**Vert.x vs Springboot**

**Aim:** To get performance metrics of API endpoints in Vert.x and Springboot applications and compare results using Jmeter.

**Configuration:**

**Hardware:**

**OS:** Windows 10 enterprise

**RAM:** 8GB

**Processor:** Intel core i3 CPU @3.2GHz

**Software:**

**Testing tool:** Jmeter

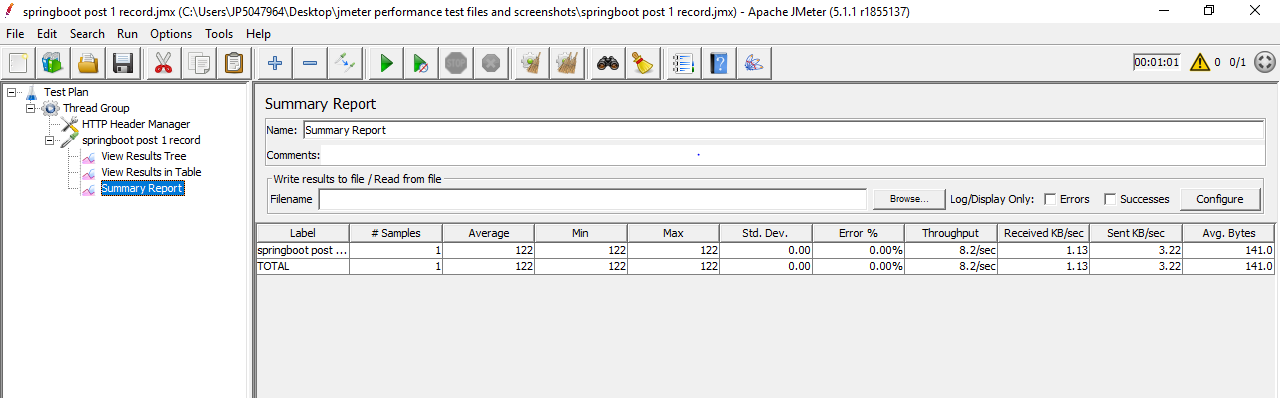
**Development IDE**: spring tool suite (STS)

**Database:** PostgreSQL

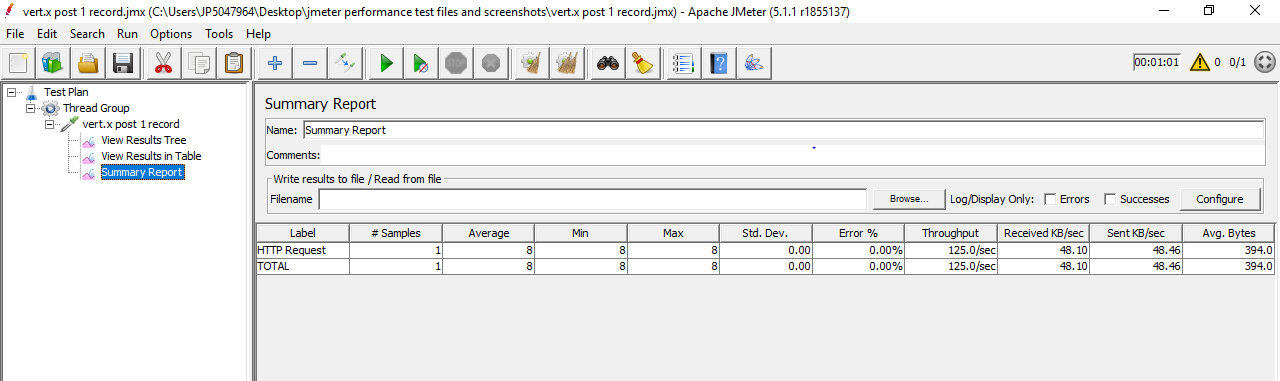
Below are test results tested for different API endpoints with screenshots copied and compared the result.

**Test1:** To post one record by 1 user into Postgre database

Springboot result:



Vert.x



|  |  |  |
| --- | --- | --- |
|  | **Vert.x** | **Springboot** |
|  |  |  |
| HTTP Request Samples | 1 | 1 |
| Average | 8 | 122 |
| Min | 8 | 122 |
| Max | 8 | 122 |
| Standard Deviation | 0 | 0 |
| Error % | 0.00% | 0.00% |
| Throughput | 125.0/sec | 8.2/sec |
| Test Execution time | 1 | 1 |

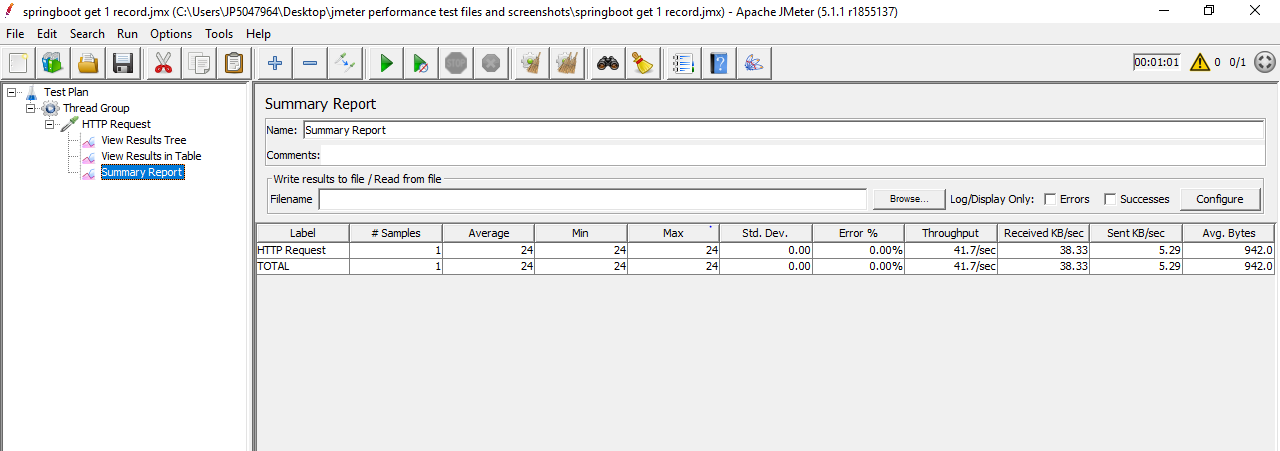
Conclusion:

Since the throughput is higher for vert.x , it performed better

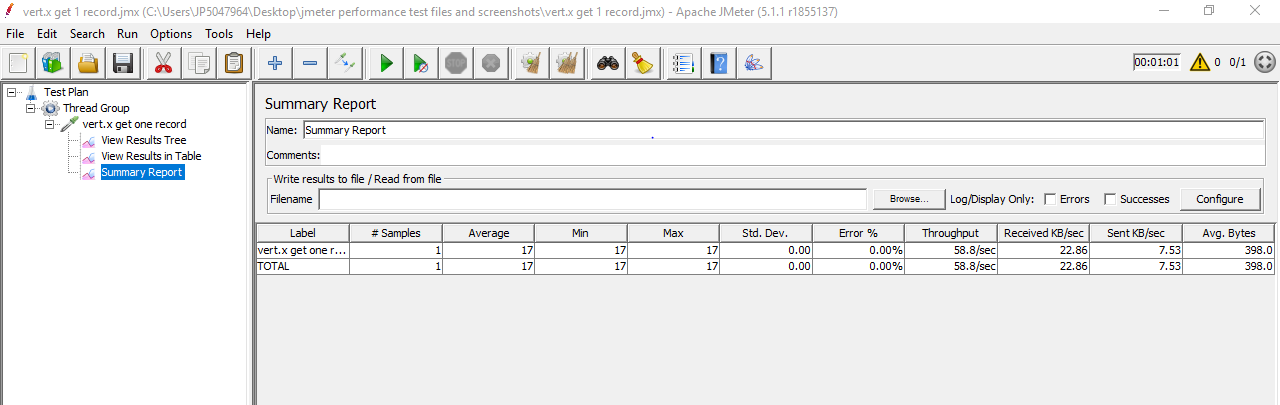
**Test 2:** To retrieve one record by 1 user from Postgre database

This is the case when only one record is present in database

Springboot result :



Vert.x result:



|  |  |  |
| --- | --- | --- |
|  | **vert.x** | **springboot** |
|  |  |  |
| HTTP Request Samples | 1 | 1 |
| Average | 17 | 24 |
| Min | 17 | 24 |
| Max | 17 | 24 |
| Standard Deviation | 0 | 0 |
| Error % | 0.00% | 0.00% |
| Throughput | 58.8/sec | 41.7/sec |
| Test Execution time | 1sec | 1 sec |

Conclusion:

Since the throughput is higher for vert.x , it performed better.

**Test3:** To post (100 records) data into Postgre database

Configuration:

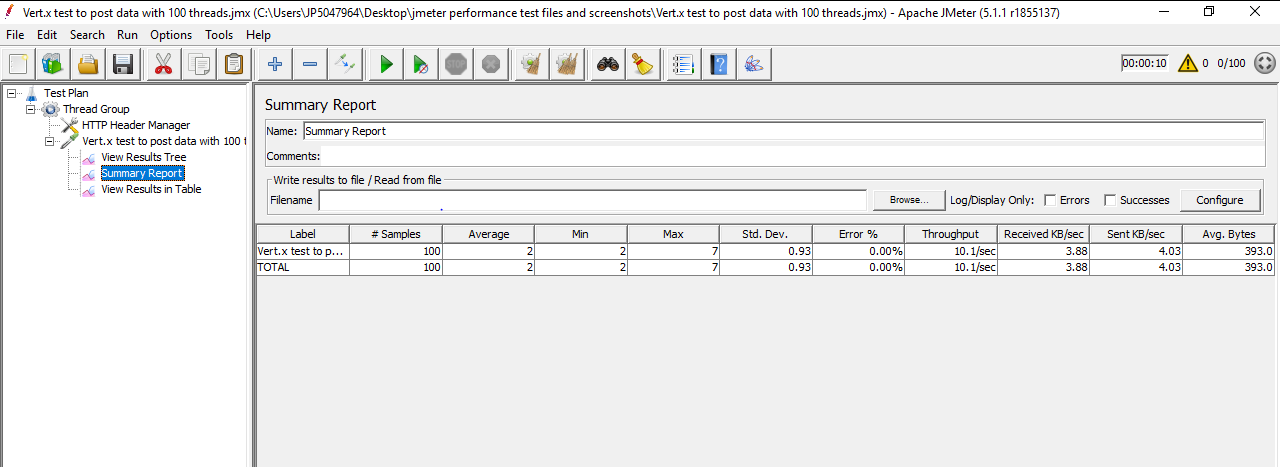
No. of threads: 100

Ramp-up period: 10sec

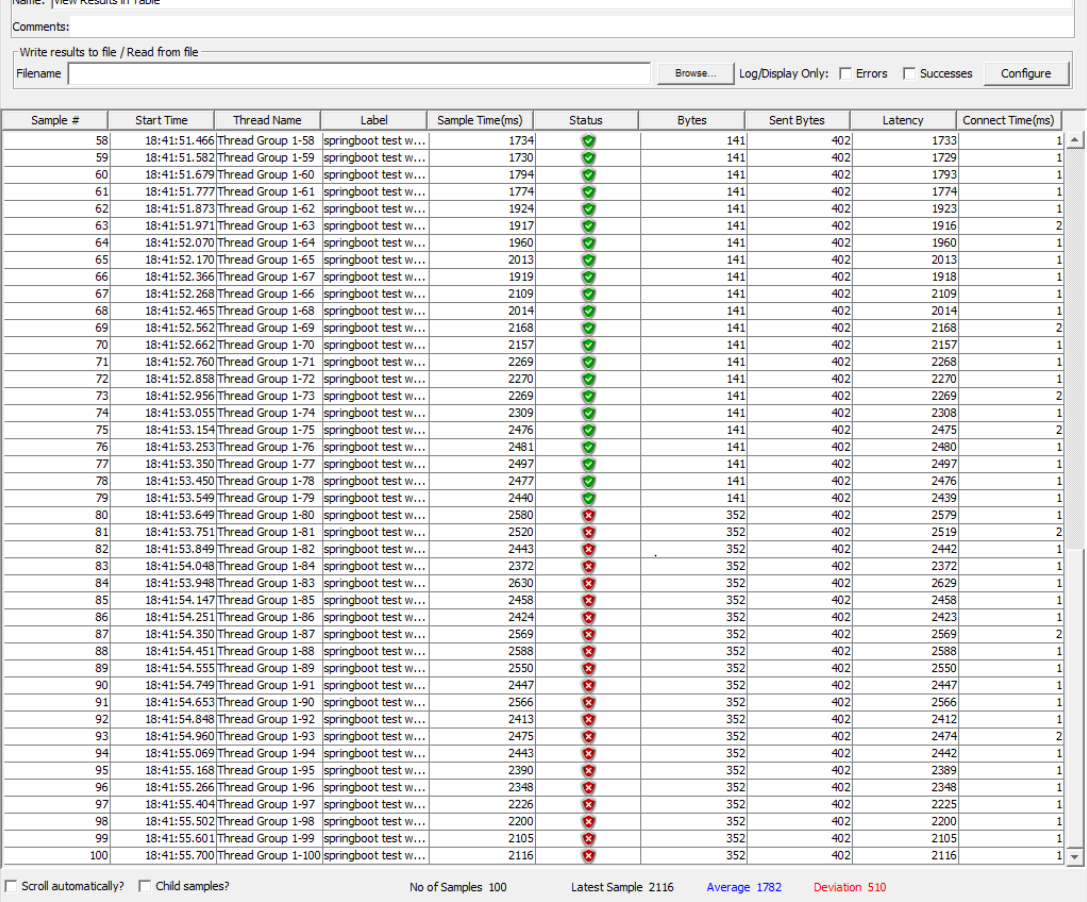
100/10=10

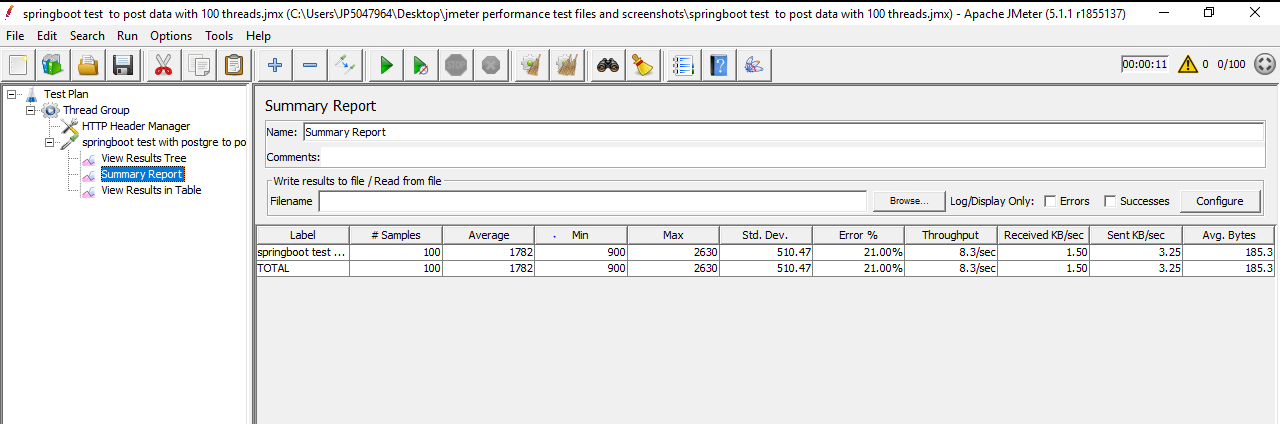
Which means for each sec 10 concurrent users will perform insertion operation. Total 100 records are inserted into database in 10 sec because ramp-up is 10sec.

Vert.x result:



Springboot result:





|  |  |  |
| --- | --- | --- |
|  | **Vert.x** | **Springboot** |
|  |  |  |
| HTTP Request Samples | 100 | 100 |
| Average | 2 | 1782 |
| Min | 2 | 900 |
| Max | 7 | 2630 |
| Standard Deviation | 0.93 | 510.47 |
| Error % | 0.00% | 21.00% |
| Throughput | 10.1/sec | 8.3/sec |
| Test Execution time | 10 sec | 11 sec |

Here some records are not inserted into database while testing for Springboot application. Tests are failed throwing exception: org.postgresql.util.PSQLException: FATAL: sorry, too many clients already

Conclusion:

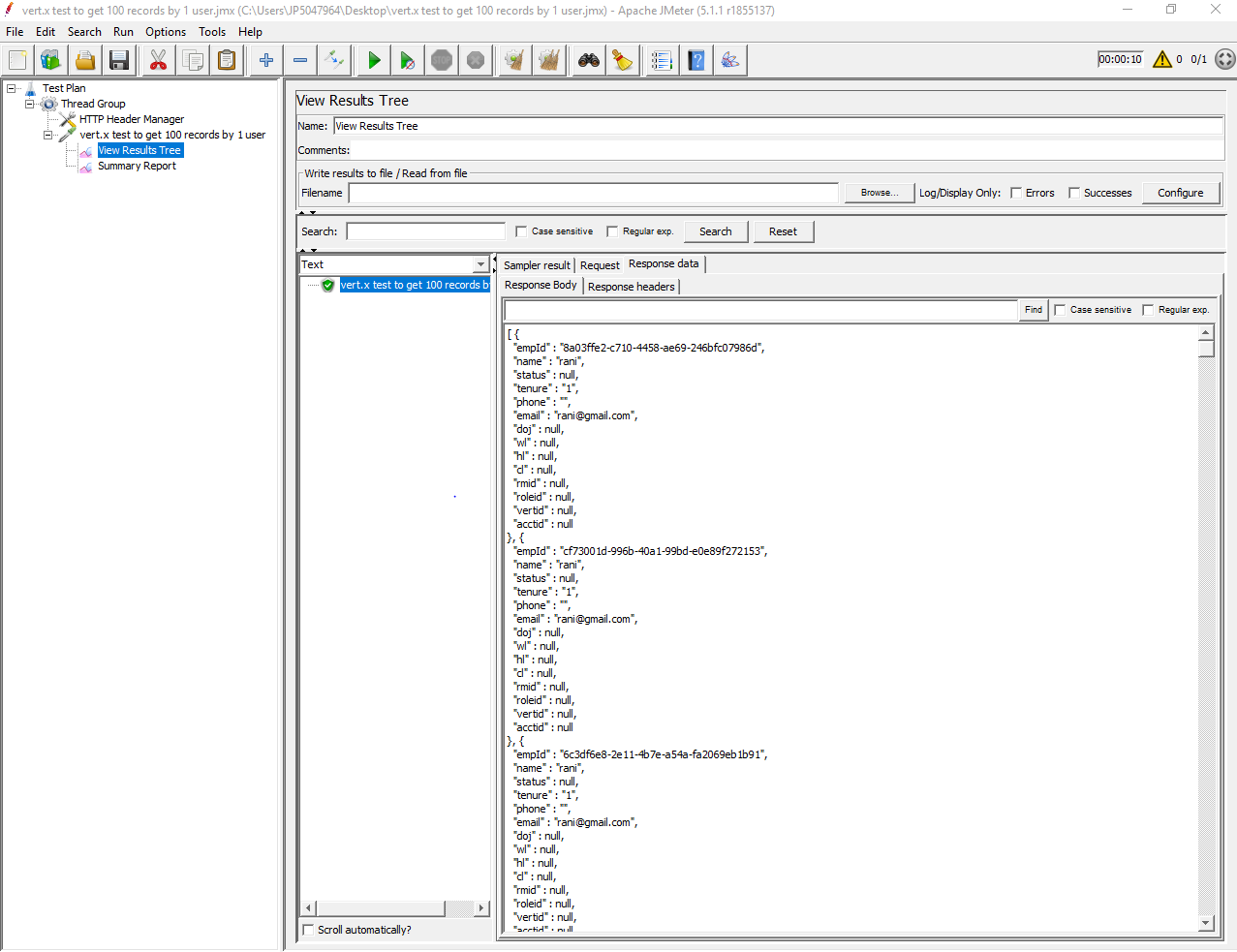
Vert.x inserted all the records and performed better than Springboot.

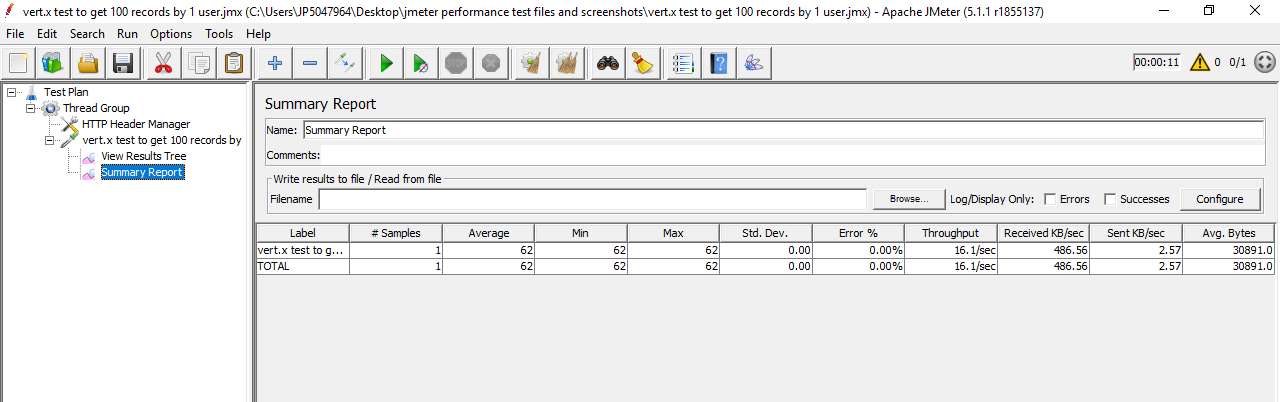
**Test4:** To retrieve all inserted 100 records by 1 user from database.

No. of threads:1

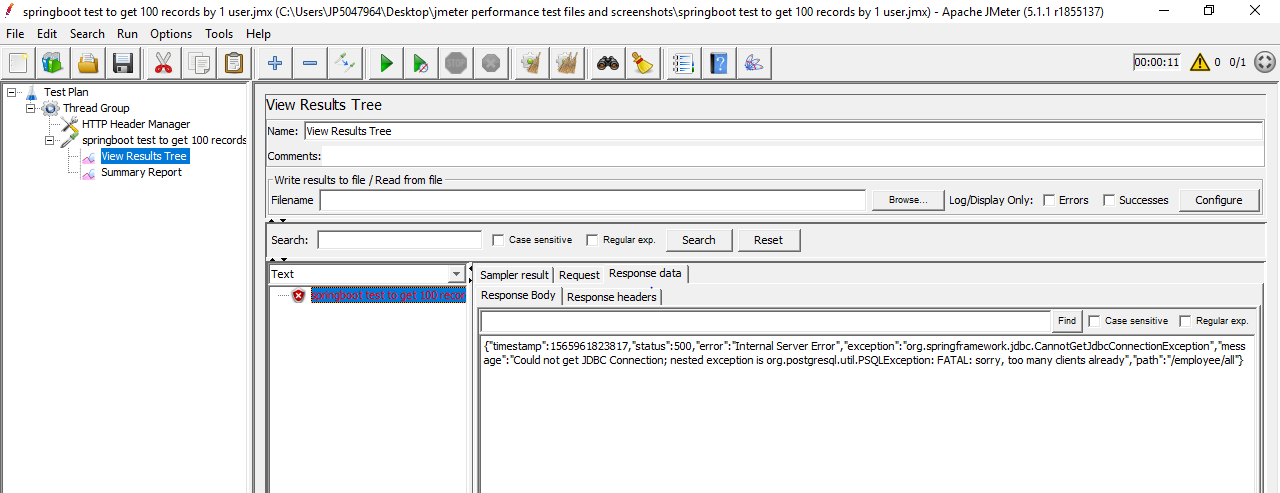
Ramp-up:1

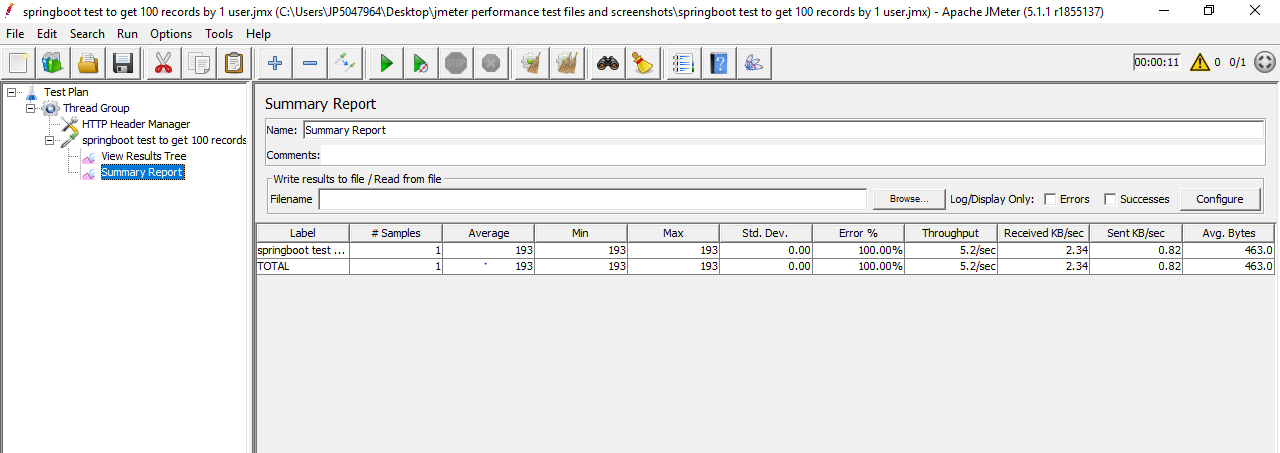
Vert.x result:





Springboot result:





|  |  |  |
| --- | --- | --- |
|  | **Vert.x** | **Springboot** |
|  |  |  |
| HTTP Request Samples | 1 | 1 |
| Average | 62 | 193 |
| Min | 62 | 193 |
| Max | 62 | 193 |
| Standard Deviation | 0 | 0 |
| Error % | 0.00% | 100.00% |
| Throughput | 16.1/sec | 5.2/sec |
| Test Execution time | 11 sec | 11 sec |

Since Springboot failed to insert 100 records in above test, now it failed to retrieve records.

Conclusion:

Vert.x retrieved all the records and performed better than Springboot.

**Test 5:** To post 1000 records into Postgre

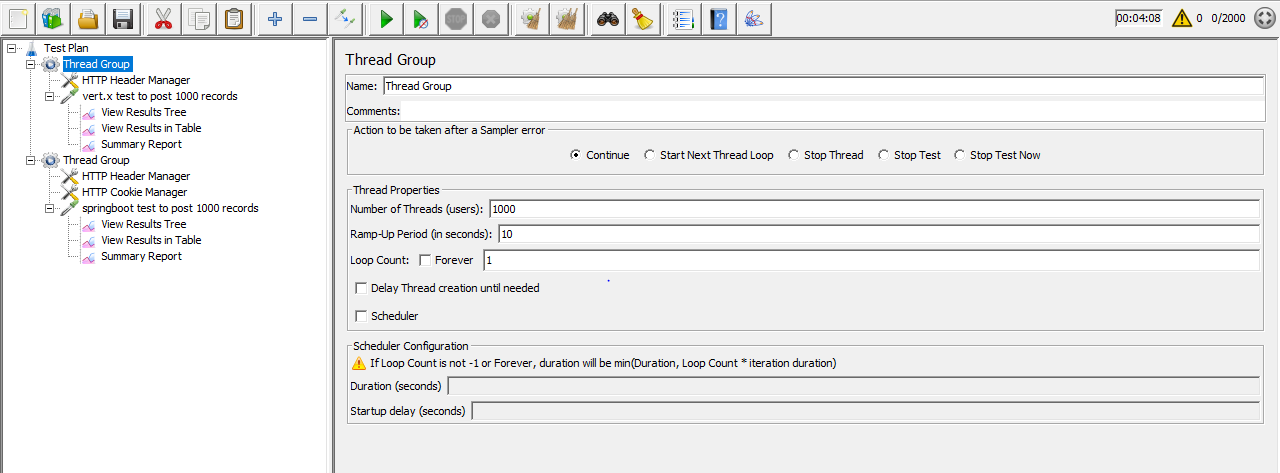
Configuration:

No. of threads:1000

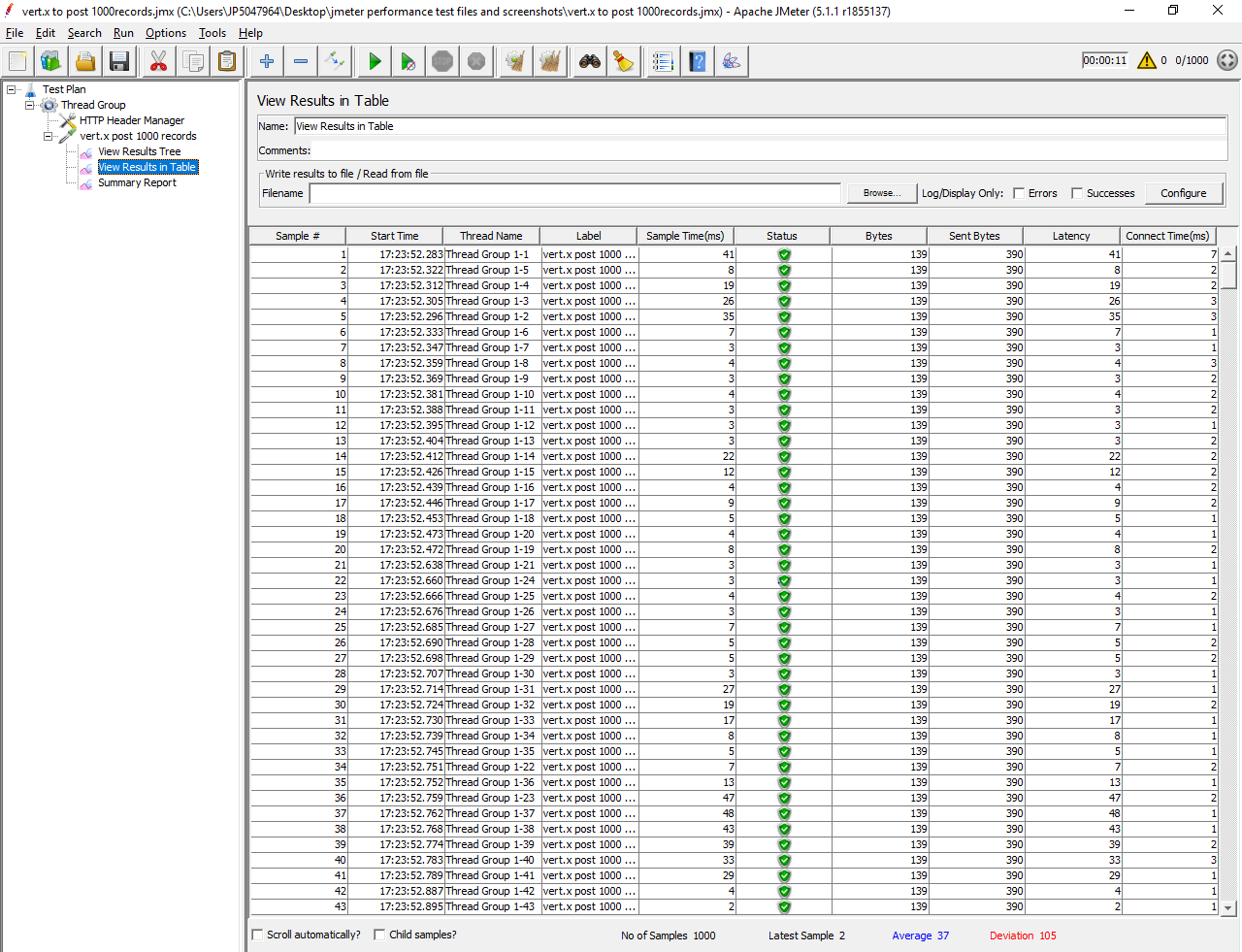
Ramp-up period:10

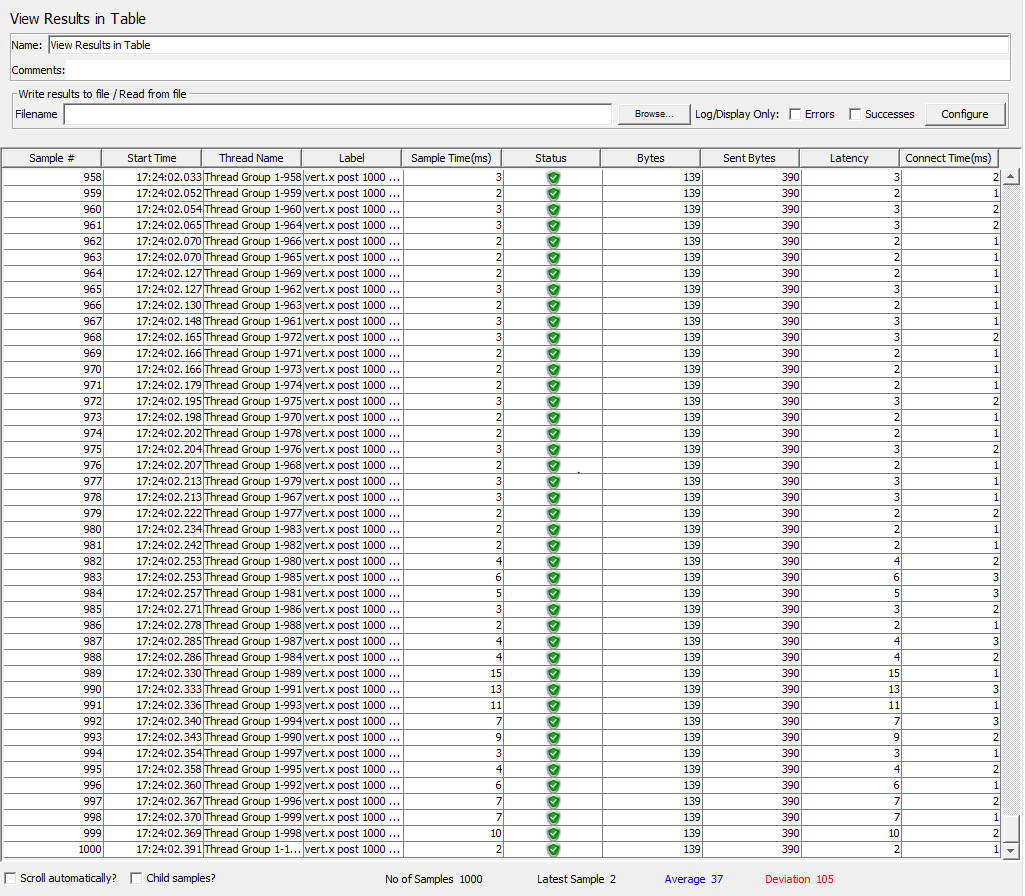
Here executing 100 parallel threads in each second for a period of 10 sec.

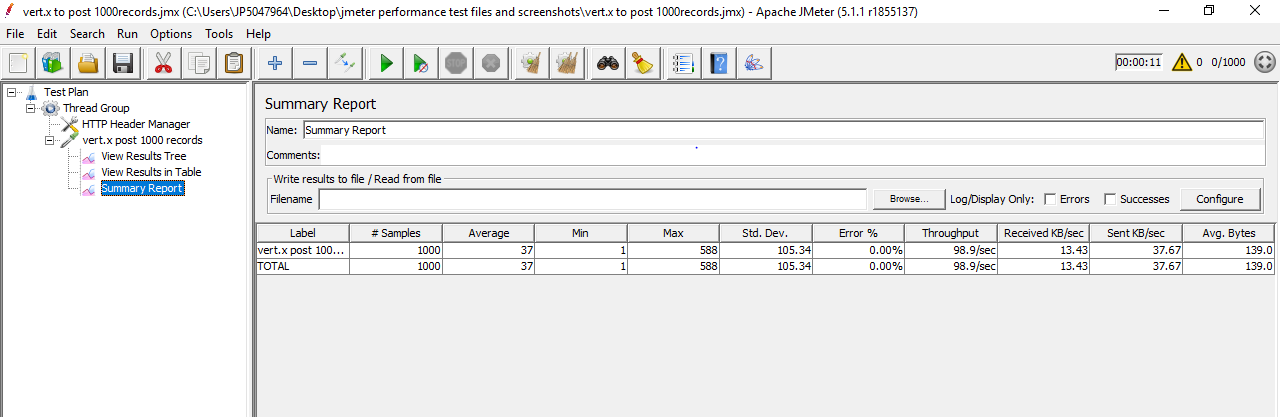
It means in each second , 100 concurrent users will come online to insert data. Totally 1000 records are inserted into database.



