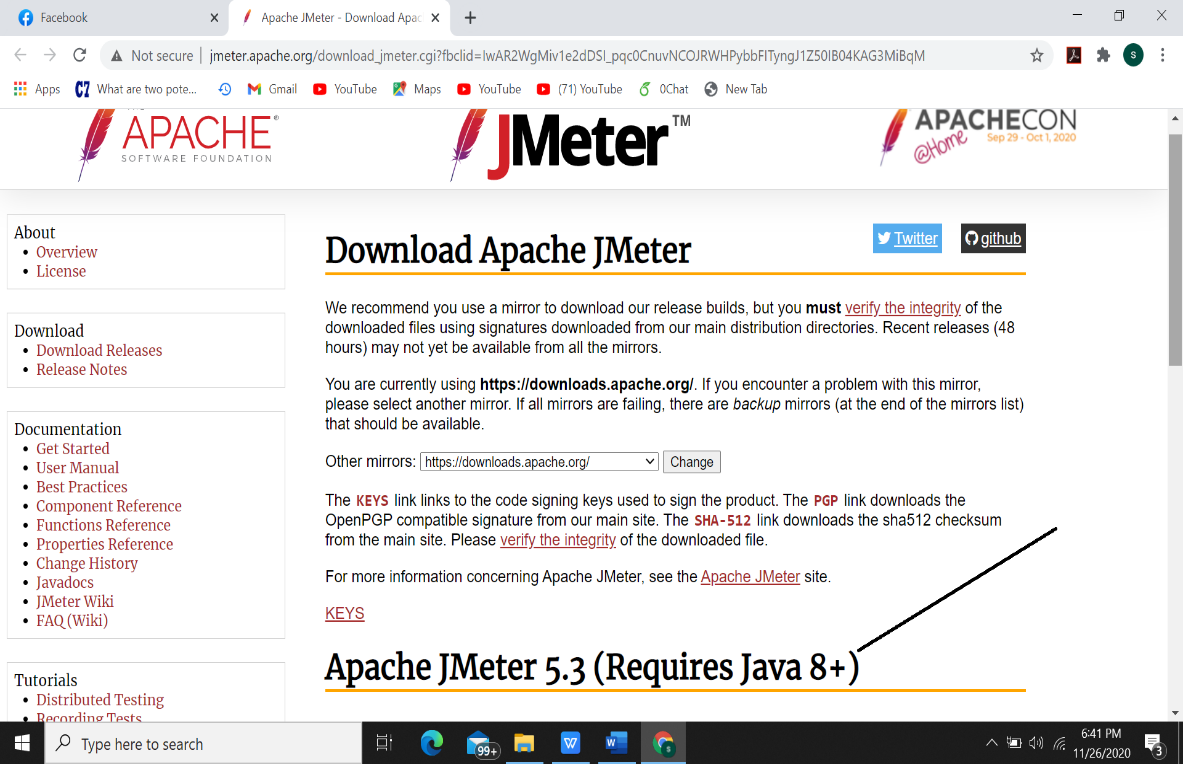
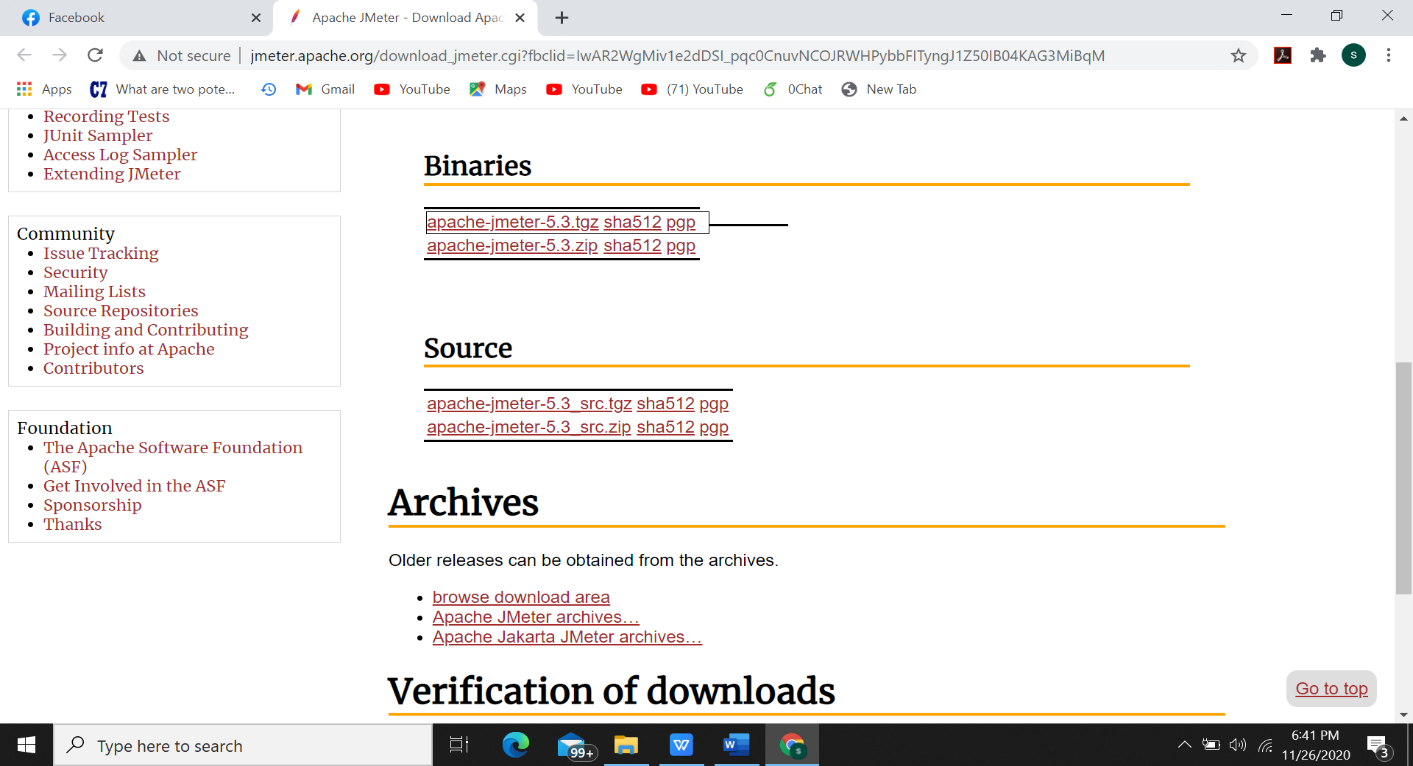
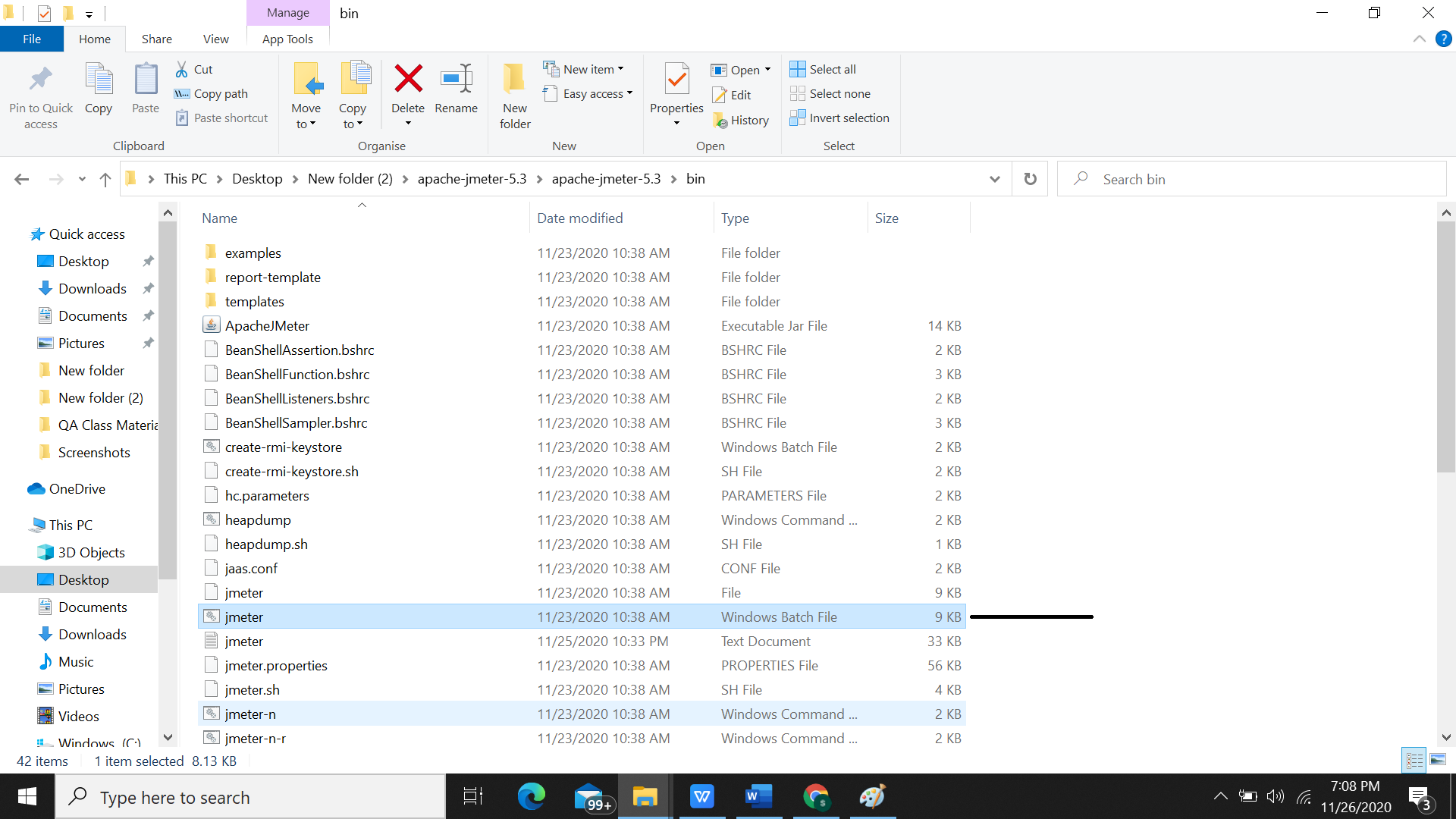
**JMETER HOMEWORK**

