Department of Computer Science and Engineering
(IoT, Cybersecurity with Blockchain Technology)
B.Tech. Sem: VII Subject: Software Testing & Quality Assurance

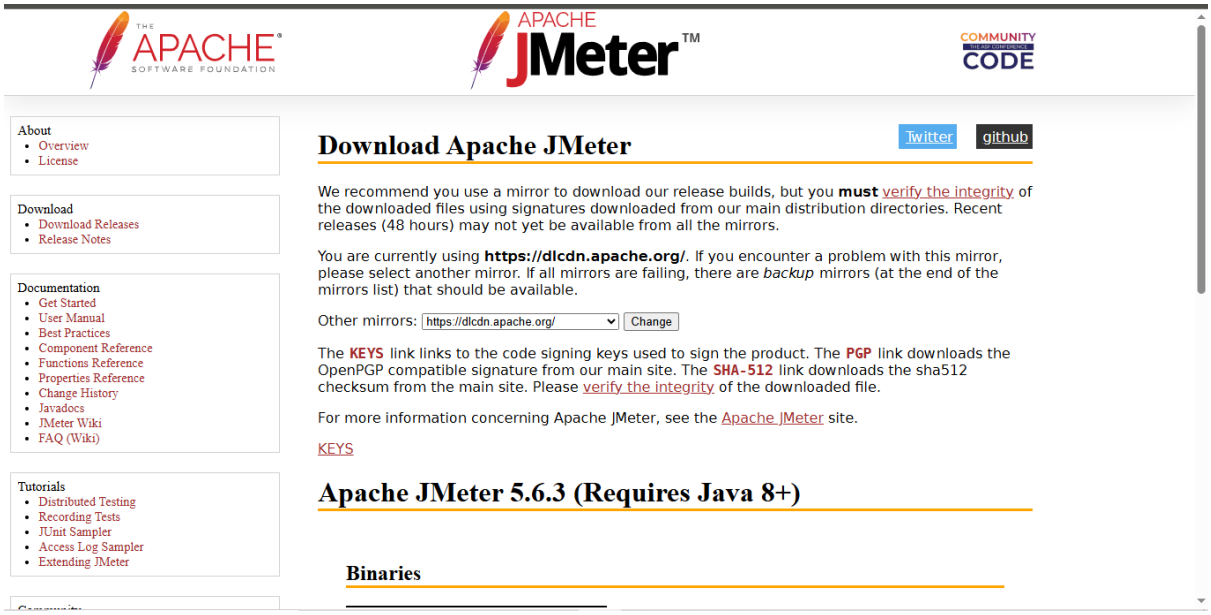
Experiment 8

Name: Hiren Darji
SAP ID: 60019230114
Aim: - Performance Testing Using Apache JMeter.

Steps:

1) **Install JMeter.**

Download Apache JMeter from https://jmeter.apache.org/download_jmeter.cgi



Click on Download releases

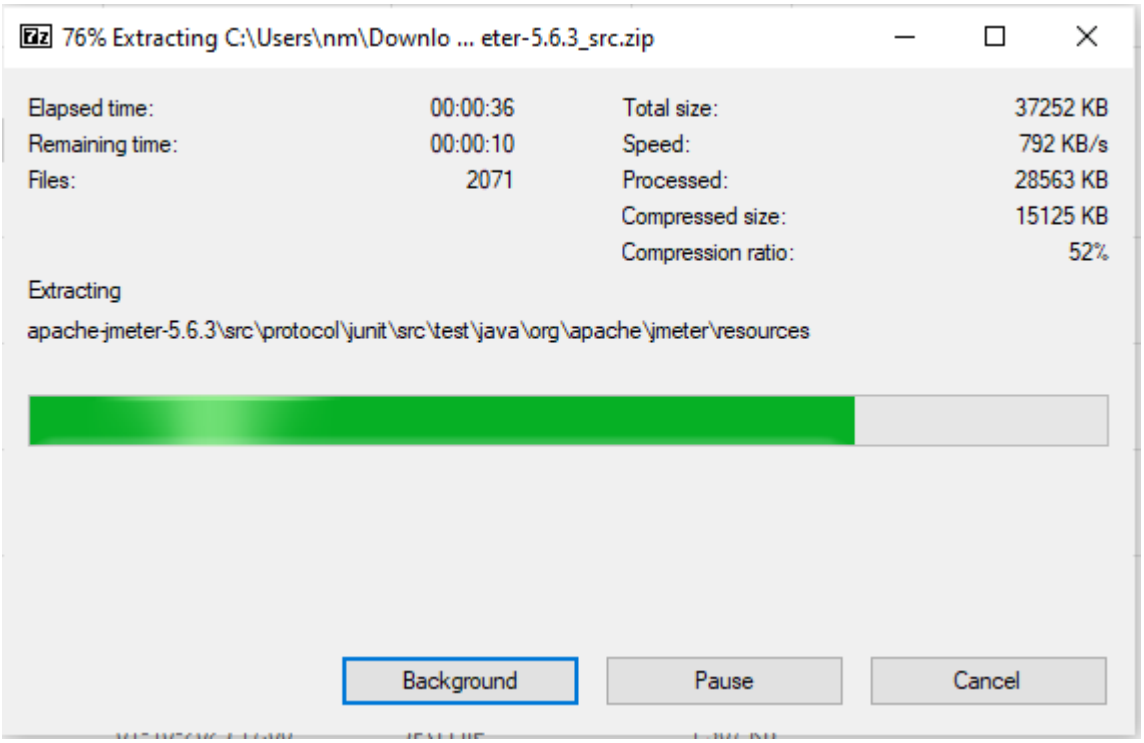
- Download .zip and Extract the ZIP file and navigate to the /bin folder.