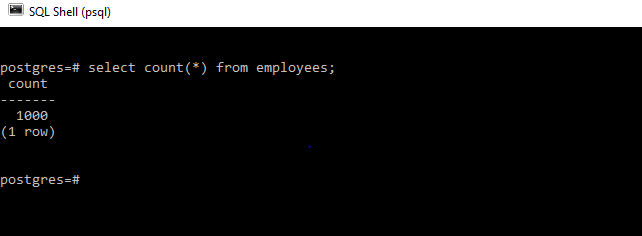
Vert.x



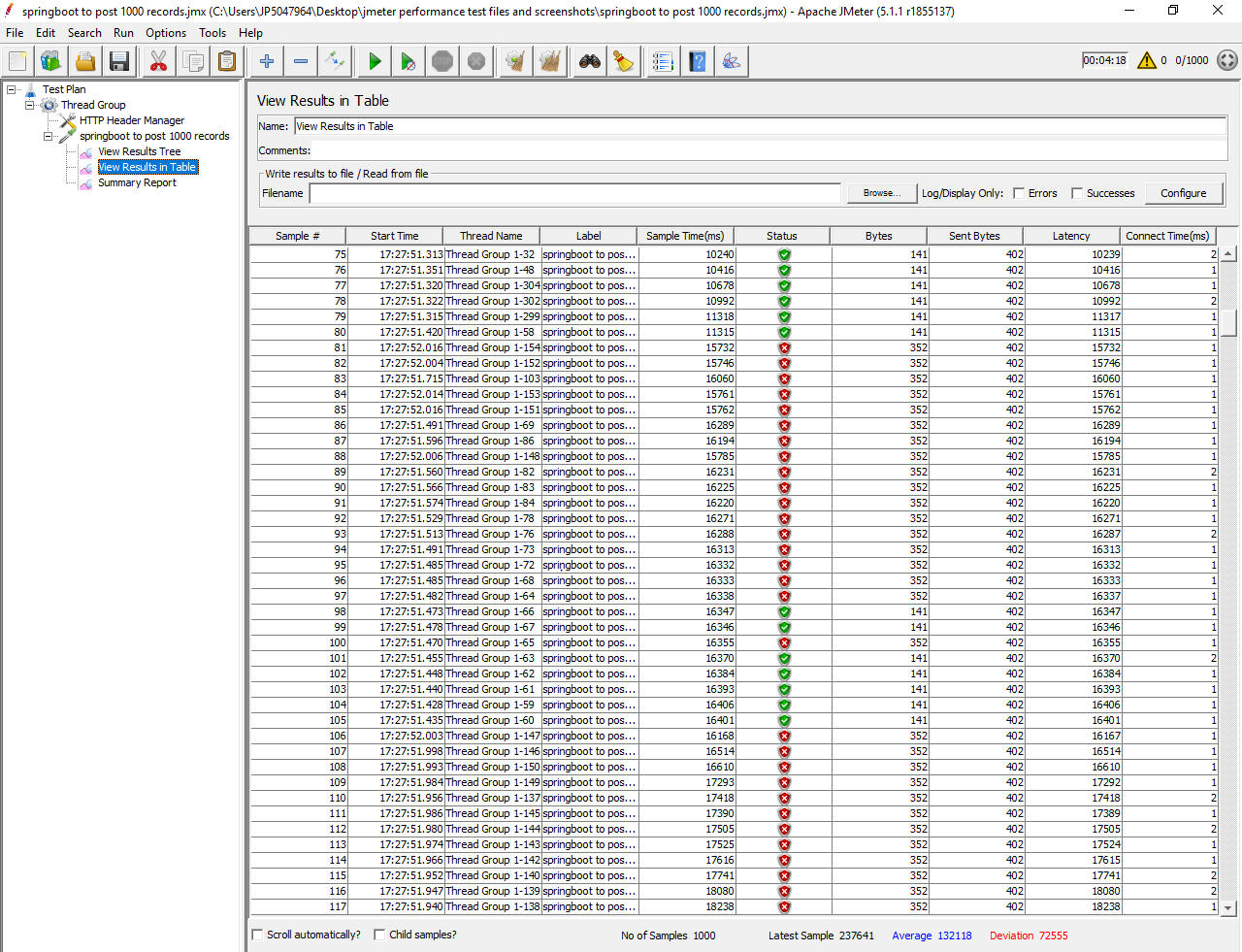


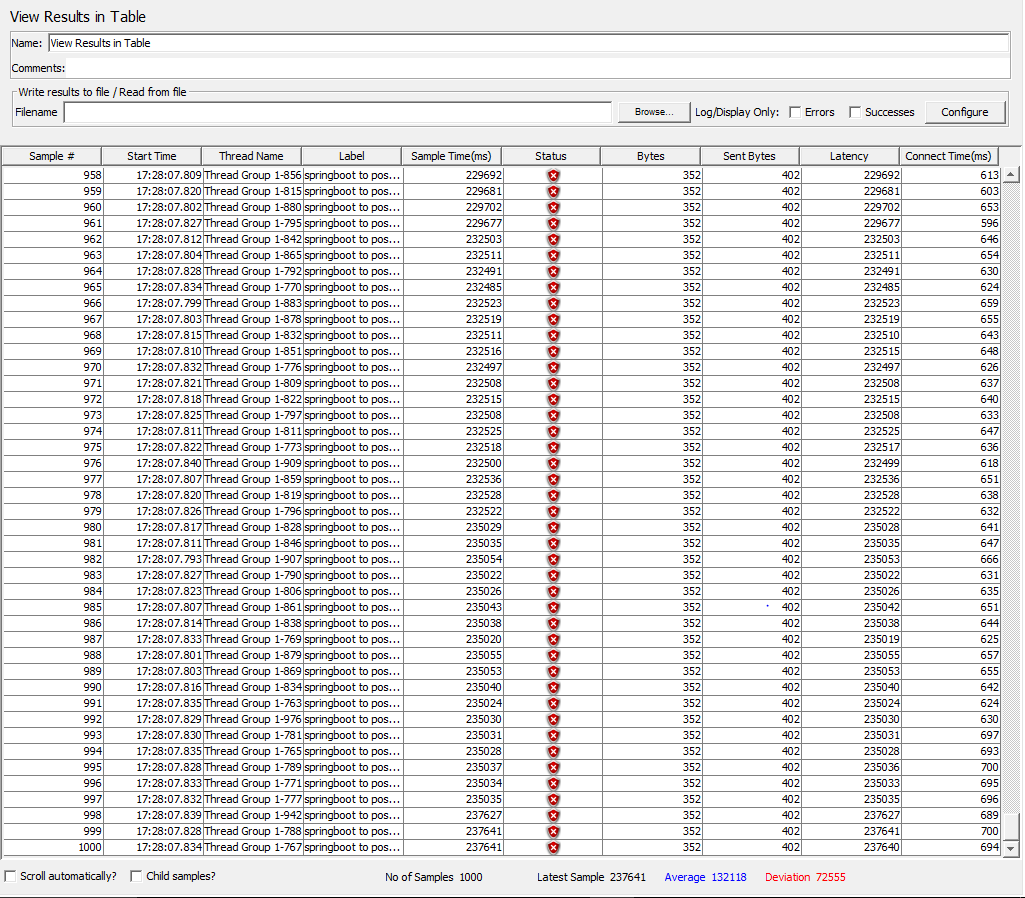


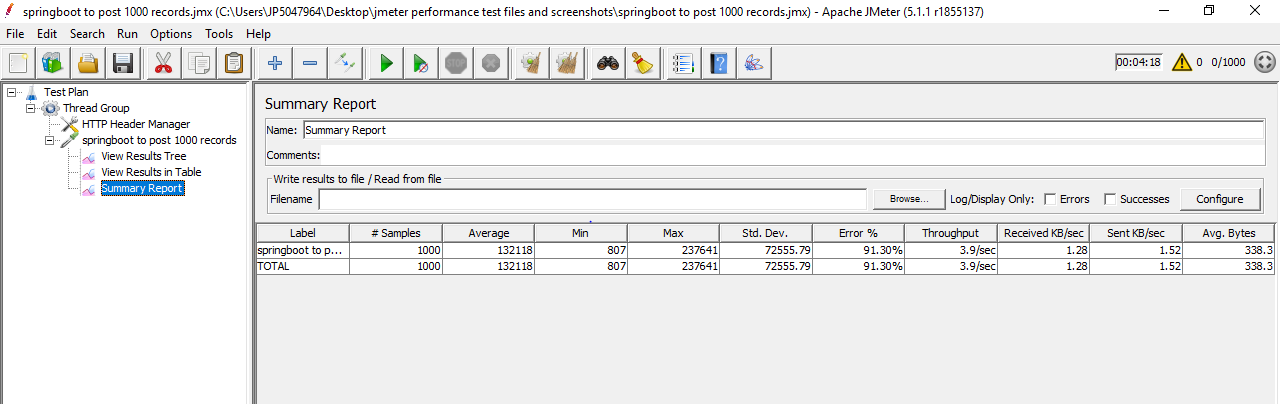
Records inserted in Postgre database:



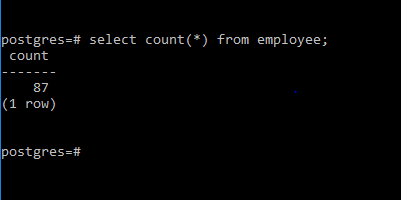
Springboot







Records inserted into Postgre database:



|  |  |  |
| --- | --- | --- |
|  | **Vert.x** | **Springboot** |
|  |  |  |
| HTTP Request Samples | 1000 | 1000 |
| Average | 37 | 132118 |
| Min | 1 | 807 |
| Max | 588 | 237641 |
| Standard Deviation | 105.34 | 72555.79 |
| Error % | 0.00% | 91.30% |
| Throughput | 98.9/sec | 3.9/sec |
| Test Execution time | 11sec | 4 min 18 sec |

This is the case in Springboot:

Out of 1000, only 87 records are inserted.

Some tests are failed and all records are not inserted due to an error: org.postgresql.util.PSQLException: FATAL: sorry, too many clients already

Conclusion:

Vert.x is performing better without any errors even under load increases.

