Experiment 4:

Obj: Analyse the performance of following website using JMeter.

Site	Website	Type
Amazon	Amazon.com	shopping
Flip kart	Flipkart.com	shopping
Railway reservation	Irctc.co.in	Ticket booking site
Train searching	Erail.in	Train searching

Description:

Apache JMeter:

Apache JMeter is pure Java-based open-source software designed to load test functional behavior and measure performance. We can use JMeter to analyze and measure the performance of web applications or a variety of services. JMeter was originally used for testing Web Applications or FTP applications. Nowadays, it is used for functional testing, database server testing, etc.

JMeter helps load test web servers, websites, and web applications by simulating real-user behaviors and testing environments. JMeter provides a user-friendly GUI, easy installation, testing strategies, simulation, and many other performance testing features. Learning JMeter helps you to ease your Performance testing tasks.

The protocols supported by JMeter are −

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

JMeter Features

Following are some of the 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 test 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 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 store its test plans in XML format. This means you can generate a test plan using a text editor.
- Its full multi-threading 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.

JMeter Working Techniques

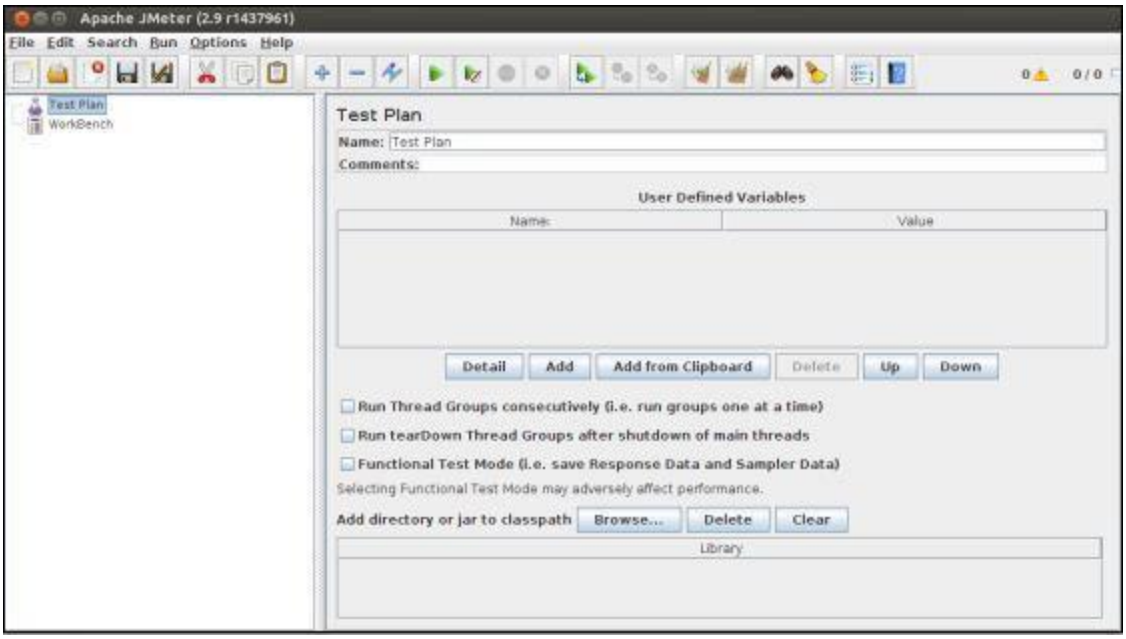
JMeter simulates a group of users sending requests to a target server, and returns statistics that show the performance/functionality of the target server/application via tables, graphs, etc.

Writing a Test Plan

Follow the steps given below to write a test plan −

Step 1: Start the JMeter Window

Open the JMeter window. The JMeter window will appear as below −



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

- **Test Plan node** − 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.

To remove an element, make sure the element is selected, right-click 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.

Step 4: Configuring the Tree Elements

Any element in the Test Plan can be configured using the controls present in JMeter's right-hand 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.,

Step 5: Saving 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 menubar.

