

Manual Tests

1. Integration Test

| JUnit.1 | Integration |
|-----------------|---|
| Requirement 1 | The web server should be responsive under high load |
| Use Cases | The system shall test the HTTP Server and the port socket. If they are un/available notify the admin. |
| Scenario | Confirming the proper function of the already existing Junit-Tests |
| Pre-condition | Server is running properly |
| Test Steps | Run the integration package as Junit Test |
| Expected result | All tests should pass except GetFromOnlineServer |
| Actual result | All the tests succeed except GetFromOnlineServer |
| Comments | Automated test package tests\se\Inu\http\integration Mac, IntelliJ snipped [pic 1], Linux, IntelliJ snipped [pic 2], Windows, IntelliJ snipped [pic 3] |

2. Response Test

| JUnit.2 | Response |
|-----------------|--|
| Requirement 2 | The web server must follow minimum requirements for HTTP 1.1 |
| Use Cases | ----- |
| Scenario | Confirming the proper function of the already existing Junit-Tests |
| Pre-condition | Server is running properly |
| Test Steps | Run the response package as Junit Test |
| Expected result | All tests should pass |
| Actual result | All the tests succeed |
| Comments | Automated test package tests\se\Inu\http\response Mac, IntelliJ snipped [pic 4], Linux,IntelliJ snipped [pic 5] Windows, IntelliJ snipped [pic 6] |

3. View Test

| JUnit.3 | View |
|-----------------|--|
| Requirement | ----- |
| Use Cases | ----- |
| Scenario | Confirming the proper function of the already existing Junit-Tests |
| Pre-condition | Server is running properly |
| Test Steps | Run the view package as Junit Test |
| Expected result | All tests should pass |
| Actual result | All the tests succeed on Linux and Mac testPortTaken failed on Windows |
| Comments | Automated test package tests\se\Inu\http\view Mac, IntelliJ snipped [pic 7], Linux,IntelliJ snipped [pic 8] Windows, IntelliJ snipped [pic 9] |

4. Group Test

| JUnit .4 | Grouped |
|-----------------|---|
| Requirement | ----- |
| Use Cases | ----- |
| Scenario | Confirming the proper function of the already existing Junit-Tests as a group of automated tests |
| Pre-condition | Server is running properly |
| Test Steps | Run the project as Junit Test |
| Expected result | All tests should pass except GetFromOnlineServer |
| Actual result | All the tests succeed except GetFromOnlineServer on Linux and Mac. GetFromOnlineServer and testPortTaken are failed on Windows |
| Comments | The group test will test all available test cases as a group inside tests package. JUnit tests finds 57 test cases which 55 tests passes, and 2 test cases fails. Automated test package tests Mac, IntelliJ snipped [pic 10], Linux,IntelliJ snipped [pic 11] Windows,IntelliJ snipped [pic 12] |

UC1 Start Server

1. Starting Server

| | |
|----------------------|---|
| TUC1.1 | Start server |
| Requirement 3 | The web server must work on Linux, Mac, Windows |
| Use Cases | UC 1 Start Server |
| Scenario | Web server starts on varies platforms |
| Pre-condition | Server is not running |
| Test Steps | <ol style="list-style-type: none">1. User opens the cmd/Terminal2. Locate the Mywebserver jar file3. Execute the Mywebserver jar file using <code>java -jar Mywebserver.jar 1412</code> and the directory for inner folder4. Open a browser5. Type "localhost:1412"6. Enter |
| Expected result | "HTTP server object constructed HTTP server started Accept" is shown in the terminal window. localhost:1412 works |
| Alternative Scenario | <p>6a. The web server could not be started due restriction on the shared resource container</p> <ol style="list-style-type: none">1. System presents an error message: "No access to folder XX" (XX is the shared resource container provided, Example "\var\www")2. Exit Use Case <p>6b. The web server could not be started due restriction on the shared resource container</p> <ol style="list-style-type: none">1. System presents an error message: "No access to folder XX" (XX is the shared resource container provided, Example "\var\www")2. Exit Use Case <p>6c. The access log could not be written to</p> <ol style="list-style-type: none">1. System presents an error message. "Cannot write to server log file log.txt"2. Exit Use Case |
| Comments | PortSocket: 1412 ShareResource: Desktop\MyWebServer-master\out\test\MyWebServer-master\se\lnu\http\resources\inner Windows,IntelliJ snipped [pic 13] |

2. Wrong Socket

| | |
|----------------------|---|
| TUC1.2 | Port is wrong |
| Requirement 3 | The web server must work on Linux, Mac, Windows |
| Use Cases | UC 1 Start Server |
| Scenario | The sever should inform the user to enter a valid port |
| Pre-condition | Server is running |
| Test Steps | <ol style="list-style-type: none">1. Administrator opens IDE Manual enter the arguments (PortSocket Desktop\MyWebServer-master\out\test\MyWebServer-master\se\lnu\http\resources\inner) for HTTPServerConsole2. Run HTTPServerConsole |
| Expected result | “Enter a valid port 1-65535 and an optional URL” pops up in the console window |
| Alternative Scenario | 2a. No warning is shown |
| Comments | PortSocket: 0 ShareResource: Desktop\MyWebServer-master\out\test\MyWebServer-master\se\lnu\http\resources\inner Windows,Mac,IntelliJ |

1. Taken Socket

| | |
|----------------------|---|
| TUC1.2 | Port is taken |
| Requirement 3 | The web server must work on Linux, Mac, Windows |
| Use Cases | UC 1 Start Server |
| Scenario | The sever should inform the port is taken |
| Pre-condition | Server is running |
| Test Steps | <ol style="list-style-type: none">1. Administrator opens IDE Manual enter the arguments (PortSocket Desktop\MyWebServer-master\out\test\MyWebServer-master\se\lnu\http\resources\inner) for HTTPServerConsole2. Run HTTPServerConsole |
| Expected result | “Port is taken” pops up in the console window |
| Alternative Scenario | 2a. The web server starts |
| Comments | PortSocket: 1412 ShareResource: Desktop\MyWebServer-master\out\test\MyWebServer-master\se\lnu\http\resources\inner Windows,Mac,Intellij snipped [pic 14] |

2. Access Log written

| | |
|----------------------|--|
| TUC1.3 | Visibility of access log |
| Requirement 5 | The access log should be viewable from a text editor |
| Use Cases | UC 1 Start Server |
| Scenario | When sever gets the request. A message will be visible to the access log |
| Pre-condition | Server is running |
| Test Steps | <ol style="list-style-type: none">1. send a request to the server2. A message will be visible in the access log and will be viewable in a text-editor |
| Expected result | System starts a web server on the given port and presents that the server was started and writes a note in the access log. |
| Alternative Scenario | <ol style="list-style-type: none">2a. Log is not createdb Log is not viewable in the text-editor |
| Comments | <p>The access log was not created on starting server and no error message was shown.</p> <p>Windows,Mac,Intellij snipped [pic 15]</p> |

UC2 Start Server

3. Stop Server

| TUC2.1 | Stop Server |
|----------------------|--|
| Requirement 3 | The web server must work on Linux, Mac, Windows |
| Use Cases | UC 2 Stop Server |
| Scenario | Web server stops when administrator Types “stop” |
| Pre-condition | Server is running |
| Test Steps | <ol style="list-style-type: none">1. Type “stop” in the terminal2. Press enter |
| Expected result | “HTTP Server Accept thread stopped HTTP Server stopped” are shown in the terminal window and web server stops |
| Alternative Scenario | 2a. web server does not terminates |
| Comments | Windows,Mac, IntelliJ snipped [pic 16] |

4. Access Log written

| | |
|----------------------|---|
| TUC2.2 | Visibility of access log |
| Requirement 5 | The access log should be viewable from a text editor |
| Use Cases | UC 2 Stop Server |
| Scenario | When sever gets the request to terminate access log is written. |
| Pre-condition | Server is running |
| Test Steps | <ol style="list-style-type: none">1. Type “stop” in the terminal window2. Press enter |
| Expected result | “HTTP Server Accept thread stopped HTTP Server stopped” are shown in the terminal window and web server stops and access log is created |
| Alternative Scenario | <ol style="list-style-type: none">2a. Log is not createdb Log is not viewable in the text-editor |
| Comments | The access log was not created on Stopping server and no error message was shown. |

UC2 Request shared resource

5. HTTP 1.1 status 200

| | |
|----------------------|---|
| TUC3.1 | HTTP 1.1 status 200 |
| Requirement 2 | The web server must follow minimum requirements for HTTP 1.1 |
| Use Cases | UC 3 Request Shared Resource |
| Scenario | Browser and System communicates using HTTP 1.1 with the status code 200 |
| Pre-condition | Server is running JMeter is running |
| Test Steps | <ol style="list-style-type: none">1. Add Sampler HTTP Request, Server name:localhost, Port Number 1412, Method GET and Path /index.html2. Add Response Assertion to test Pattern 2003. Start the test |
| Expected result | Server responds with status code 200 |
| Alternative Scenario | 2a. The server responds with different status code |
| Comments | Success response tested using JMeter: 200-OK Snipped [pic 17] |

