🧩Thread Count, Ramp-up Period, and Loop Count from Command Line

Use the \_\_P(propertyName, defaultValue) function to dynamically pass values from the command line:

text

CopyEdit

Number of Threads (Users): ${\_\_P(user,1)}

Ramp-up Period (seconds): ${\_\_P(rampup,1)}

Loop Count: ${\_\_P(loop,1)}

**💻 2. Run JMeter from Command Line**

Use the -J flag to override values directly from the terminal.

**🐧 Linux/macOS:**

bash

CopyEdit

sh jmeter.sh -n -t "/path/to/test.jmx" -l "/path/to/results.csv" -e -o "/path/to/report" -Juser=10 -Jrampup=5 -Jloop=3

**🪟 Windows:**

bash

CopyEdit

jmeter -n -t "C:\path\to\test.jmx" -l "C:\path\to\results.csv" -e -o "C:\path\to\report" -Juser=10 -Jrampup=5 -Jloop=3

**🧠 3. Explanation of Command-Line Arguments**

| **🏷️ Flag** | **📝 Description** |
| --- | --- |
| -n | Run in **non-GUI** mode |
| -t | Path to your .jmx **test plan** |
| -l | Output **results file** (CSV or JTL) |
| -e | **Generate** dashboard report |
| -o | Output **folder** for dashboard (must be empty or not exist) |
| -Juser=X | Set number of **Threads (Users)** |
| -Jrampup=Y | Set **Ramp-up** period in seconds |
| -Jloop=Z | Set **Loop Count** |

**🧪 4. Example: Load Test with Custom Parameters**

You want:

* 10 users
* 20 seconds ramp-up
* 5 loops per user

**Command:**

bash

CopyEdit

jmeter -n -t "test.jmx" -l "results.csv" -e -o "report" -Juser=10 -Jrampup=20 -Jloop=5

**💡 5. Pro Tip: Create a Reusable Script**

Automate your runs with a shell or batch script:

**Example (Linux/macOS):**

bash

CopyEdit

#!/bin/bash

USER\_COUNT=$1

RAMPUP=$2

LOOPS=$3

sh jmeter.sh -n -t test.jmx -l results.csv -e -o report -Juser=$USER\_COUNT -Jrampup=$RAMPUP -Jloop=$LOOPS

Run it like:

bash

CopyEdit

sh run\_test.sh 50 10 2