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.

jMeter - Test Plan Elements

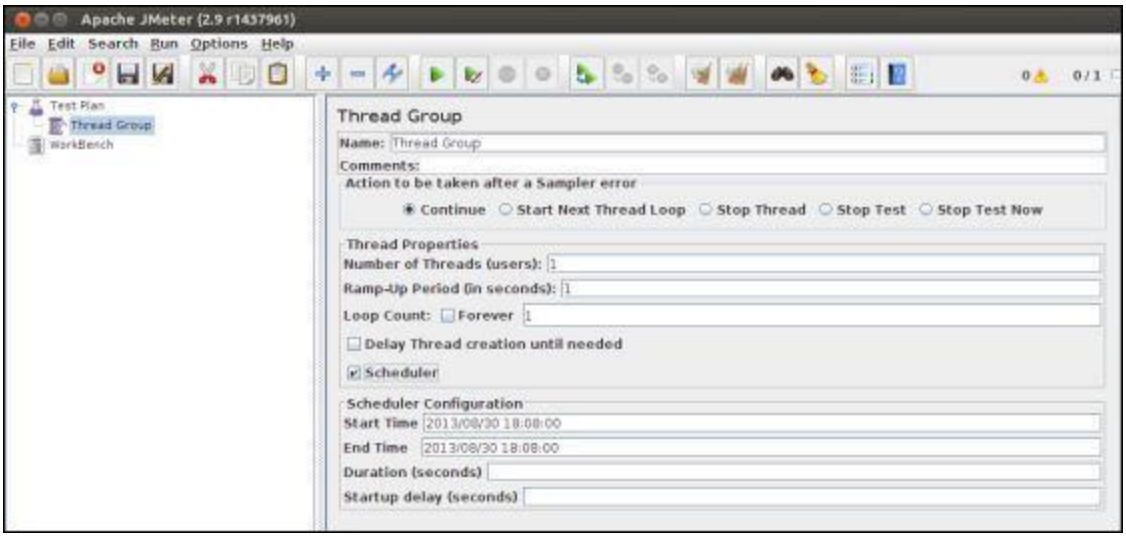
A JMeter Test Plan comprises of test elements discussed below. A Test Plan comprises of at least one Thread Group. Within each Thread Group, we may place a combination of one or more of other elements − Sampler, Logic Controller, Configuration Element, Listener, and Timer. Each Sampler can be preceded by one or more Pre-processor element, followed by Post-processor element, and/or Assertion element. Let us see each of these elements in detail −

Thread Group

Thread Group elements are the beginning points of your test plan. As the name suggests, the thread group elements control the number of threads JMeter will use during the test. We can also control the following via the Thread Group −

- Setting the number of threads
- Setting the ramp-up time
- Setting the number of test iterations

The Thread Group Control Panel looks like this −



The Thread Group Panel holds the following components −

- **Action to be taken after a Sampler error** − In case any error occurs during test execution, you may let the test either −
    - **Continue** to the next element in the test
    - **Stop Thread** to stop the current Thread.
    - **Stop Test** completely, in case you want to inspect the error before it continues running.
- **Number of Threads** − Simulates the number of users or connections to your server application.
- **Ramp-Up Period** Defines how long it will take JMeter to get all threads running.
- **Loop Count** − Defines the number of times to execute the test.
- **Scheduler checkbox** − Once selected, the Scheduler Configuration section appears at the bottom of the control panel.
- **Scheduler Configuration** − You can configure the start and end time of running the test.

Controllers

JMeter has two types of Controllers − *Samplers* and *Logic Controllers*.