6. HTTP 1.1 status 400

| | |
|----------------------|--|
| TUC3.2 | HTTP 1.1 status 400 |
| Requirement 2 | The web server must follow minimum requirements for HTTP 1.1 |
| Use Cases | UC 3 Request Shared Resource |
| Scenario | The web server should be able to respond with the status code 400 Using System communicates HTTP 1.1 |
| Pre-condition | Server is running JMeter is running |
| Test Steps | <ol style="list-style-type: none">1. Add Sampler HTTP Request, Server name:localhost, Port Number 1412, Method GET and Path /index.html2. Add “Daniel” in Body Data3. Run the test |
| Expected result | Server responds with status code 400 |
| Alternative Scenario | 2a. The server responds with different status code |
| Comments | Success response tested using JMeter: 400-Bad request Snipped [pic 18] |

7. HTTP 1.1 status 403

| | |
|----------------------|---|
| TUC3.3 | HTTP 1.1 status 403 |
| Requirement 2 | The web server must follow minimum requirements for HTTP 1.1 |
| Use Cases | UC 3 Request Shared Resource |
| Scenario | The web server should be able to respond with the status code 403 Using System communicates HTTP 1.1 |
| Pre-condition | Server is running JMeter is running |
| Test Steps | <ol style="list-style-type: none">1. Add Sampler HTTP Request, Server name:localhost, Port Number 1412, method GET and Path ../../secret.html2. Run the test |
| Expected result | Server responds with status code 403 |
| Alternative Scenario | 2a. The server responds with different status code |
| Comments | Success response tested using JMeter: 403-Forbidden Snipped [pic 19] |

8. HTTP 1.1 status 404

| | |
|----------------------|--|
| TUC3.4 | HTTP 1.1 status 404 |
| Requirement 2 | The web server must follow minimum requirements for HTTP 1.1 |
| Use Cases | UC 3 Request Shared Resource |
| Scenario | The web server should be able to respond with the status code 404 Using System communicates HTTP 1.1 |
| Pre-condition | Server is running JMeter is running |
| Test Steps | <ol style="list-style-type: none">1. Add Sampler HTTP Request, Server name:localhost, Port Number 1412, method GET and Path /daniel.html2. Run the test |
| Expected result | Server responds with status code 404 |
| Alternative Scenario | 2a. The server responds with different status code |
| Comments | Success response tested using JMeter: 404-Not Found Snipped [pic 20] |

9. HTTP 1.1 status 405

| | |
|----------------------|--|
| TUC3.5 | HTTP 1.1 status 405 |
| Requirement 2 | The web server must follow minimum requirements for HTTP 1.1 |
| Use Cases | UC 3 Request Shared Resource |
| Scenario | The web server should be able to respond with the status code 405 Using System communicates HTTP 1.1 |
| Pre-condition | Server is running JMeter is running |
| Test Steps | <ol style="list-style-type: none"> 1. Add Sampler HTTP Request, Server name: localhost, Port Number 1412, Method Post and Path /index.html 2. Run the test |
| Expected result | Server responds with status code 405 |
| Alternative Scenario | 2a. The server responds with different status code |
| Comments | Success response tested using JMeter: 405 Method not supported Snipped [pic 21] |

The web server should be responsive under high load

1. High load 2000

| THL.1.1 | High Load 2000 |
|----------------------|---|
| Requirement 1 | The web server should be responsive under high load |
| Use Cases | -- |
| Scenario | The web server should be responsive under high load |
| Pre-condition | Server is running JMeter is running |
| Test Steps | <ol style="list-style-type: none">1. Create 2000 users Thread Group2. Number of loops: 13. Add Sampler HTTP Request, Server name:localhost, Port Number 1412, Method GET and Path /index.html4. Run the test |
| Expected result | Server responds properly |
| Alternative Scenario | 3a. The server not responsive under high load |
| Comments | Server responded using JMeter Snipped [pic 22] |

2. High load 200

| THL.1.2 | High Load 200 |
|----------------------|--|
| Requirement 1 | The web server should be responsive under high load |
| Use Cases | -- |
| Scenario | The web server should be responsive under high load |
| Pre-condition | Server is running JMeter is running |
| Test Steps | <ol style="list-style-type: none">1. Create 200 users Thread Group2. Number of loops: 2503. Add Sampler HTTP Request, Server name:localhost, Port Number 1412, Method GET and Path /index.html4. Run the test |
| Expected result | Server responds properly |
| Alternative Scenario | 3a. The server not responsive under high load |
| Comments | Server responded using JMeter Snipped [pic 23] |

The web server must work on Linux, Mac, Windows

| TOS 1.1 | Test Operating System |
|----------------------|--|
| Requirement 3 | The web server must work on Linux, Mac, Windows |
| Use Cases | ----- |
| Scenario | The web server must work on Linux, Mac, Windows |
| Pre-condition | All three Operating systems are available |
| Test Steps | <ol style="list-style-type: none">1. Run the server in windows2. Run the server in Linux3. Run the server in Mac |
| Expected result | Server should respond properly in all the operating systems |
| Alternative Scenario | 3a. The server not responsive under high load |
| Comments | The server is up and running in all three operating systems |

Additional Tests

Browser test

| | |
|----------------------|---|
| AT.1.1 | Test browser |
| Requirement | ----- |
| Use Cases | ----- |
| Scenario | The web server should accessible in varies internet browsers |
| Pre-condition | Server is running |
| Test Steps | <ol style="list-style-type: none">1. Open google Chrome Enter localhost:14122. Open Firefox Enter localhost:14123. Open Microsoft Edge Enter localhost:14124. Open safari Enter localhost:1412 |
| Expected result | An image “it works” is shown on all the browser |
| Alternative Scenario | ----- |
| Comments | Tests passes in all browsers |

Acceptance

| AT.1.2 | Acceptance |
|----------------------|---|
| Requirement | ----- |
| Use Cases | |
| Scenario | The web server has been tested and is ready for deployment |
| Pre-condition | None |
| Test Steps | 5. Run Junit tests 6. Run Manual tests |
| Expected result | All test should pass |
| Alternative Scenario | |
| Comments | The goal of Acceptance test is to evaluate whether it fulfils the requirements and whether it is acceptable for deployments |

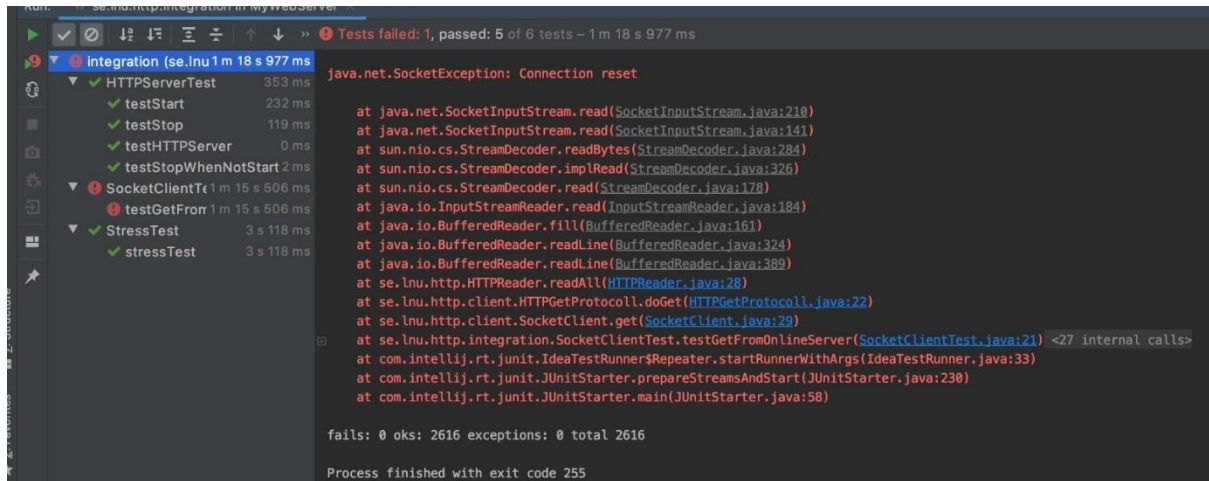
The source code should be released under GPL-2.0

| TGPL 2.0 | GPL-2.0 |
|----------------------|---|
| Requirement 4 | The source code should be released under GPL-2.0 |
| Use Cases | -- |
| Scenario | The server should be under a GPL 2.0 source license “code”. |
| Pre-condition | None |
| Test Steps | <ol style="list-style-type: none">1. Open MyWebServer-master/LICENSE2. Check whether the license of the server is GPL2.0 |
| Expected result | The web server is under GPL license |
| Alternative Scenario | |
| Comments | ----- |

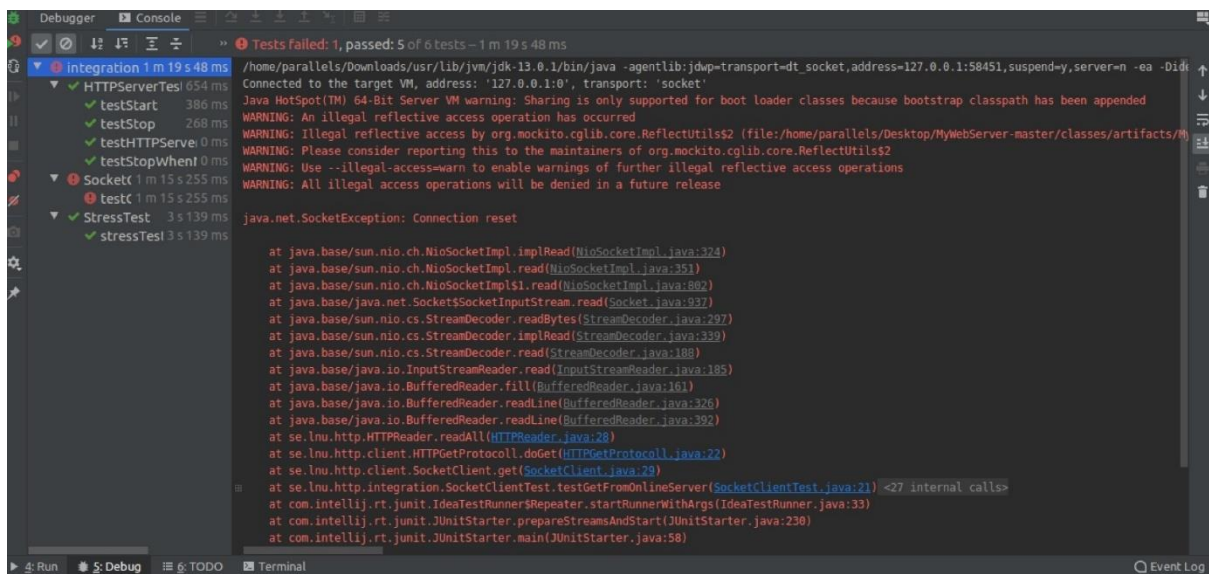
Test results

1. Integration Test

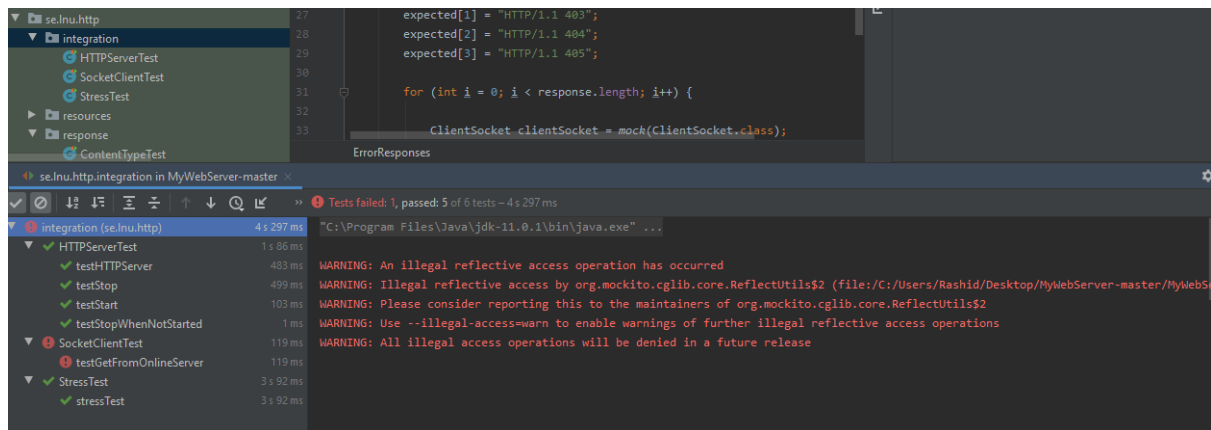
- Mac, IntelliJ snipped [pic 1]



- Linux, IntelliJ snipped [pic 2]

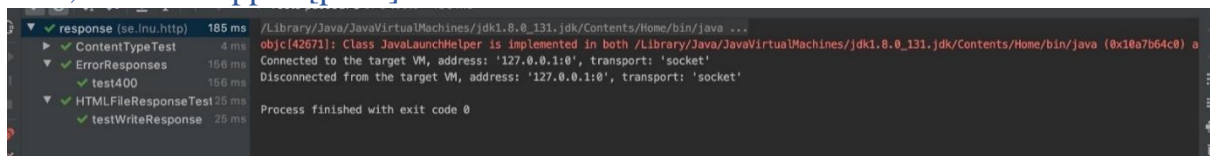


- Windows, IntelliJ snipped [pic 3]

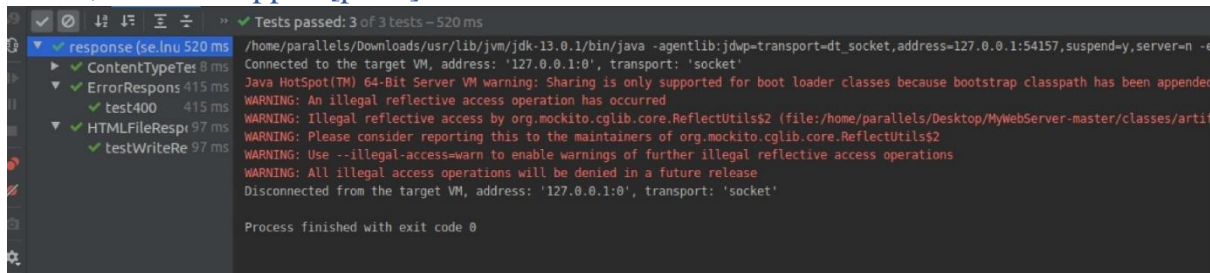


2. Response Test

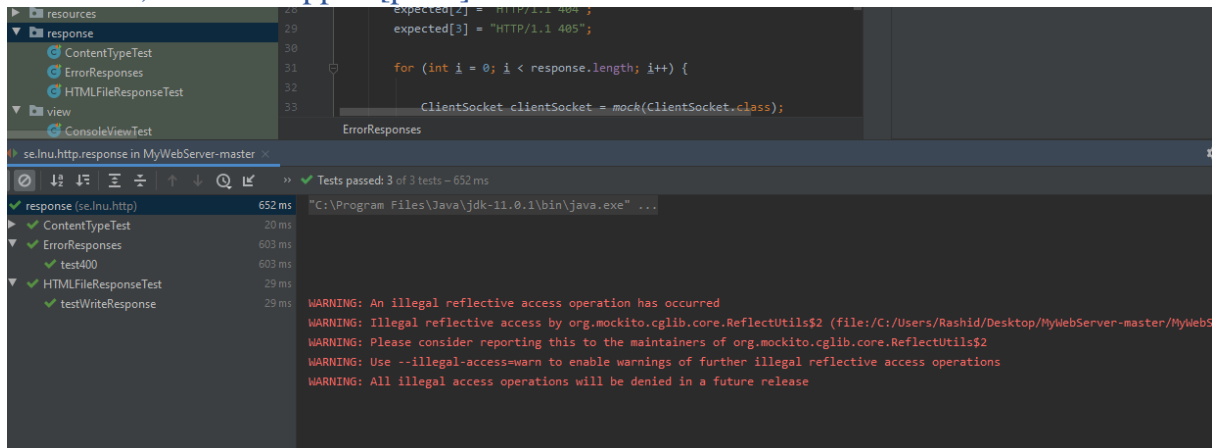
Mac, IntelliJ snipped [pic 4]



Linux, IntelliJ snipped [pic 5]

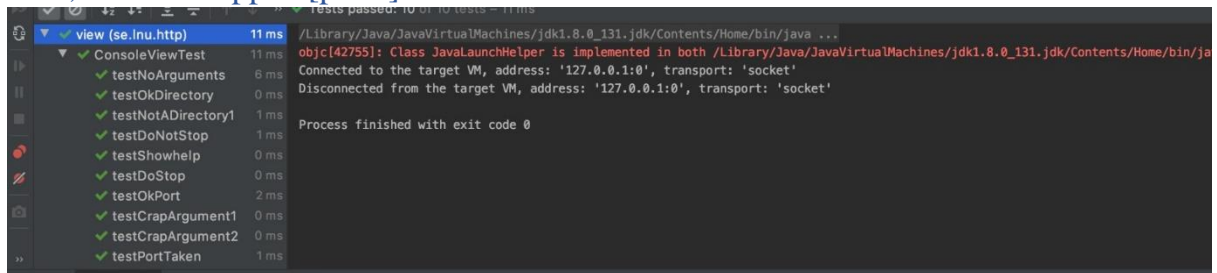


Windows, IntelliJ snipped [pic 6]

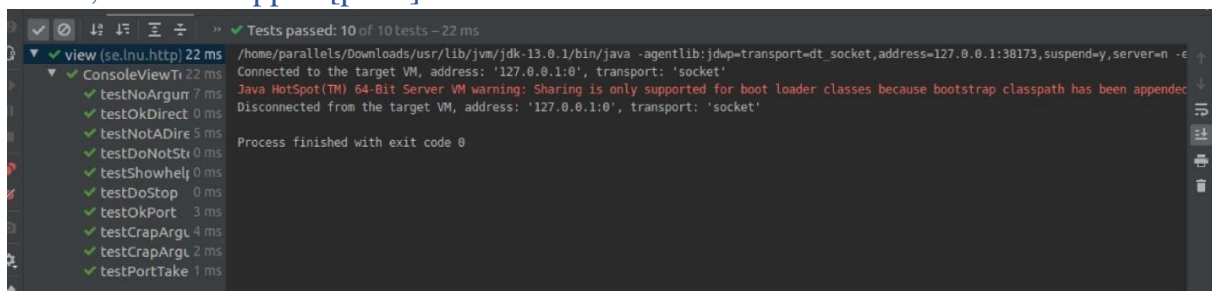


3. View Test

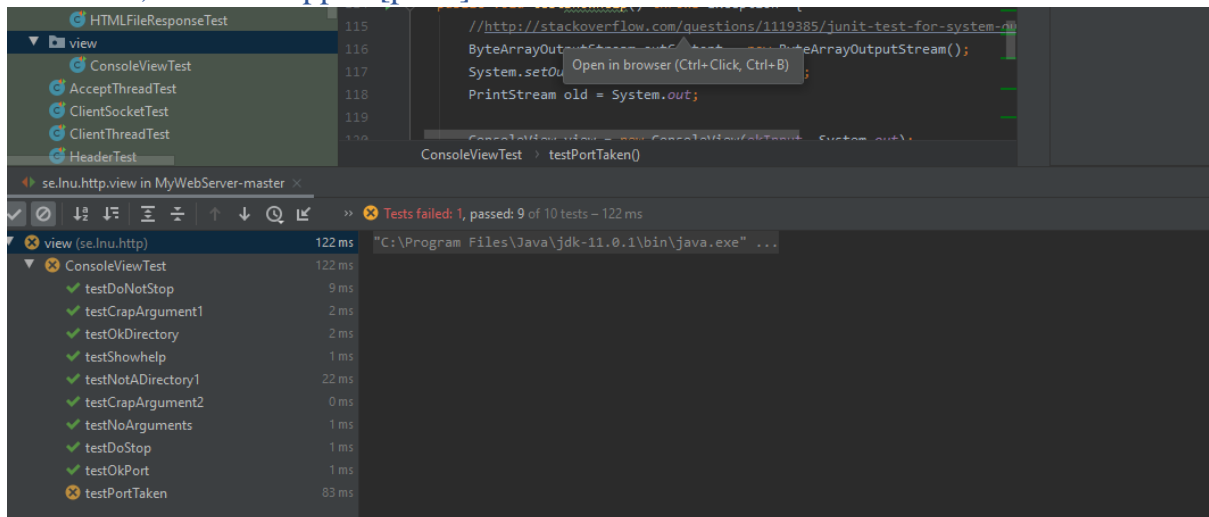
Mac, IntelliJ snipped [pic 7]



Linux, IntelliJ snipped [pic 8]

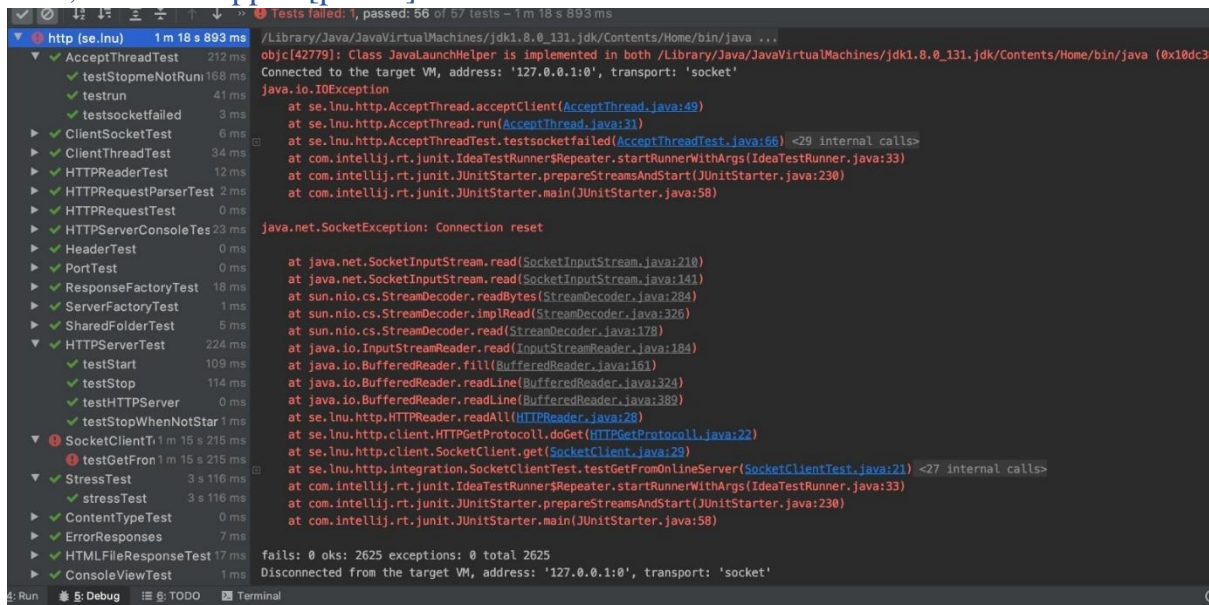


Windows, IntelliJ snipped [pic 9]

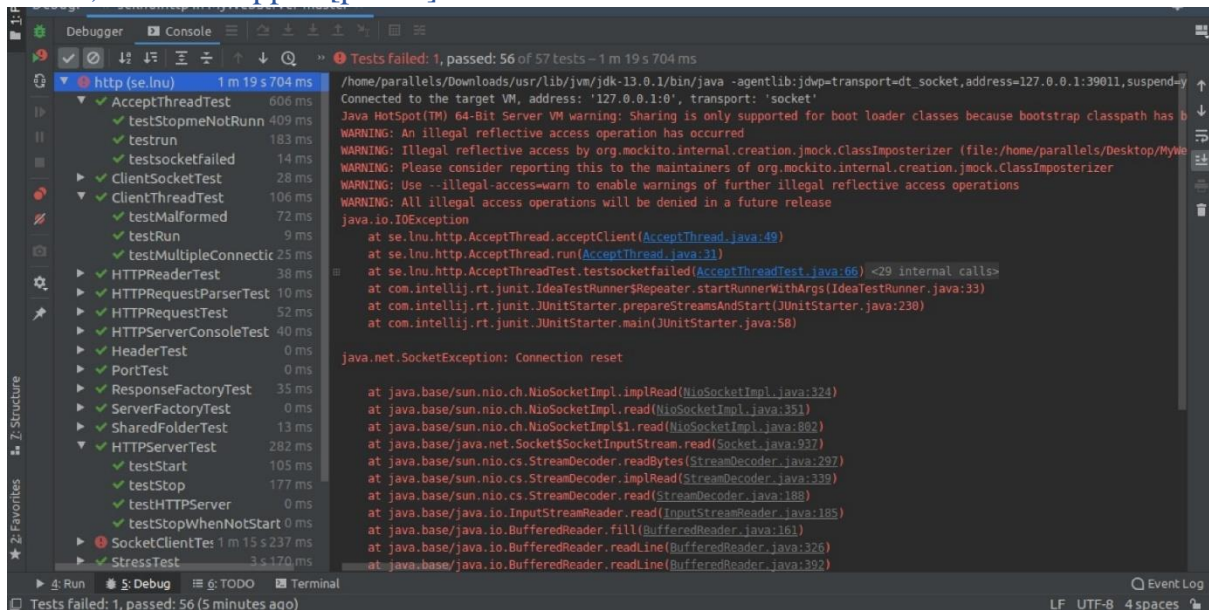


4. Group Test

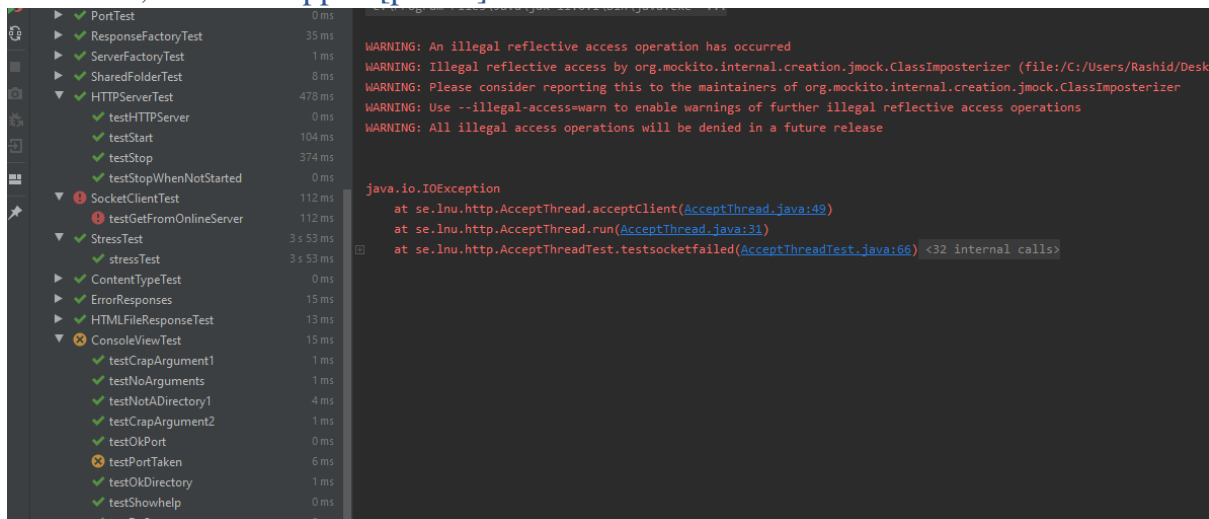
Mac, IntelliJ snipped [pic 10].



