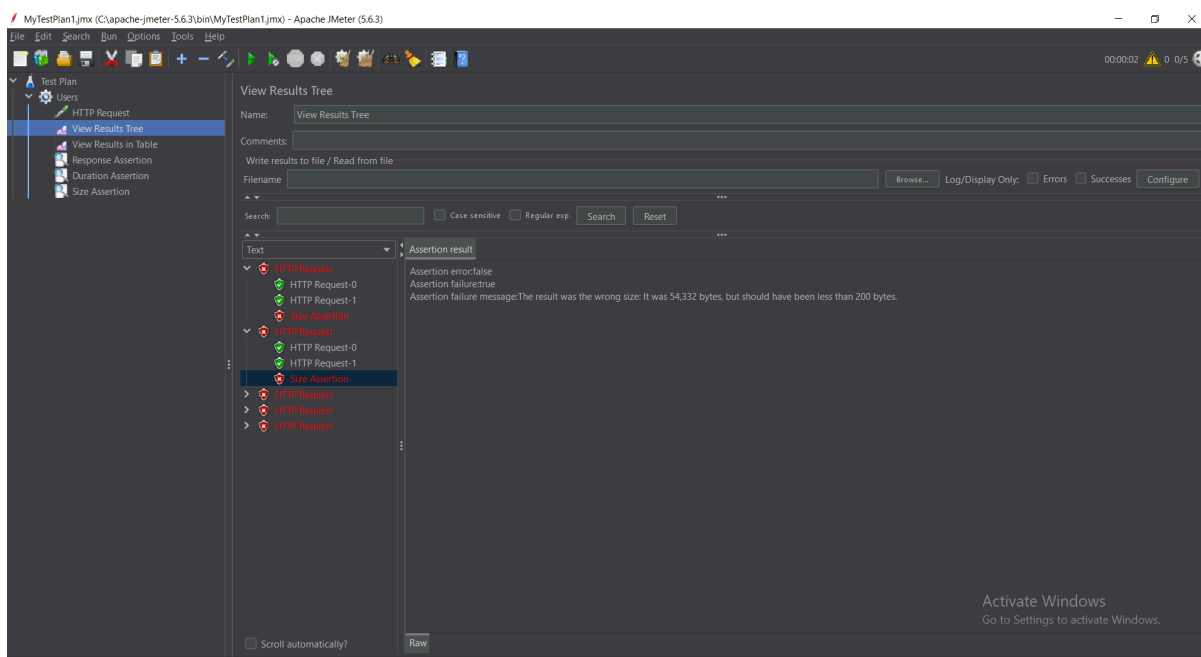
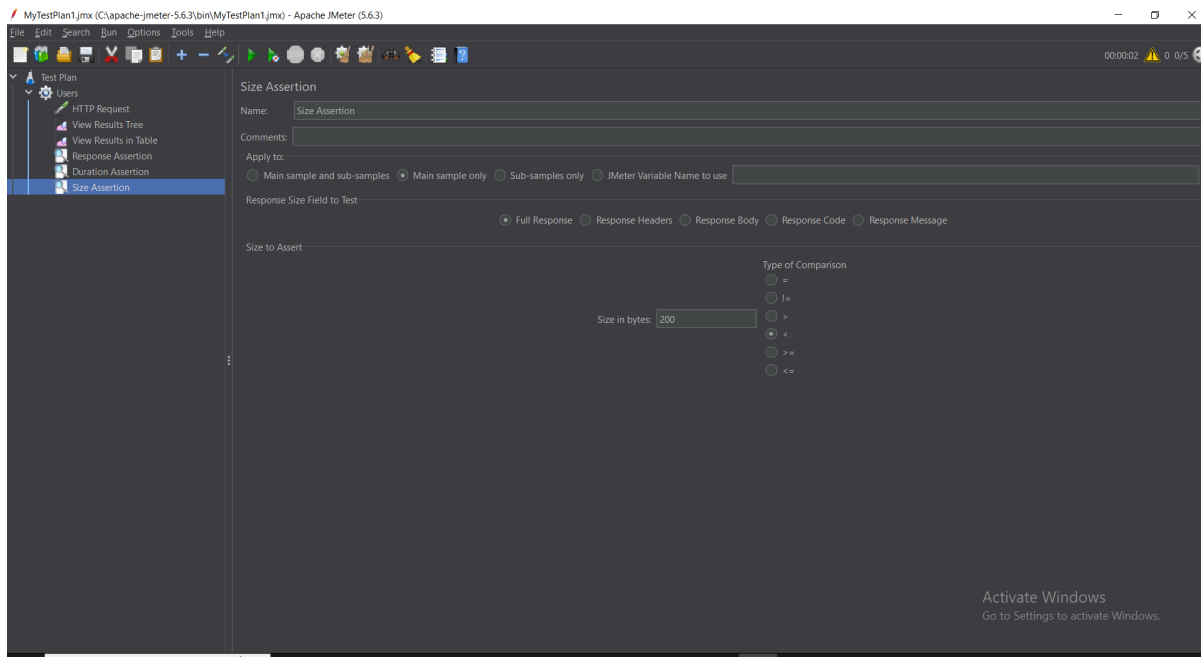
Use a **Duration Assertion** to ensure the response time is within a specified limit.

