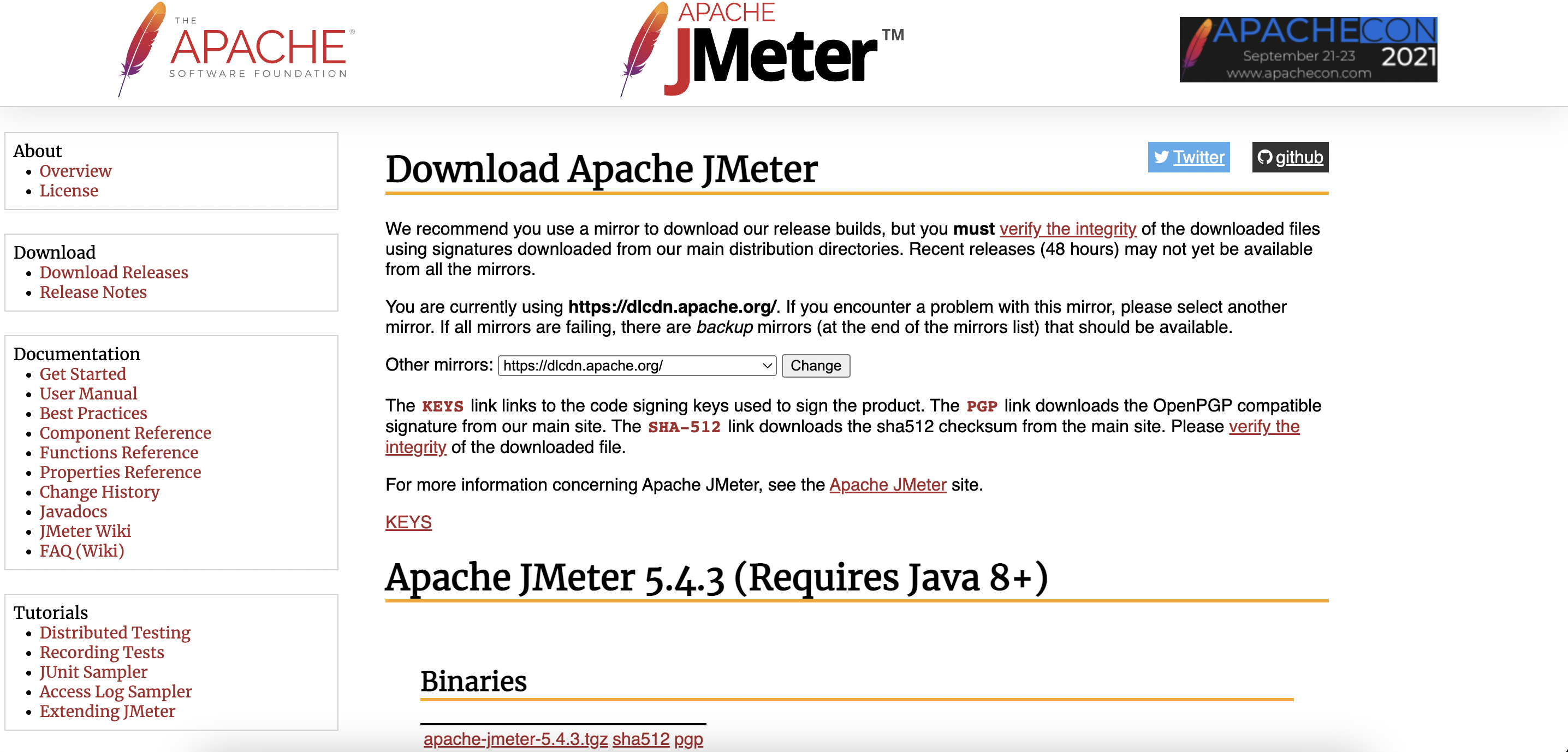
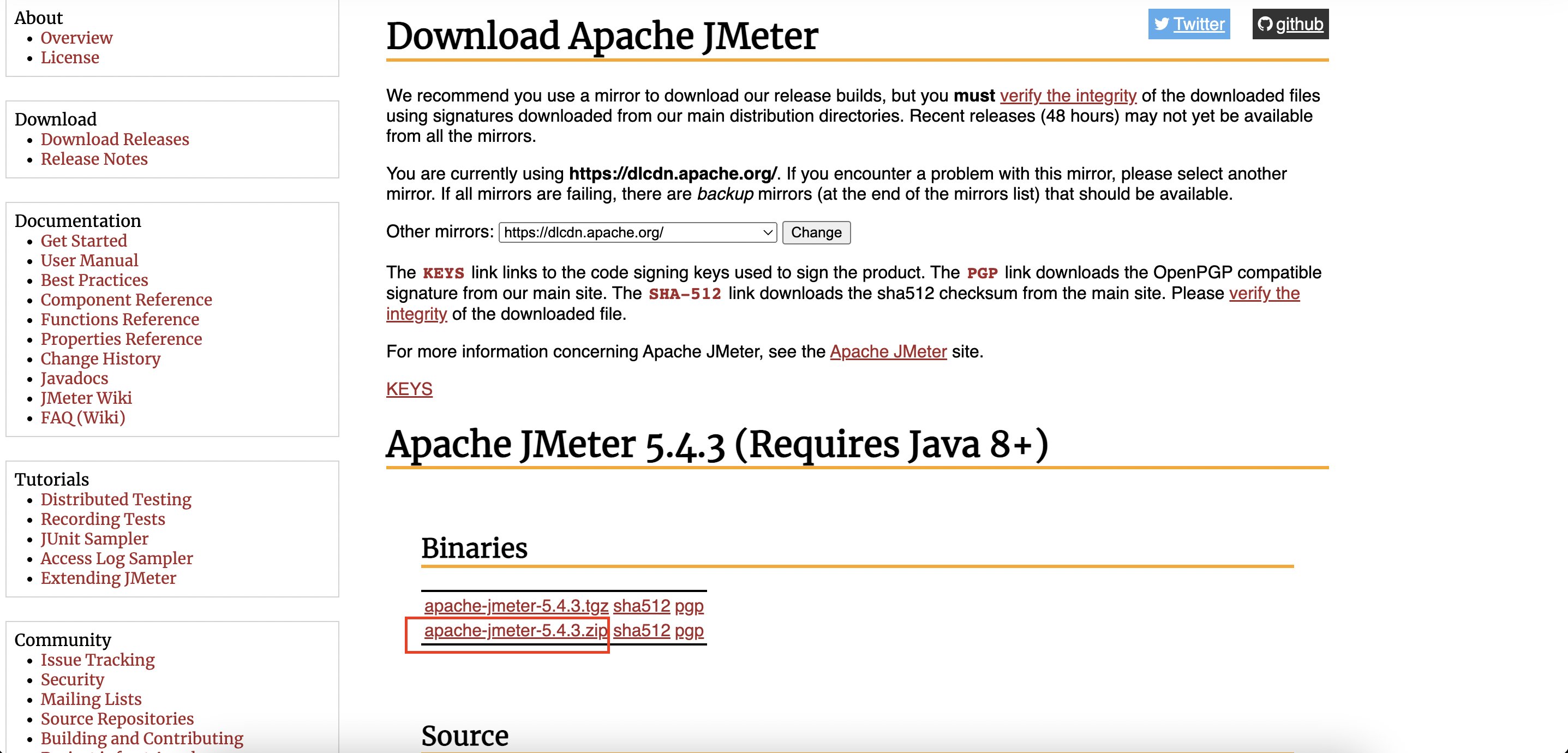
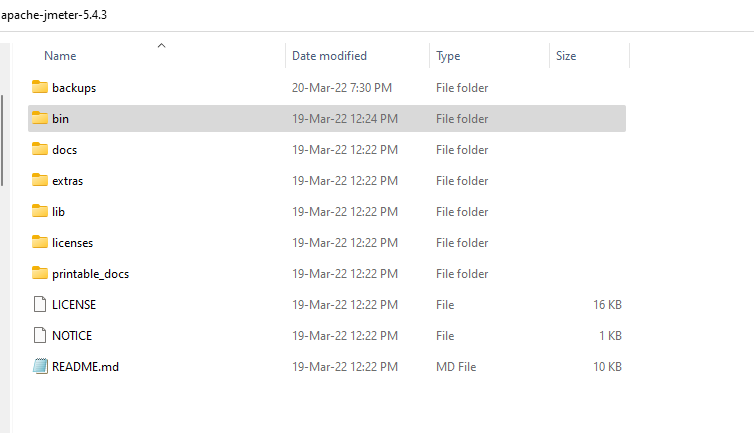
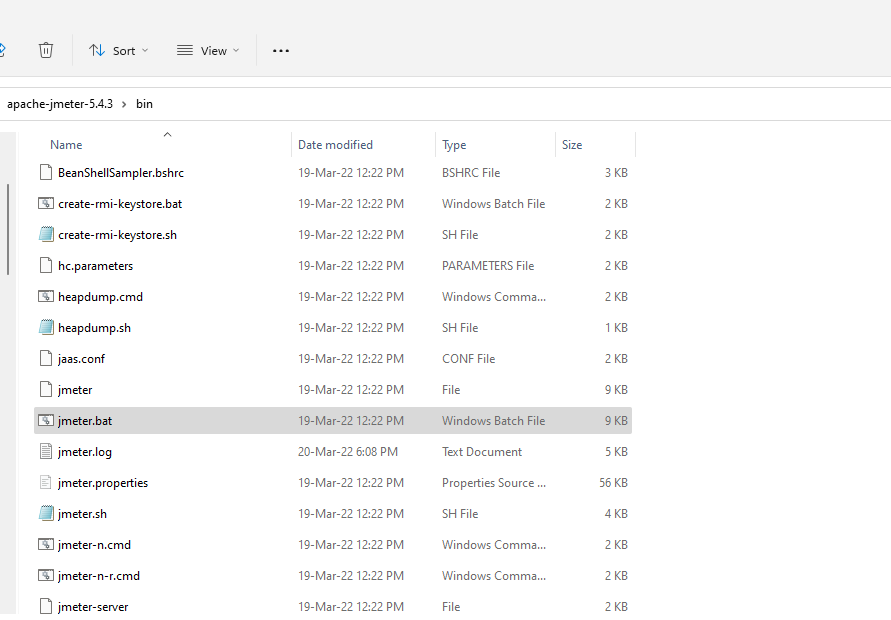
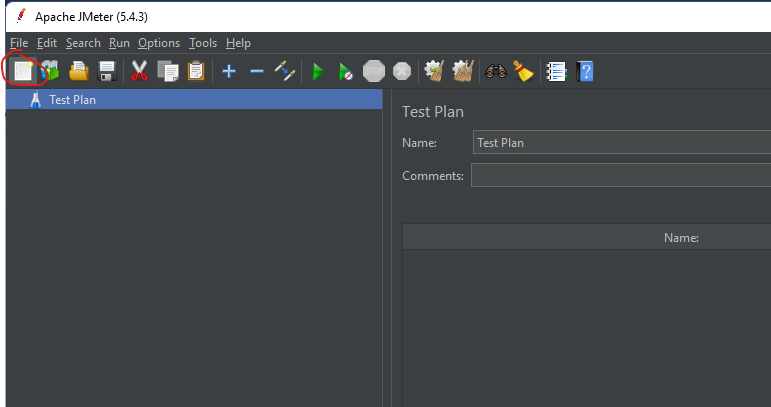
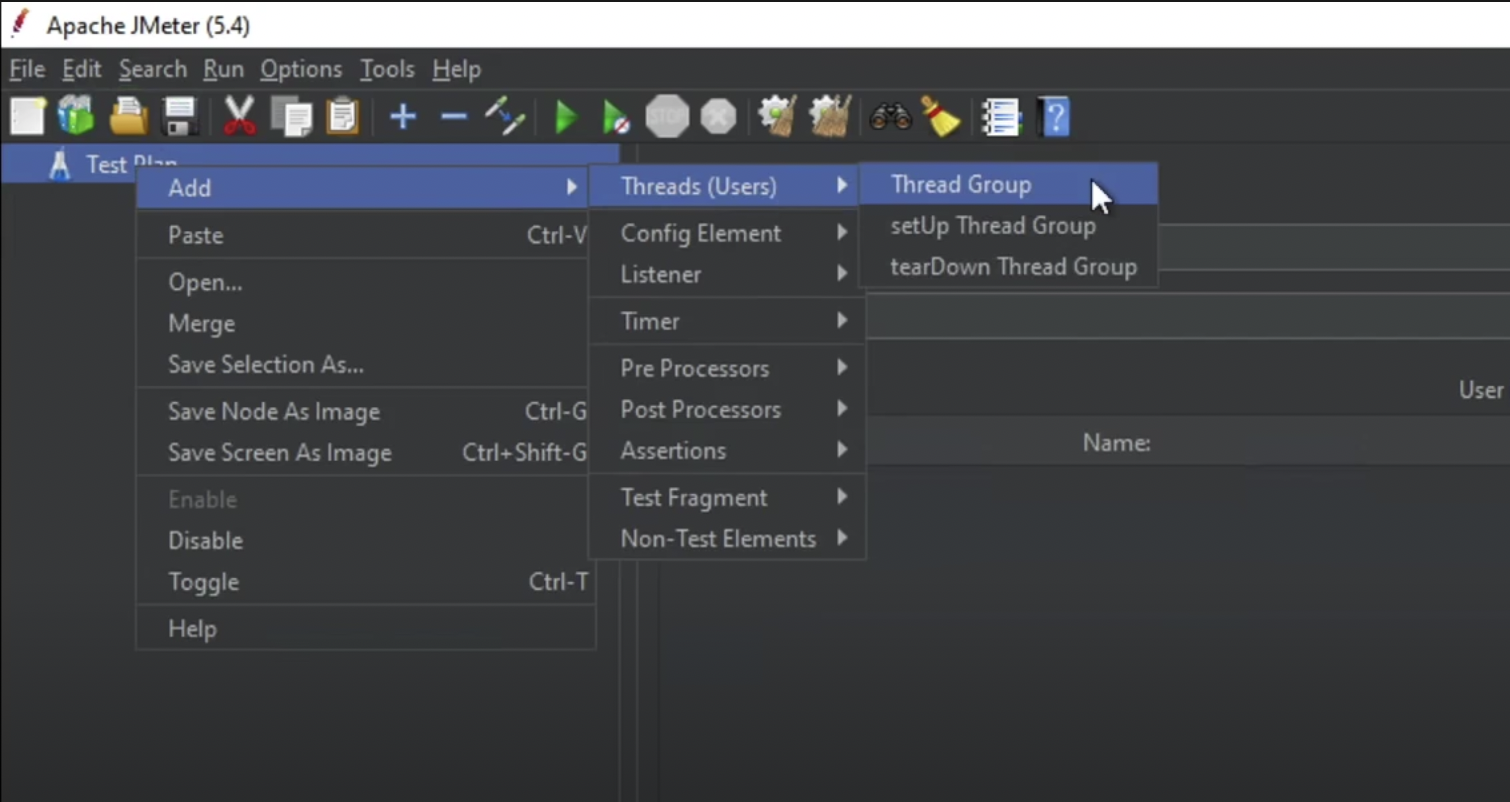
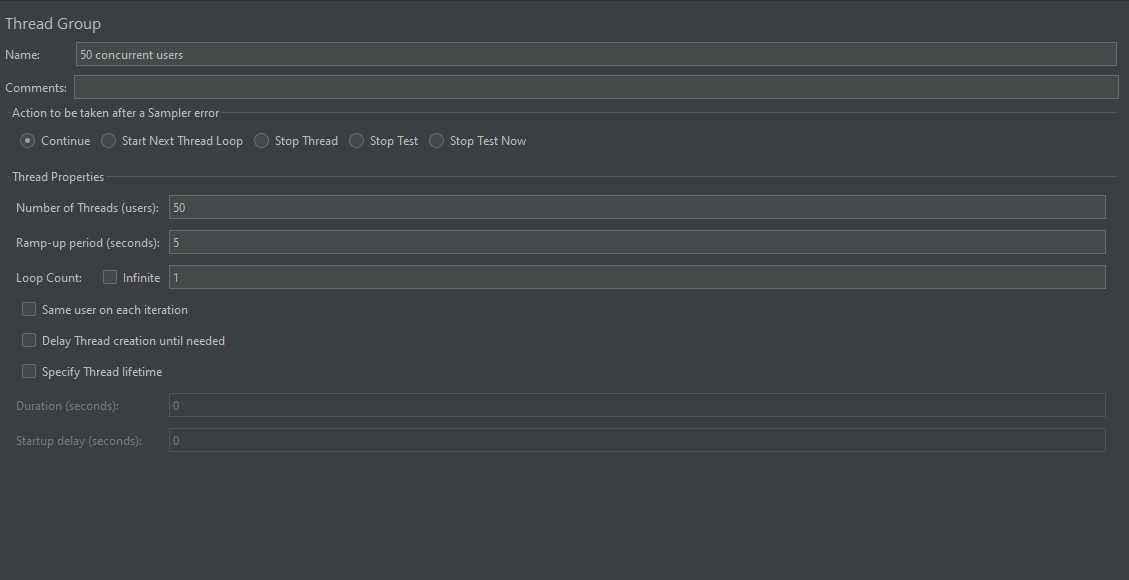
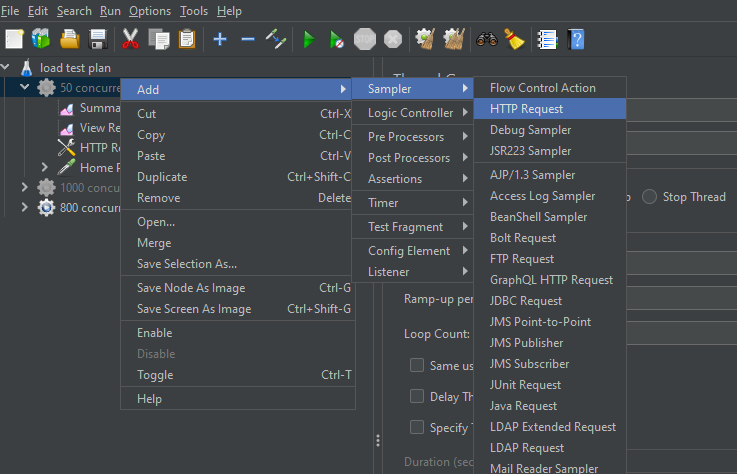
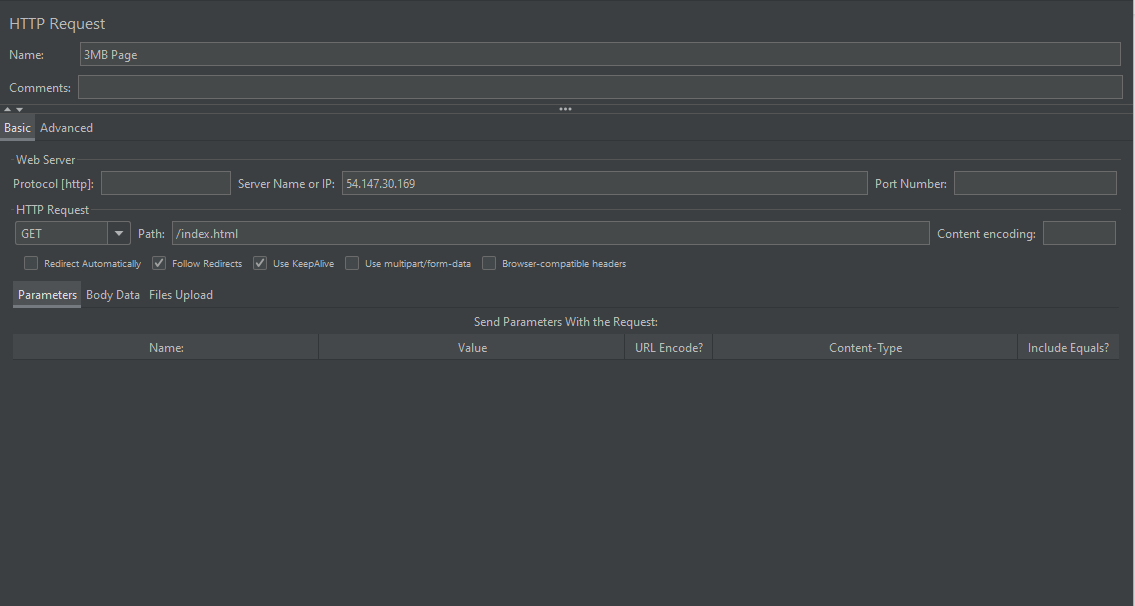
**JMeter and test set up**

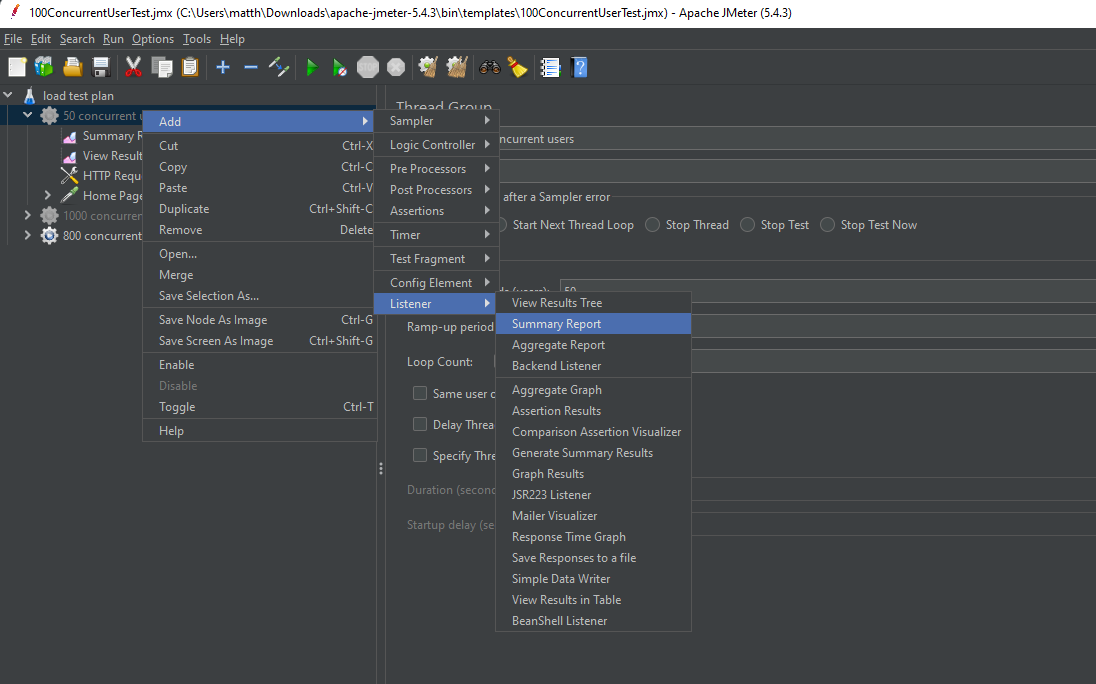
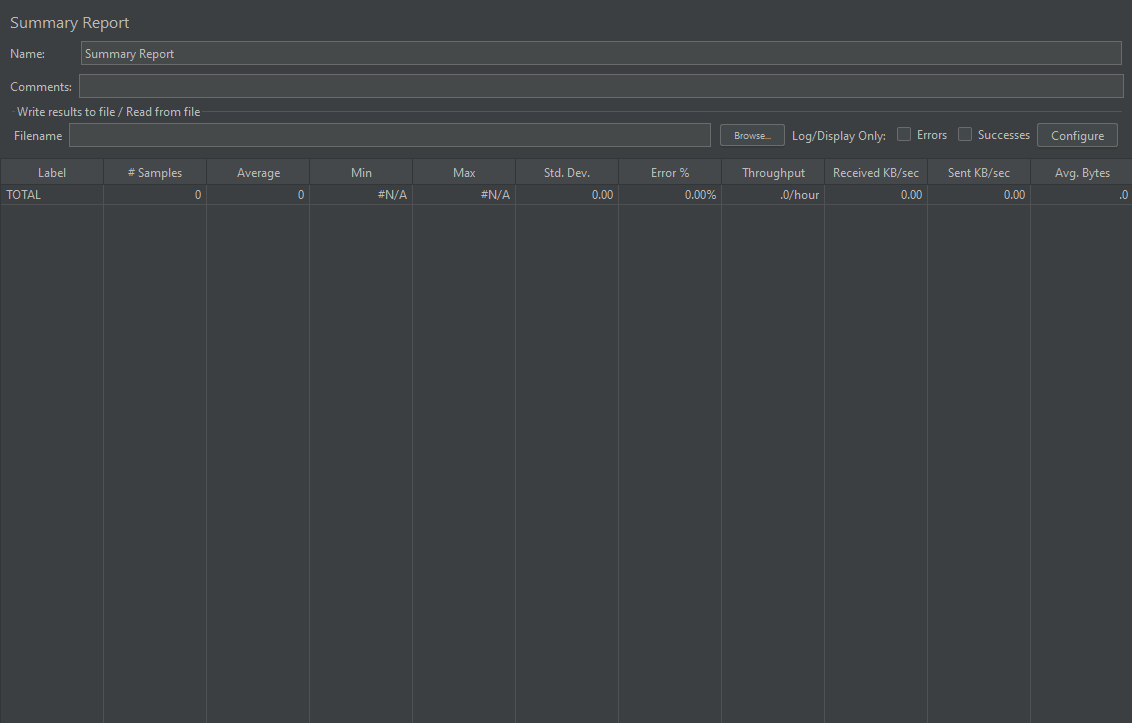
The JMeter is being run on a Window 11 PC and JAVA version 8.   
For Mac, it may have a bit of a difference.  
Please download JAVA before doing the following