Apache JMeter 5.6.3 (Requires Java 8+)

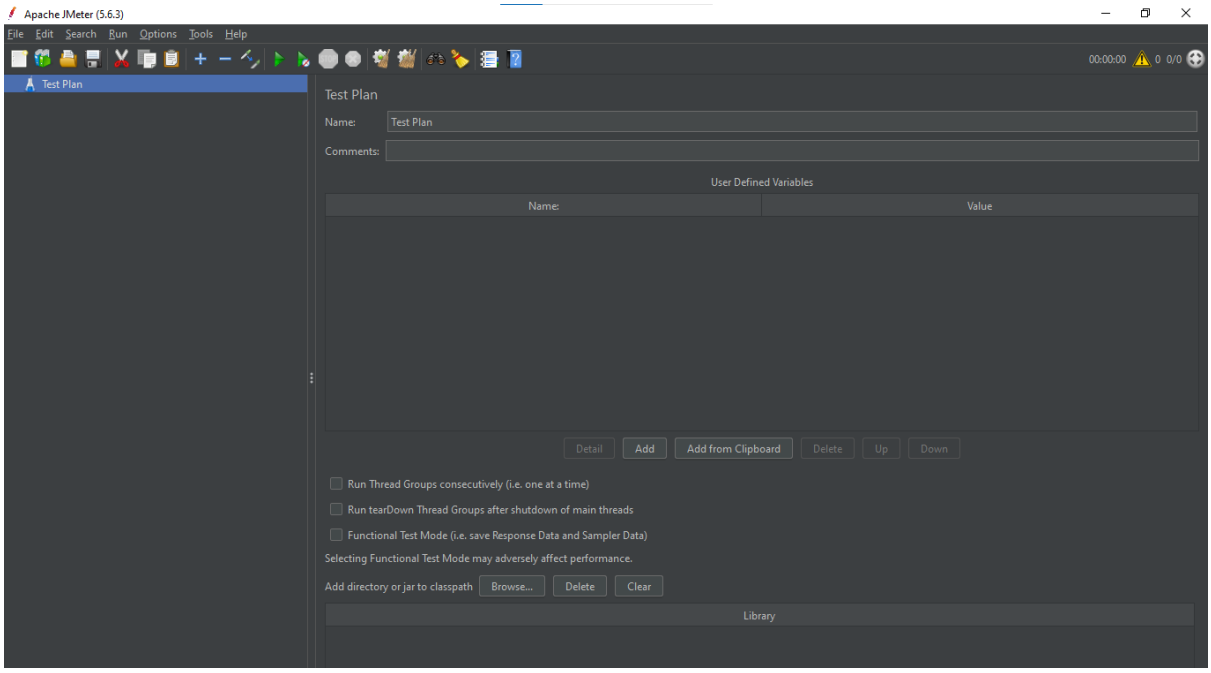
Binaries

apache-jmeter-5.6.3.tgz sha512 pgp
apache-jmeter-5.6.3.zip sha512 pgp

Step 2: Extract the JMeter



Step 3: Download .zip and Extract the ZIP file and navigate to the /bin folder. • Run jmeter.bat

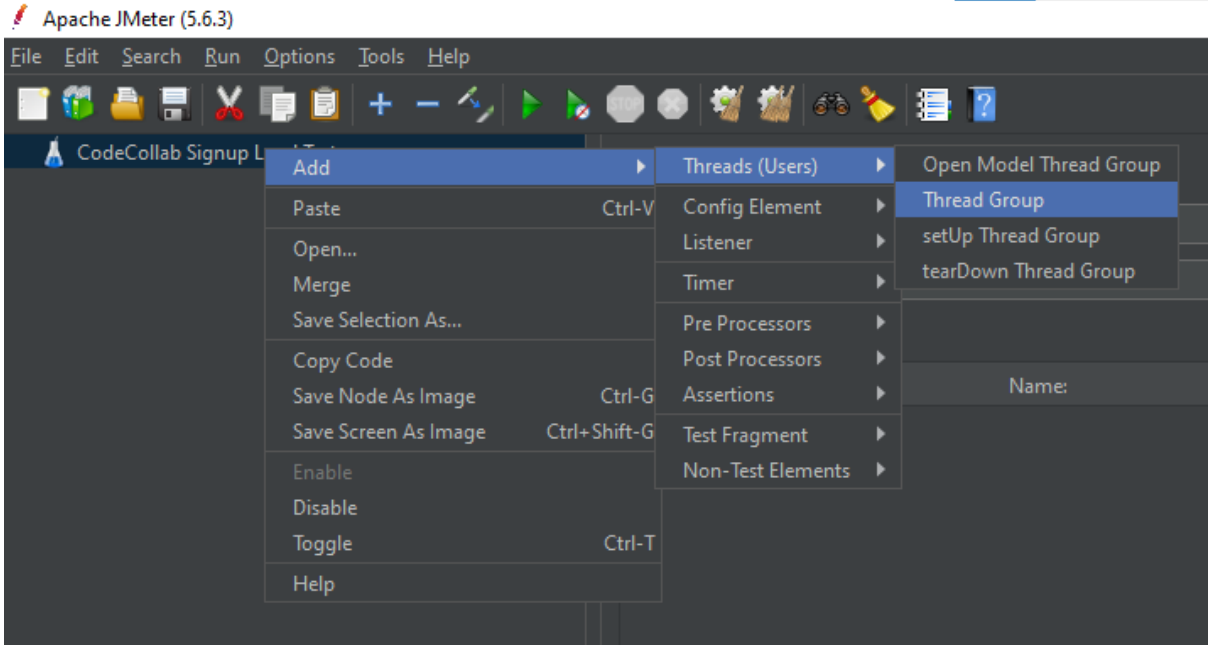


4 — New Test Plan

- Menu: **File → New**
- In left tree, right-click **Test Plan → Rename** → e.g. CodeCollab Signup Load Test