Samplers

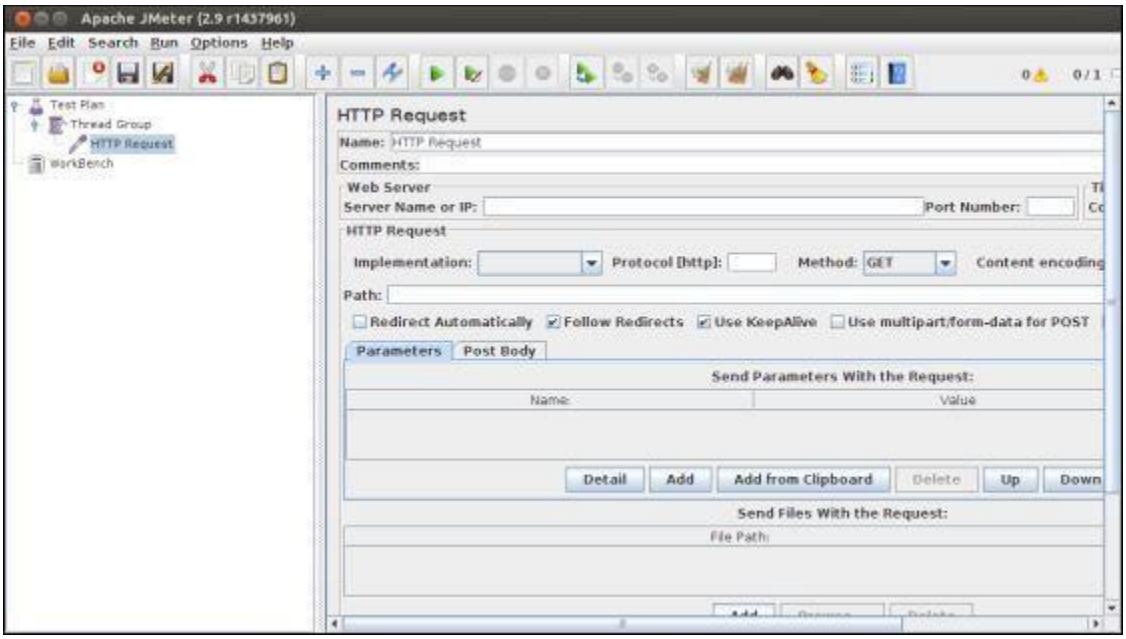
Samplers allow JMeter to send specific types of requests to a server. They simulate a user request for a page from the target server. For example, you can add a HTTP Request sampler if you need to perform a POST, GET, or DELETE on a HTTP service.

Some useful samplers are −

- HTTP Request

- FTP Request
- JDBC Request
- Java Request
- SOAP/XML Request
- RPC Requests

The following screenshot shows an HTTP Request Sampler Control Panel −



Listeners

Listeners let you view the results of Samplers in the form of tables, graphs, trees, or simple text in some log files. They provide visual access to the data gathered by JMeter about the test cases as a Sampler component of JMeter is executed.

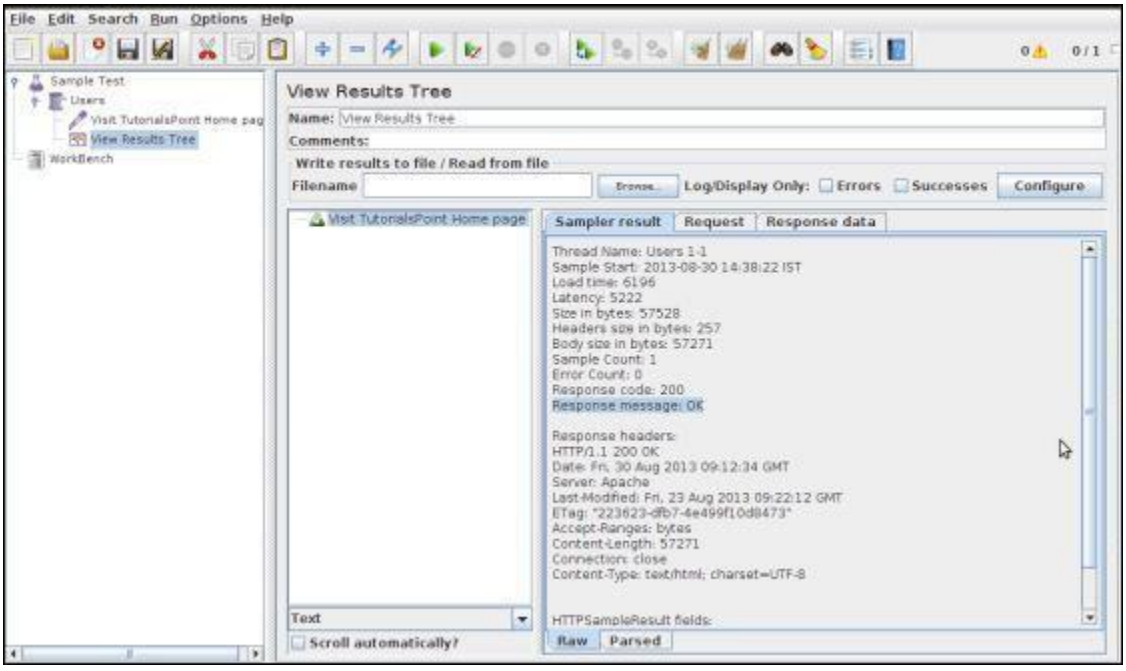
Listeners can be added anywhere in the test, including directly under the test plan. They will collect data only from elements at or below their level. The following list consists of all the Listeners JMeter provides −