**Download and Installation:**

Have to install java in that device which is used to run JMETER



Download this binary file



After opening the bin file open this windows batch file to open the ui of jmeter

After extracting the jar file open the jmeter 5.3 folder and go to this bin folder

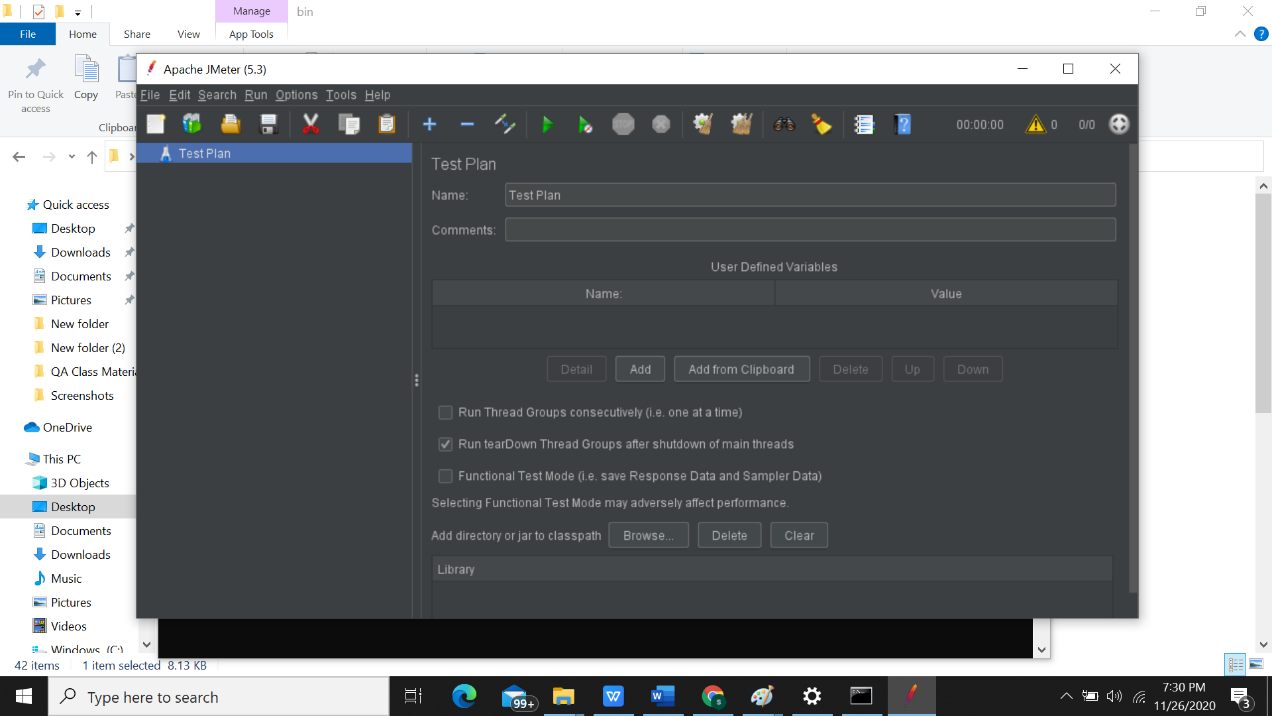


Fig: Apache Jmeter user interface

So, installation is successful.

**Short description of JMeter Homepage:**

It is divided into two work frames.

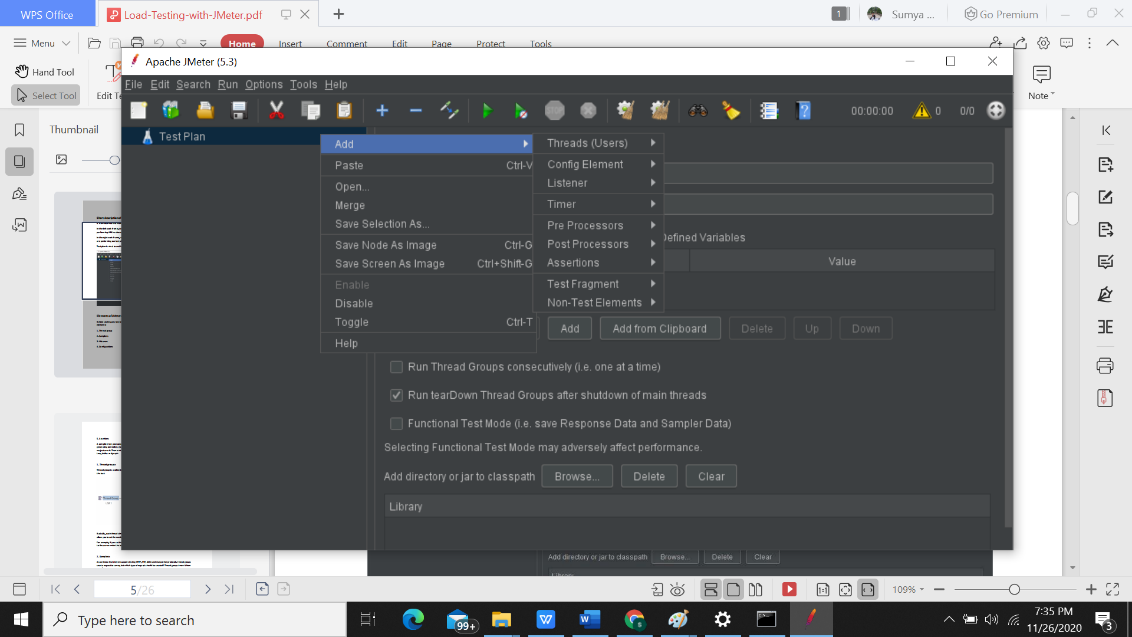
In the left work frame, we add the threads, groups, test plan name, listener, sampler or whatever we are

performing. All the element of JMeter are here.

In the right work frame, we add all the values of the number of threads or users or in which website we

are performing our test plan. This consists of the configuration.

**Test plan** is what to and how to test.



Picture 7: Apache JMeter GUI

**Elements of JMeter:**

Before starting any test lets discuss some important elements of Jmeter. There are 4 important

elements:

1. Thread group

2. Samplers

3. Listeners

4. Configurations

5. Assertions

A complete test plan consists of one or more elements such as thread groups, logic controllers, sample

generating controllers etc. It describes the behavior of the element. Once we configure the elements,

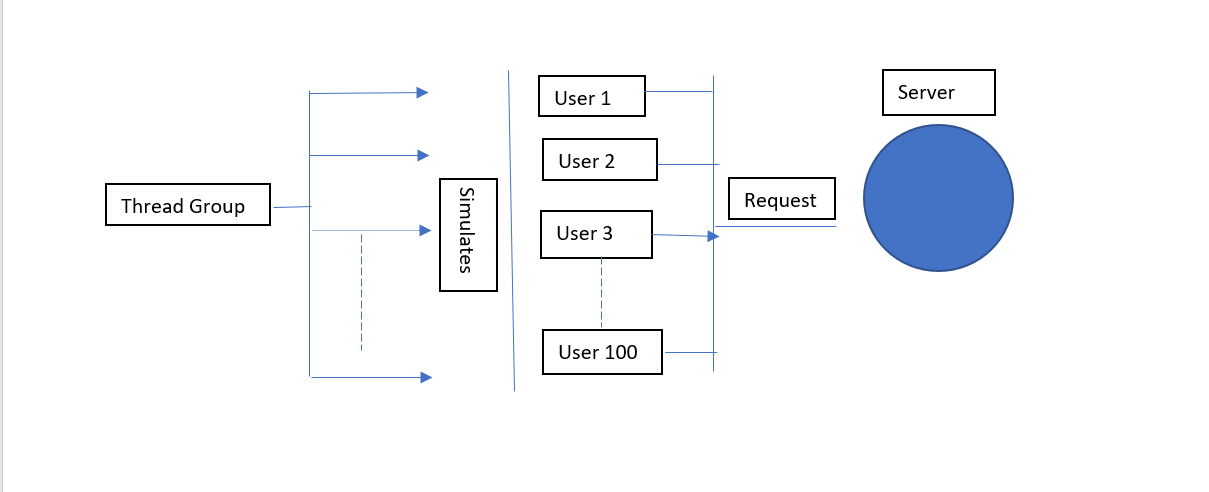
we just save it. Then run the test plan and analyze the test results from various graphical formats such as

tree, tables and graph.