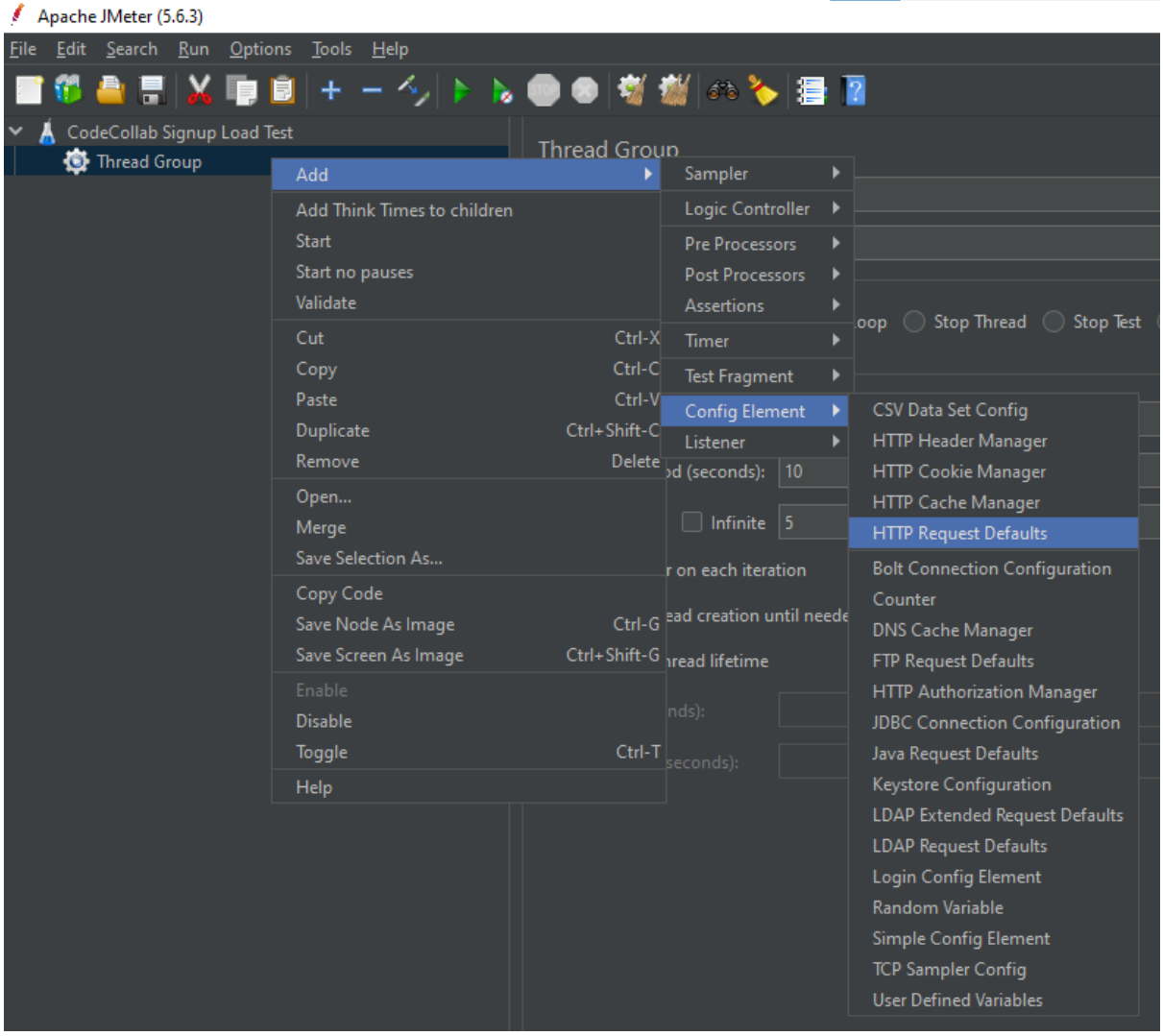
5 — Add a Thread Group (virtual users)

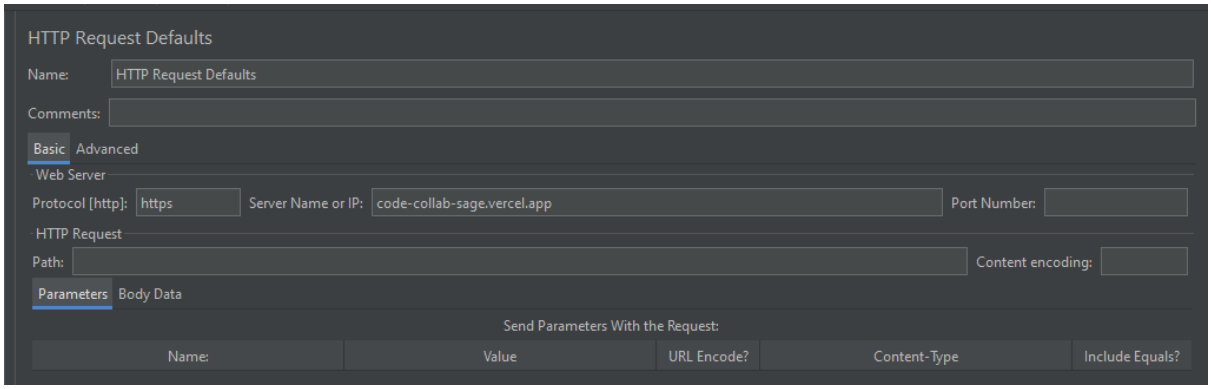
- Right-click **Test Plan → Add → Threads (Users) → Thread Group**
- Click **Thread Group** node and set:
  - **Number of Threads (users)**: 50 (example)
  - **Ramp-Up Period (seconds)**: 10
  - **Loop Count**: 5 (or check **Forever** for long runs)
  - *(This simulates 50 users arriving over 10s, each doing test 5 times.)*