- Sample Result Save Configuration
- Graph Full Results
- Graph Results
- Spline Visualizer
- Assertion Results
- View Results Tree
- Aggregate Report
- View Results in Table

- Simple Data Writer
- Monitor Results
- Distribution Graph (alpha)
- Aggregate Graph
- Mailer Visualizer
- BeanShell Listener
- Summary Report

View the Output

We have kept the setting of the thread group as single thread (one user only) and loop for 1 time (run only one time), hence we will get the result of one single transaction in the View Result Tree Listener.



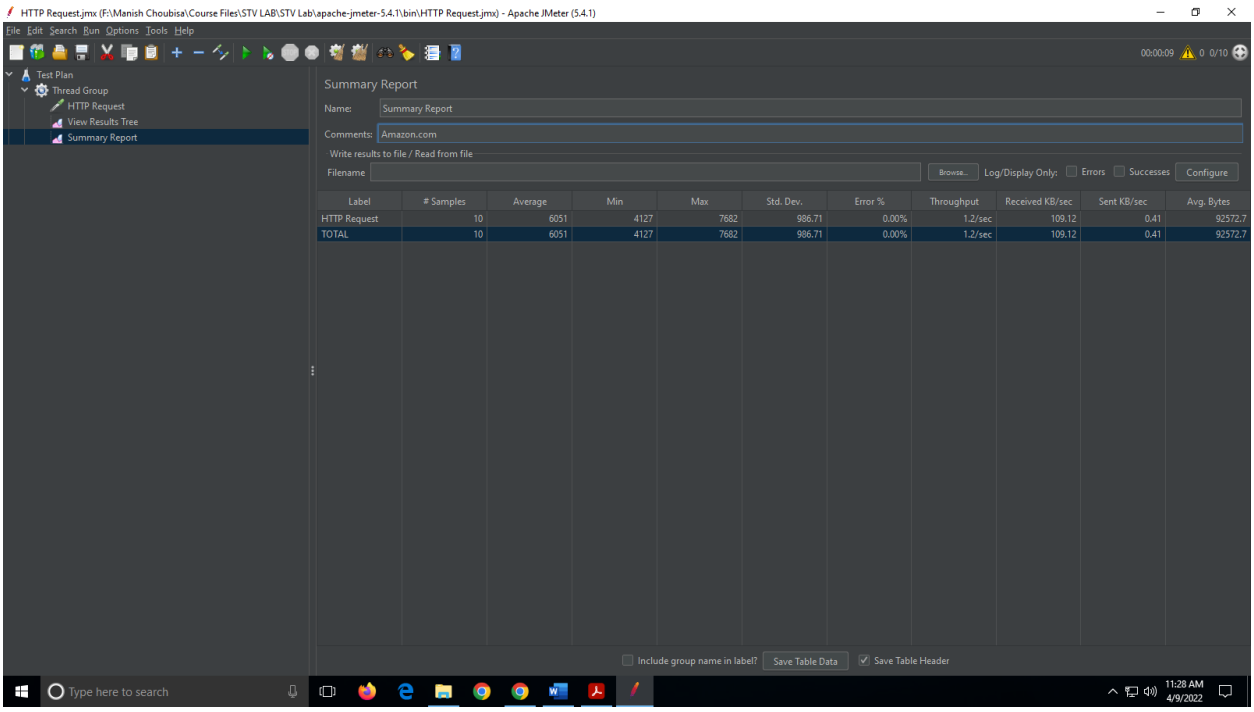
Program Outputs:

The performance Analyse of following website using JMeter.