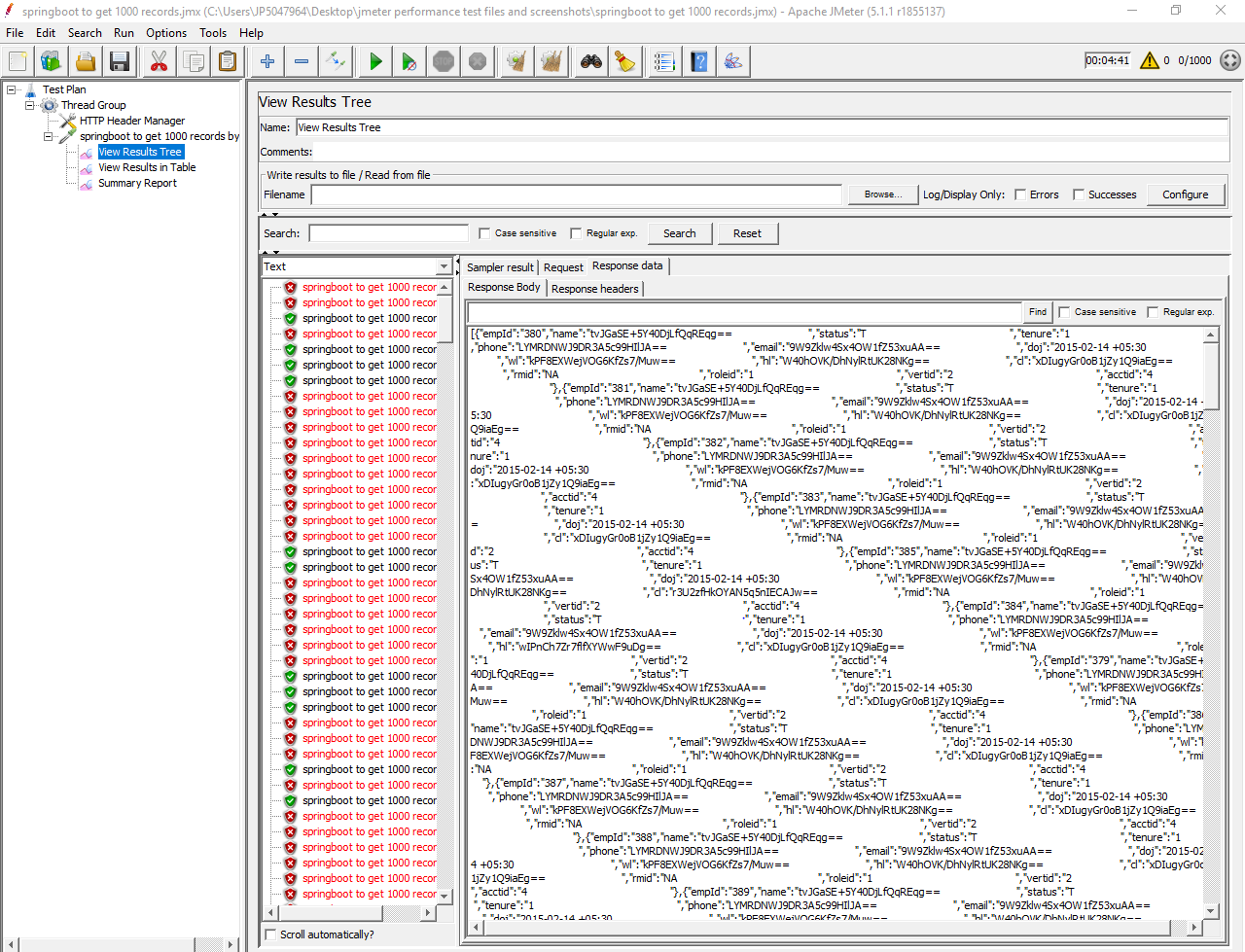
**Test6:** To get all records by 1000 users from Postgre database.

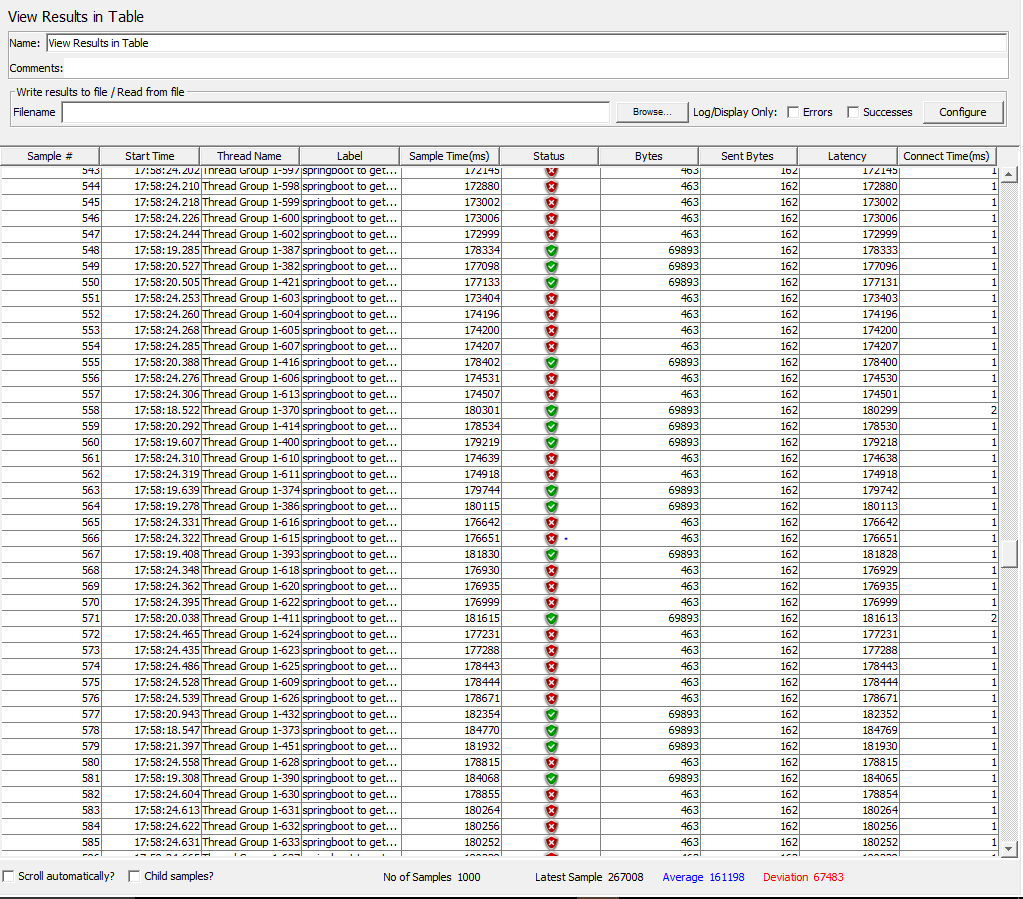
No. of threads:1000

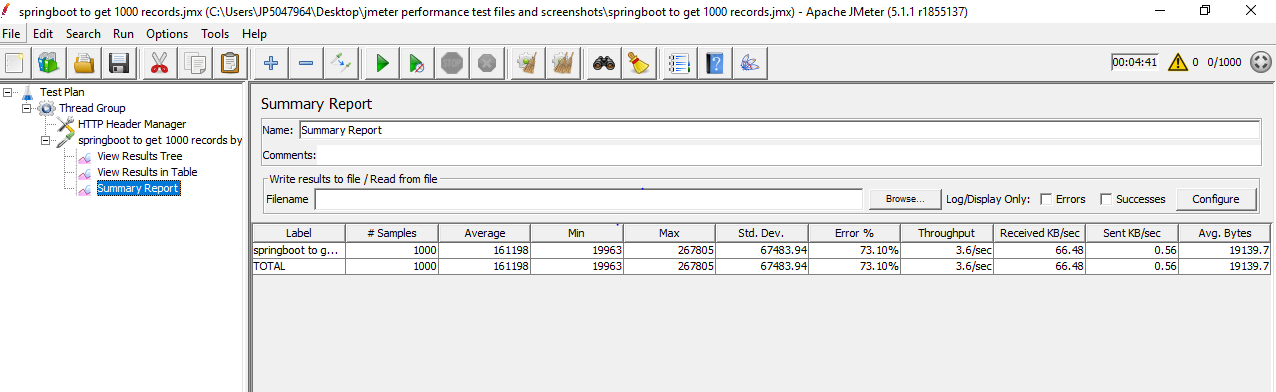
Ramp-up:10

Which means 100 users will come online to get all records from database in each sec