6 — Configure HTTP Request Defaults (so each sampler doesn't repeat host)

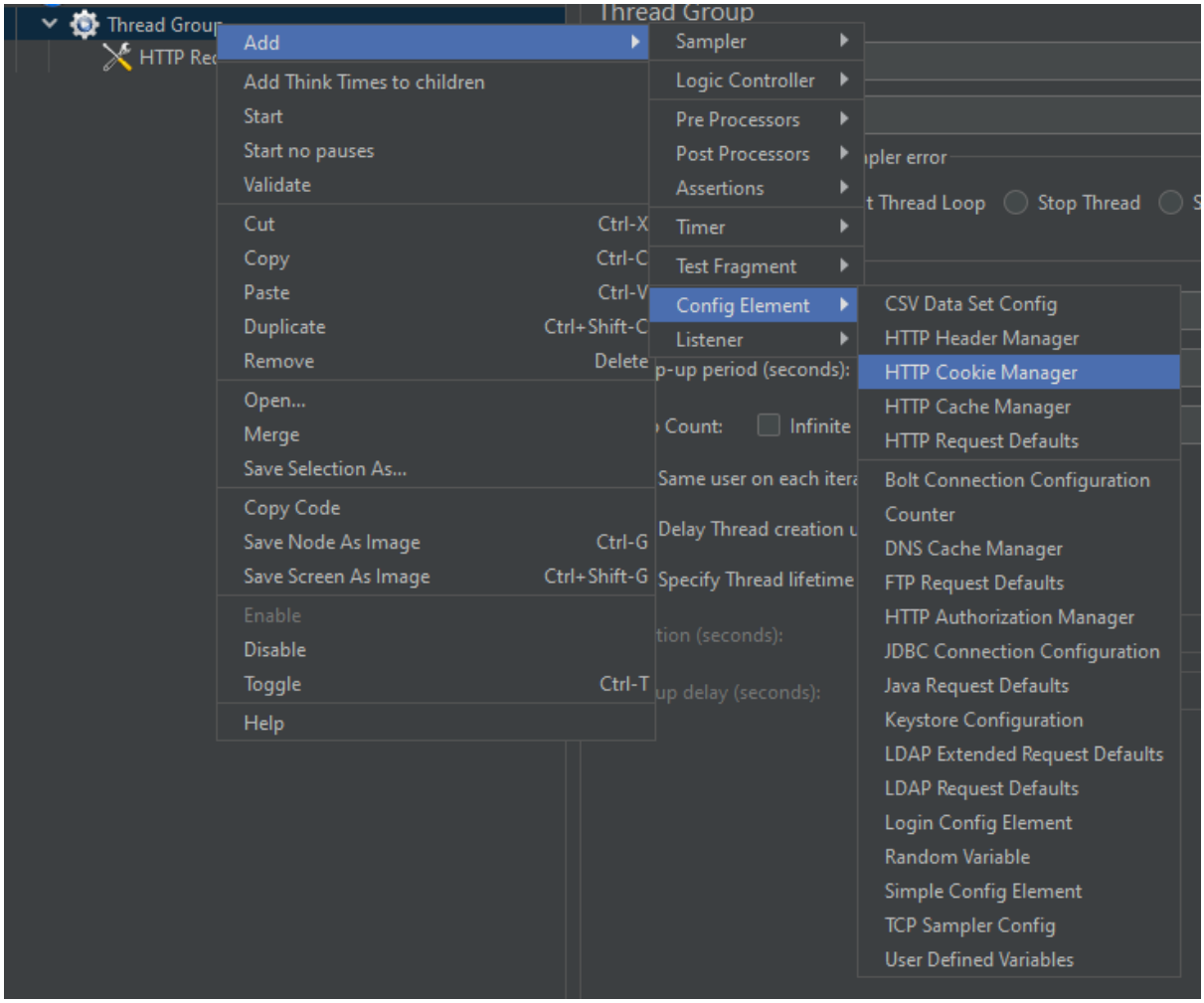
- Right-click **Thread Group → Add → Config Element → HTTP Request Defaults**
  - **Server Name or IP**: code-collab-sage.vercel.app
  - **Protocol**: https
  - **Port Number**: leave blank (default for HTTPS)
  - **Path**: leave blank (you'll set path per sampler)
- Save — this reduces typing for each HTTP Request.