**1. Thread groups:**

**Thread group is a collection of threads. Each thread represents one user using the application under**

**the test.**



Picture 8: Thread groups

Basically, each thread simulates one real user request to the server and the controls for a thread groups

allow you to set the number of threads for each group.

For example, if you set the number of threads as 100, JMeter will create and simulate 100 user requests

to the server under the test.

**2. Samplers:**

As we know that Jmeter supports testing HTTP, FTP, JDBC and many other protocols. thread groups

create request to server, but which type of request should be created? Thread groups know it from

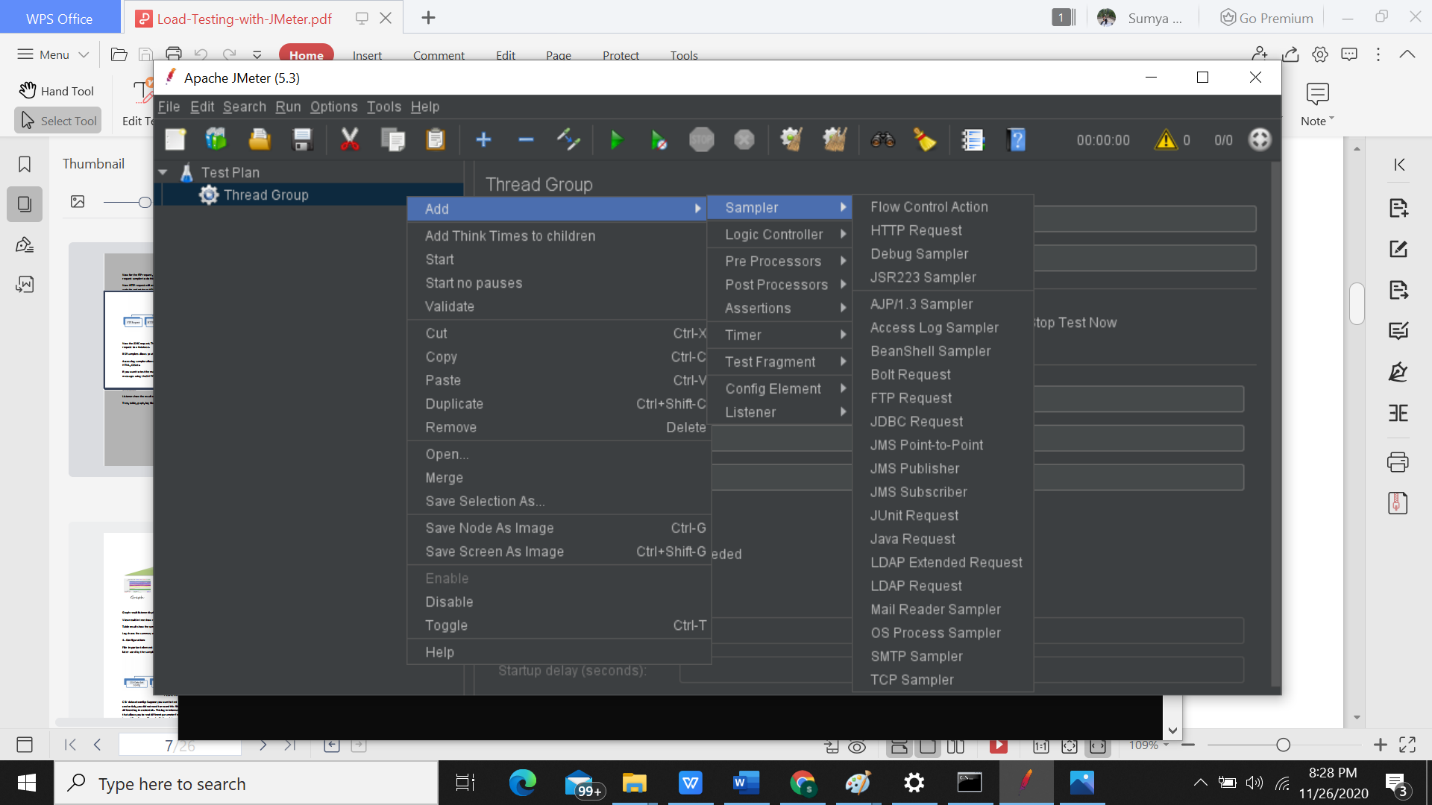
samplers.

Now for the **FTP** request, if you want to create an FTP request to the server then you need to use an FTP

request sampler to do this task.

Now **HTTP** request will send **HTTP/HTTPS** request to the server. JMeter sends an http request to google

website and retrieves HTML files or image from this website.



Now the **JDBC** request. This sampler lets you execute database performance testing. It sends JDBC

request to a Database.

**BSF** samplers allows you to write BSF scripting language.

**Access log sampler** allows you to read access log and generate http request. The log could be images,

HTML, CSS etc.

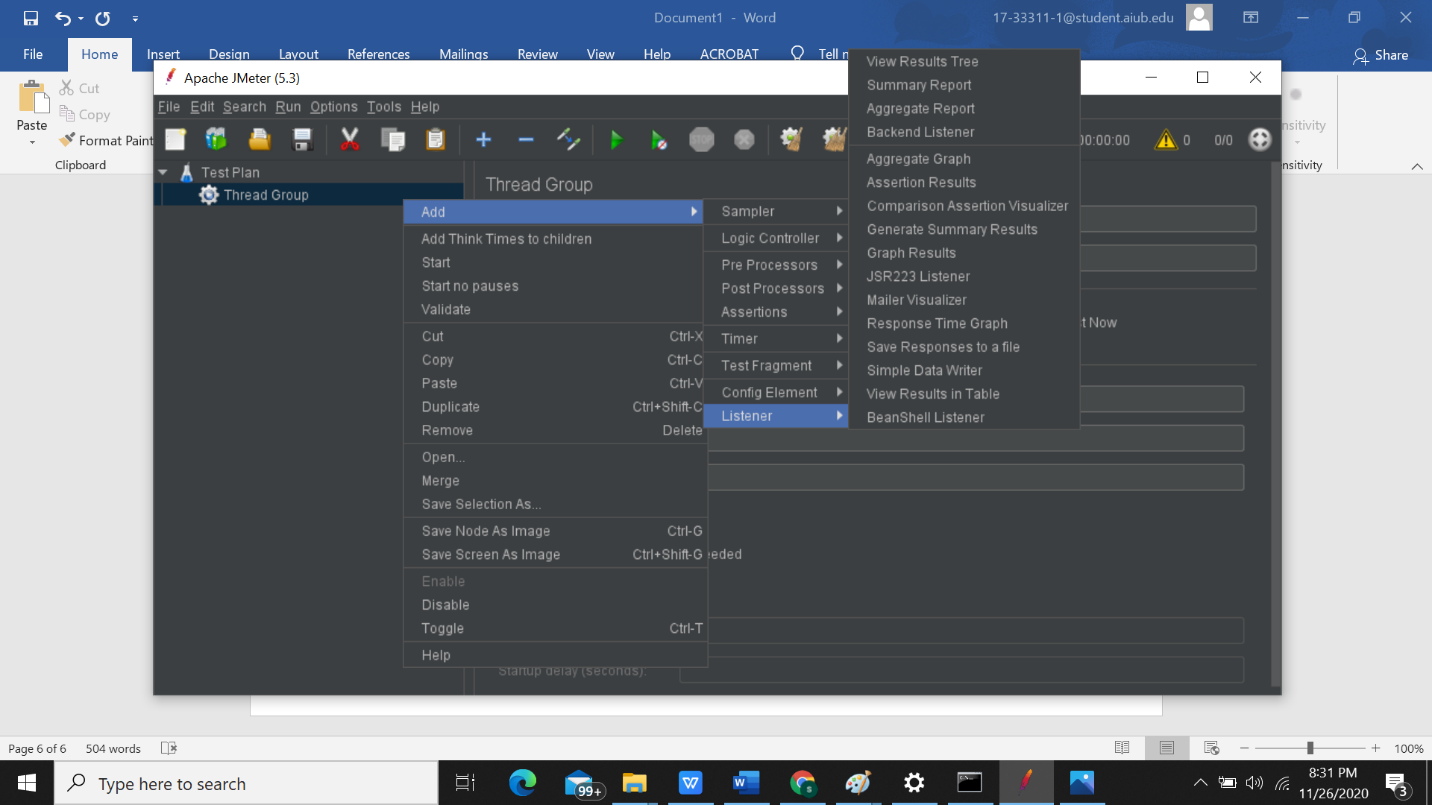
If you want to test the mail server, you can use **SMTP** sampler. Tis sampler is used to send an email

messages using the SMTP protocol.

**3. Listener:**

Listener show the result of the test execution. They can show result in different format such as

Tree, table, graph, log file.



Picture 10: Listener

Graph result listener display the server response time on a graph view.

View result in tree show result of the user request in basic HTML format.