Linux, IntelliJ snipped [pic 11]

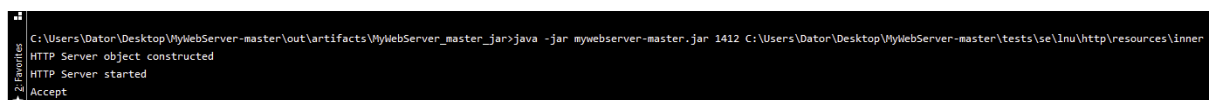


Windows, IntelliJ snipped [pic 12].



Windows,Mac, IntelliJ snipped [pic 13]

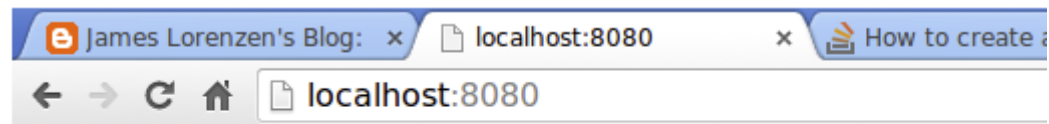
HTTP Server started



```
[...]@MacBook-Air:mywebserver_jar$ java -jar mywebserver.jar 1412  
/Users/mu.../Downloads/MyWebServer/tests/se/lnu/http/resources/inner  
HTTP Server object constructed  
HTTP Server started  
Accept
```

← → ↻ ⓘ localhost:1412

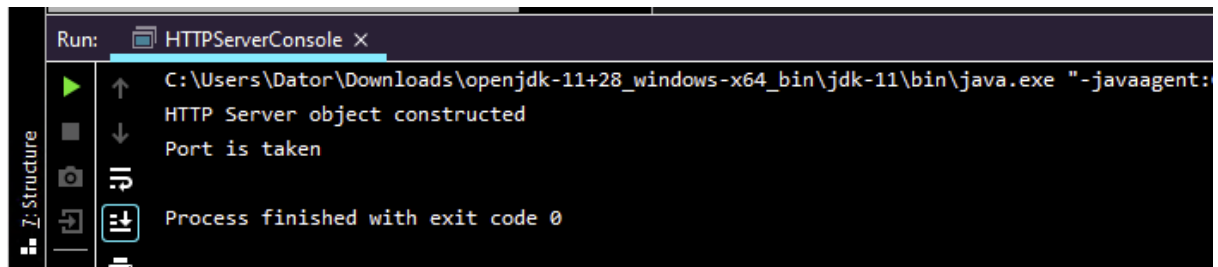
It works



It works

Windows,Mac IntelliJ snipped [pic 14]

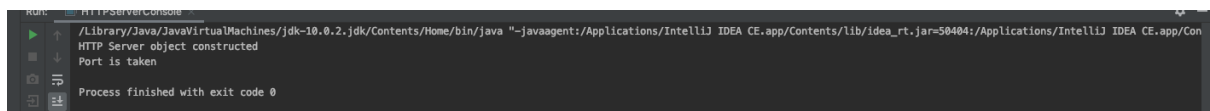
Port is taken



The image shows a screenshot of the IntelliJ Run console for a Windows environment. The console title is "Run: HTTPServerConsole X". The output text is as follows:

```
C:\Users\Dator\Downloads\openjdk-11+28_windows-x64_bin\jdk-11\bin\java.exe "-javaagent:  
HTTP Server object constructed  
Port is taken  
Process finished with exit code 0
```

On the left side of the console, there is a vertical toolbar with icons for running, stepping through code, and other debugging actions. The "Process finished" icon at the bottom is highlighted with a blue square.



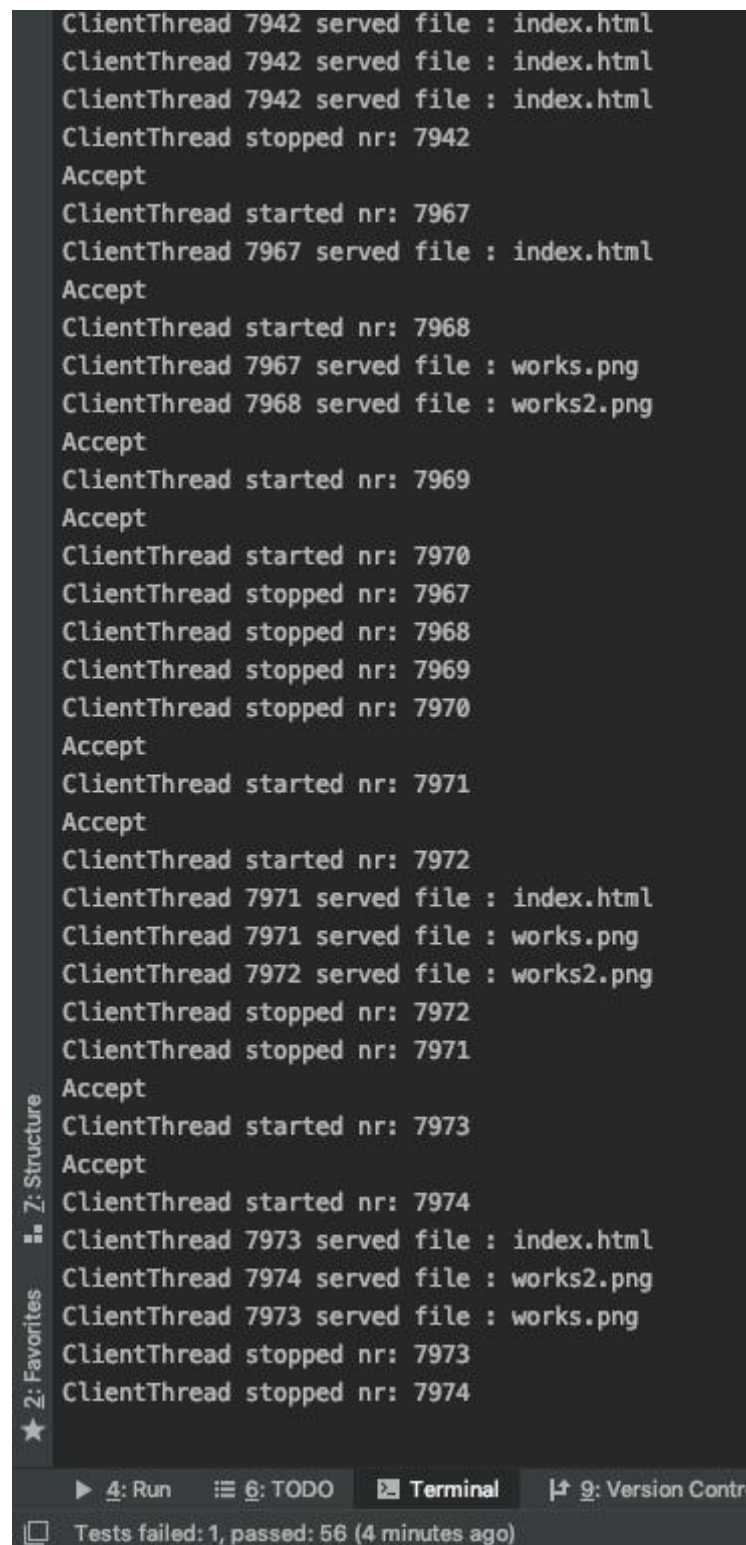
The image shows a screenshot of the IntelliJ Run console for a Mac environment. The console title is "Run: HTTPServerConsole". The output text is as follows:

```
/Library/Java/JavaVirtualMachines/jdk-10.0.2.jdk/Contents/Home/bin/java "-javaagent:/Applications/IntelliJ IDEA CE.app/Contents/lib/idea_rt.jar=50404:/Applications/IntelliJ IDEA CE.app/Con  
HTTP Server object constructed  
Port is taken  
Process finished with exit code 0
```

Similar to the Windows screenshot, there is a vertical toolbar on the left with debugging icons. The "Process finished" icon at the bottom is highlighted with a blue square.

Windows, IntelliJ snipped [pic 15]

Access log



```
ClientThread 7942 served file : index.html
ClientThread 7942 served file : index.html
ClientThread 7942 served file : index.html
ClientThread stopped nr: 7942
Accept
ClientThread started nr: 7967
ClientThread 7967 served file : index.html
Accept
ClientThread started nr: 7968
ClientThread 7967 served file : works.png
ClientThread 7968 served file : works2.png
Accept
ClientThread started nr: 7969
Accept
ClientThread started nr: 7970
ClientThread stopped nr: 7967
ClientThread stopped nr: 7968
ClientThread stopped nr: 7969
ClientThread stopped nr: 7970
Accept
ClientThread started nr: 7971
Accept
ClientThread started nr: 7972
ClientThread 7971 served file : index.html
ClientThread 7971 served file : works.png
ClientThread 7972 served file : works2.png
ClientThread stopped nr: 7972
ClientThread stopped nr: 7971
Accept
ClientThread started nr: 7973
Accept
ClientThread started nr: 7974
ClientThread 7973 served file : index.html
ClientThread 7974 served file : works2.png
ClientThread 7973 served file : works.png
ClientThread stopped nr: 7973
ClientThread stopped nr: 7974
```

★ 2: Favorites 7: Structure

▶ 4: Run ≡ 6: TODO ☒ Terminal ↗ 9: Version Contr

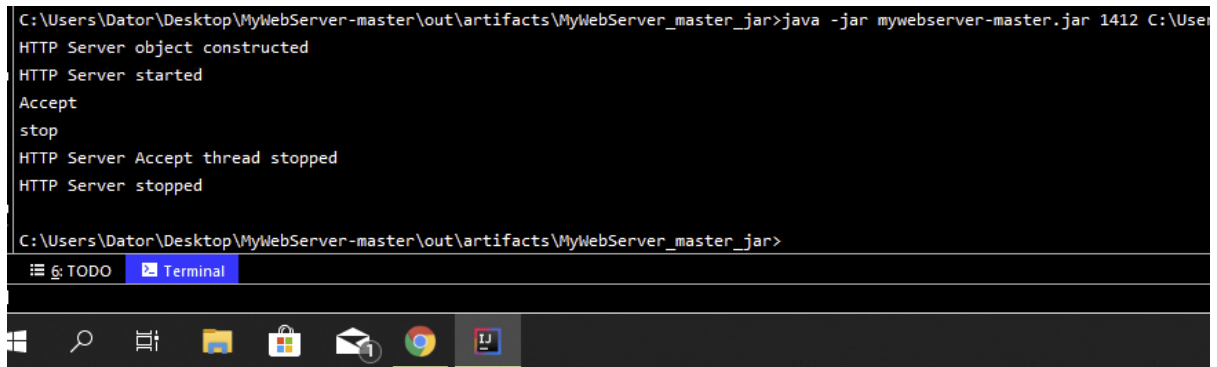
☐ Tests failed: 1, passed: 56 (4 minutes ago)

Windows, Mac, IntelliJ snipped [pic 16].

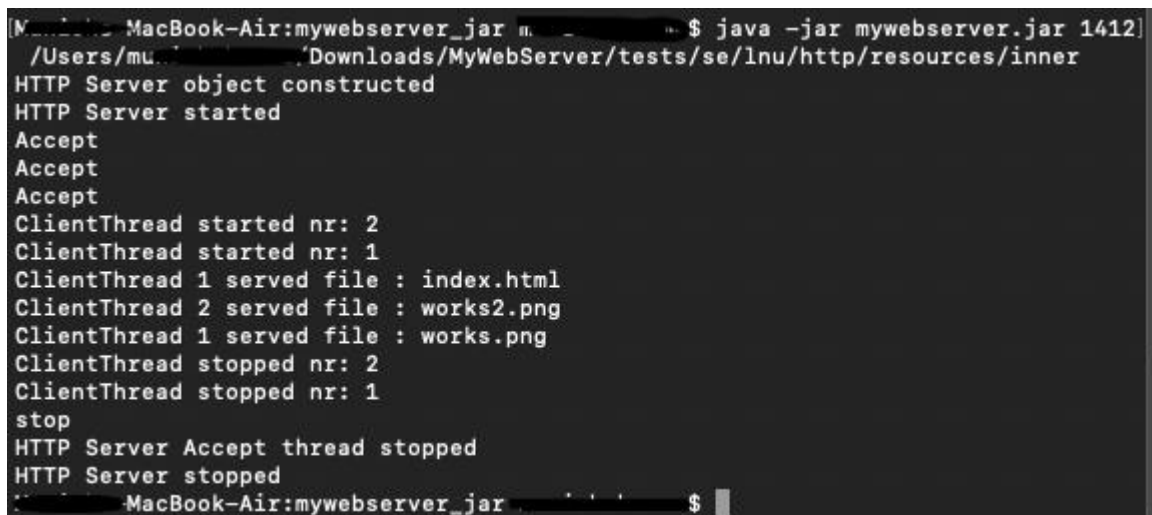
HTTP Server Stopped

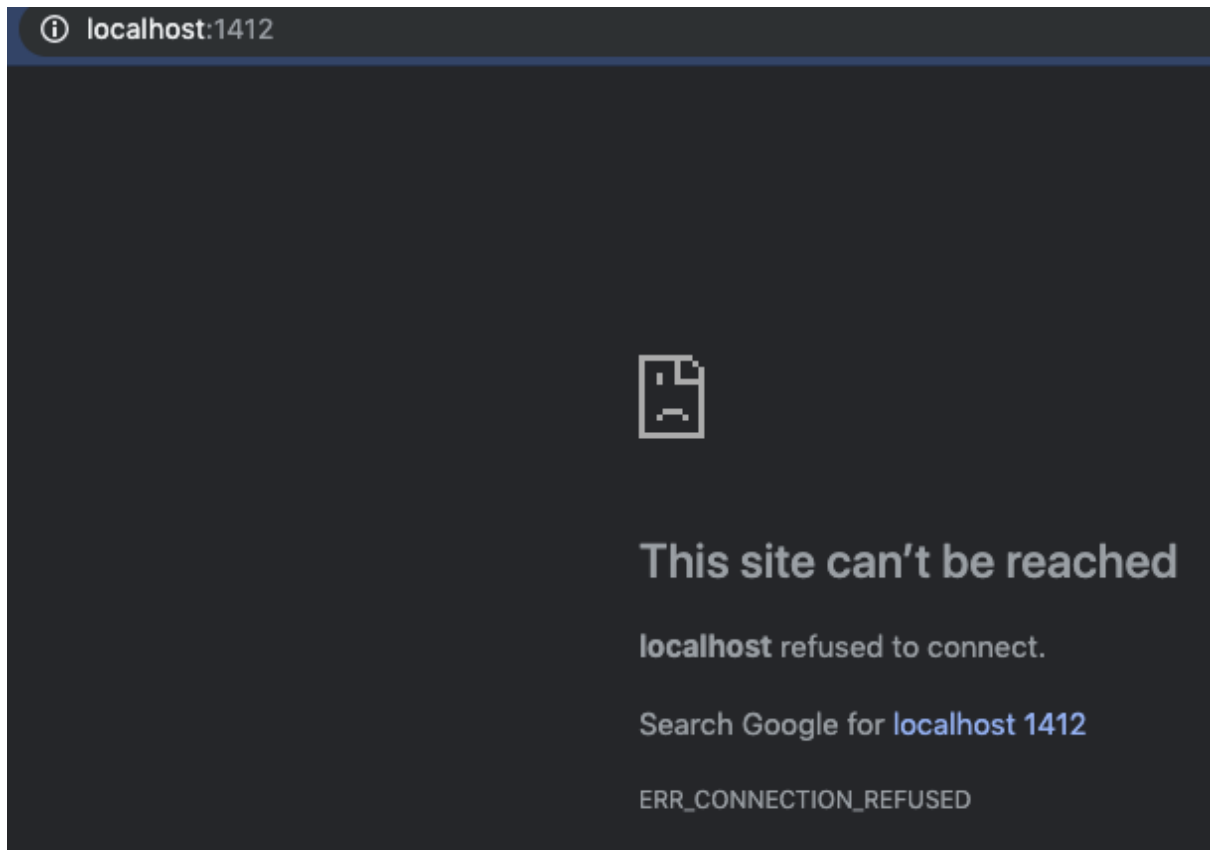
```
C:\Users\Dator\Desktop\MyWebServer-master\out\artifacts\MyWebServer_master_jar>java -jar mywebserver-master.jar 1412 C:\User
HTTP Server object constructed
HTTP Server started
Accept
stop
HTTP Server Accept thread stopped
HTTP Server stopped

C:\Users\Dator\Desktop\MyWebServer-master\out\artifacts\MyWebServer_master_jar>
```