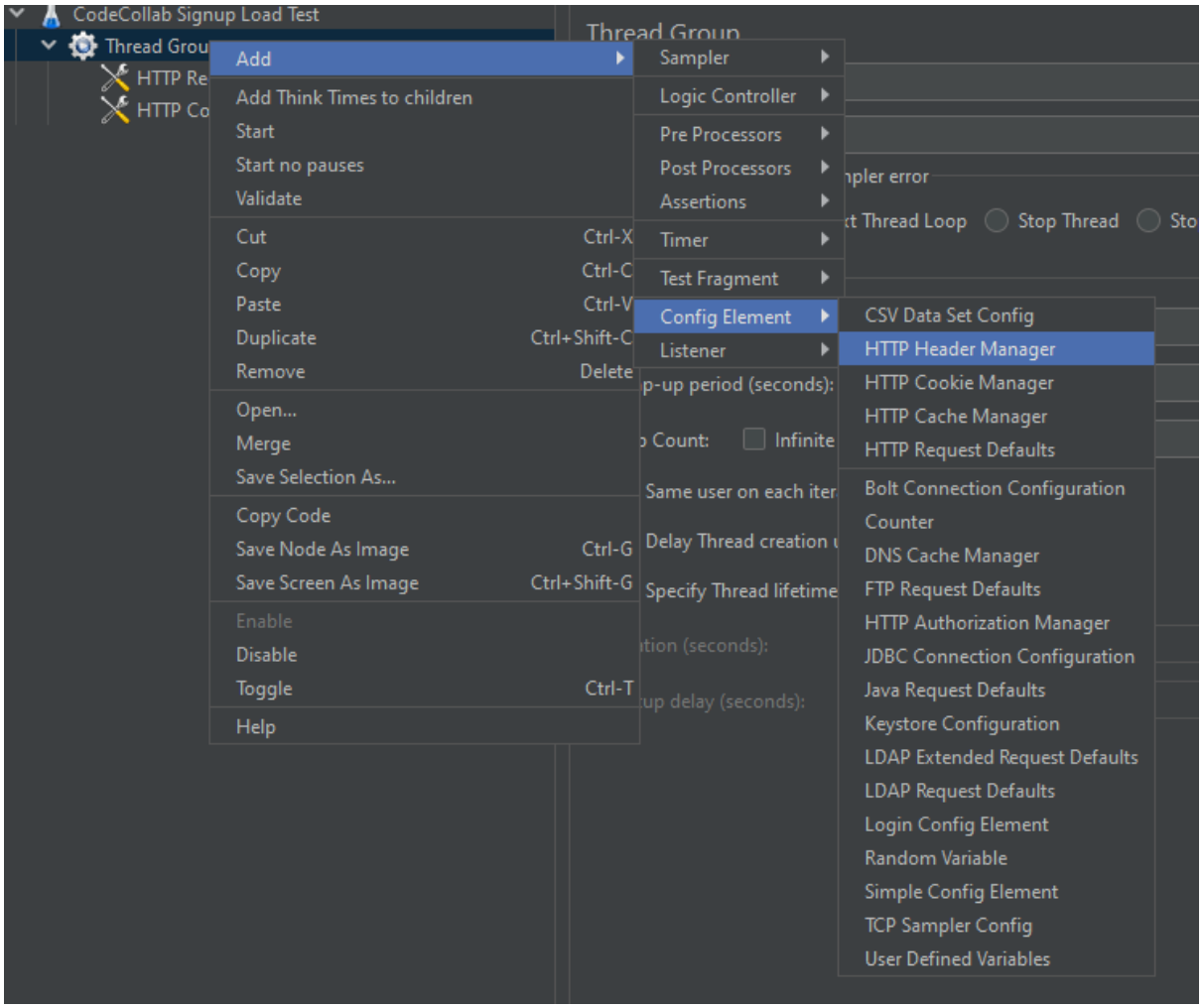




7 — Add HTTP Cookie Manager & Header Manager

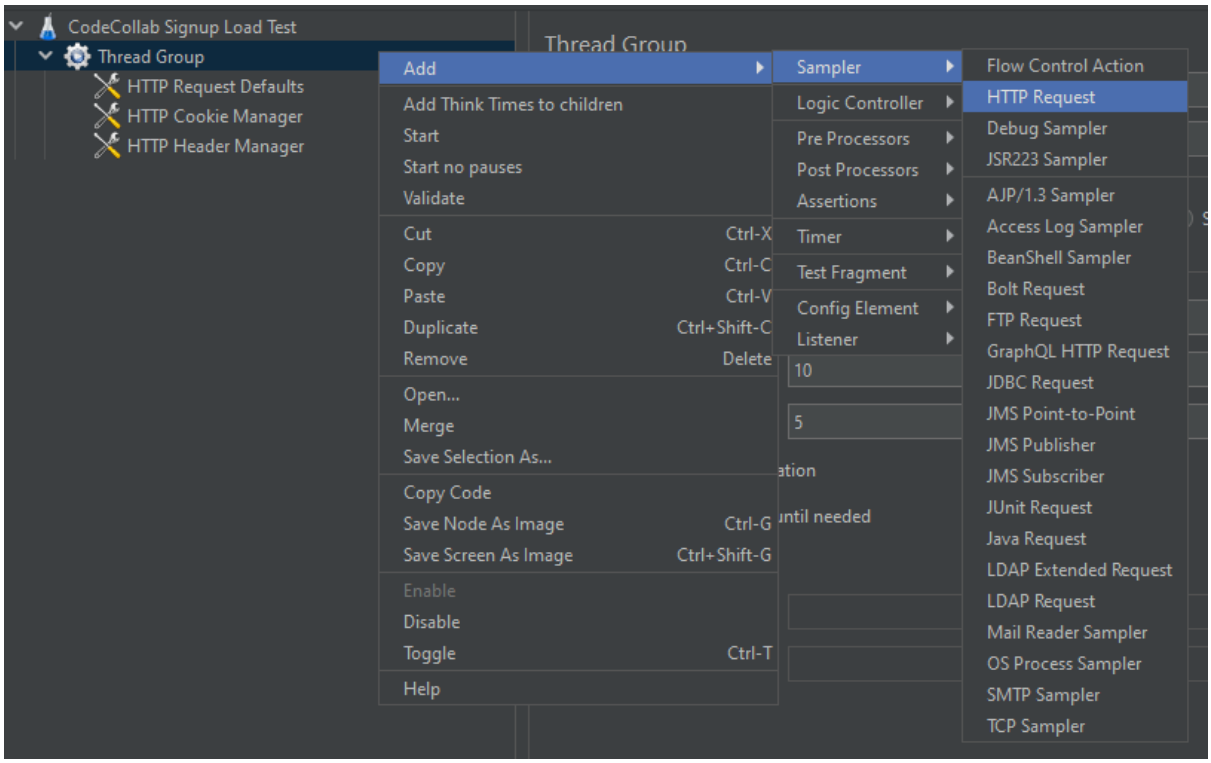
- Right-click **Thread Group → Add → Config Element → HTTP Cookie Manager** (handles cookies like a browser).
- Right-click **Thread Group → Add → Config Element → HTTP Header Manager**
  - Click Header Manager → **Add** a header: Content-Type = application/json (if endpoints accept JSON).
  - Add any auth headers if your app requires tokens already.

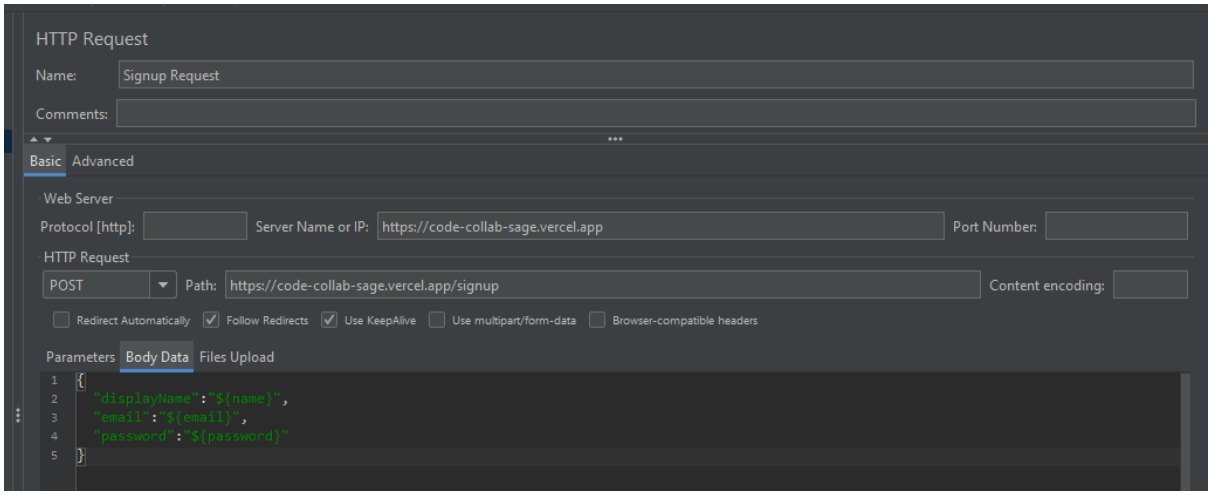

Thread Group
Add
Add Think Times to children
Start
Start no pauses
Validate
Cut Ctrl-X
Copy Ctrl-C
Paste Ctrl-V
Duplicate Ctrl+Shift-C
Remove Delete
Open...
Merge
Save Selection As...
Copy Code
Save Node As Image Ctrl-G
Save Screen As Image Ctrl+Shift-G
Enable
Disable
Toggle Ctrl-T
Help
Sampler
Logic Controller
Pre Processors
Post Processors
Assertions
Timer
Test Fragment
Config Element
Listener
CSV Data Set Config
HTTP Header Manager
HTTP Cookie Manager
HTTP Cache Manager
HTTP Request Defaults
Bolt Connection Configuration
Counter
DNS Cache Manager
FTP Request Defaults
HTTP Authorization Manager
JDBC Connection Configuration
Java Request Defaults
Keystore Configuration
LDAP Extended Request Defaults
LDAP Request Defaults
Login Config Element
Random Variable
Simple Config Element
TCP Sampler Config
User Defined Variables
Thread Loop
Stop Thread
Infinite



8 — Add HTTP Request Sampler(s)

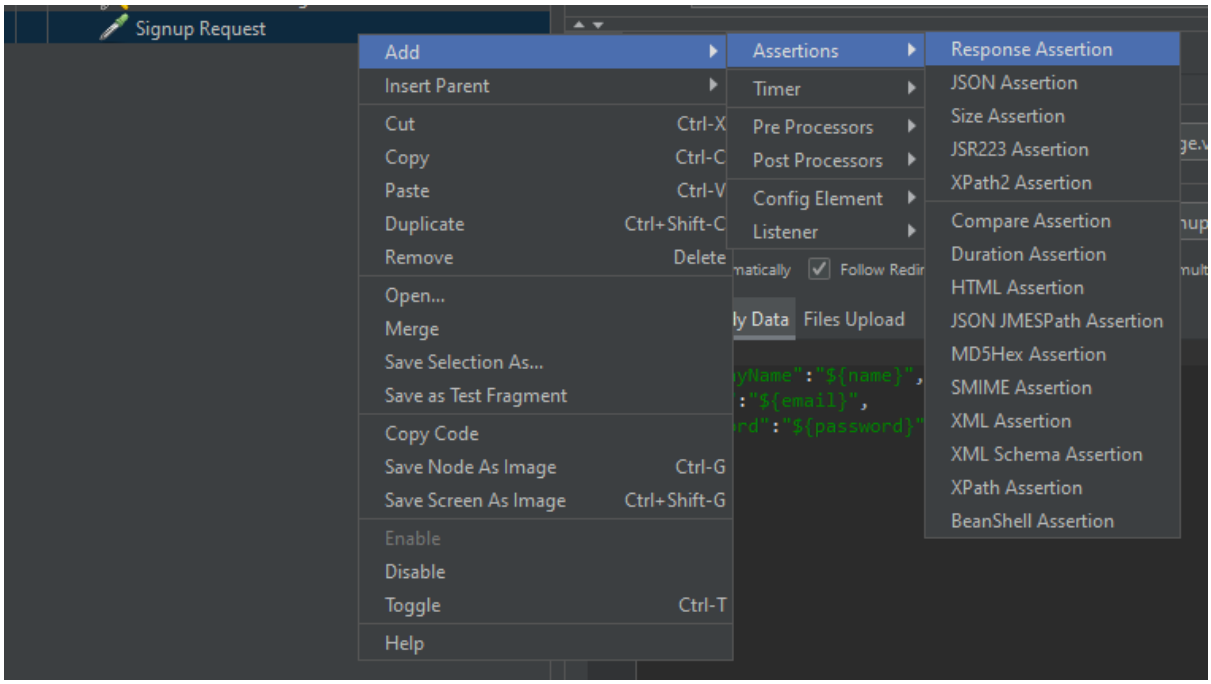
- Right-click **Thread Group → Add → Sampler → HTTP Request**
  - **Name**: Signup Request
  - **Method**: POST (or GET depending endpoint)
  - **Path**: / or specific endpoint, e.g. /api/v1/users (use your actual sign-up endpoint)
  - **Body Data**: paste JSON payload, e.g.
  - {
  - "displayName":"${name}",
  - "email":"${email}",
  - "password":"${password}"
  - }
  - If using form data, use **Parameters** table instead of Body Data.





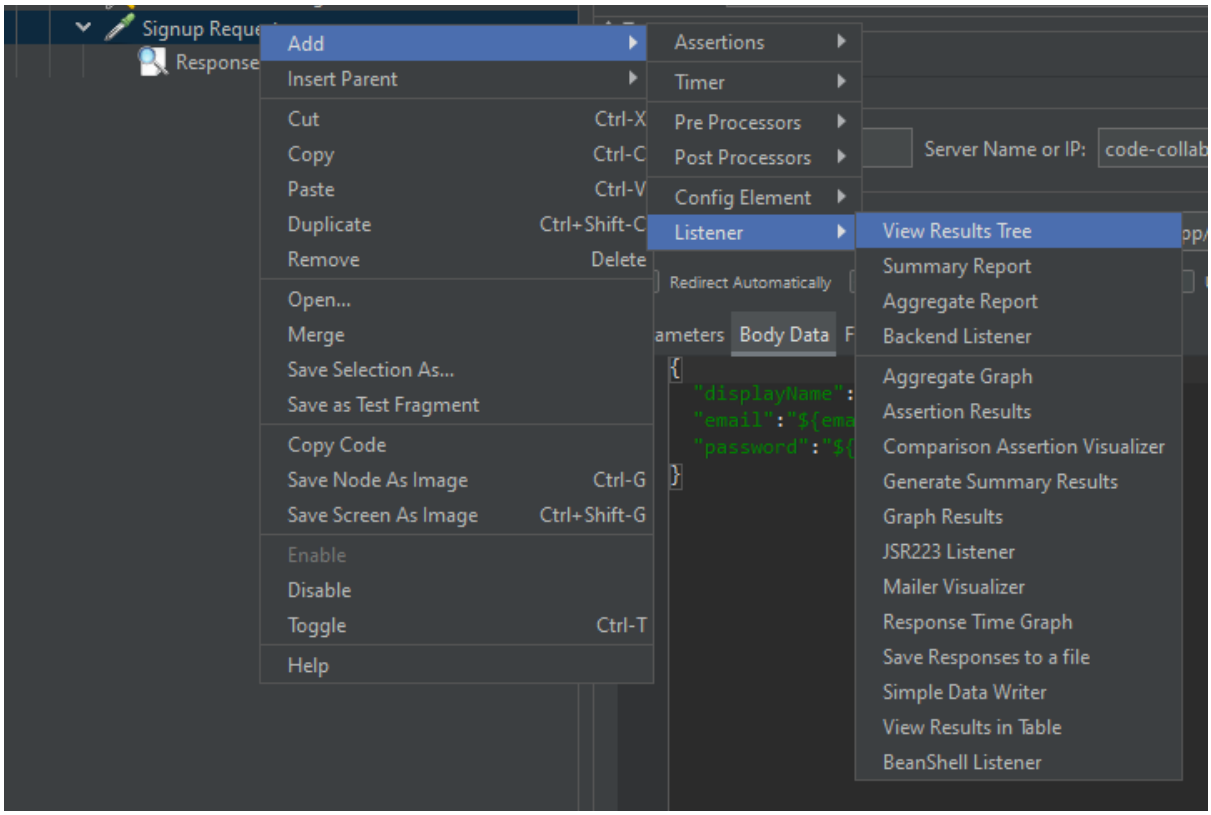
9 — Add Assertions (verify success/failure)

- Right-click **HTTP Request → Add → Assertions → Response Assertion**
  - Choose **Response Text** contains expected success text (e.g., "user created" or success code).
  - This allows JMeter to mark failures where functionally incorrect.