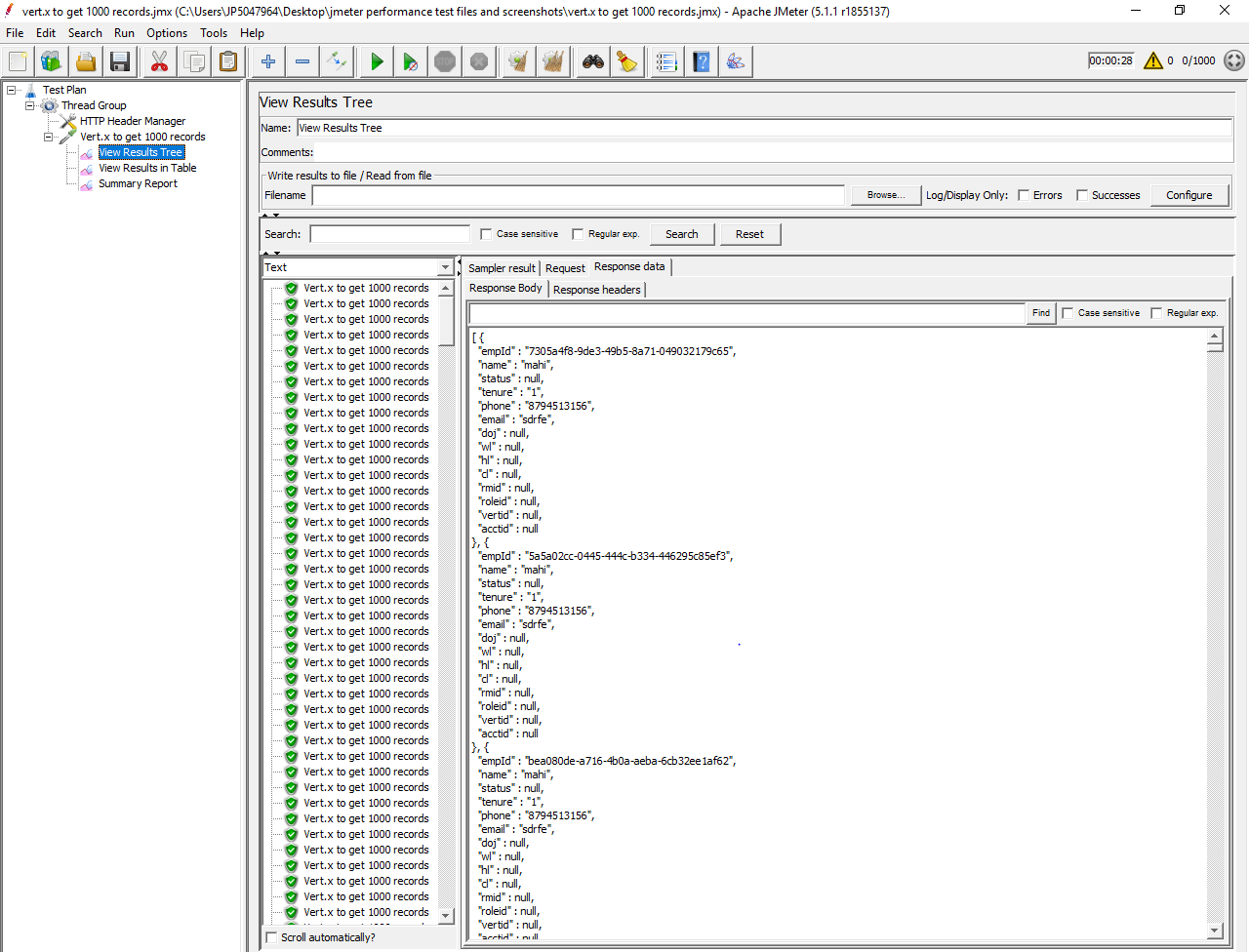
Springboot

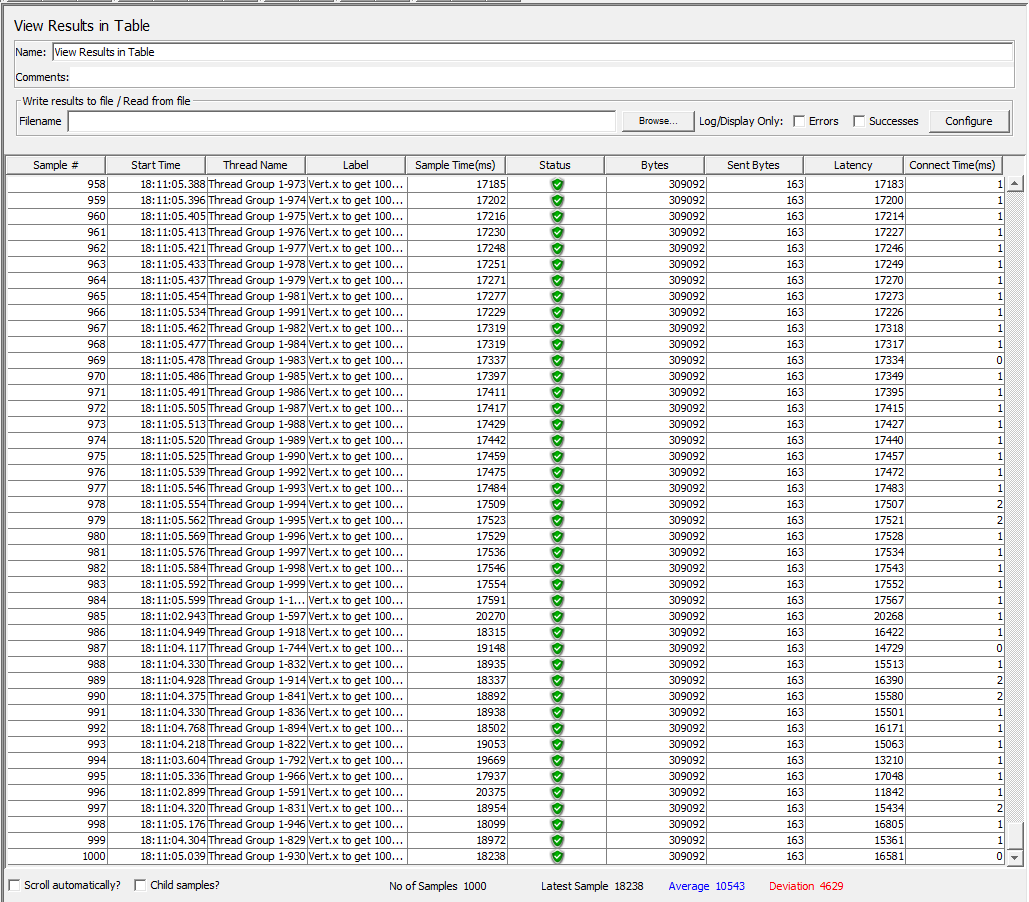


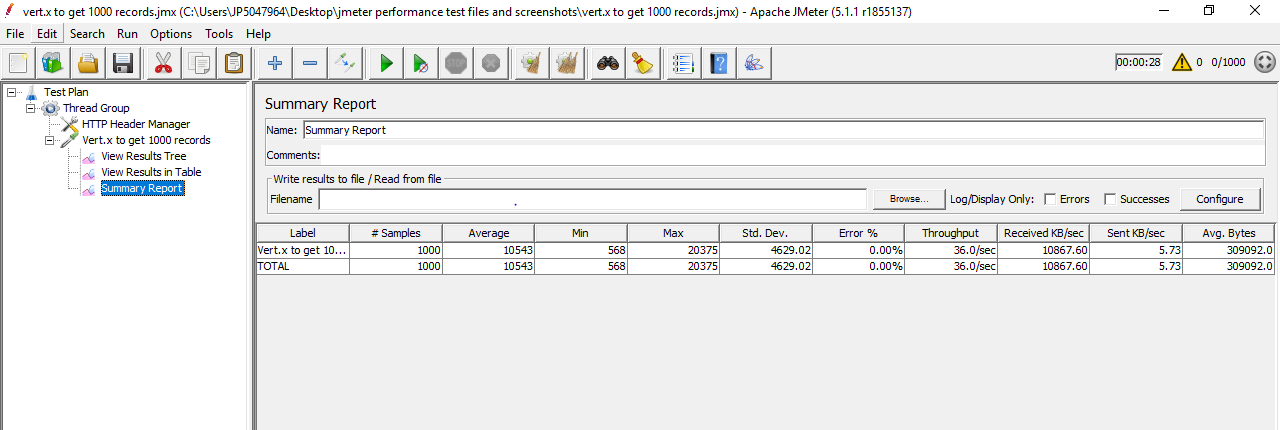




Vert.x







|  |  |  |
| --- | --- | --- |
|  | **vert.x** | **springboot** |
|  |  |  |
| HTTP Request Samples | 1000 | 1000 |
| Average | 10543 | 161198 |
| Min | 568 | 19963 |
| Max | 20375 | 267805 |
| Standard Deviation | 4629.02 | 67483.94 |
| Error % | 0.00% | 73.10% |
| Throughput | 36.0/sec | 3.6/sec |
| Test Execution time | 28 sec | 4 min 41 sec |

This is in case of Springboot:

Since many records are not properly inserted into database .It cannot retrieve all records

Conclusion:

Vert.x is performing better than Springboot.

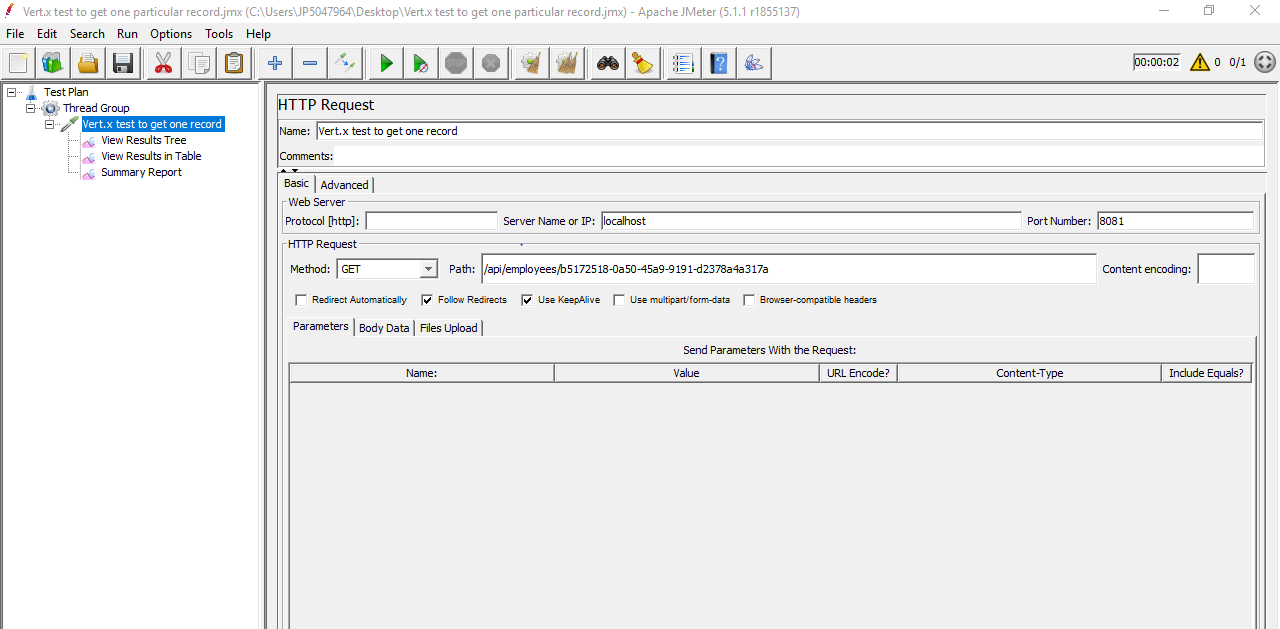
**Test 7:** To retrieve one particular record among all records in database

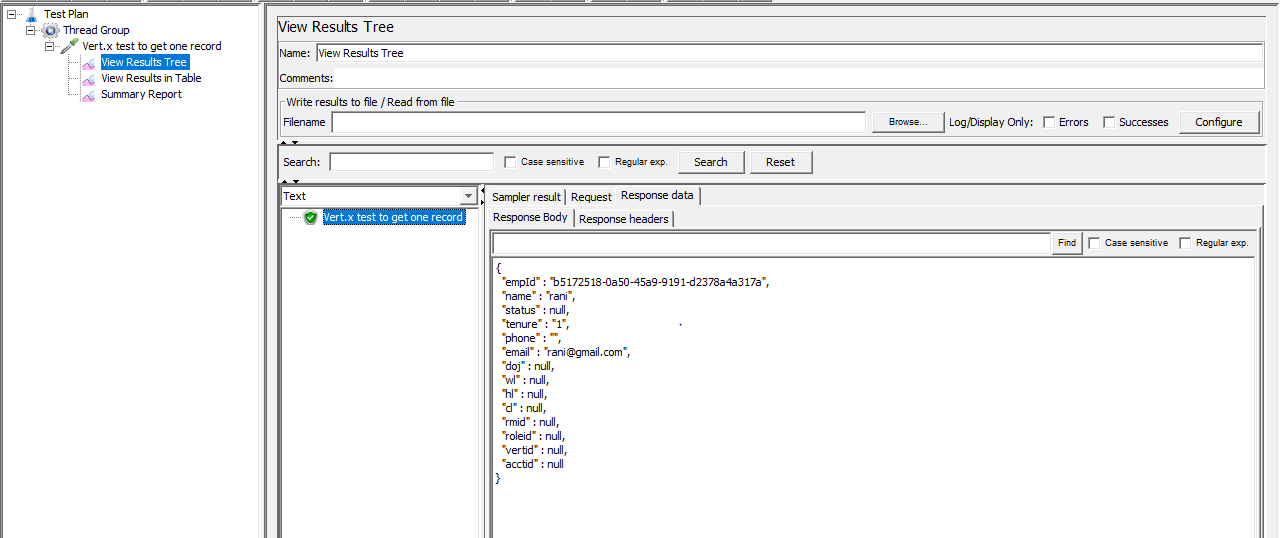
Configuration:

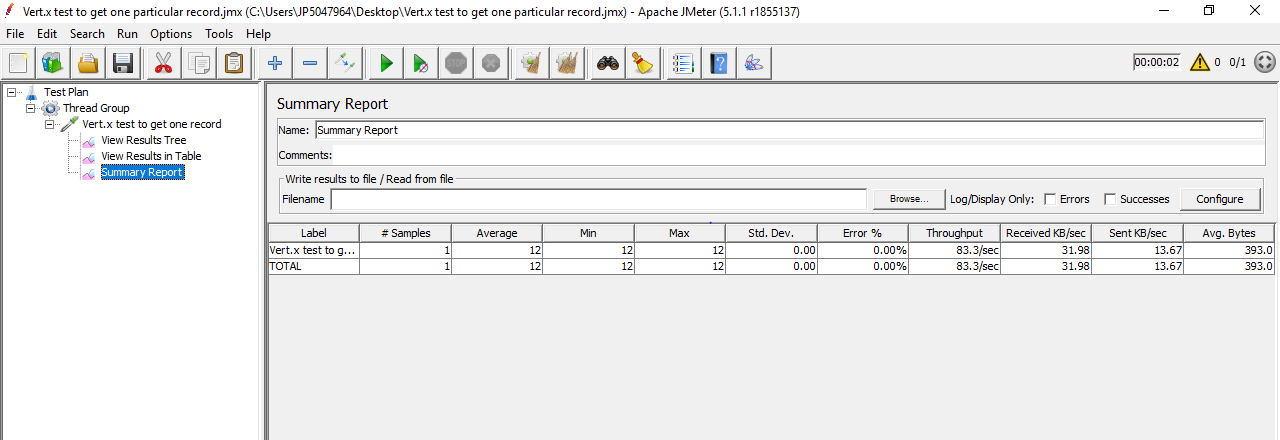
No. of threads:1

Ramp-up period:1

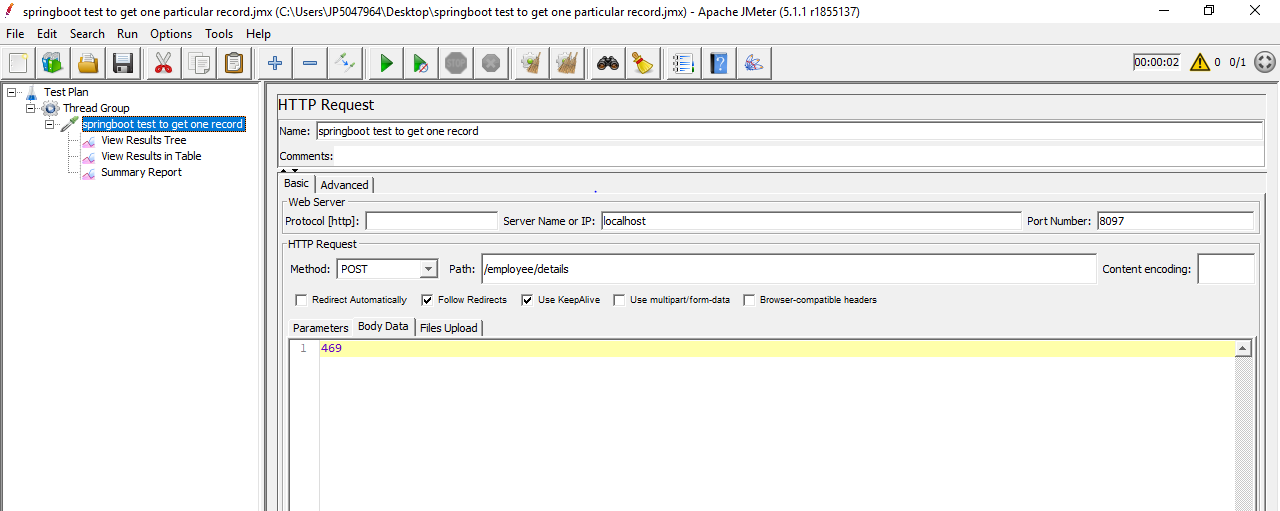
Vert.x :