1. **Add a Duration Assertion:**
 - Right-click on the **Sampler** → **Add** → **Assertions** → **Duration Assertion**.
2. **Configure the Duration Assertion:**
 - **Duration in milliseconds:** Set the maximum allowed response time (e.g., 1000 ms).
3. **Run the Test:**
 - If the response time exceeds 1000 ms, the assertion fails.



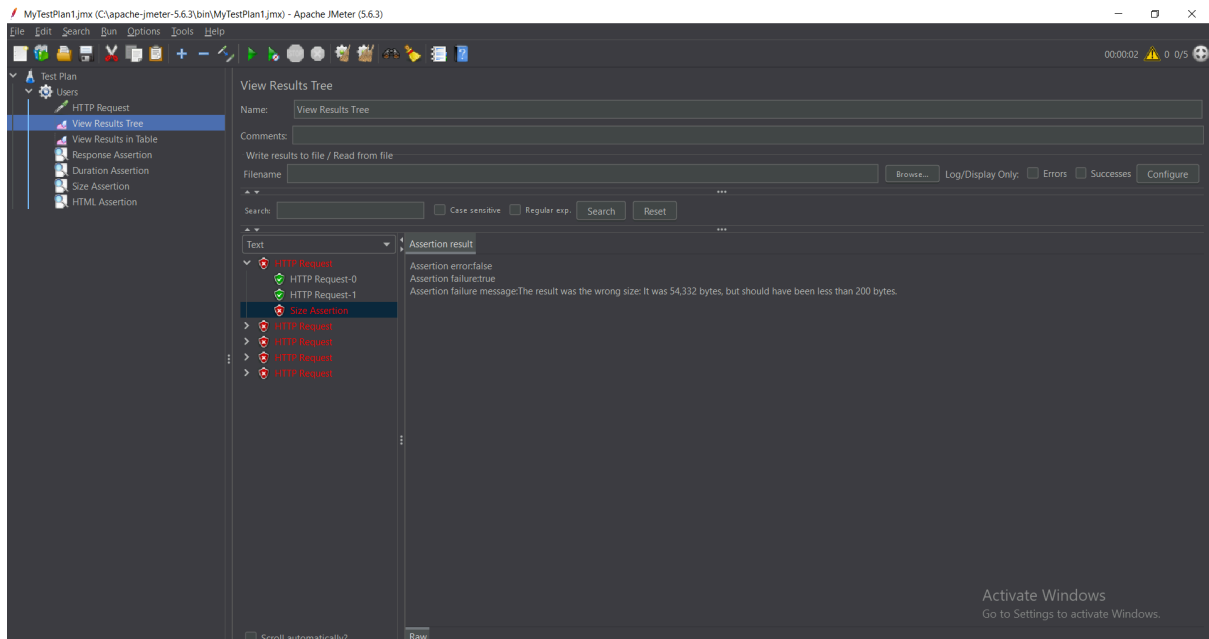
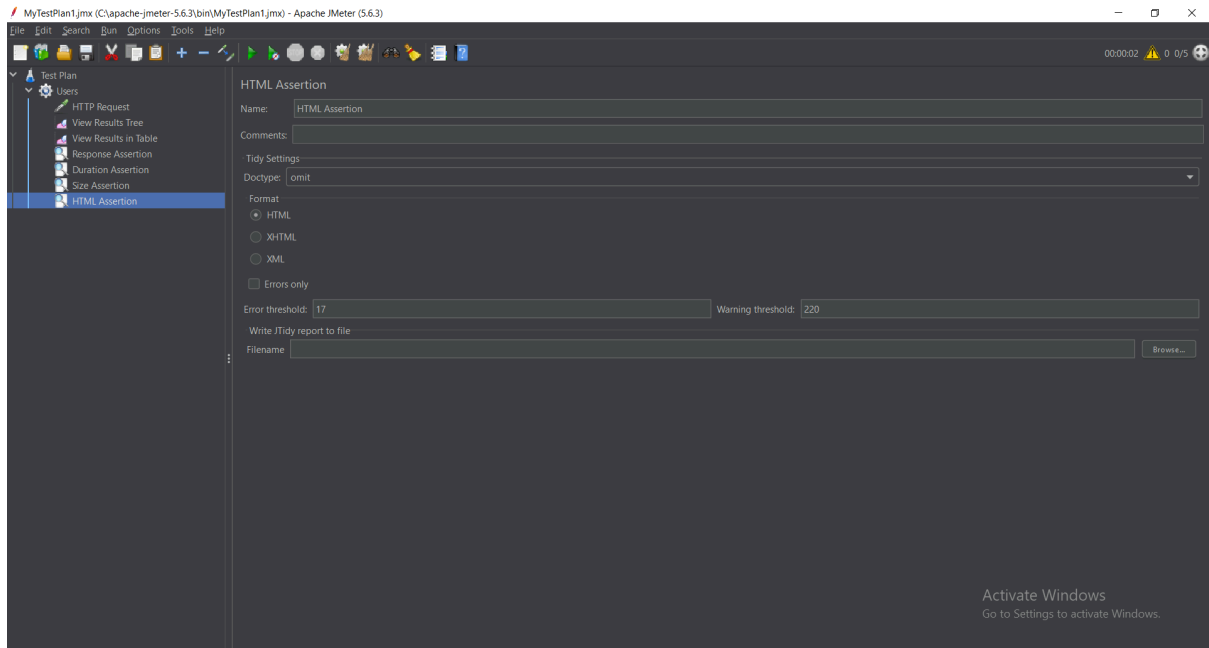
Size Assertion