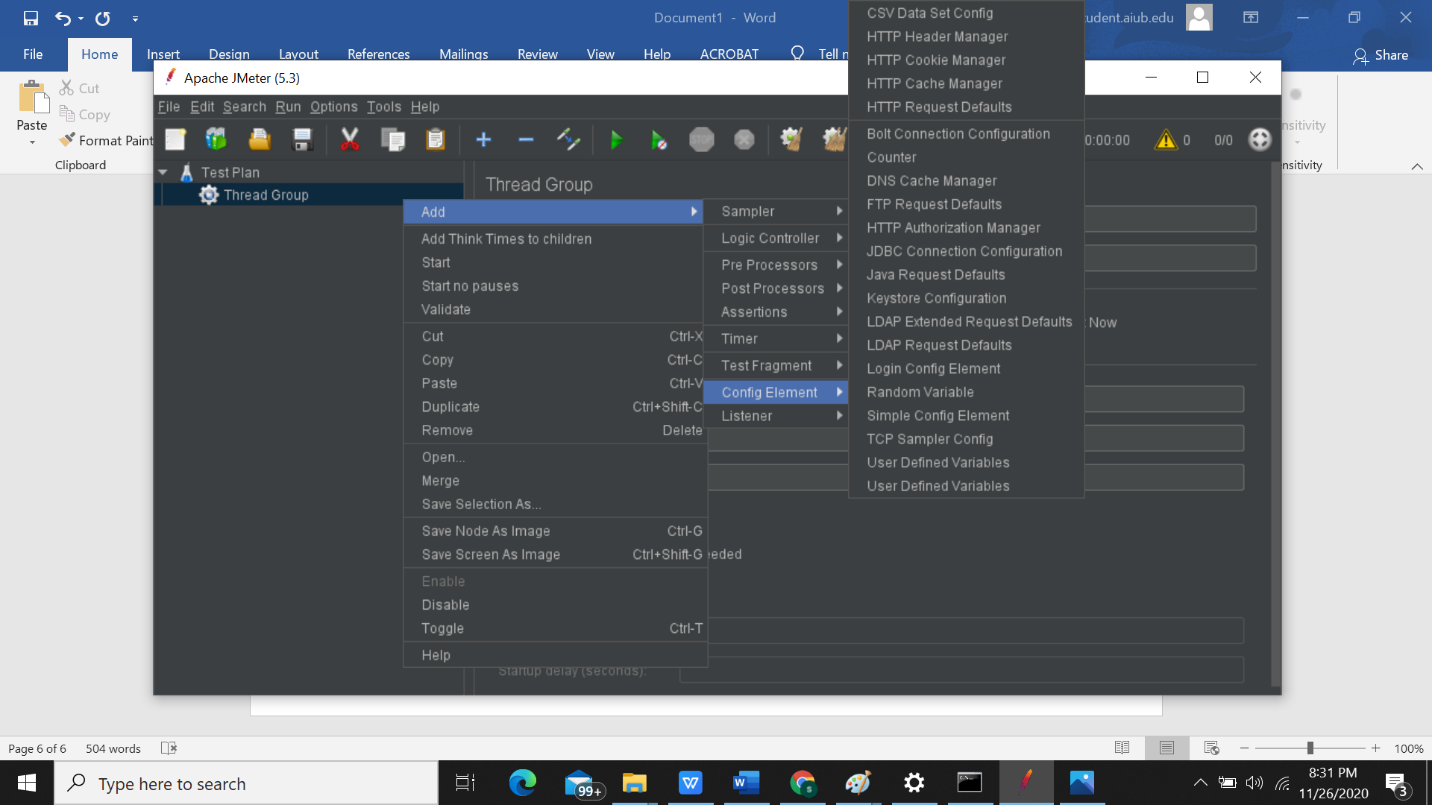
Table result show the summary of the test result in table format.

Log shows the summary of test result in the text file.

**4. Configuration:**

File important element is the configuration element. It basically sets in default and variable for

later used by the samplers.



Picture 11: Configuration elements

**CSV dataset config**: Suppose you want to test e website for 100 users signing in with the different

credentials, you did not need to record this file for 100 times. You can parameterize the script to enter

different log in credentials. This log in information could be stored in a text file. JMeter has an element

that allows you to read different parameter from that text file. It is the CSV data set config which is used

to read lines from a file and split them into variable. **HTTP cookie manager:** The cookie manager automatically stores the cookie and will use it for all future

request for that website.

**HTTP request default:** It lets you set the default value for HTTP request controller uses. When we need

to send 100 HTTP request to the server google.com then, we can add a single HTTP default with the

server name or IP field that is equal to google.com.

**Login config element:** It lets you add or override the username and password settings in samplers. For

example, if you simulate one user log in to the website facebook.com with user and password, you can

use the log in config element to add the username and password setting in a user request.

**5. Assertion:**

Assertion help verifies that your server under test returns the **expected** results.

Types of assertion:

1. Response assertion: The response assertion lets you add pattern strings to be compared against

various fields of the server response.

For example, you send a user request to the website http://www.google.com and get the server

response. You can use Response Assertion to verify if the server response **contains** expected pattern

string (e.g. "OK").

2. Duration assertion: The Duration Assertion tests that each server response was received within

a **given amount** of time. Any response that takes longer than the given number of milliseconds

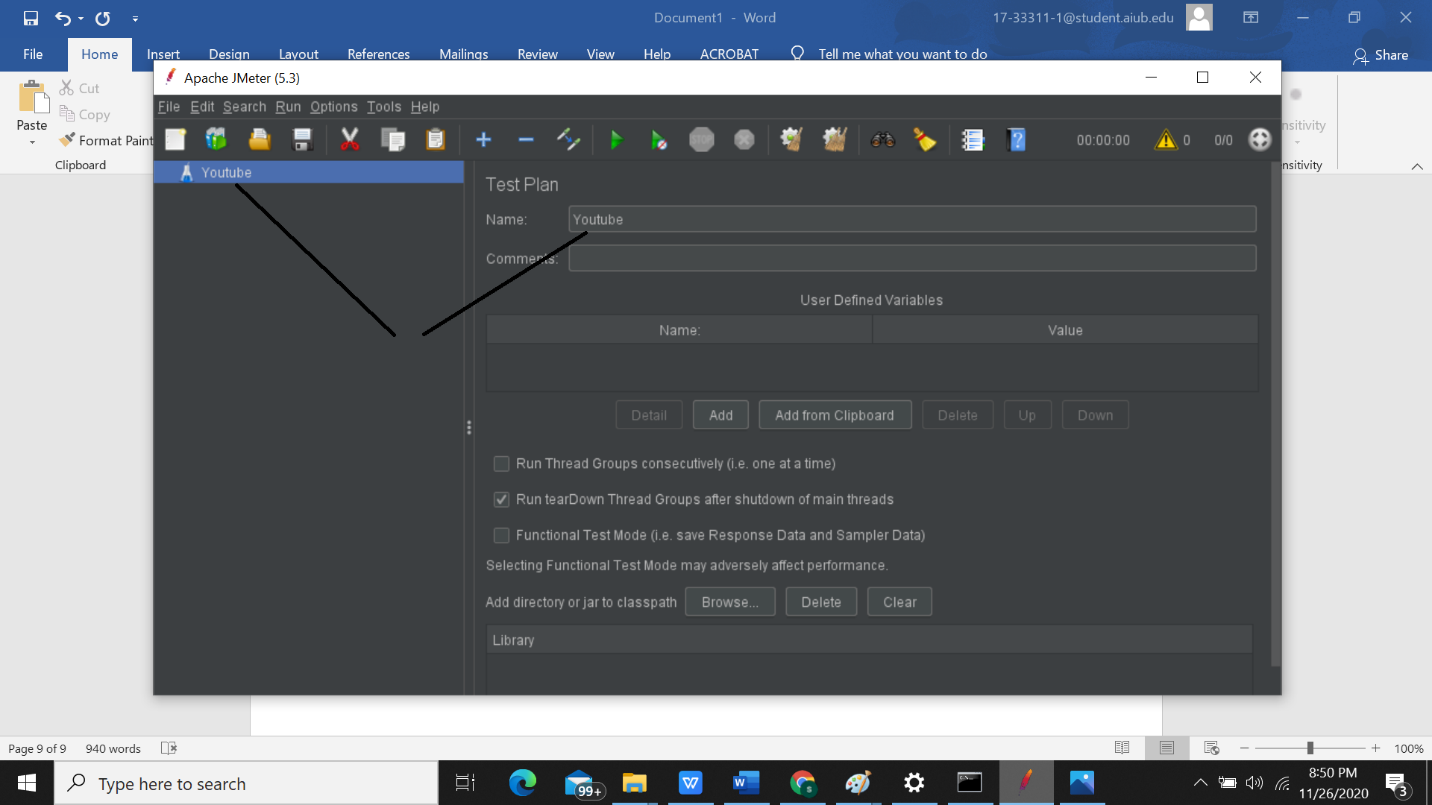
(specified by the user) is marked as a failed response.

3.HTML assertion: The HTML Assertion allows the user to check the HTML syntax of the response data.

It means the response data must be met the HTML syntax.