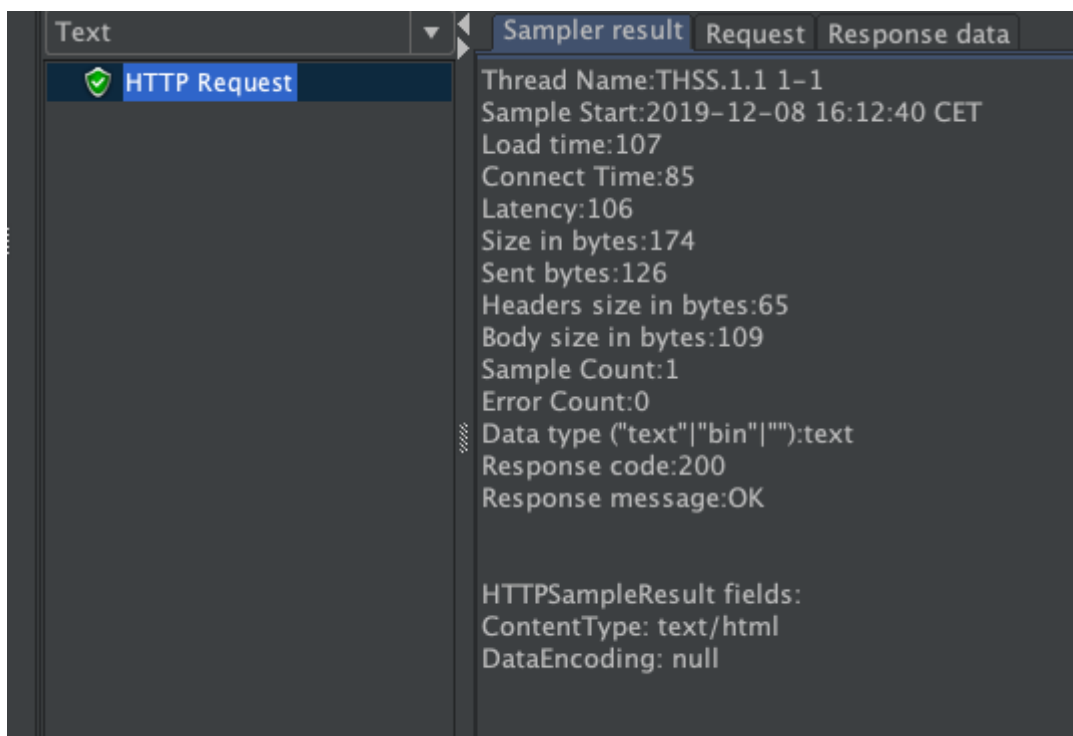
```
MacBook-Air:mywebserver_jar m... $ java -jar mywebserver.jar 1412]
/Users/mu.../Downloads/MyWebServer/tests/se/lnu/http/resources/inner
HTTP Server object constructed
HTTP Server started
Accept
Accept
Accept
ClientThread started nr: 2
ClientThread started nr: 1
ClientThread 1 served file : index.html
ClientThread 2 served file : works2.png
ClientThread 1 served file : works.png
ClientThread stopped nr: 2
ClientThread stopped nr: 1
stop
HTTP Server Accept thread stopped
HTTP Server stopped
MacBook-Air:mywebserver_jar ... $
```





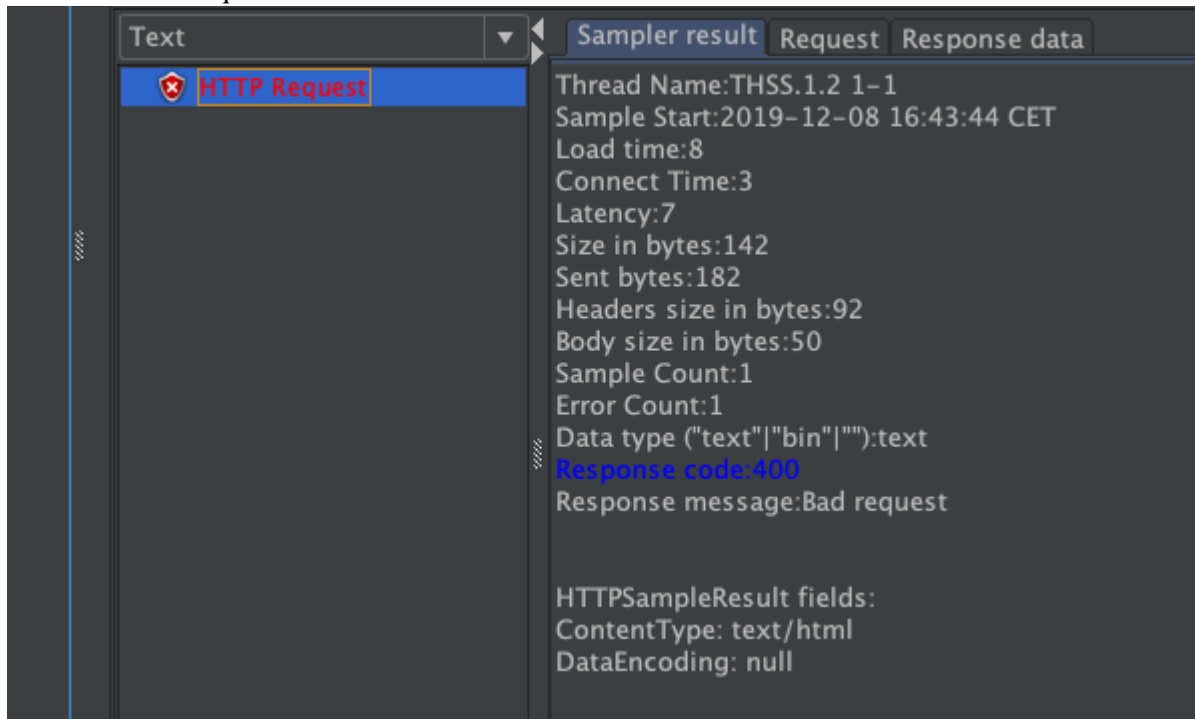
Windows,Mac,IntelliJ snipped [pic 17]

Status 200



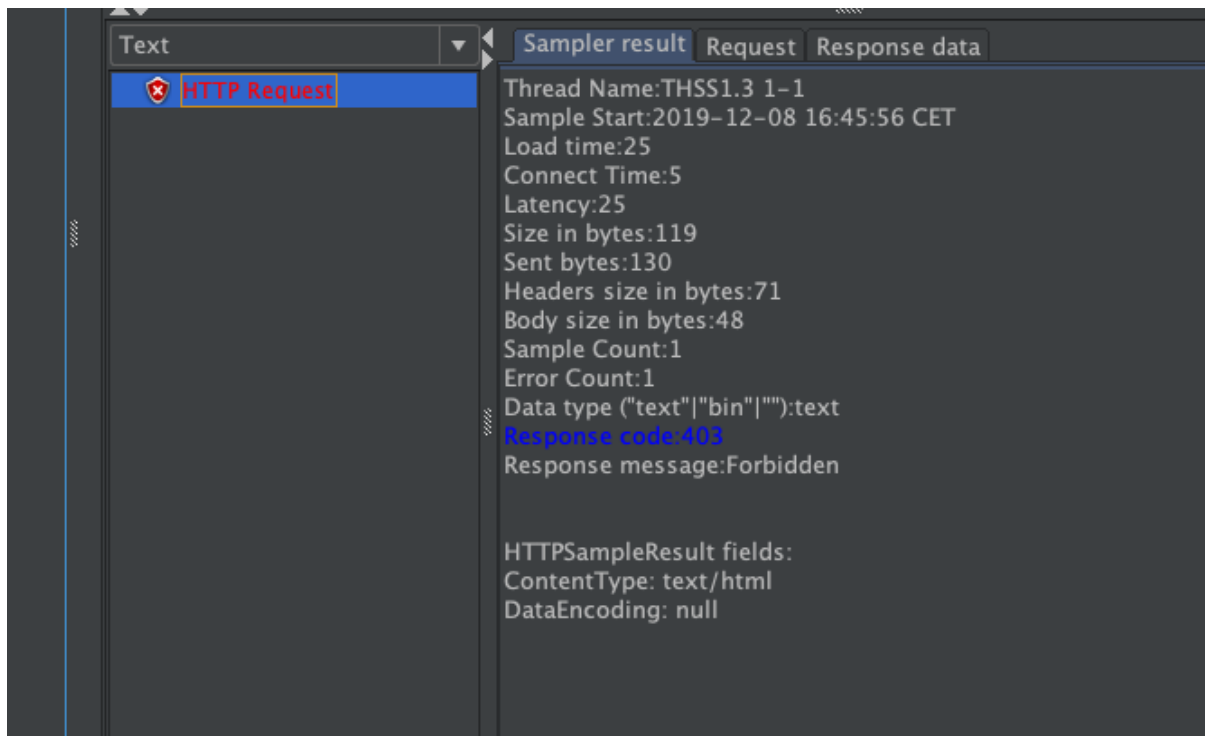
Windows,Mac,IntelliJ snipped [pic 18]