1. **Select a Sampler** (e.g., HTTP Request).
2. **Right-click** on the Sampler → **Add** → **Assertions** → **Size Assertion**.



HTML Assertion

1. **Select a Sampler** (e.g., HTTP Request).
2. **Right-click** on the Sampler → **Add** → **Assertions** → **HTML Assertion**.

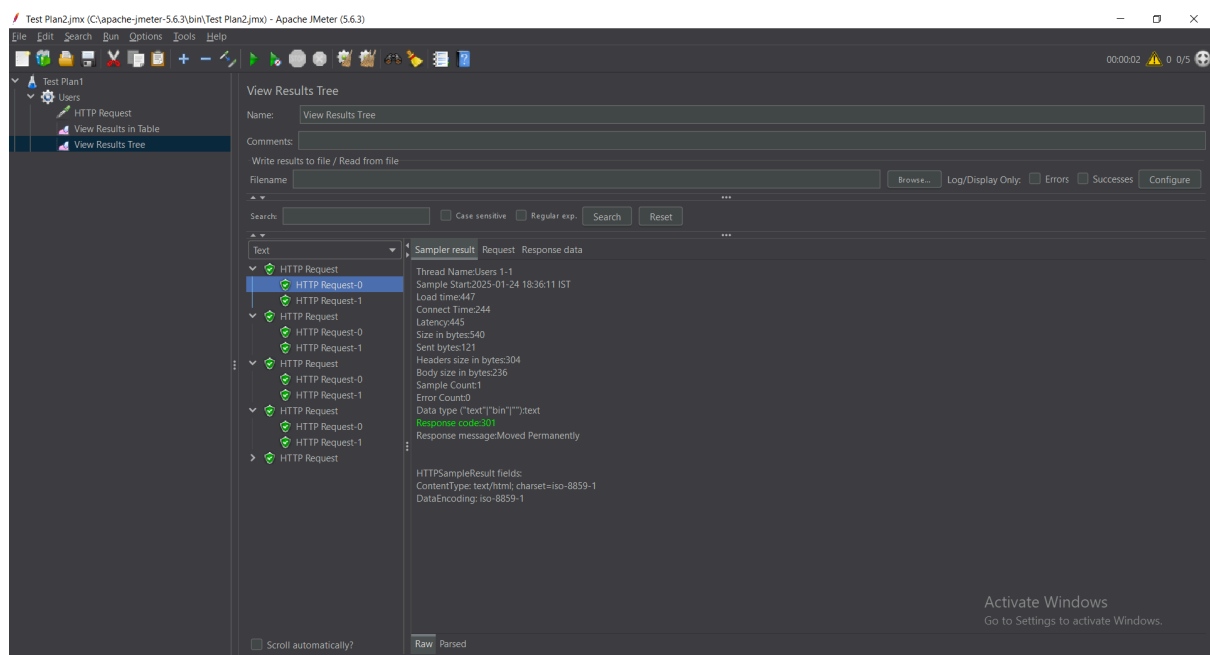


Listeners

- Listeners are the elements that gather information about the performance test.
- It used to view results / metrics of the test

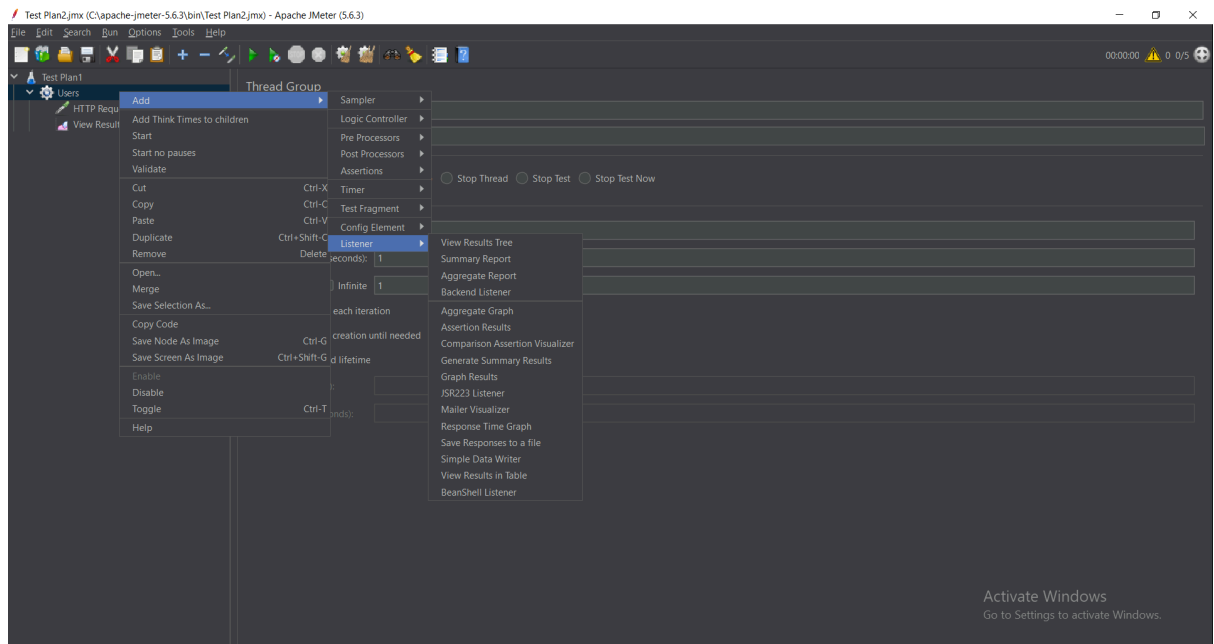
View Results Tree

- **Purpose:** Displays detailed information about each request and response.
- **Use Case:** Useful for debugging and verifying the correctness of requests and responses.
- **Data Shown:** Request headers, response data, response codes, and more.
- **Caution:** Avoid using this listener during high-load tests, as it consumes a lot of memory.



View Results in Table

- **Purpose:** Displays test results in a tabular format.
- **Use Case:** Provides a summary of each request, including response times, status codes, and latency.
- **Data Shown:** Sample time, latency, response code, and more.



Test Plan2.jmx (C:\apache-jmeter-5.6.3\bin\Test Plan2.jmx) - Apache JMeter (5.6.3)

View Results in Table

Name: View Results in Table

Comments:

Write results to file / Read from file

Filename: Browse... Log/Display Only: ☐ Errors ☐ Successes ☐ Configure

Sample #	Start Time	Thread Name	Label	Sample Time(ms)	Status	Bytes	Sent Bytes	Latency	Connect Time(ms)
1	10:14:41.407	Users 1-1		0		875	0	0	0
2	10:14:41.614	Users 1-2		0		875	0	0	0
3	10:14:41.808	Users 1-3		0		875	0	0	0
4	10:14:42.007	Users 1-4		0		875	0	0	0
5	10:14:42.207	Users 1-5		0		875	0	0	0

☐ Scroll automatically? ☐ Child samples?

No of Samples: 5 Latest Sample: 0 Average: 0 Deviation: 0

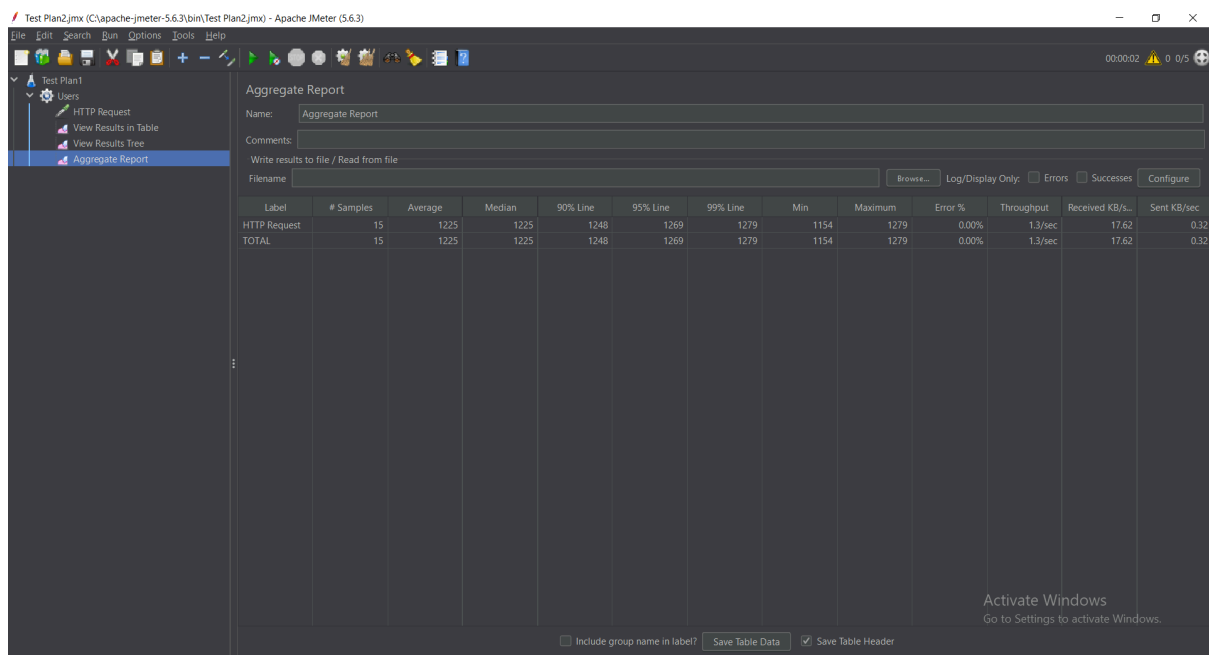
Activate Windows
Go to Settings to activate Windows.

Column Name	Description
Start Time	Execution started time
Thread Name	Iteration of user n(ex: 1st iteration of user 1)
Label	Name of request
Sample Time(ms)	Response time of request
Status	Assertion whether pass or fail

Bytes	Size of the request
Latency	Time take to arrive first Byte of the response

Aggregate Report

- **Purpose:** Similar to the Summary Report but provides more detailed statistics.
- **Use Case:** Useful for comparing performance across different samplers or requests.
- **Data Shown:** Median, 90th percentile, throughput, and error rates.



Graph Results

- **Purpose:** Displays performance metrics in a graphical format.
- **Use Case:** Visualizes trends in response times, throughput, and other metrics over time.
- **Data Shown:** Response times, throughput, and deviations.