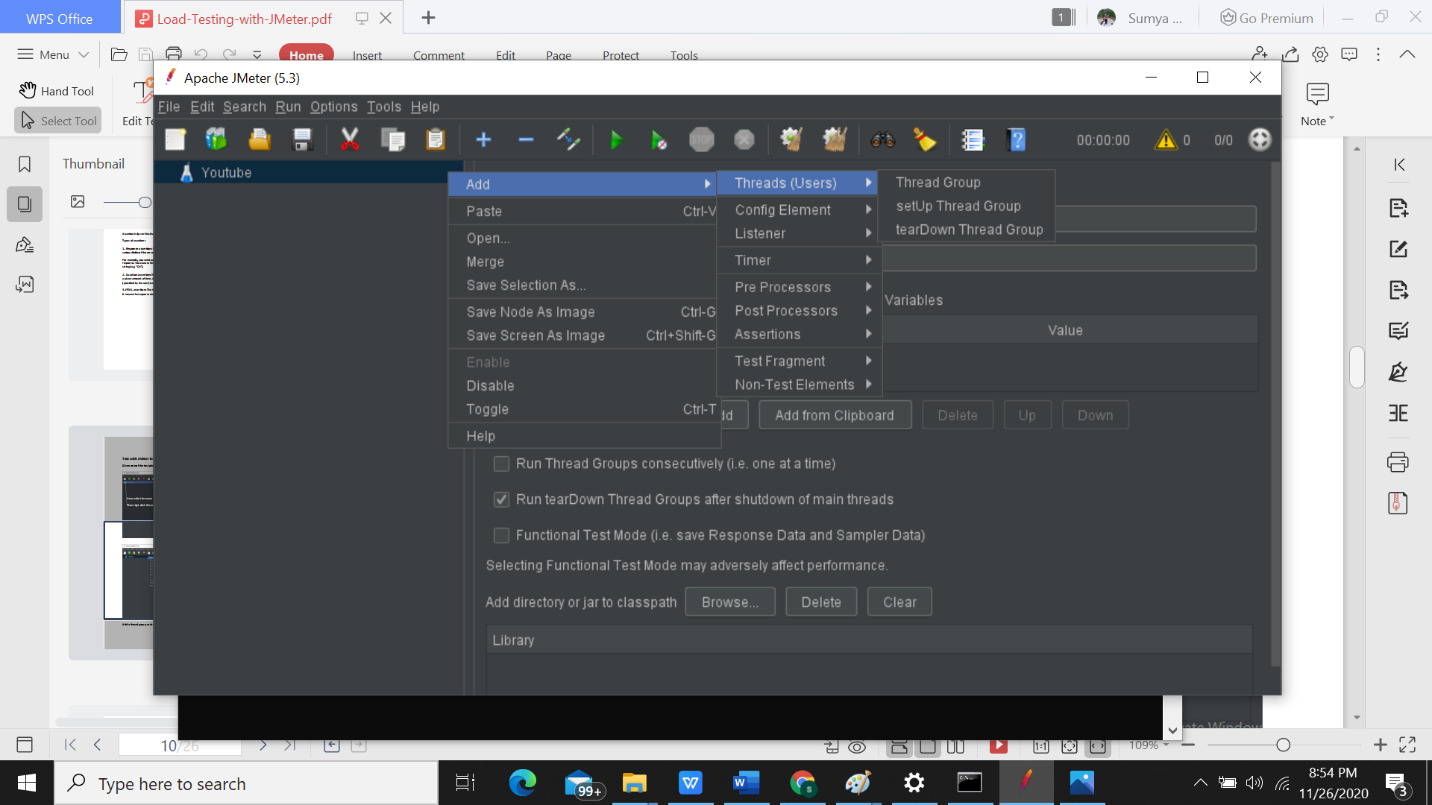
**Test with JMeter in GUI:**

Give name of the test plan.



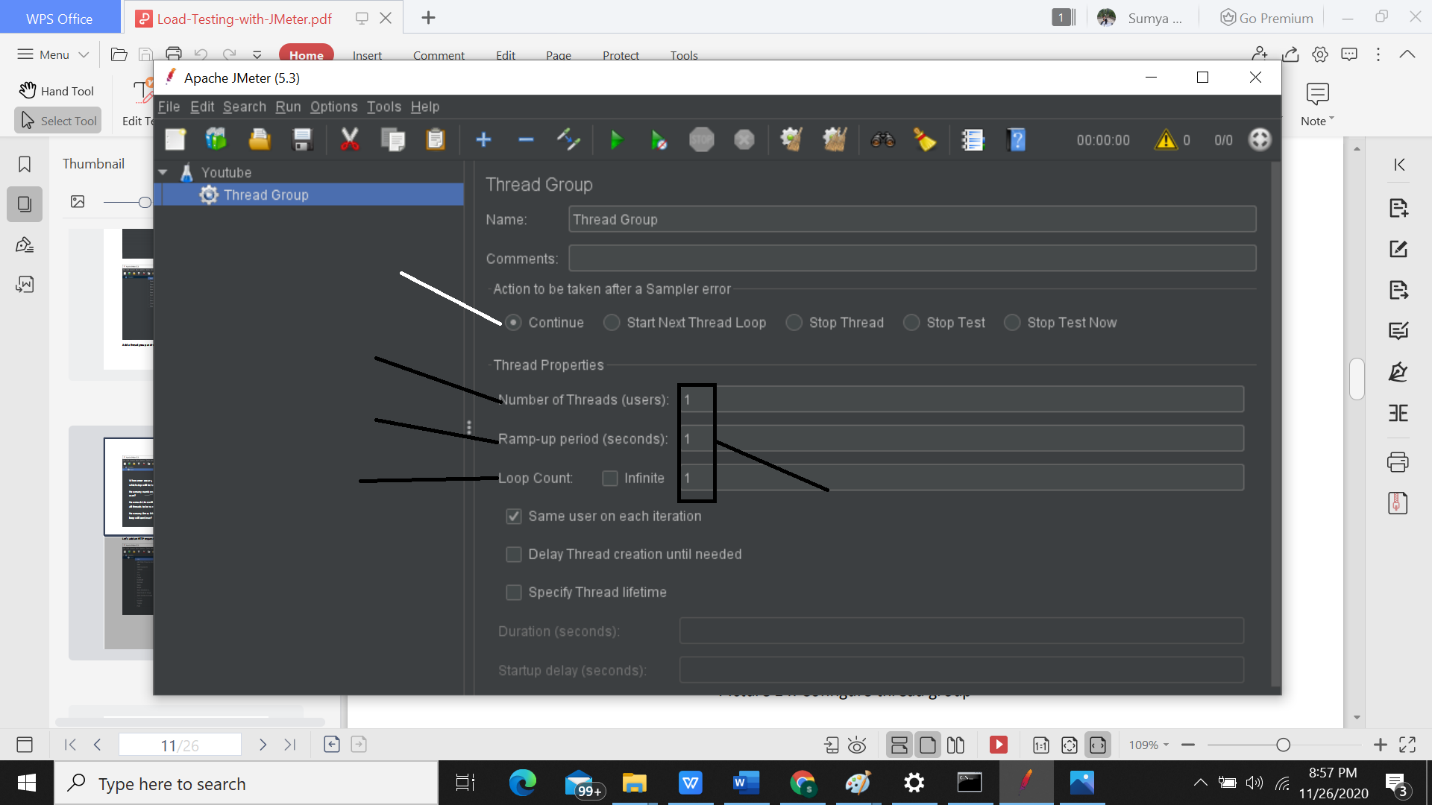
Here write the name and right click this area

Picture : Test name



Picture 13: Add thread group

Add a thread group and rename the group as user.



Here we want to see the

performance of one user

How many times this loop will continue?

How much time will get all threads to be run?

How many numbers of

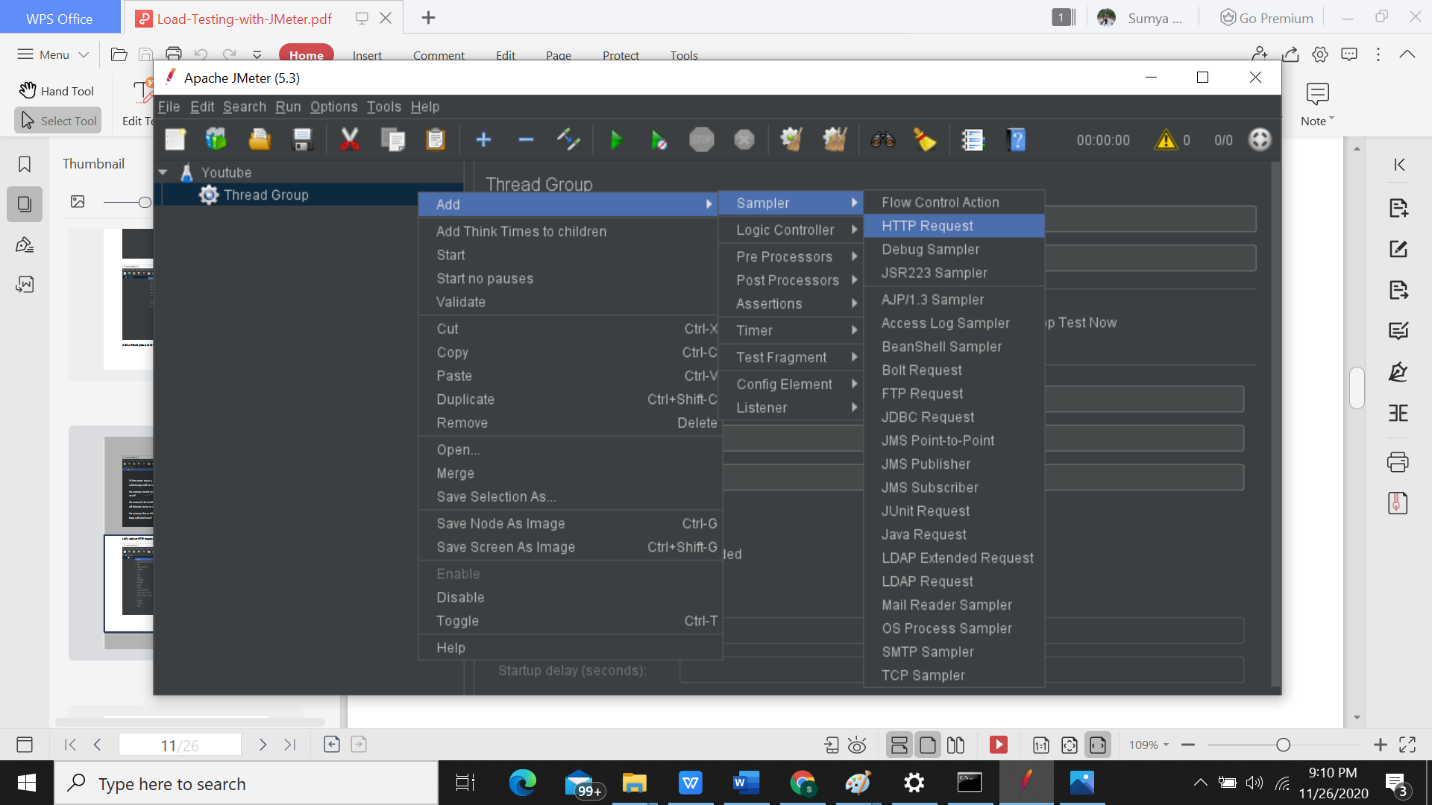
user?

When error occurs,

which step will be taken

Picture 14: Configure thread group

Let’s add an HTTP request sampler. This sampler let you send http or https request to the server.



Picture : Add sampler (HTTP request)

We want to test youtube homepage. So, go to youtube and copy homepage URL. The URL is

https://www.youtube.com/

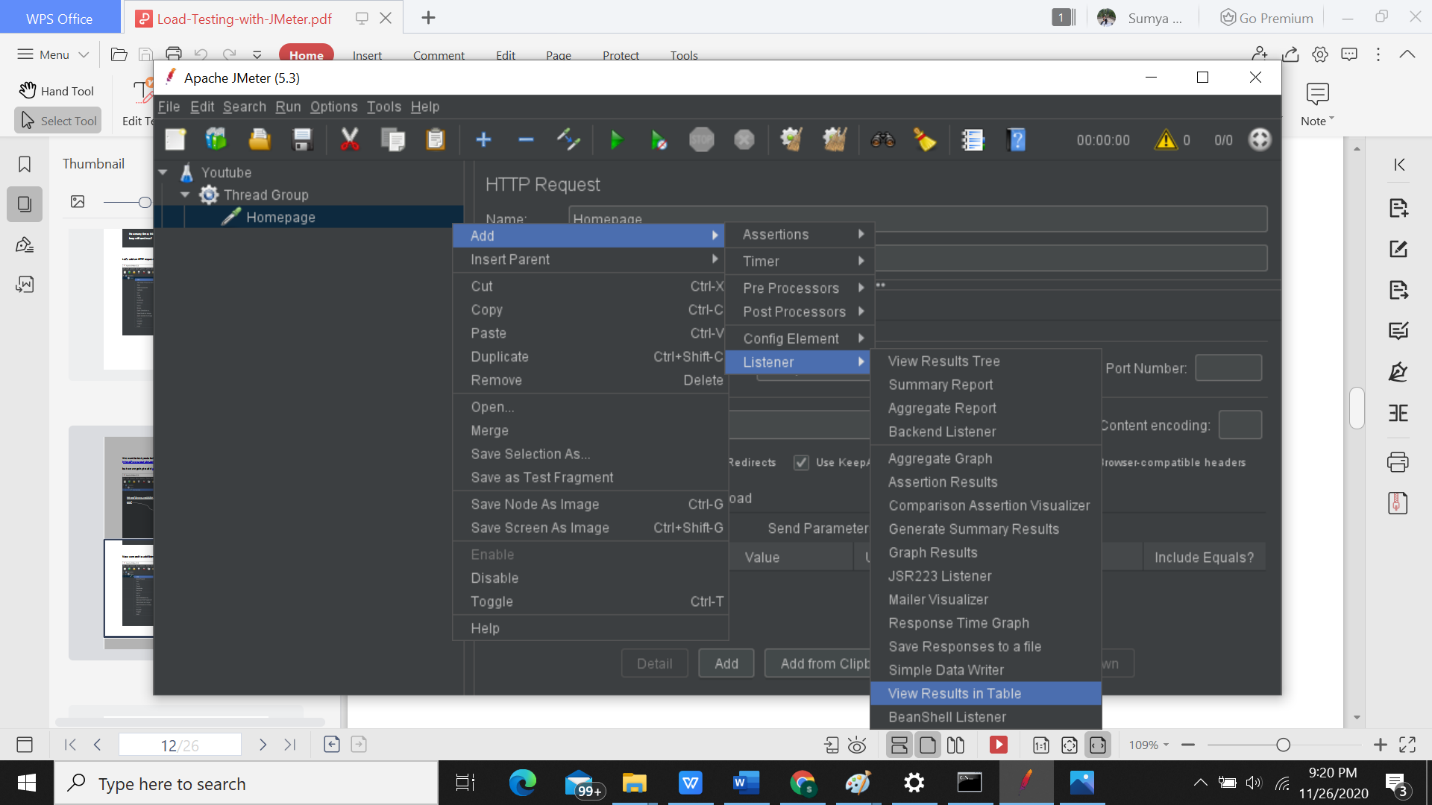
but we are going to add www.youtube.com on server name or IP part.

https://www.youtube.c

om/

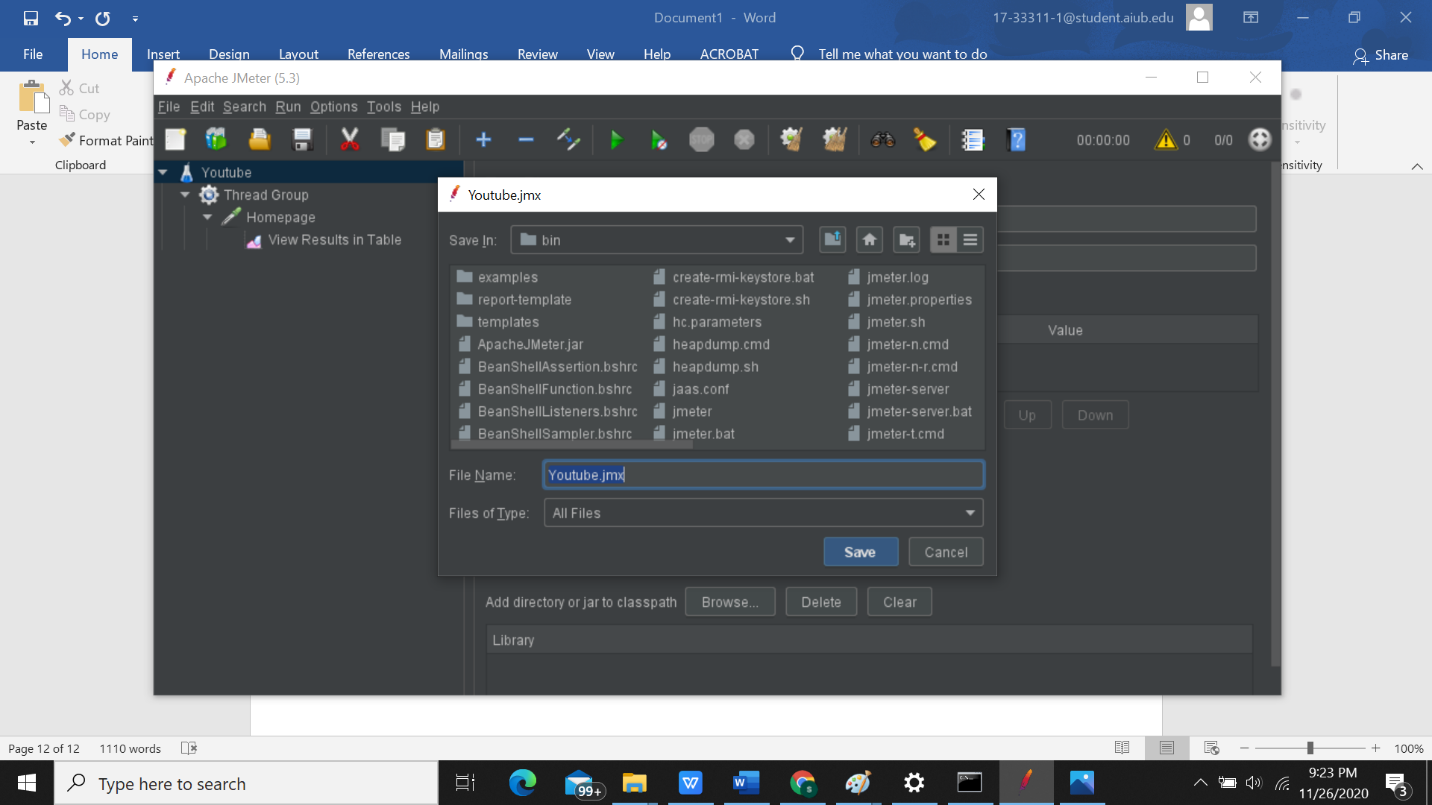
Picture 16: Add HTTP request configuration

Now we need to add listener. Initially we select table format to see result.



Picture: Add listener

So, before running the test plan we need to save it. Then click on run button.