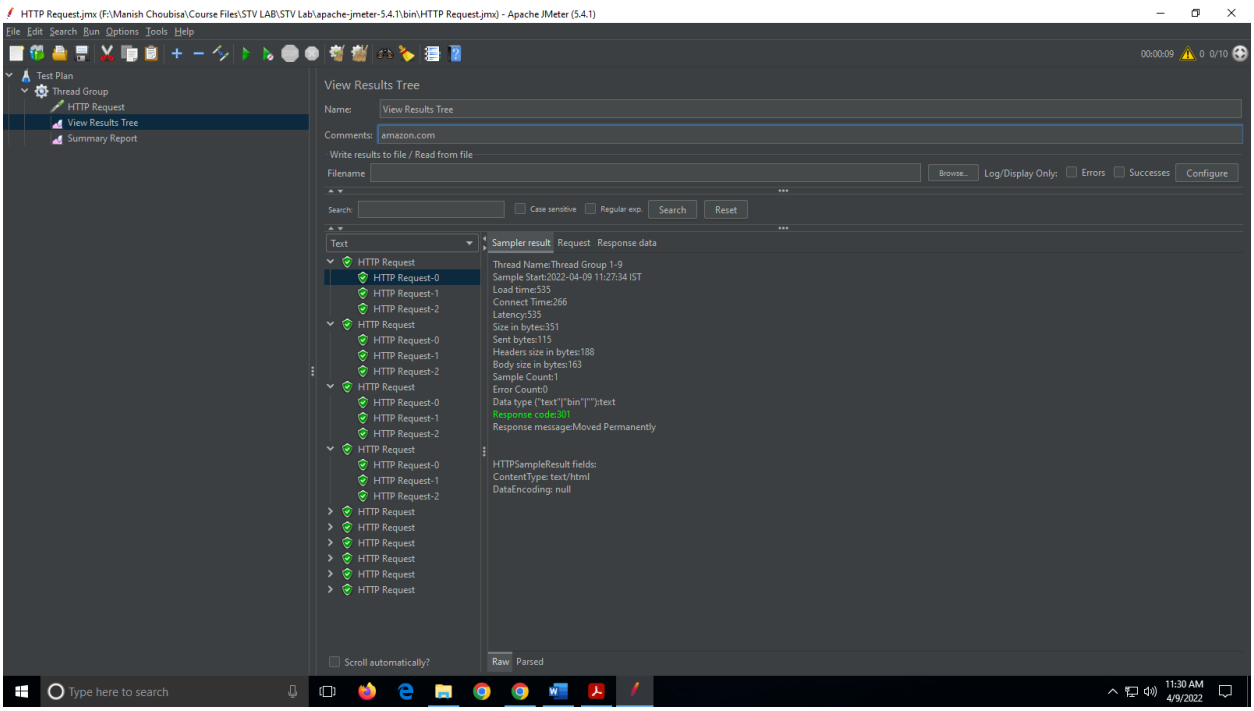
Site	Out put
Amazon.com	
Flipkart.com	
Irctc.co.in	
Erail.in	

1. Amazon.com

Summary Report:

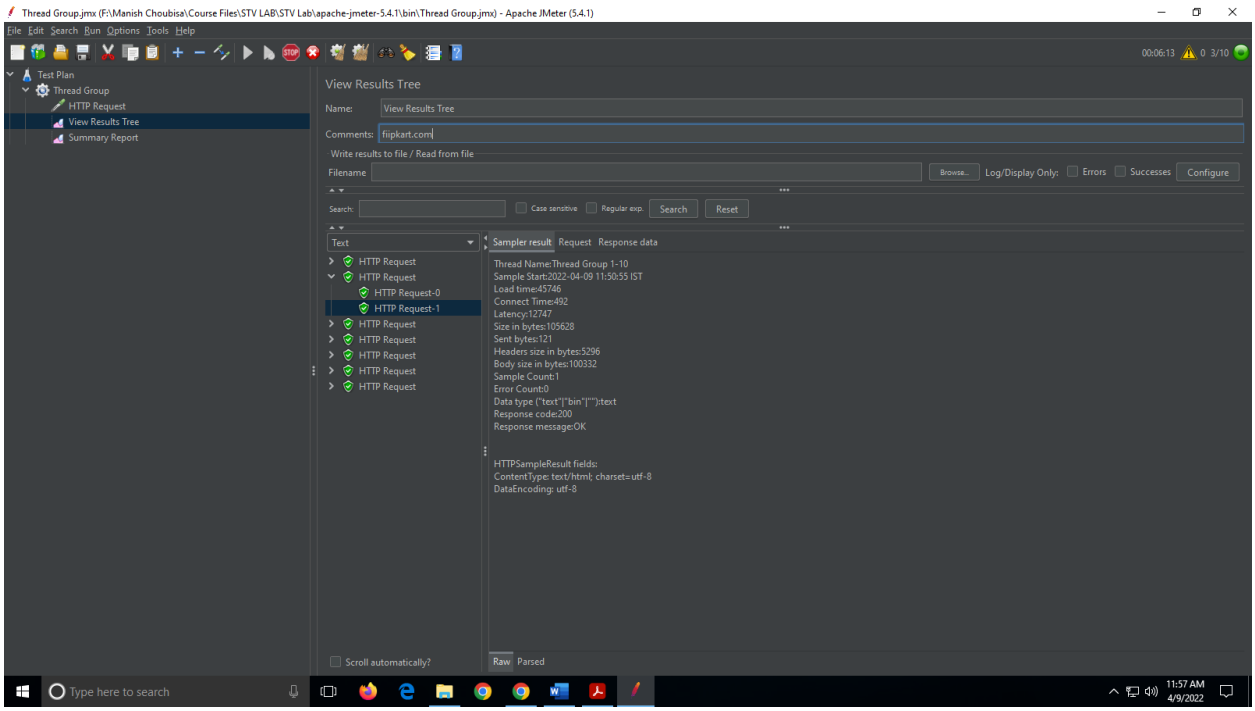


View result Tree:

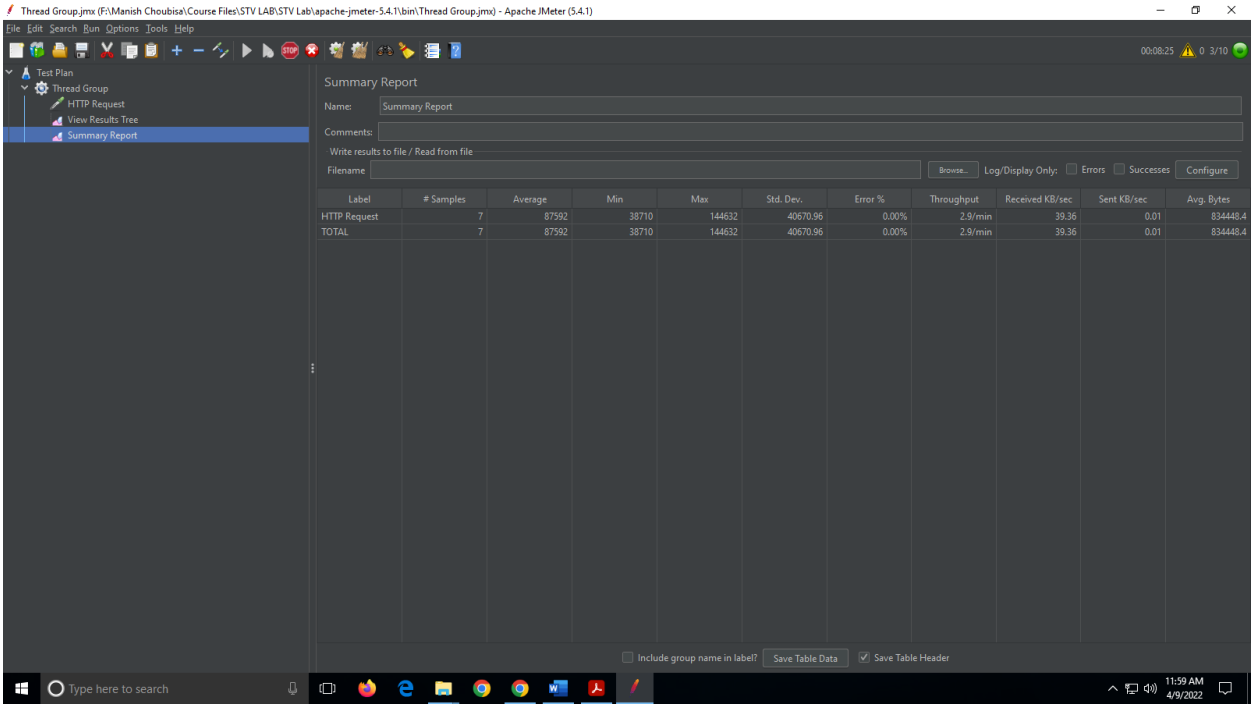


2. Flipkart.com

- View result Tree:



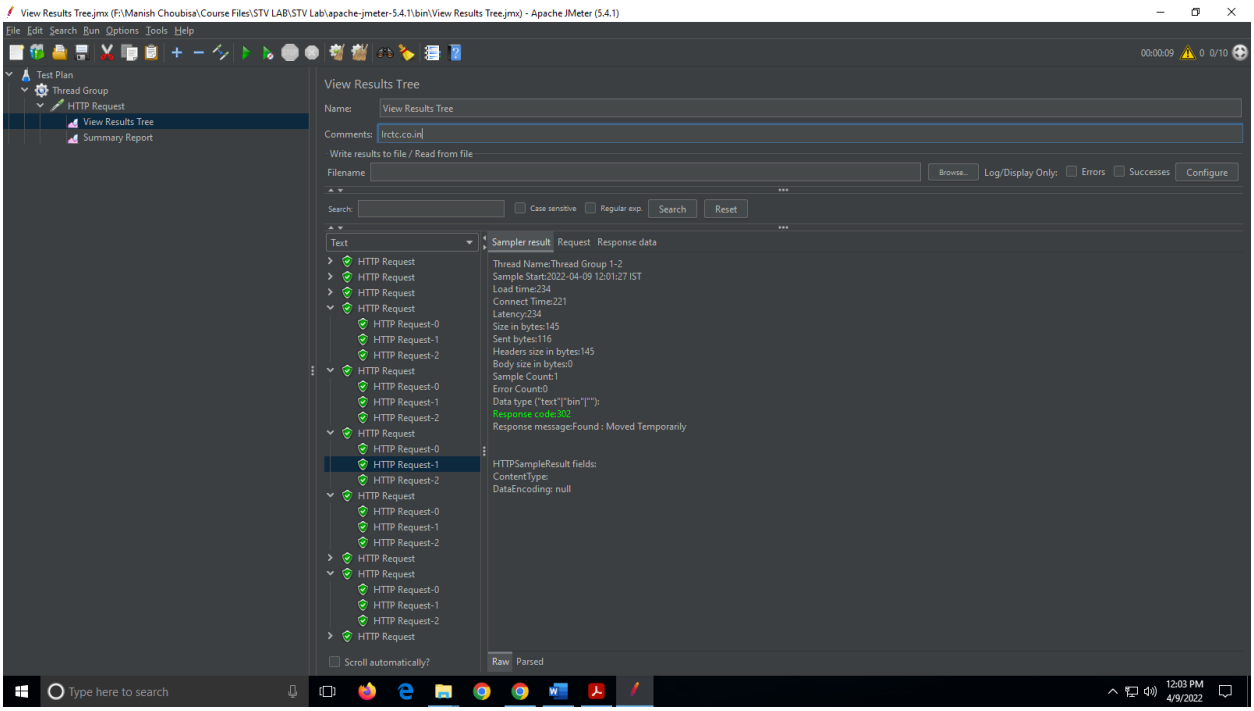
- Summary Report:



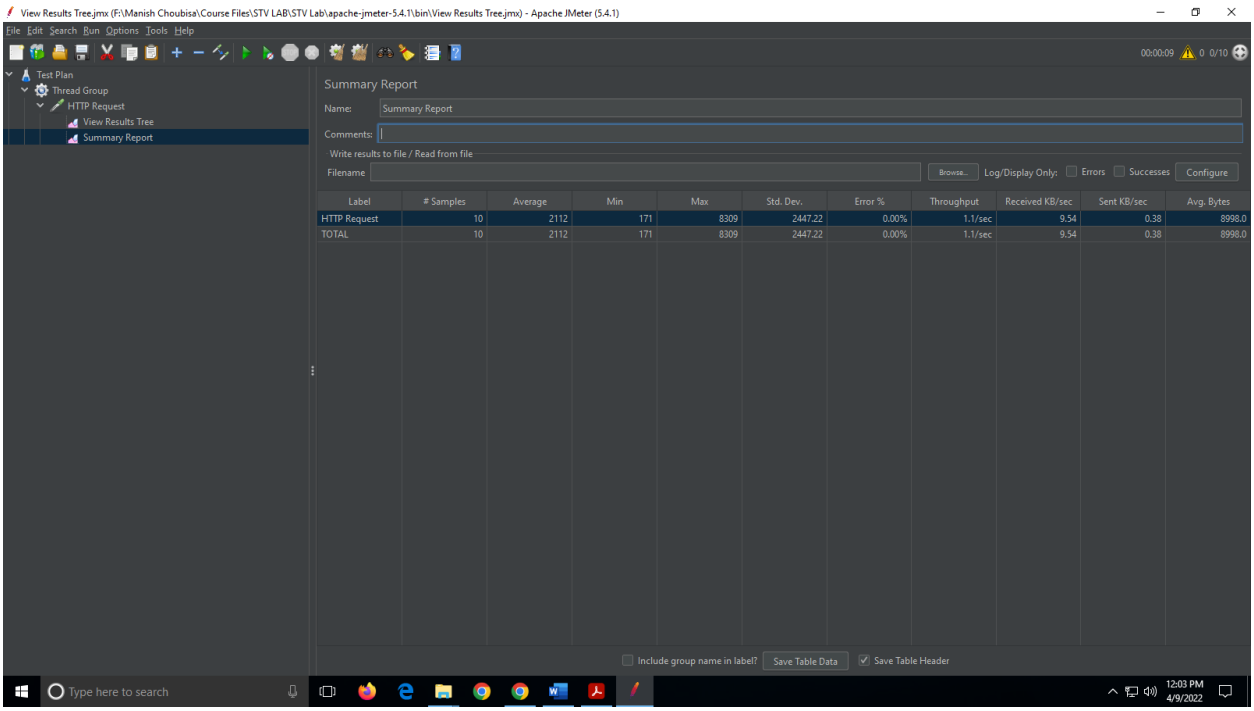
-

3. Irctc.co.in

- View result Tree:

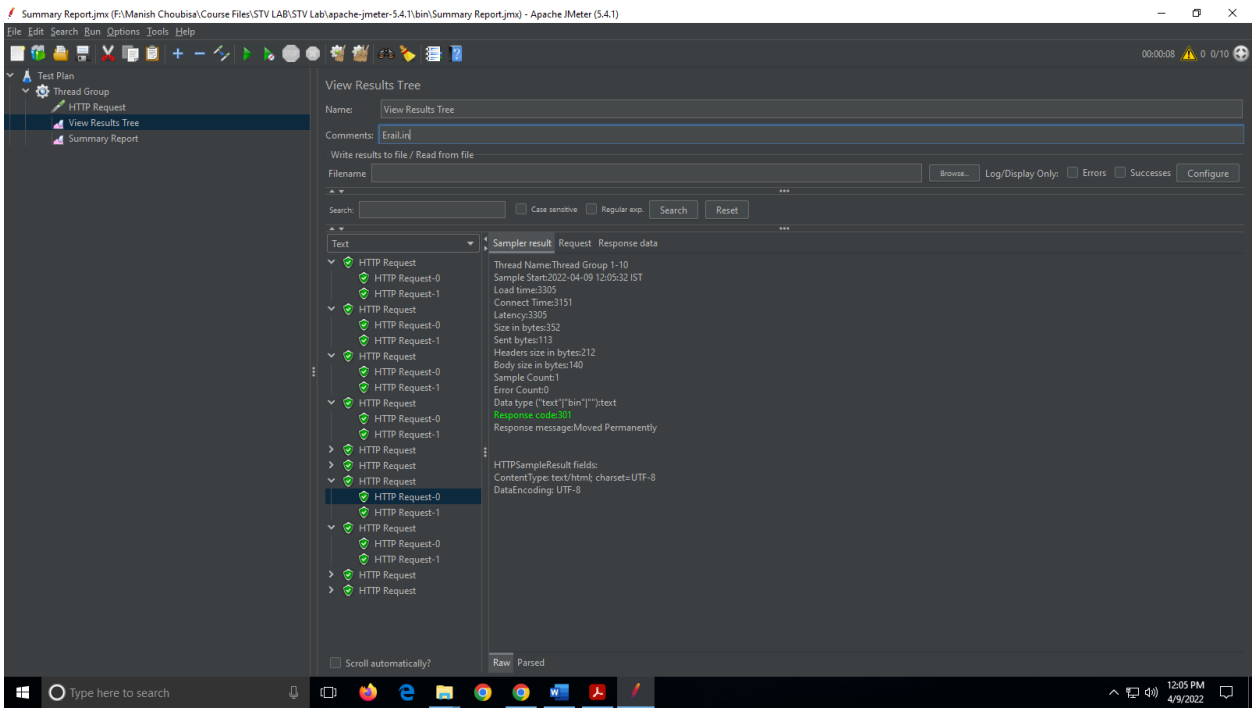


- Summary Report:



4. Erail.in

- View result Tree:



- Summary Report: