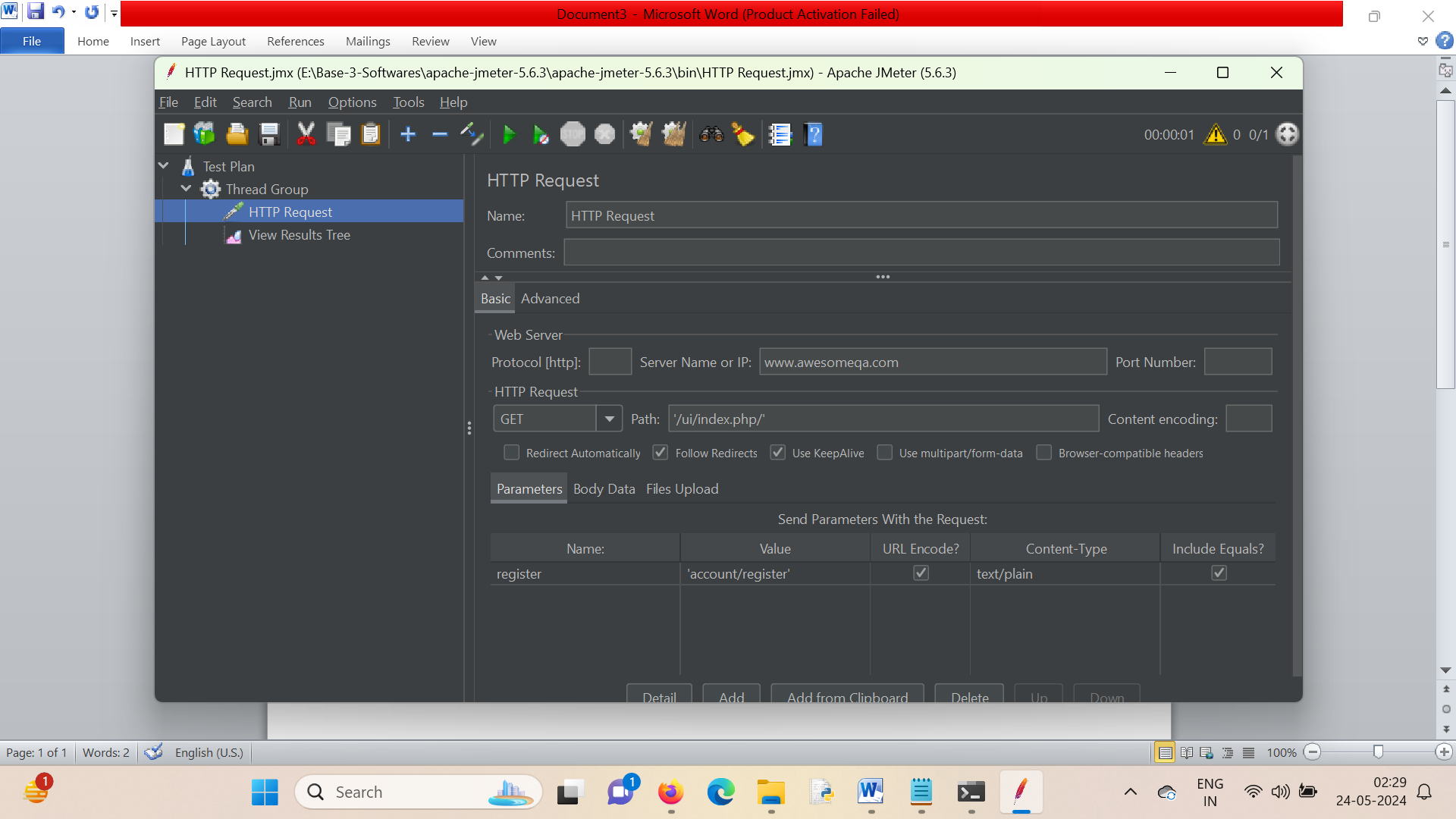
E:\Base-2\Self Learning \_ Dec2023\Phase-3-NonFunctional Testing\Jmeter 5.6 Content

Scenario-1:

<https://awesomeqa.com/ui/index.php?route=account/register>



Scenario-Test-001

To perform performance testing on the URL https://awesomeqa.com/ui/ with 10 users using Apache JMeter, follow these steps:

1. **Download and Install JMeter:**
   * Ensure you have Java installed on your machine.
   * Download JMeter from the [official website](https://jmeter.apache.org/download_jmeter.cgi).
   * Extract the downloaded file and navigate to the bin directory.
   * Launch JMeter by running the jmeter.bat (Windows) or jmeter (Linux/Mac) file.
2. **Set Up JMeter Test Plan:**
   * Open JMeter.
   * Right-click on **Test Plan** in the left panel.
   * Select **Add > Threads (Users) > Thread Group**.
3. **Configure Thread Group:**
   * Set **Number of Threads (users)** to 10 (this represents the 10 concurrent users).
   * Set **Ramp-Up Period (in seconds)** to an appropriate value (e.g., 10). This determines how long JMeter will take to start all 10 users. A ramp-up period of 10 seconds means it will start one user per second.
   * Leave **Loop Count** set to 1 if you want each user to run only once. Adjust if you need repeated iterations.
4. **Add HTTP Request Defaults:**
   * Right-click on **Thread Group**.
   * Select **Add > Config Element > HTTP Request Defaults**.
   * Set **Server Name or IP** to awesomeqa.com.
   * Set **Protocol** to https.
   * Set **Path** to /ui/.
5. **Add HTTP Request:**
   * Right-click on **Thread Group**.
   * Select **Add > Sampler > HTTP Request**.
   * This will inherit the defaults set in HTTP Request Defaults. Ensure the path /ui/ is correctly set.
6. **Add HTTP Header Manager (Optional):**
   * If you need to add specific headers (like User-Agent, Content-Type, etc.):
   * Right-click on **HTTP Request**.
   * Select **Add > Config Element > HTTP Header Manager**.
   * Add necessary headers by clicking the **Add** button and specifying the header name and value.
7. **Add Listeners:**
   * Listeners are used to view and analyze the results.
   * Right-click on **Thread Group**.
   * Select **Add > Listener > View Results Tree**.
   * Add other listeners as needed (e.g., Summary Report, Aggregate Report).
8. **Save the Test Plan:**
   * Save your test plan by clicking **File > Save**.
9. **Run the Test:**
   * Click the **Start** button (green triangle) on the toolbar to start the test.
   * Monitor the test in the listeners you added.
10. **Analyze the Results:**
    * After the test completes, analyze the results using the listeners.
    * Look for metrics like response time, throughput, error percentage, etc.

Here's a summary of the key steps visually:

1. **Thread Group Configuration:**
2. **HTTP Request Defaults:**
3. **HTTP Request:**
4. **View Results Tree:**

This setup will simulate 10 users accessing https://awesomeqa.com/ui/ concurrently, allowing you to analyze the performance and identify any bottlenecks.

how to provide the http rewuest as https://awesomeqa.com/ui/index.php?route=account/register in jmeter for performance testing with 10 users

Scenario-Test-002:

To set up Apache JMeter to test the URL https://awesomeqa.com/ui/index.php?route=account/register with 10 users, follow these detailed steps:

1. **Download and Install JMeter:**
   * Ensure Java is installed on your machine.
   * Download JMeter from the [official website](https://jmeter.apache.org/download_jmeter.cgi).
   * Extract the downloaded file and navigate to the bin directory.
   * Launch JMeter by running the jmeter.bat (Windows) or jmeter (Linux/Mac) file.
2. **Create a New Test Plan:**
   * Open JMeter.
   * By default, a new test plan is created. If not, create a new one by clicking **File > New**.
3. **Add a Thread Group:**
   * Right-click on **Test Plan**.
   * Select **Add > Threads (Users) > Thread Group**.
   * Configure the Thread Group:
     + **Number of Threads (users):** Set to 10.
     + **Ramp-Up Period (in seconds):** Set to 10 (this will start 1 user per second).
     + **Loop Count:** Set to 1 (adjust if you need repeated iterations).
4. **Add HTTP Request Defaults:**
   * Right-click on **Thread Group**.
   * Select **Add > Config Element > HTTP Request Defaults**.
   * Configure the HTTP Request Defaults:
     + **Server Name or IP:** awesomeqa.com
     + **Protocol:** https
     + **Path:** Leave this blank (will be set in the HTTP Request).
5. **Add HTTP Request Sampler:**
   * Right-click on **Thread Group**.
   * Select **Add > Sampler > HTTP Request**.
   * Configure the HTTP Request:
     + **Name:** Give a name to the request (e.g., "Account Register Request").
     + **Path:** /ui/index.php
     + **Method:** GET
     + **Parameters:** Add a parameter for the query string.
       - Click on **Add**.
       - **Name:** route
       - **Value:** account/register
6. **Add HTTP Header Manager (Optional):**
   * If you need to add specific headers (like User-Agent, Content-Type, etc.):
   * Right-click on **HTTP Request**.
   * Select **Add > Config Element > HTTP Header Manager**.
   * Add necessary headers by clicking the **Add** button and specifying the header name and value.
7. **Add Listeners:**
   * Listeners are used to view and analyze the results.
   * Right-click on **Thread Group**.
   * Select **Add > Listener > View Results Tree**.
   * Add other listeners as needed (e.g., Summary Report, Aggregate Report).
8. **Save the Test Plan:**
   * Save your test plan by clicking **File > Save**.
9. **Run the Test:**
   * Click the **Start** button (green triangle) on the toolbar to start the test.
   * Monitor the test in the listeners you added.
10. **Analyze the Results:**
    * After the test completes, analyze the results using the listeners.
    * Look for metrics like response time, throughput, error percentage, etc.

### Visual Summary:

1. **Thread Group Configuration:**
2. **HTTP Request Defaults:**
3. **HTTP Request:**
   * Add a parameter for the query string:
4. **View Results Tree:**

### Example Configuration in Detail:

**Thread Group:**

* **Number of Threads (users):** 10
* **Ramp-Up Period (in seconds):** 10
* **Loop Count:** 1

**HTTP Request Defaults:**

* **Server Name or IP:** awesomeqa.com
* **Protocol:** https
* **Path:** (leave blank)

**HTTP Request:**

* **Name:** Account Register Request
* **Path:** /ui/index.php
* **Method:** GET
* **Parameters:**
  + **Name:** route
  + **Value:** account/register

By following these steps, you will be able to configure JMeter to test the performance of the URL https://awesomeqa.com/ui/index.php?route=account/register with 10 users concurrently.

How to perform registration in Jmeter

Scenario-2

To perform a registration process using JMeter, you'll typically need to simulate a form submission with required user details. Assuming https://awesomeqa.com/ui/index.php?route=account/register is the registration URL, here’s how you can configure JMeter to perform the registration process:

1. **Download and Install JMeter:**
   * Ensure Java is installed on your machine.
   * Download JMeter from the [official website](https://jmeter.apache.org/download_jmeter.cgi).
   * Extract the downloaded file and navigate to the bin directory.
   * Launch JMeter by running the jmeter.bat (Windows) or jmeter (Linux/Mac) file.
2. **Create a New Test Plan:**
   * Open JMeter.
   * By default, a new test plan is created. If not, create a new one by clicking **File > New**.
3. **Add a Thread Group:**
   * Right-click on **Test Plan**.
   * Select **Add > Threads (Users) > Thread Group**.
   * Configure the Thread Group:
     + **Number of Threads (users):** Set to 10.
     + **Ramp-Up Period (in seconds):** Set to 10 (this will start 1 user per second).
     + **Loop Count:** Set to 1 (adjust if you need repeated iterations).
4. **Add HTTP Request Defaults:**
   * Right-click on **Thread Group**.
   * Select **Add > Config Element > HTTP Request Defaults**.
   * Configure the HTTP Request Defaults:
     + **Server Name or IP:** awesomeqa.com
     + **Protocol:** https
     + **Path:** Leave this blank (will be set in the HTTP Request).
5. **Add HTTP Request for Registration:**
   * Right-click on **Thread Group**.
   * Select **Add > Sampler > HTTP Request**.
   * Configure the HTTP Request:
     + **Name:** Give a name to the request (e.g., "Registration Request").
     + **Method:** POST
     + **Path:** /ui/index.php?route=account/register
     + **Parameters:** Add the necessary parameters that the registration form requires (e.g., first name, last name, email, password, etc.).
       - Click on **Add** for each parameter and provide the name and value.
       - **Example parameters might include:**
         * **Name:** firstname, **Value:** TestFirstName
         * **Name:** lastname, **Value:** TestLastName
         * **Name:** email, **Value:** testuser@example.com
         * **Name:** password, **Value:** TestPassword
         * **Name:** confirm, **Value:** TestPassword
         * **Name:** telephone, **Value:** 1234567890
6. **Add HTTP Header Manager (Optional):**
   * Right-click on **HTTP Request**.
   * Select **Add > Config Element > HTTP Header Manager**.
   * Add necessary headers, such as:
     + **Content-Type:** application/x-www-form-urlencoded
     + Other headers as needed (e.g., User-Agent).
7. **Add User Defined Variables (Optional for Dynamic Data):**
   * To use dynamic data for each registration, you can add a CSV Data Set Config.
   * Right-click on **Thread Group**.
   * Select **Add > Config Element > CSV Data Set Config**.
   * Configure the CSV Data Set Config to read from a CSV file containing user data.
   * Reference the variables in the HTTP Request parameters (e.g., ${firstname}, ${lastname}, ${email}, etc.).
8. **Add Listeners:**
   * Listeners are used to view and analyze the results.
   * Right-click on **Thread Group**.
   * Select **Add > Listener > View Results Tree**.
   * Add other listeners as needed (e.g., Summary Report, Aggregate Report).
9. **Save the Test Plan:**
   * Save your test plan by clicking **File > Save**.
10. **Run the Test:**
    * Click the **Start** button (green triangle) on the toolbar to start the test.
    * Monitor the test in the listeners you added.
11. **Analyze the Results:**
    * After the test completes, analyze the results using the listeners.
    * Look for metrics like response time, throughput, error percentage, etc.

### Example Configuration in Detail:

**Thread Group:**

* **Number of Threads (users):** 10
* **Ramp-Up Period (in seconds):** 10
* **Loop Count:** 1

**HTTP Request Defaults:**

* **Server Name or IP:** awesomeqa.com
* **Protocol:** https
* **Path:** (leave blank)

**HTTP Request:**

* **Name:** Registration Request
* **Method:** POST
* **Path:** /ui/index.php?route=account/register
* **Parameters:**
  + **Name:** firstname, **Value:** TestFirstName
  + **Name:** lastname, **Value:** TestLastName
  + **Name:** email, **Value:** testuser@example.com
  + **Name:** password, **Value:** TestPassword
  + **Name:** confirm, **Value:** TestPassword
  + **Name:** telephone, **Value:** 1234567890

**HTTP Header Manager (Optional):**

* **Content-Type:** application/x-www-form-urlencoded

### Adding Dynamic Data Using CSV Data Set Config:

1. **Prepare a CSV file** (e.g., user\_data.csv) with the following columns:

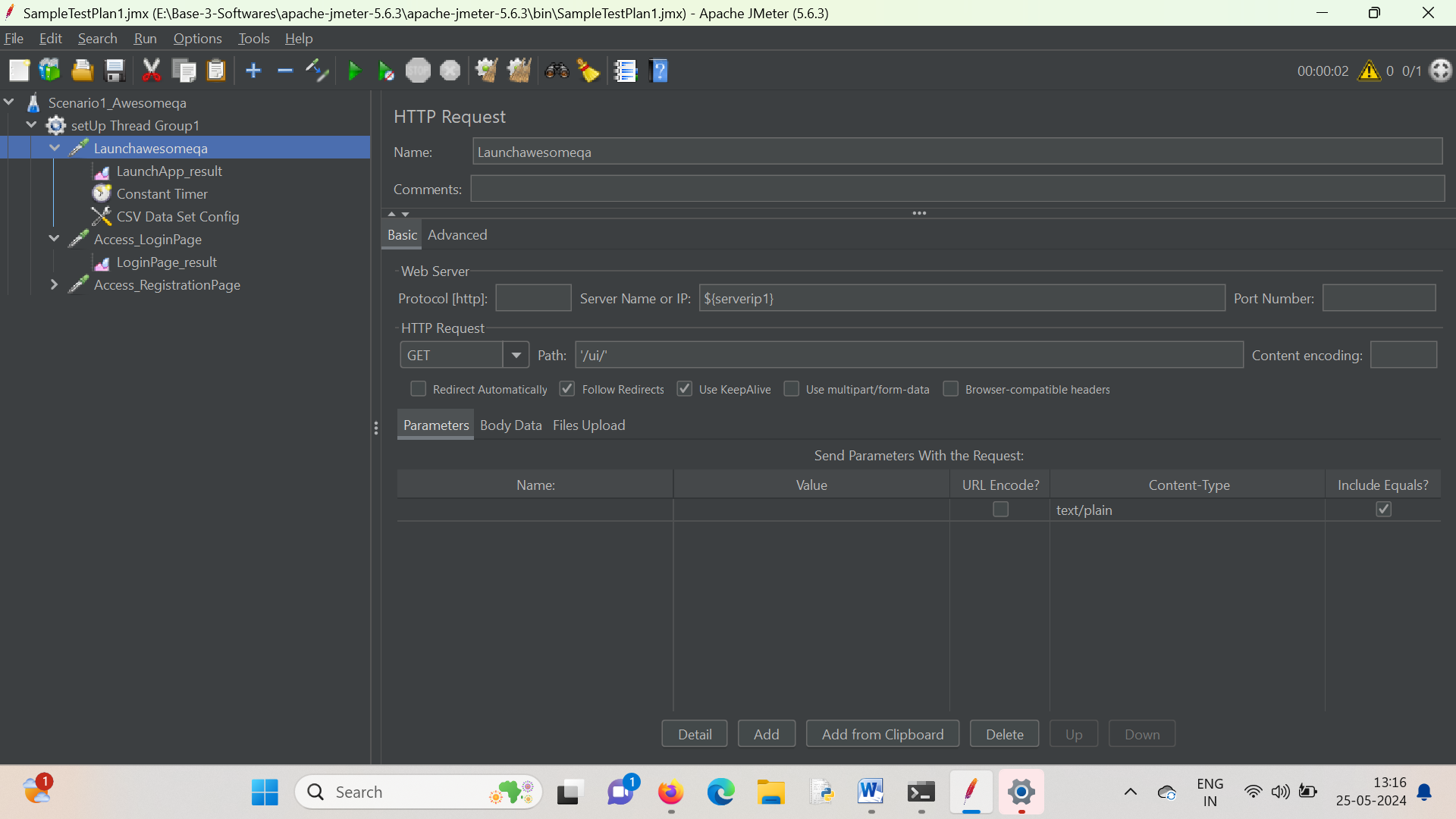
graphql

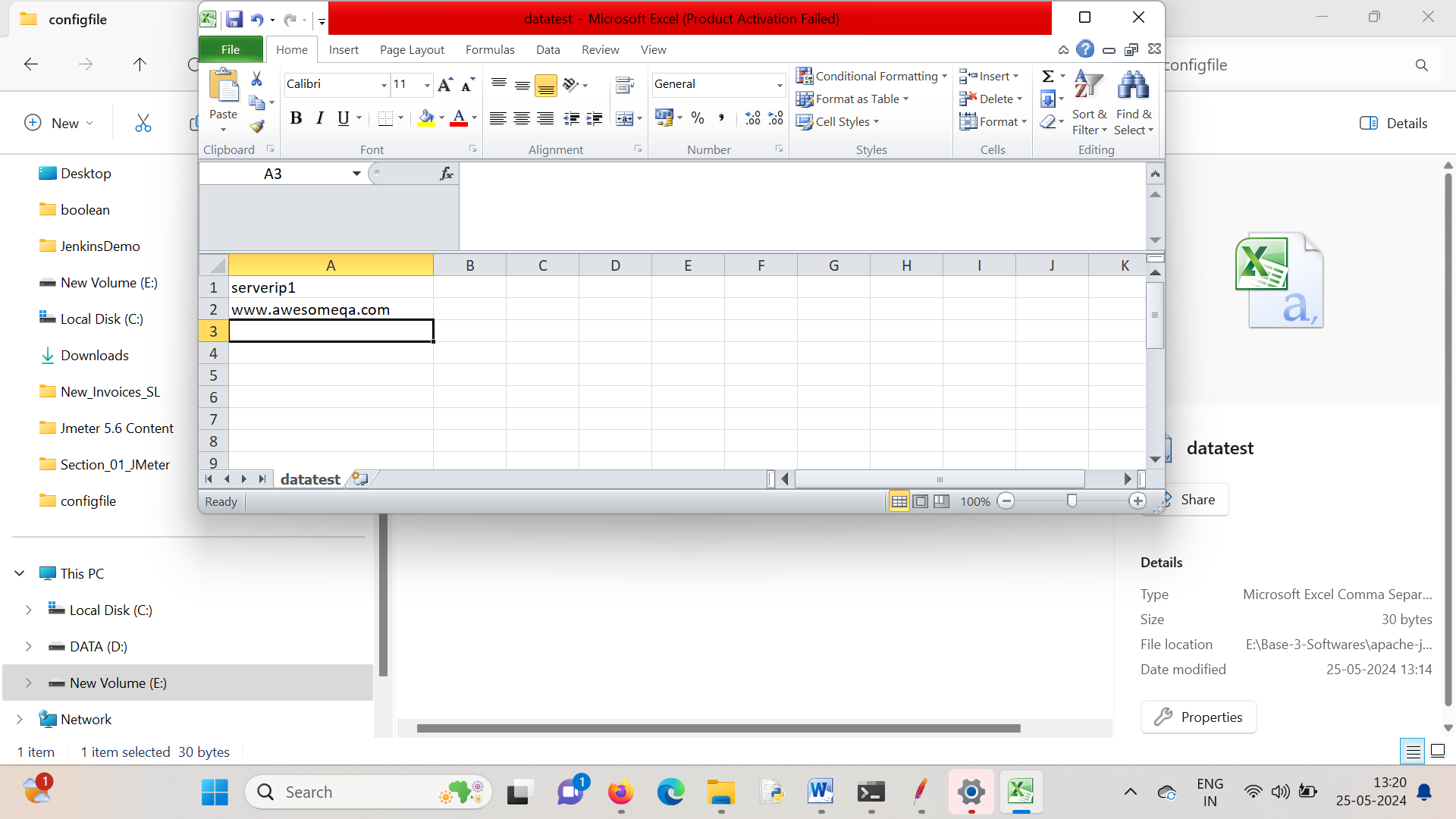
1. firstname,lastname,email,password,telephone
2. TestFirstName1,TestLastName1,testuser1@example.com,TestPassword1,1234567890
3. TestFirstName2,TestLastName2,testuser2@example.com,TestPassword2,1234567891
4. ...
5. **CSV Data Set Config:**
   * **Filename:** Path to your CSV file (e.g., user\_data.csv)
   * **Variable Names:** firstname,lastname,email,password,telephone
   * **Delimiter:** ,
   * **Recycle on EOF?** True
   * **Stop thread on EOF?** False
   * **Sharing Mode:** All threads
6. **HTTP Request Parameters:**
   * **Name:** firstname, **Value:** ${firstname}
   * **Name:** lastname, **Value:** ${lastname}
   * **Name:** email, **Value:** ${email}
   * **Name:** password, **Value:** ${password}
   * **Name:** confirm, **Value:** ${password}
   * **Name:** telephone, **Value:** ${telephone}

By following these steps, you will be able to configure JMeter to simulate user registration on https://awesomeqa.com/ui/index.php?route=account/register with dynamic user data for performance testing with 10 concurrent users.

ConfigElement :

Csv data config





Note : use the file path in file config

Scenario – 5

HTTPHeader Config

Connection: keep-alive  
Host: www.google.com  
User-Agent: Apache-HttpClient/4.5.14 (Java/11.0.17)