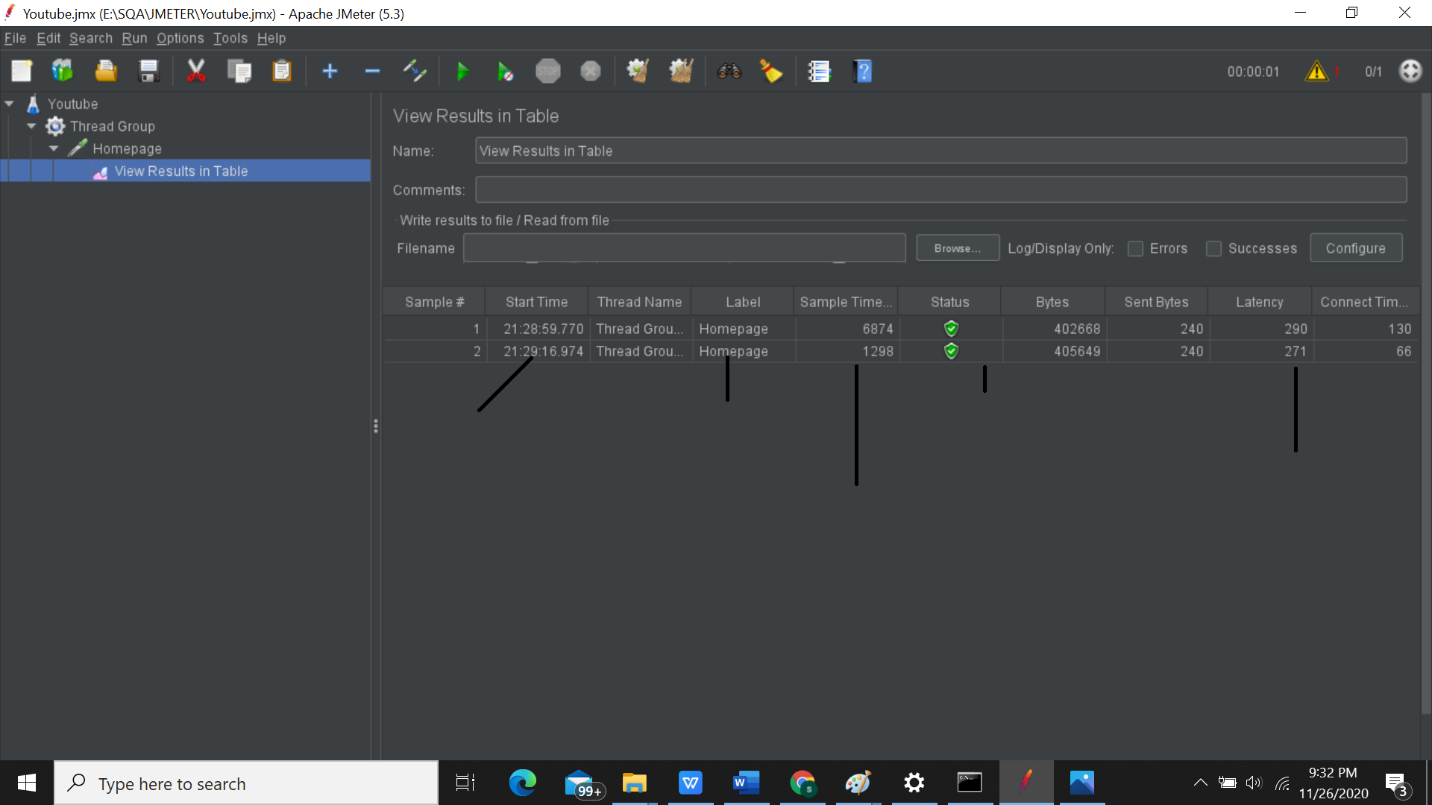
Picture 18: Save Test plan

This file extension is .jmx

I have saved this in E:\SQA\JMETER location of my pc. You can save it any location. But remember this

location because we need to go there.

Now run the test plan then see the result.



Time duration between

before sending request

and get the 1st response

Status is OK

Total time

When the process starts?

We save the level name as Homepage

Picture: Report

We can also see the result in different format using other listeners.

Now we will check the result using 5 threads and loop count 2. So, go to the Thread(user) configuration and edit it.

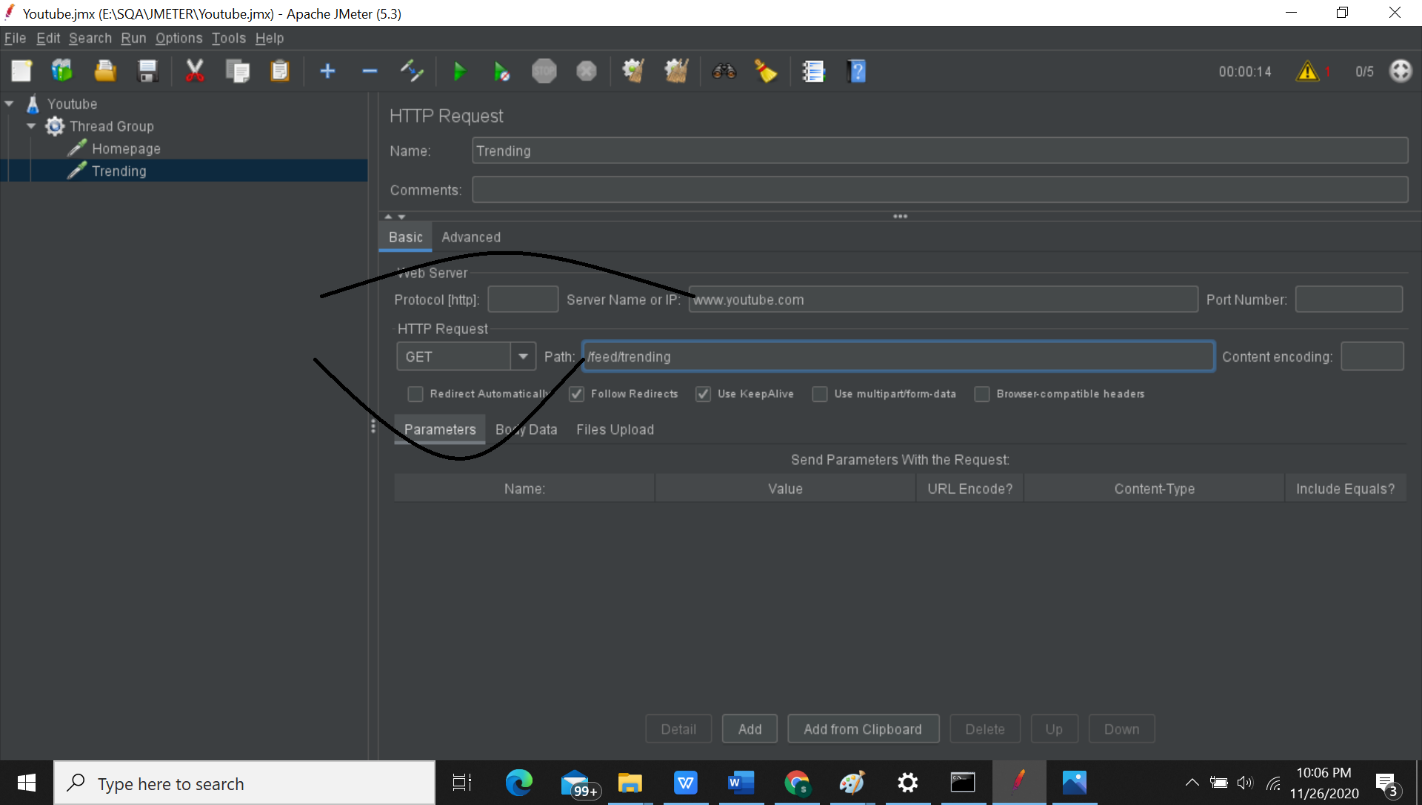
Then run the test.



Picture: Result of 5 threads

Now we add more HTTP request of you tube’s other pages where we can go from the homepage in

same way but this time path will different.