10 — Add Listeners (to view results)

- Right-click **Thread Group → Add → Listener → View Results Tree** (debugging only)
- Add **Summary Report**, **Aggregate Report**, **Graph Results**, **Response Time Graph**.
  - Important: use **Summary / Aggregate Report** for summary metrics. Avoid View Results Tree for heavy loads (it stores full response content and eats memory).

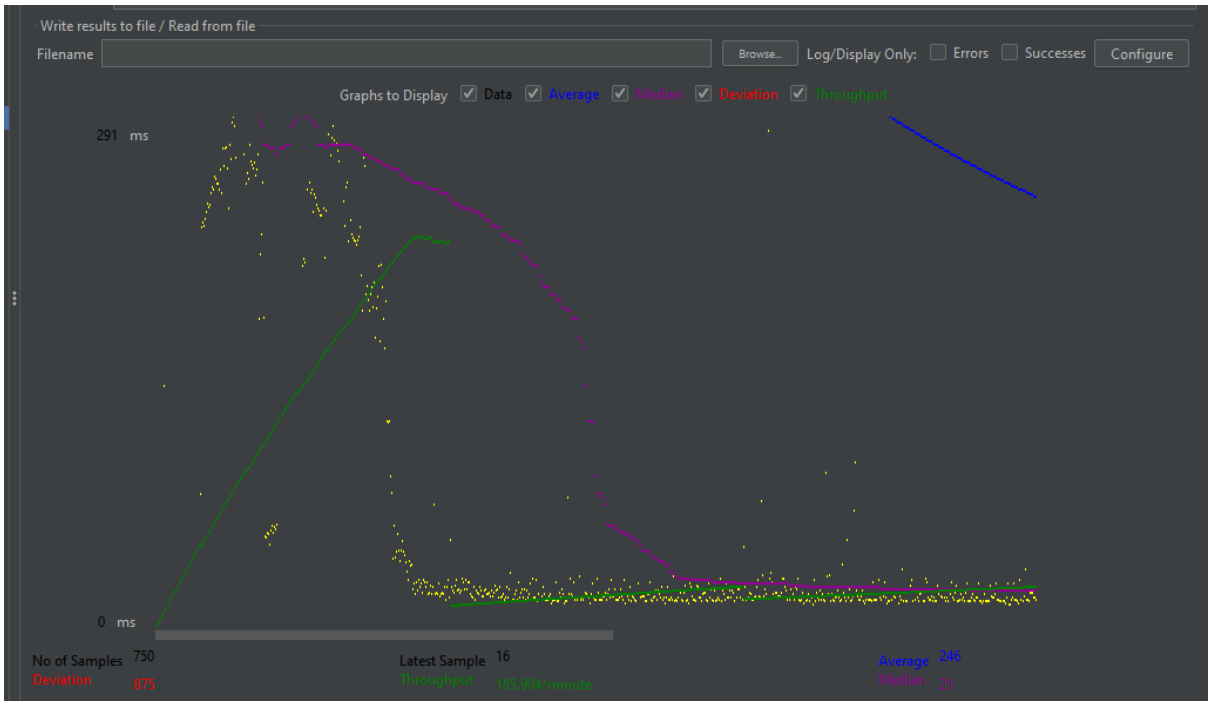


11— Save and Run Tests

Run in GUI (for building & debug)

- Click the **green Start (▶)** button in toolbar.
- Watch results in Listeners. For moderate loads GUI is OK.

Response Graph



Results Tree Table

Sample #	Start Time	Thread Name	Label	Sample Time(...	Status	Bytes	Sent Bytes	Latency	Connect Time(...
1	18:26:22.984	Thread Group ...	Signup Request	5900		103	271	5730	5613
2	18:26:27.742	Thread Group ...	Signup Request	1141		103	271	1009	855
3	18:26:27.140	Thread Group ...	Signup Request	1744		103	271	1574	1457
4	18:26:26.544	Thread Group ...	Signup Request	2340		103	271	2170	2053
5	18:26:24.350	Thread Group ...	Signup Request	4535		103	271	4401	4247
6	18:26:28.542	Thread Group ...	Signup Request	341		103	271	209	55
7	18:26:28.747	Thread Group ...	Signup Request	138		103	271	36	29
8	18:26:22.984	Thread Group ...	Signup Request	5899		103	271	5730	5613
9	18:26:23.357	Thread Group ...	Signup Request	5524		103	271	5356	5240
10	18:26:22.984	Thread Group ...	Signup Request	5897		103	271	5766	5615
11	18:26:24.150	Thread Group ...	Signup Request	4730		103	271	4564	4447
12	18:26:25.746	Thread Group ...	Signup Request	3134		103	271	3005	2851
13	18:26:25.945	Thread Group ...	Signup Request	2935		103	271	2806	2652
14	18:26:25.349	Thread Group ...	Signup Request	3531		103	271	3402	3248

Conclusion:

I have successfully performed the complete performance testing of the signup API using Apache JMeter. The test plan was designed with dynamic inputs, appropriate headers, and JSON payloads to simulate real user behavior. Assertions and listeners were added to accurately measure responsiveness, throughput, and success rates of the API. This experiment helped validate the system's stability under load and ensured reliable performance for multiple concurrent user registrations