Status 400 Bad request



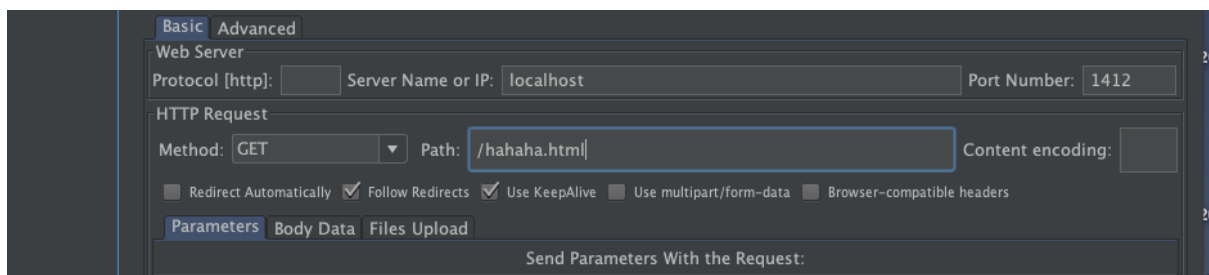
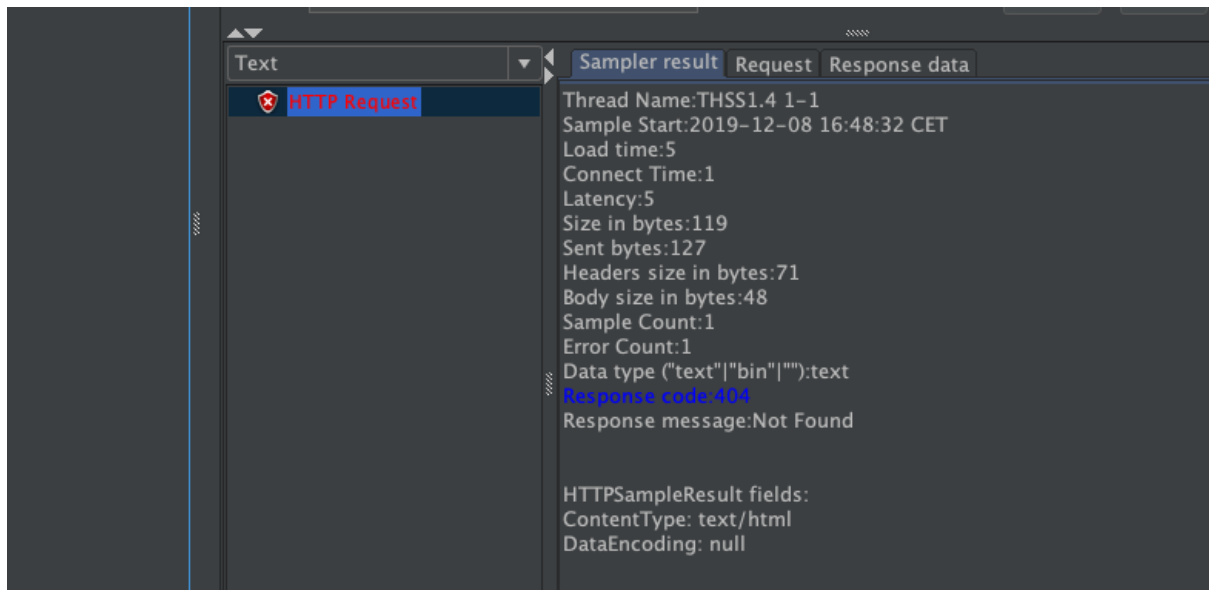
Windows,Mac,IntelliJ snipped [pic 19]

Status 403 Forbidden



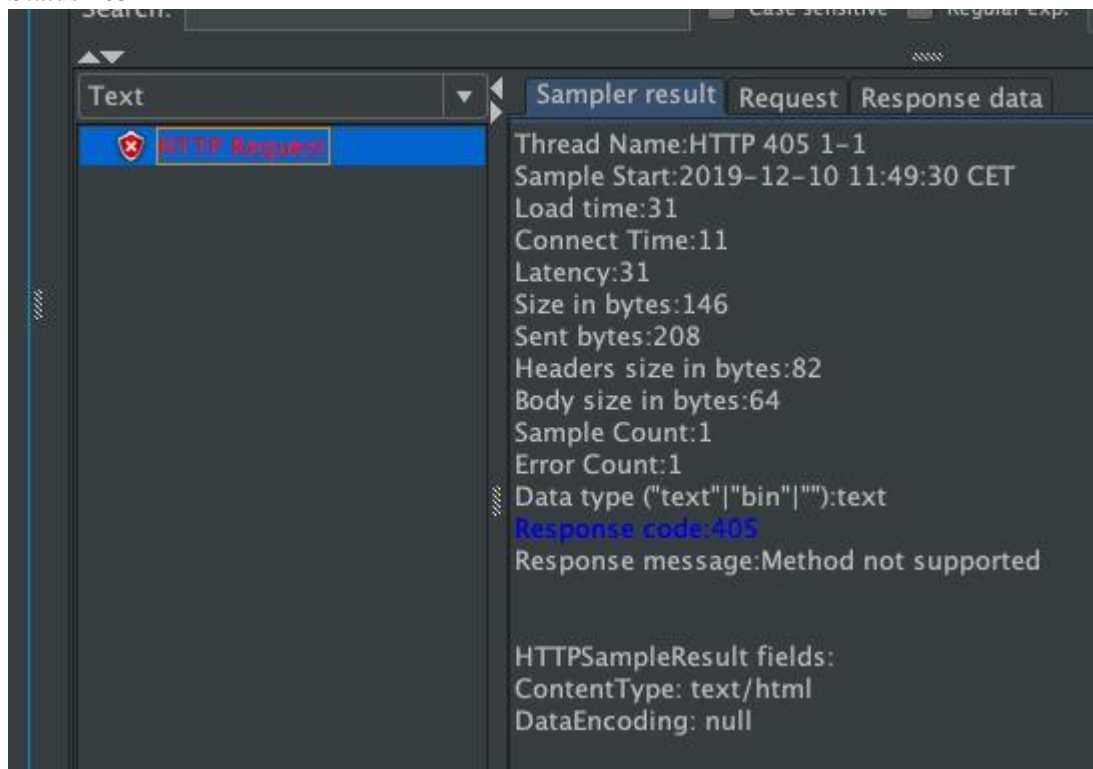
Windows,Mac,IntelliJ snipped [pic 20]

Status 404 Not Found



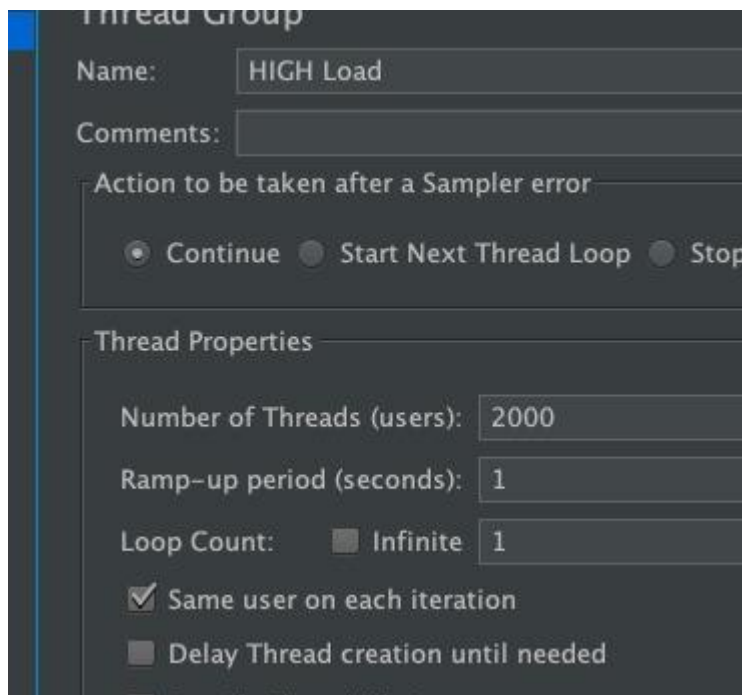
Snipped [pic 21]

Status 405



Snipped [pic 22]

High load 2000



| | | | | | | |
|------|--------------|-------------------------------|------|---|-----|-----|
| 1988 | 13:08:27.412 | HIGH Load 1-1... HTTP Request | 4576 | ✓ | 174 | 126 |
| 1989 | 13:08:27.412 | HIGH Load 1-1... HTTP Request | 4576 | ✓ | 174 | 126 |
| 1990 | 13:08:27.507 | HIGH Load 1-1... HTTP Request | 4554 | ✓ | 174 | 126 |
| 1991 | 13:08:26.305 | HIGH Load 1-7... HTTP Request | 5756 | ✓ | 174 | 126 |
| 1992 | 13:08:26.304 | HIGH Load 1-7... HTTP Request | 5757 | ✓ | 174 | 126 |
| 1993 | 13:08:27.512 | HIGH Load 1-1... HTTP Request | 4553 | ✓ | 174 | 126 |
| 1994 | 13:08:26.309 | HIGH Load 1-7... HTTP Request | 5757 | ✓ | 174 | 126 |
| 1995 | 13:08:27.090 | HIGH Load 1-1... HTTP Request | 5636 | ✓ | 174 | 126 |
| 1996 | 13:08:27.092 | HIGH Load 1-1... HTTP Request | 5635 | ✓ | 174 | 126 |
| 1997 | 13:08:27.094 | HIGH Load 1-1... HTTP Request | 5635 | ✓ | 174 | 126 |
| 1998 | 13:08:27.093 | HIGH Load 1-1... HTTP Request | 5636 | ✓ | 174 | 126 |
| 1999 | 13:08:25.907 | HIGH Load 1-3... HTTP Request | 7825 | ✓ | 174 | 126 |
| 2000 | 13:08:25.915 | HIGH Load 1-4... HTTP Request | 7827 | ✓ | 174 | 126 |

☐ Scroll automatically?
 ☐ Child samples?
 No of Samples 2000
 Latest Sample 7827
 Average 3147

Snipped [pic 23]

High load 200

Thread Group

Name: HIGH Load

Comments:

Action to be taken after a Sampler error

☒ Continue

☐ Start Next Thread Loop

☐ Stop Thre

Thread Properties

Number of Threads (users): 200

Ramp-up period (seconds): 1

Loop Count:

☐ Infinite
 250

☒ Same user on each iteration

☐ Delay Thread creation until needed

☐ Specify Thread lifetime

Duration (seconds):

Startup delay (seconds):