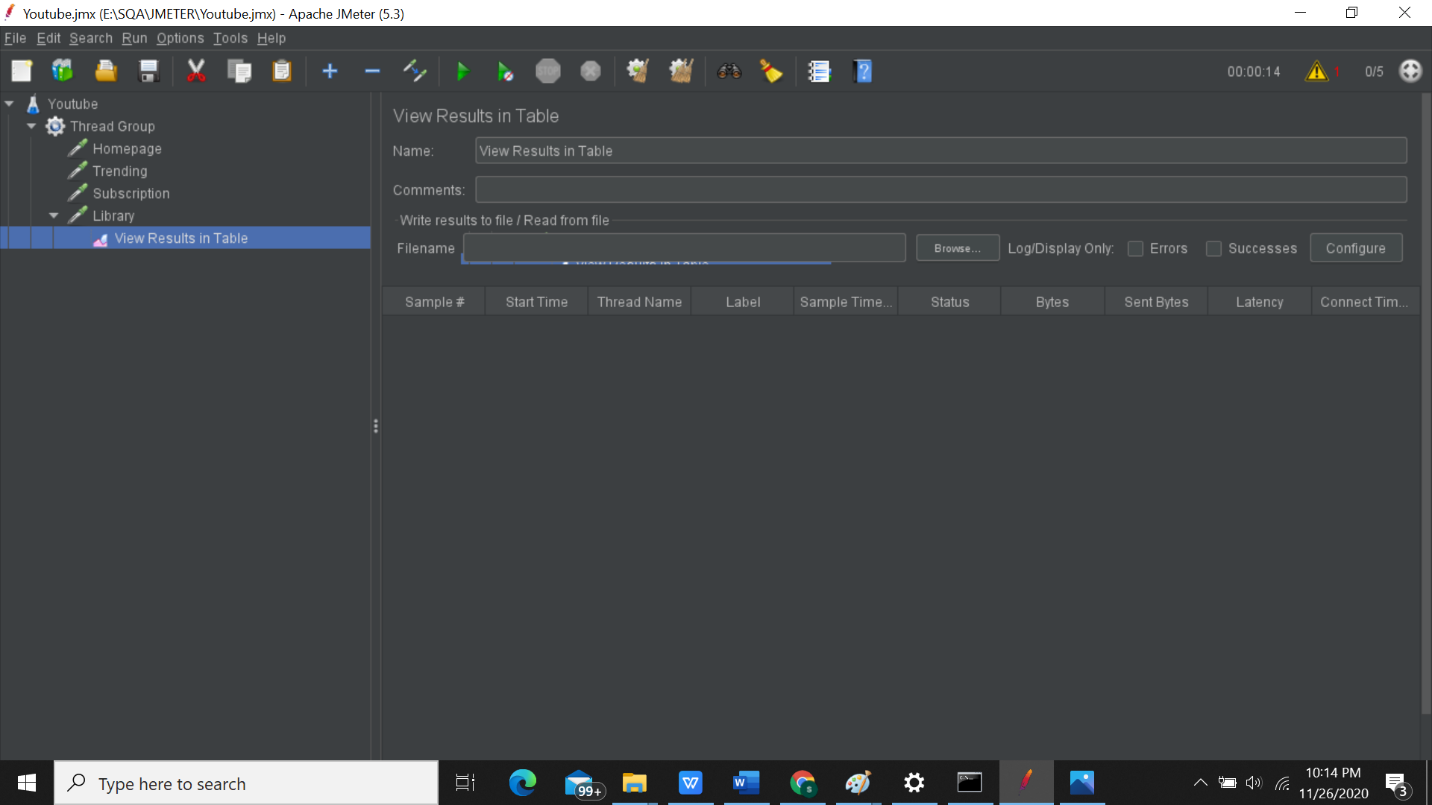


www.youtube.com/feed/trending

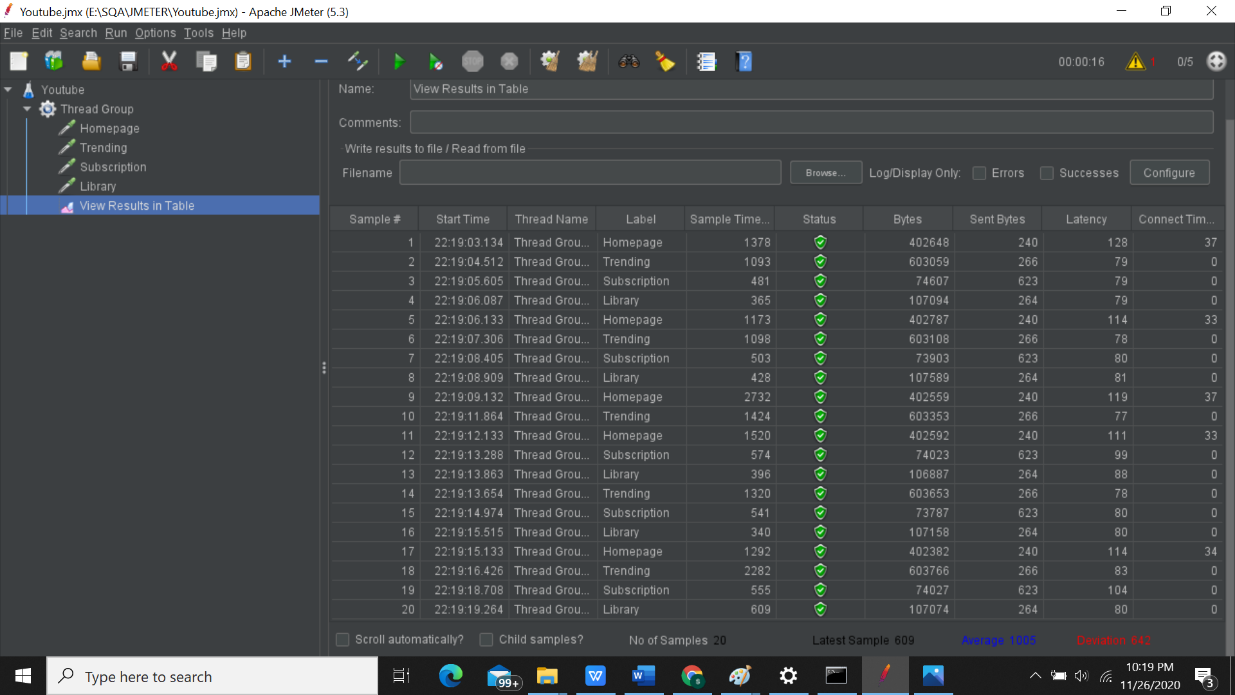
Picture 21: Add another HTTP request

We add YouTube trending website request. Now also add two or three HTTP request of you tube in

same way. Then save it and add table format as listener.



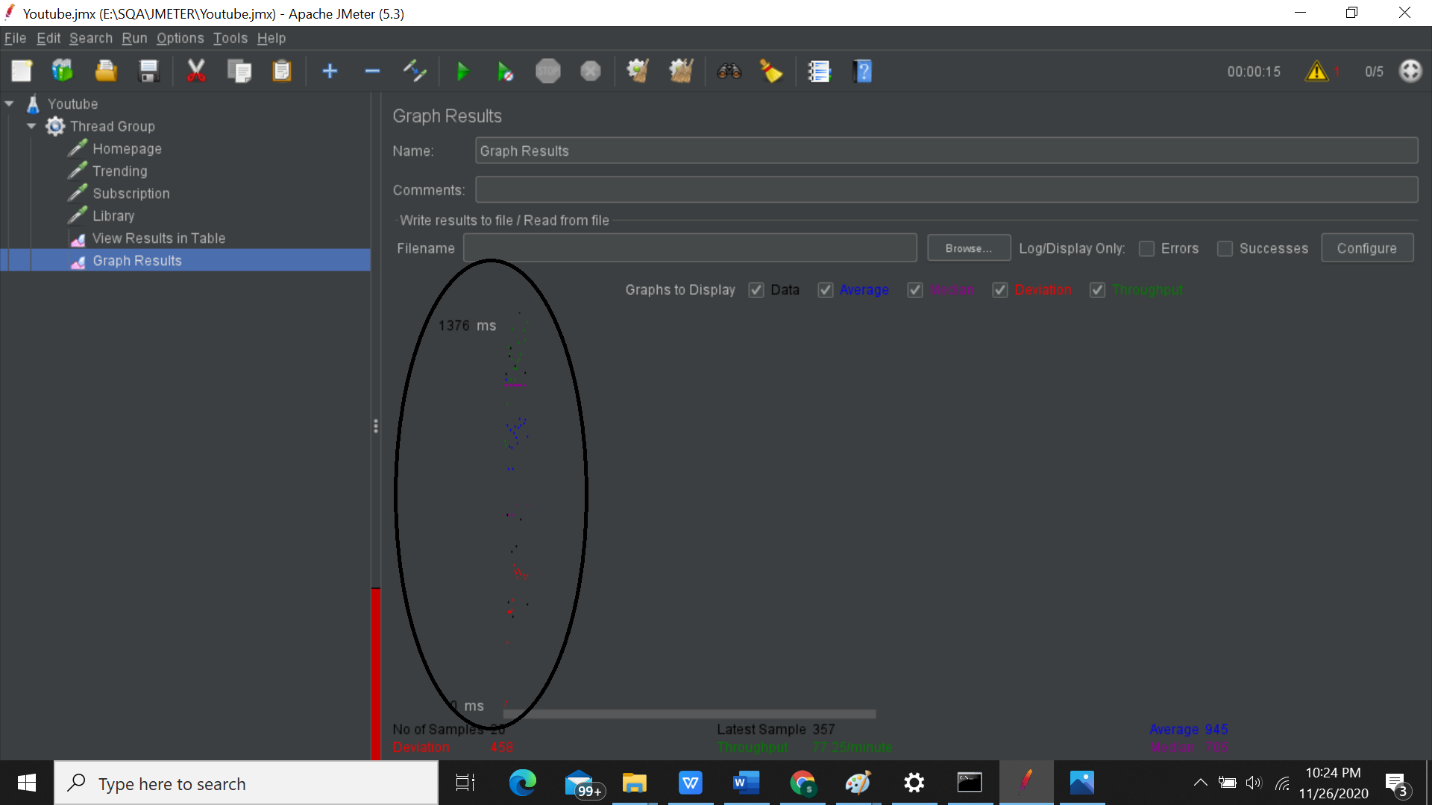
Picture 22: Show all result in table format

Save it and Run.

Picture: Result

Here 5 threads of each makes in total 20 threads. All are successfully run.

Now see the result in another format like graph format.



Picture 24: Graphical representation

So now we can see the result in different format using different listeners. But we already feel trouble to

see the result in GUI format. We cannot see the result properly. That’s why non-GUI Format is come.