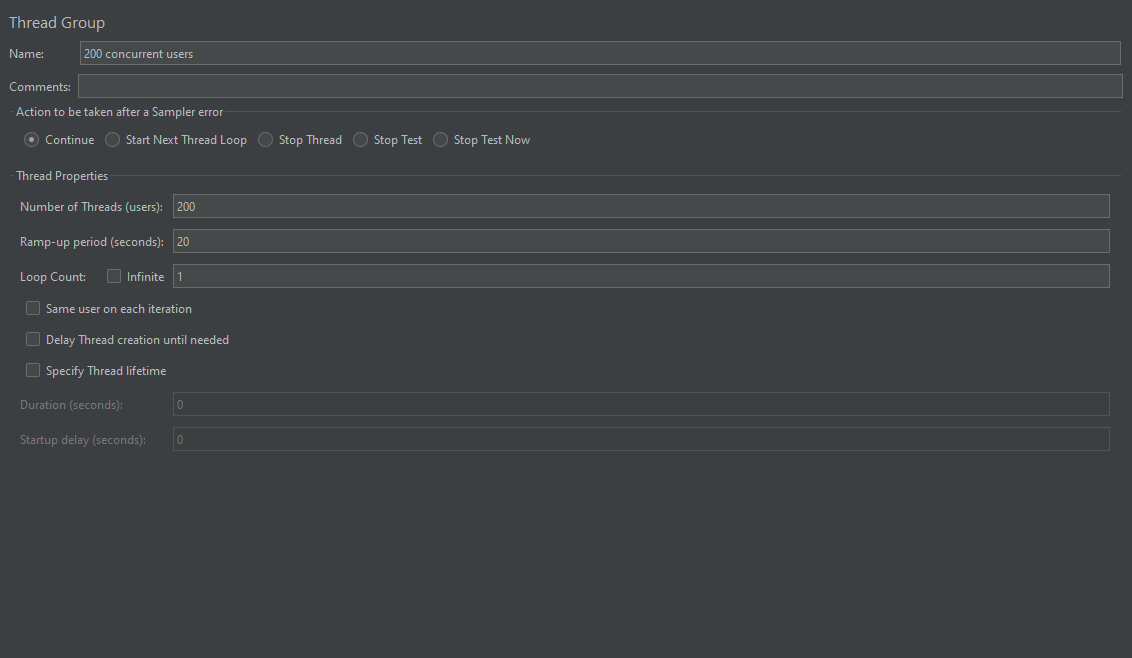
1. Please check whether Java is being installed in the PC, if not, please download Java via the official site <https://www.java.com/download/ie_manual.jsp>
2. After the Java installation has been completed or it has been installed before, please follow the steps below
3. Access the official download site of JMeter (<https://jmeter.apache.org/download_jmeter.cgi>)  
   
4. Download the Apache JMeter 5.4.3 (Requires Java 8+)  
   apache-jmeter-5.4.3.zip  
   
5. Unzip the “apache-jmeter-5.4.3.zip”  
   
6. To start the JMeter application , double click the “jmeter.bat” in the “apache-jmeter-5.4.3/bin”  
     
     
   The cmd and GUI of JMeter should be loaded as below  
   
7. Create a new Test plan by clicking the top left blank paper icon   
   
8. Add a Thread Group by right clicking the “Test plan” created in previous step, hover on “Add” and then select “Thread Group”  
   
9. In the thread group settings, we will have the following settings:
   1. Name: “50 concurrent user”
   2. Action to be taken after a sample error
   3. Thread properties
      1. Number of threads
         1. 50
      2. Ramp up period
         1. 5
      3. Loop count
         1. 1
   4. Same user on each iteration
      1. Untick
   5. Delay thread creation until needed
      1. Untick
   6. Specify Thread lifetime
      1. Untick



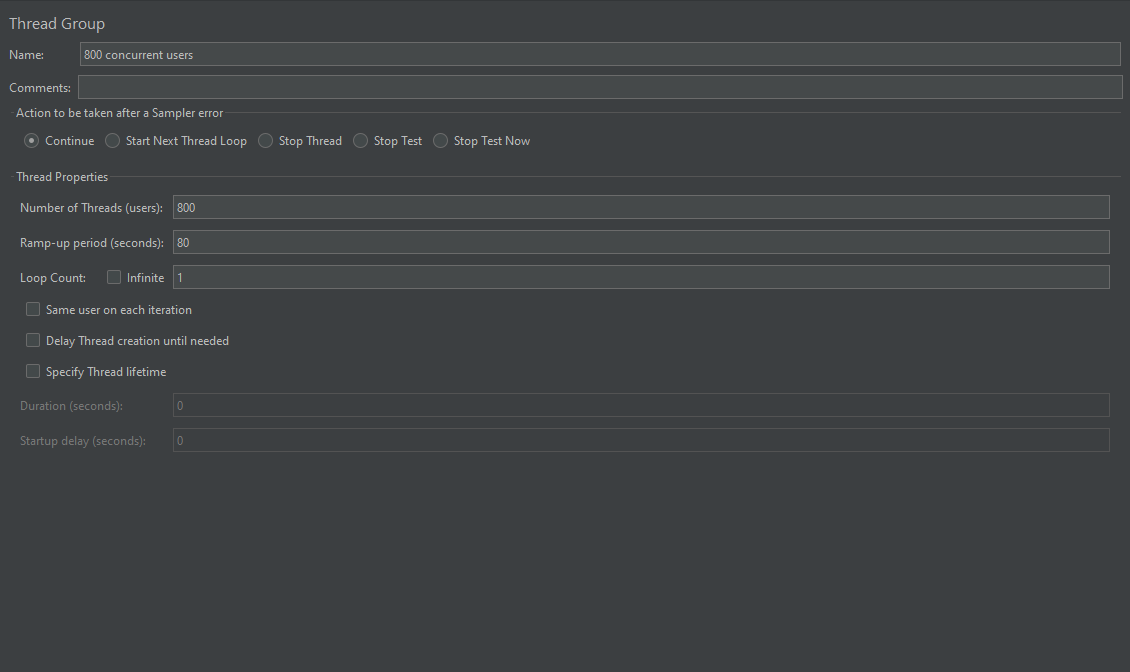
1. Add a new HTTP request by right clicking the “50 concurrent user” thread group, hover on “Add”, then hover on “Sampler”, lastly select “HTTP Request”  
   
2. In the “HTTP Request”, we will have the following settings:
   1. Name: “3 MB page”
   2. Basic
      1. Web server
         1. Protocol (http):
         2. Server Name or IP
            1. (Please insert the public IP address of one of the EC2 servers)
         3. Port number
            1. (In the case of Nginx, u may need to insert the respective port number that u have set for Nginx load balancer)
         4. HTTP request
            1. GET
         5. Path
            1. /index.html
         6. Parameter (all blanks)

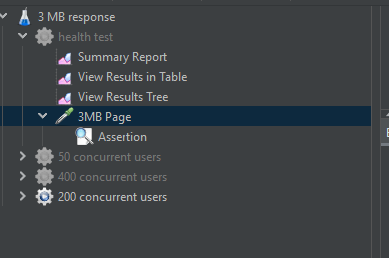
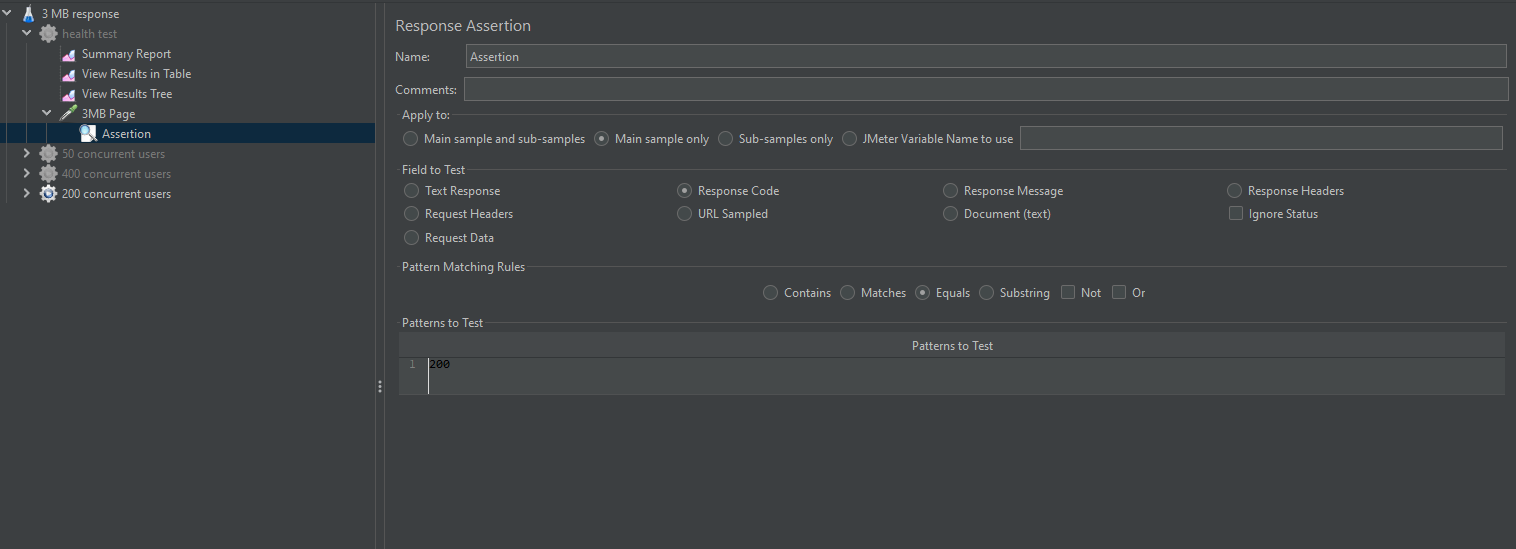


1. Add a new HTTP request by right clicking the “50 concurrent user” thread group, hover on “Add”, then hover on “Listener”, lastly select “Summary Report”  
   
2. The result of the test can be found in the “Summary Report”  
   
3. For establishing the test on 200 concurrent users, repeat step 8 – 13, but with different settings in the thread group settings, we will have the following settings:
   1. Name: “200 concurrent user”
   2. Action to be taken after a sample error
   3. Thread properties
      1. Number of threads
         1. 200
      2. Ramp up period
         1. 20
      3. Loop count
         1. 1
   4. Same user on each iteration
      1. Untick
   5. Delay thread creation until needed
      1. Untick
   6. Specify Thread lifetime
      1. Untick

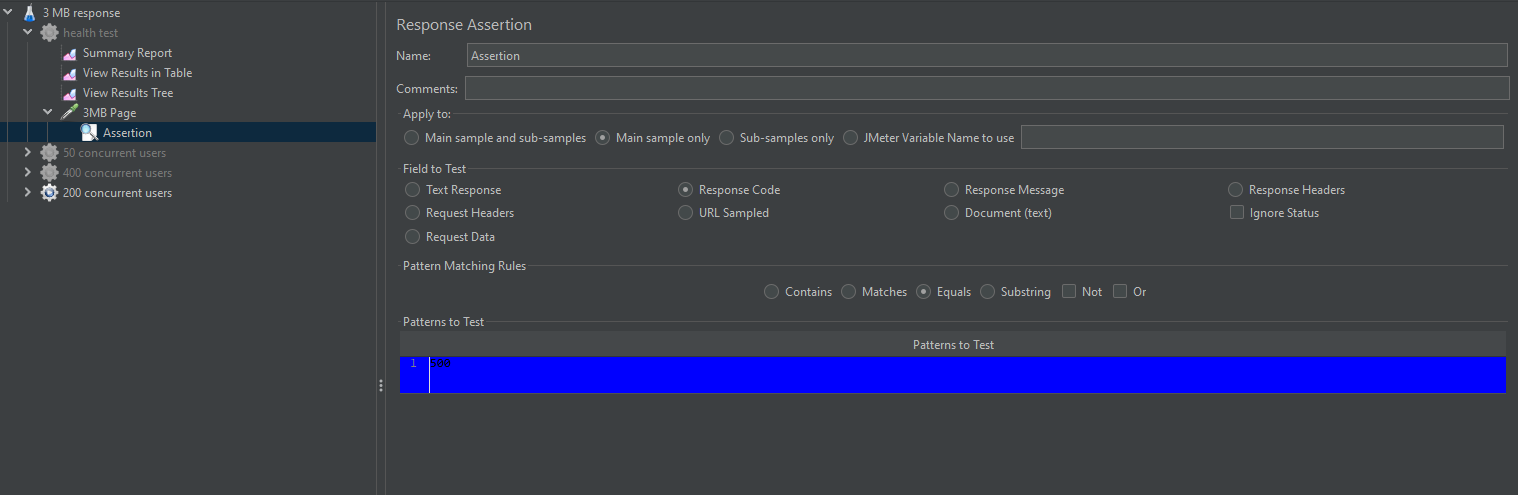
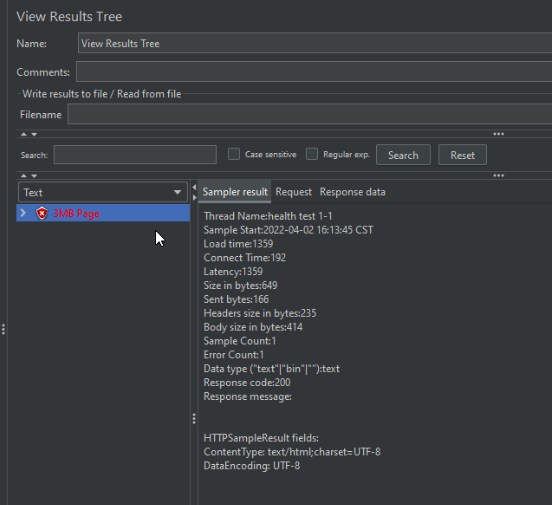
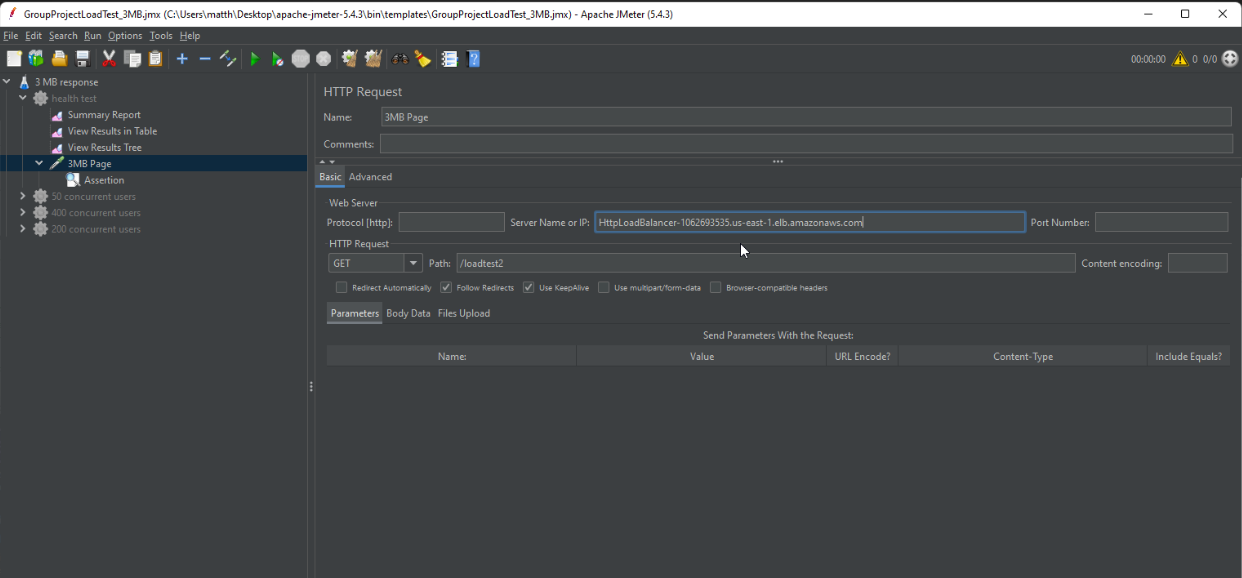
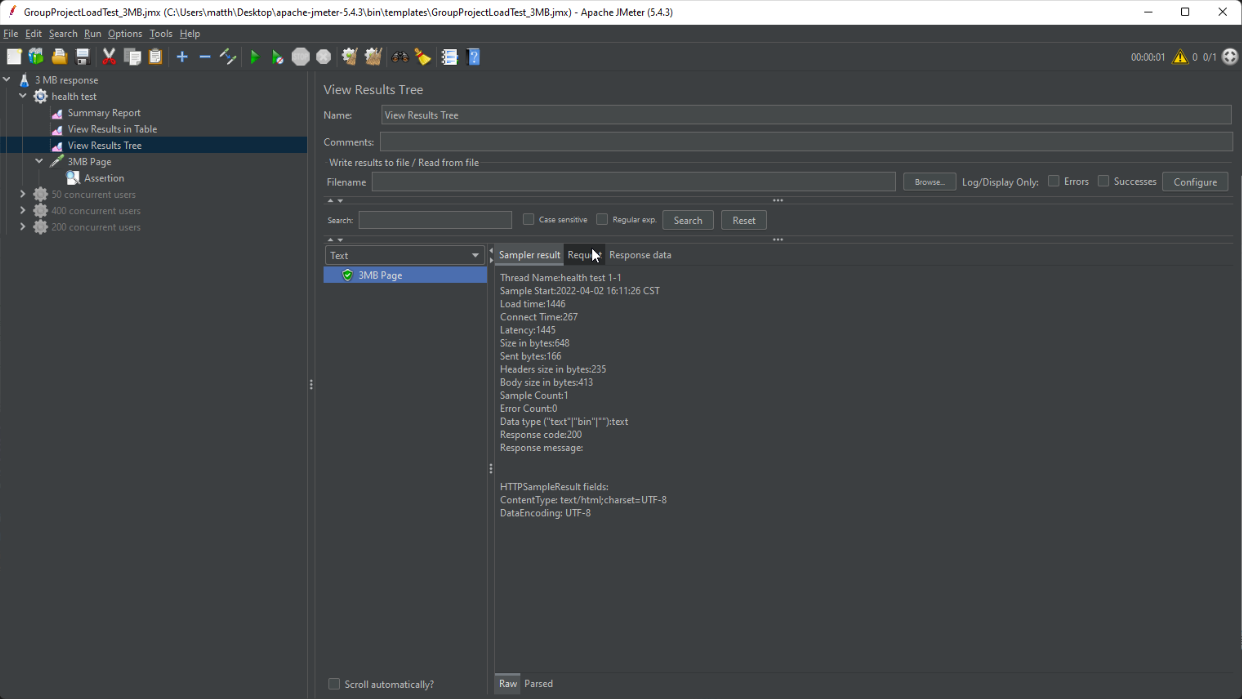
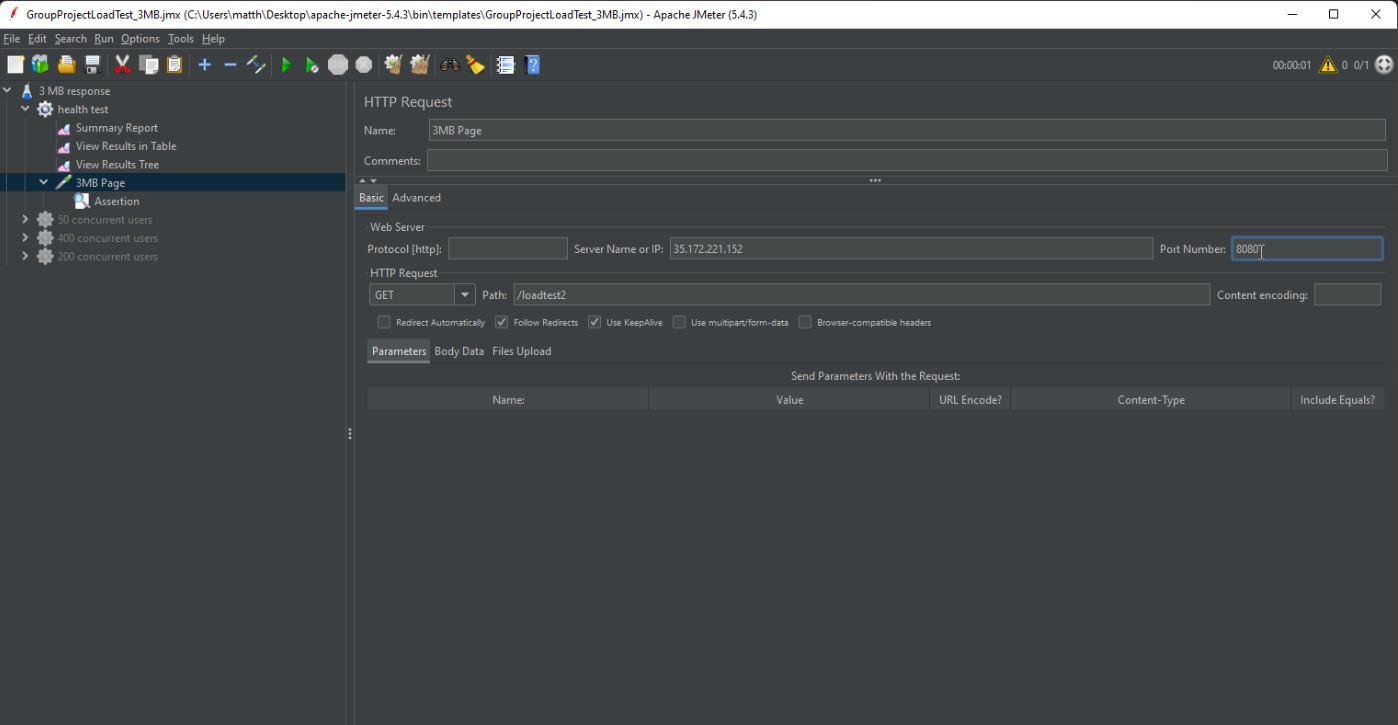
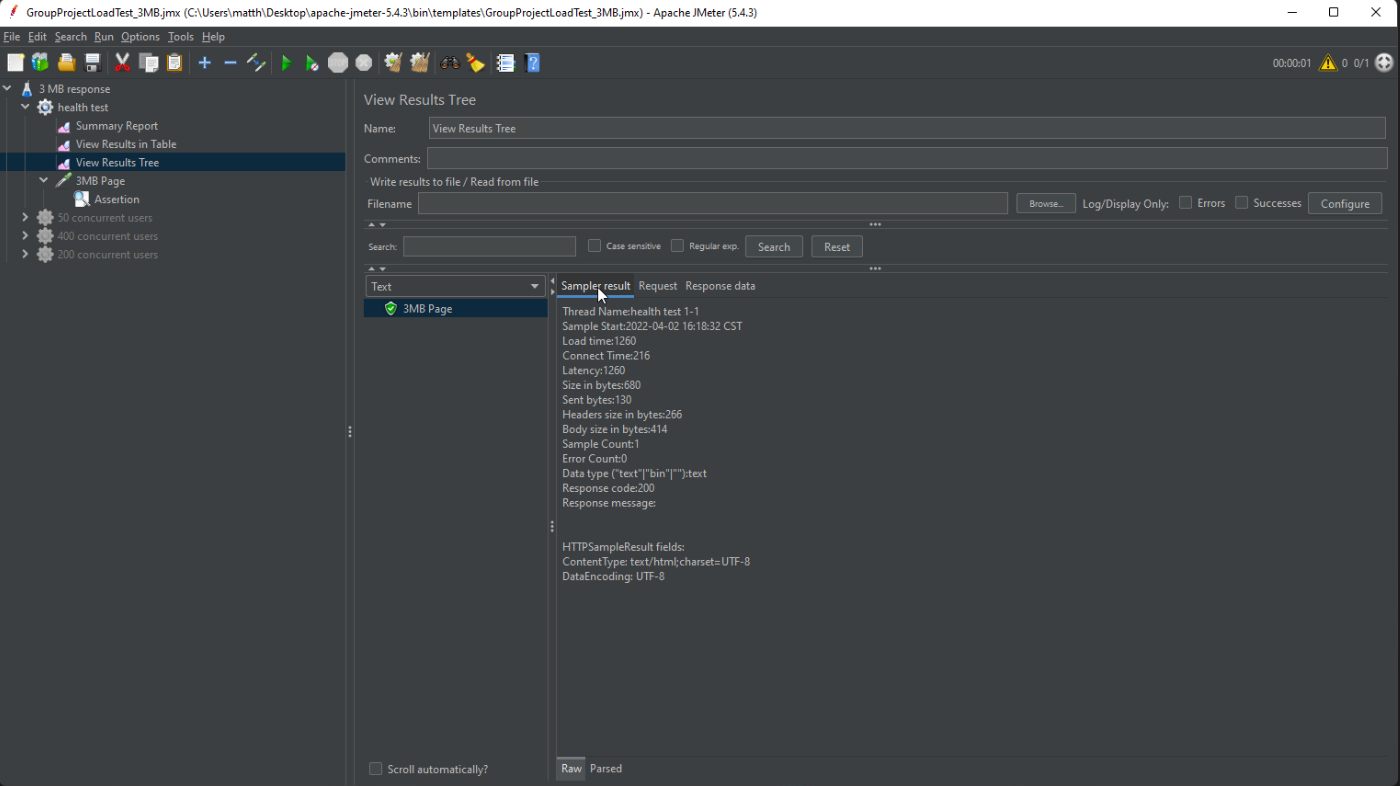
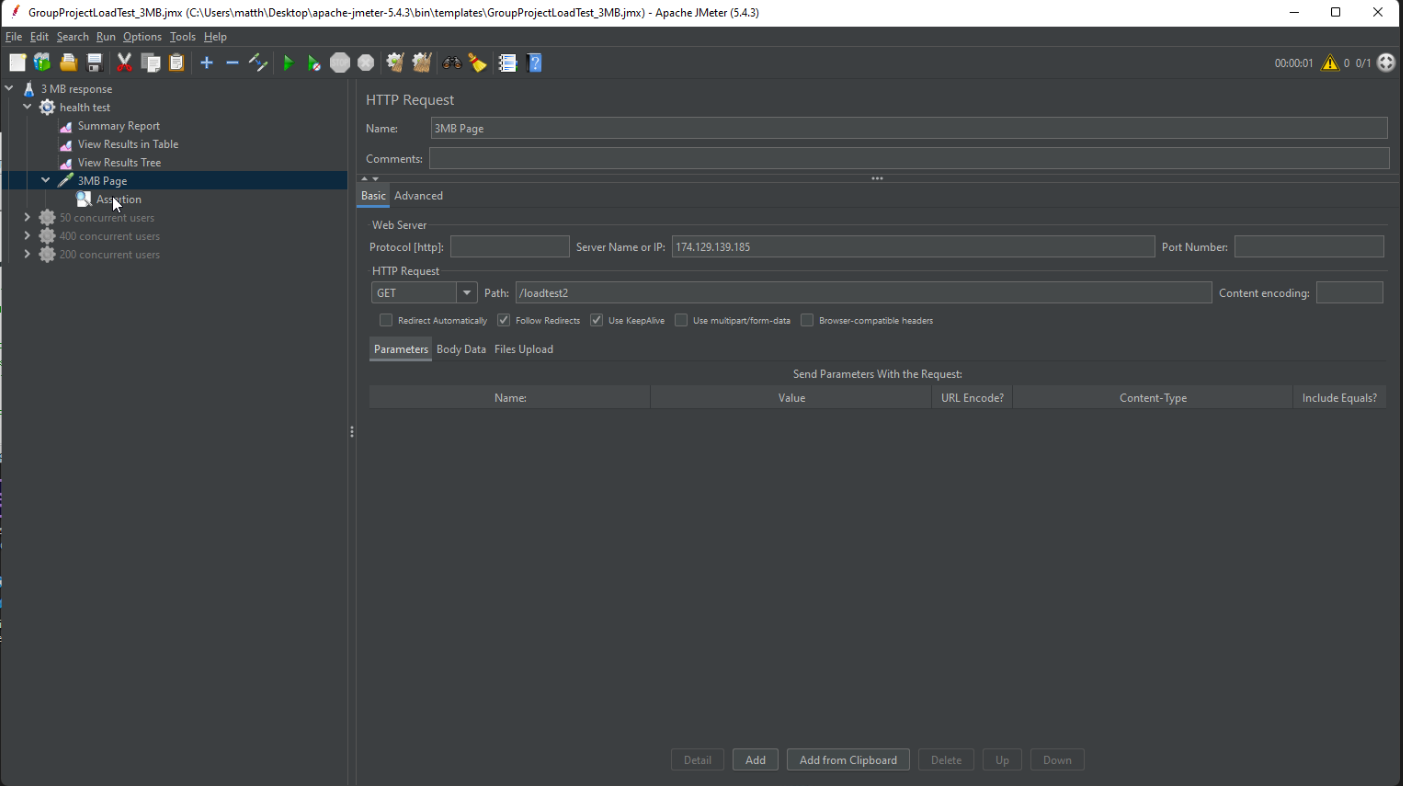
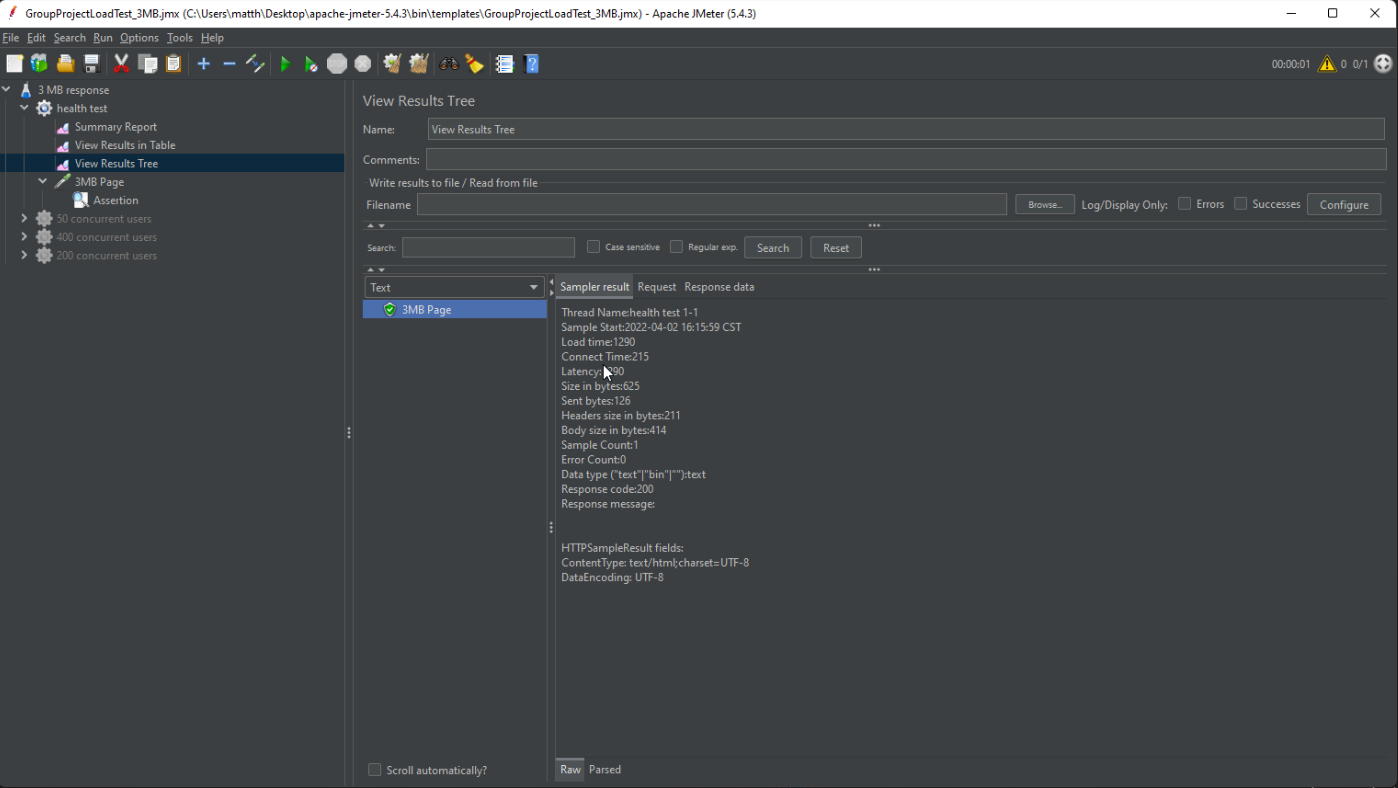


1. For establishing the test on 800 concurrent users, repeat step 8 – 13, but with different settings in the thread group settings, we will have the following settings:
   1. Name: “800 concurrent user”
   2. Action to be taken after a sample error
   3. Thread properties
      1. Number of threads
         1. 800
      2. Ramp up period
         1. 80
      3. Loop count
         1. 1
   4. Same user on each iteration
      1. Untick
   5. Delay thread creation until needed
      1. Untick
   6. Specify Thread lifetime
      1. Untick



1. For testing 5MB response size, change the settings in the “HTTP Request”, we will have the following settings:
   1. Name: “5MB page”
   2. Basic
      1. Web server
         1. Protocol (http):
         2. Server Name or IP
            1. (Please insert the public IP address of one of the EC2 servers)
         3. Port number
            1. (In the case of Nginx, u may need to insert the respective port number that u have set for Nginx load balancer)
         4. HTTP request
            1. GET
         5. Path
            1. /index.html
         6. Parameter (all blanks)
2. For testing 1MB response size, change the settings in the “HTTP Request”, we will have the following settings:
   1. Name: “1MB page”
   2. Basic
      1. Web server
         1. Protocol (http):
         2. Server Name or IP
            1. (Please insert the public IP address of one of the EC2 servers)
         3. Port number
            1. (In the case of Nginx, u may need to insert the respective port number that u have set for Nginx load balancer)
         4. HTTP request
            1. GET
         5. Path
            1. /index.html
         6. Parameter (all blanks)
3. For setting the assertion of the webpage, first left click on the HTTP Request top show the assertion section  
    
4. Then, choose “response code” in the “Field to test”, select “Equals” in the “Pattern Matching Rules” and type “200” in “Pattern to Test”. The reason is that whenever the webpage response with 200 we treat it as a success, otherwise, it will be an error.  
   

**JMeter health checks**

1. Assertion check
   1. As to ensure the assertion is set correctly, here is a test to ensure the assertion is working as intended.
      1. We first type “500” in “Pattern to Test”.  
         
      2. Then we run the test. Afterwards, you could check at the “View Results Tree”. If the webpage is returning 200, it should treated as an error. This shows that the assertion is working as intended.  
         
      3. Please edit the “500” back to “200” in the “Pattern to Test” of the assertion page.
2. ALB health check
   1. As to ensure the servers configurated with ALB can be accessed by the JMeter, here is a test to ensure the the ALB is working as intended.
      1. In the “HTTP Request” page, change the “Server Name or IP” field to the ALB domain name.  
         
      2. Run the test and view the “View Results Tree”. There should be a correct result.  
         
3. Nginx health check
   1. As to ensure the servers configured with Nginx can be accessed by the JMeter, here is a test to ensure it is working as intended.
      1. In the “HTTP Request” page, change the “Server Name or IP” field to the IP and port number of the server that installed the Nginx.  
         
      2. Run the test and view the “View Results Tree”. There should be a correct result.  
         
4. HAproxy health check
   1. As to ensure the servers configured with HAproxy can be accessed by the JMeter, here is a test to ensure it is working as intended.
      1. In the “HTTP Request” page, change the “Server Name or IP” field to the IP and port number of the server that installed the HAproxy.  
         
      2. Run the test and view the “View Results Tree”. There should be a correct result.  
         

**JMeter example setup**

There are example setups in the github project directory:  
<https://github.com/CityUGroup3/CityUGroup3/tree/main/JMeter_set_up_procedure/example>

You could reference the settings there. But please be reminded that the “Server Name or IP” and “Port number” in the “HTTP Request” section will need to be changed to your destinated servers.