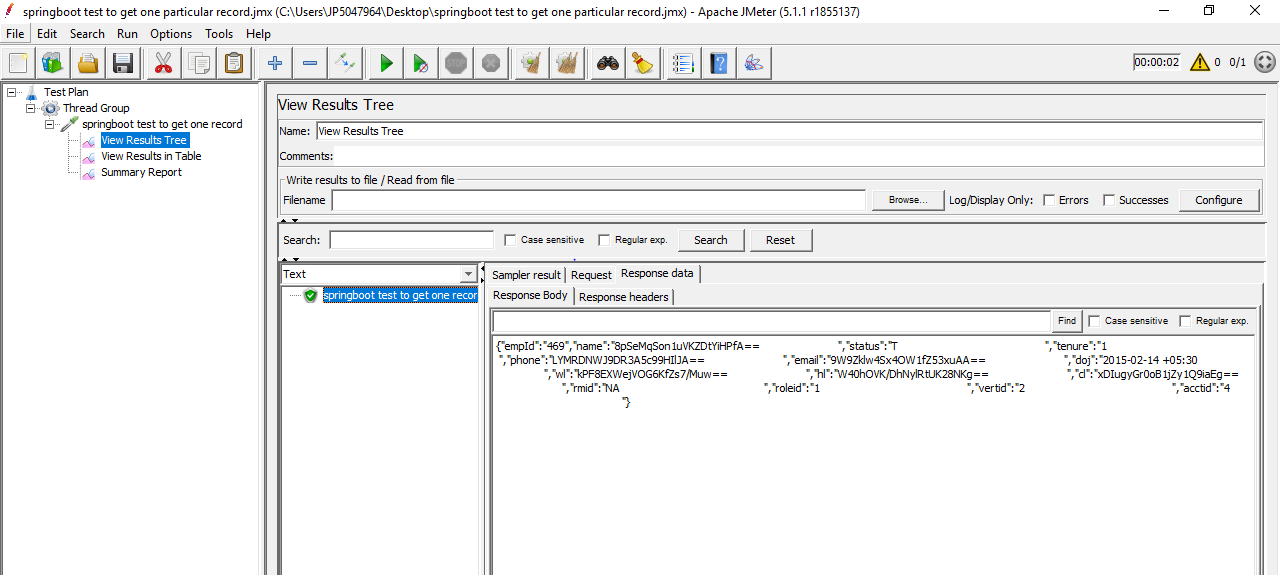


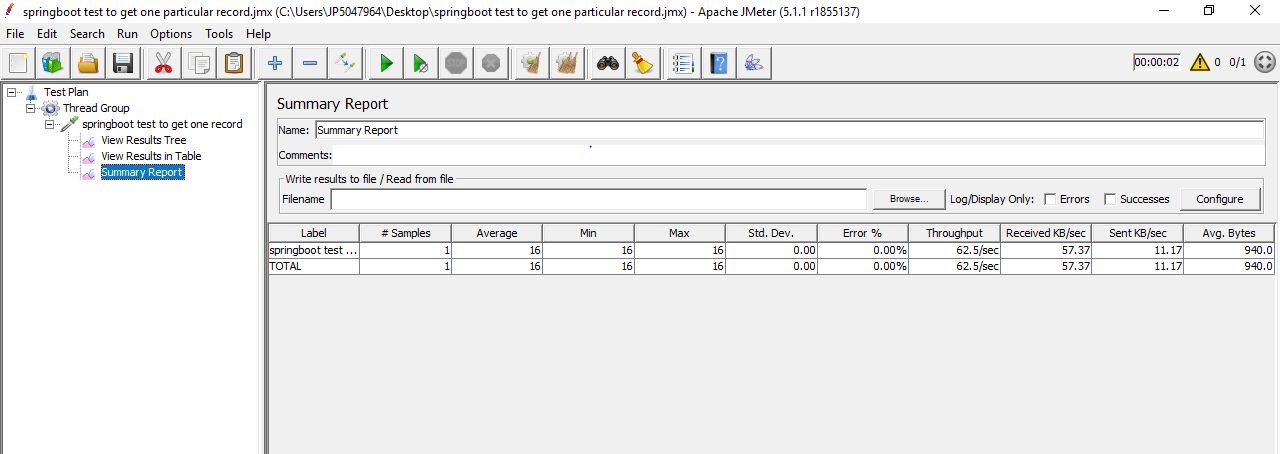




Springboot:







|  |  |  |
| --- | --- | --- |
|  | **vert.x** | **springboot** |
|  |  |  |
| HTTP Request Samples | 1 | 1 |
| Average | 12 | 16 |
| Min | 12 | 16 |
| Max | 12 | 16 |
| Standard Deviation | 0 | 0 |
| Error % | 0.00% | 0.00% |
| Throughput | 83.3/sec | 62.5/sec |
| Test Execution time | 2 sec | 2sec |

Conclusion:

Since the throughput is higher for vert.x it is performing better than springboot.

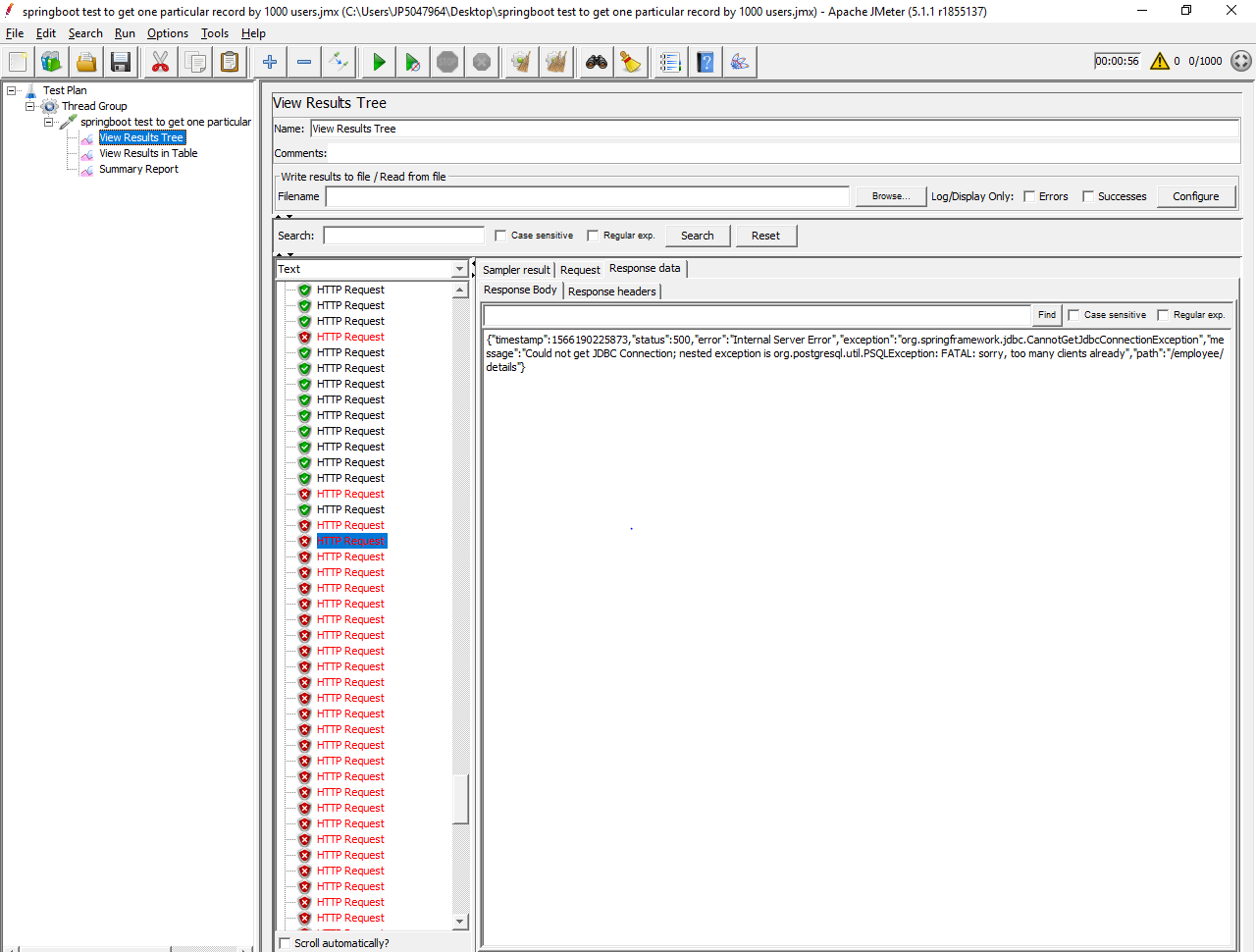
**Test 8:** To retrieve one particular record by 1000 users.

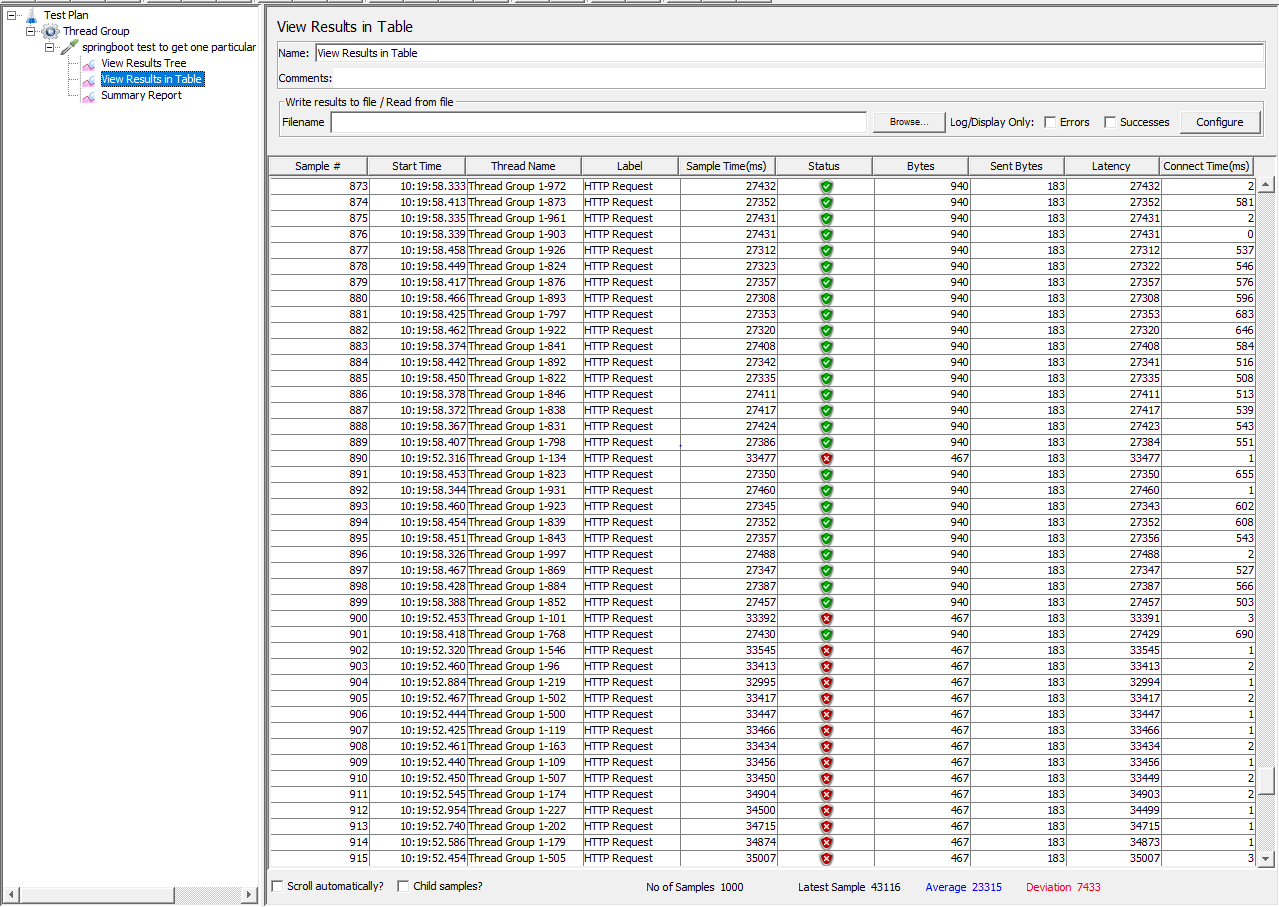
Configuration:

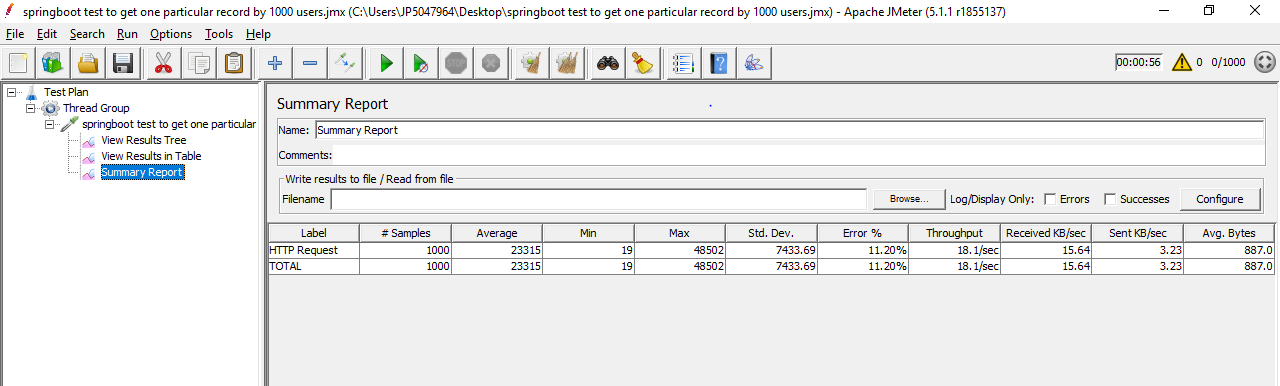
No. of threads:1000

Ramp-up period:10

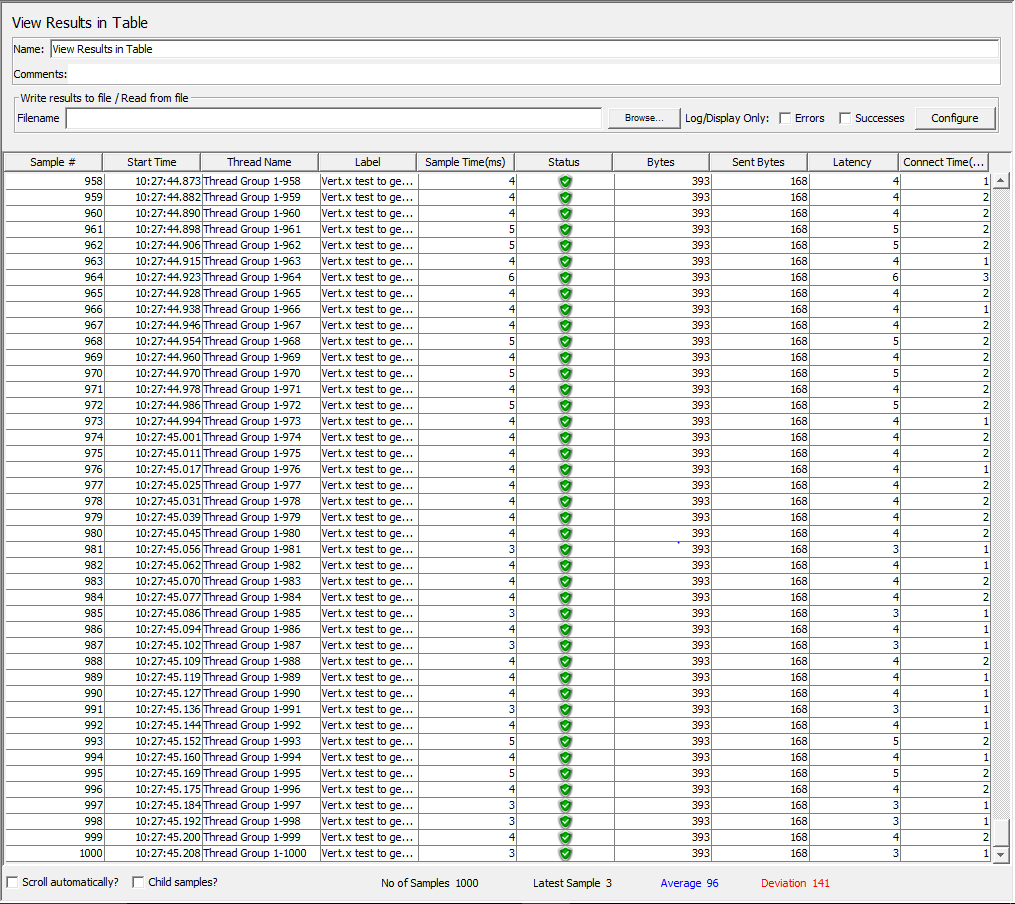
Springboot:

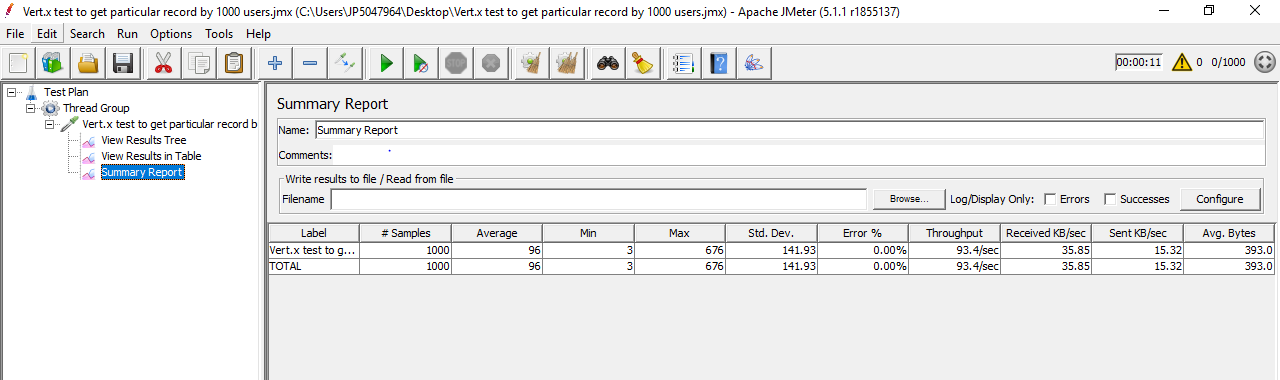






Vert.x





|  |  |  |
| --- | --- | --- |
|  | **vert.x** | **springboot** |
|  |  |  |
| HTTP Request Samples | 1000 | 1000 |
| Average | 96 | 23315 |
| Min | 3 | 19 |
| Max | 676 | 48502 |
| Standard Deviation | 141.93 | 7433.69 |
| Error % | 0.00% | 11.20% |
| Throughput | 93.4/sec | 18.0/sec |
| Test Execution time | 11sec | 56sec |

In case of Springboot, error occurred when it connected with many users. i.e: org.postgresql.util.PSQLException: FATAL: sorry, too many clients already

Conclusion:

Vert.x is better than Springboot.

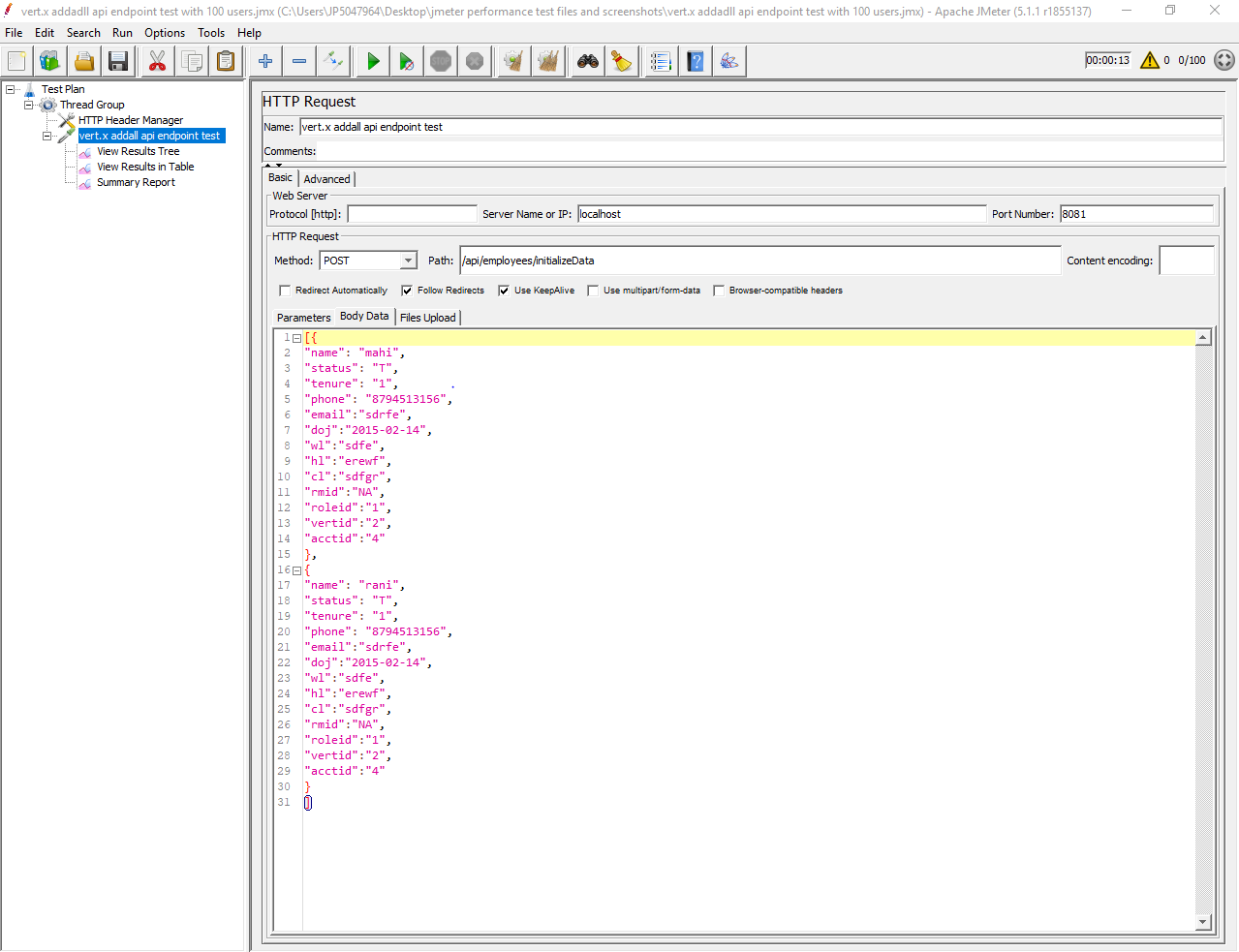
**Test 9:** To test addall api endpoint with 100 users.

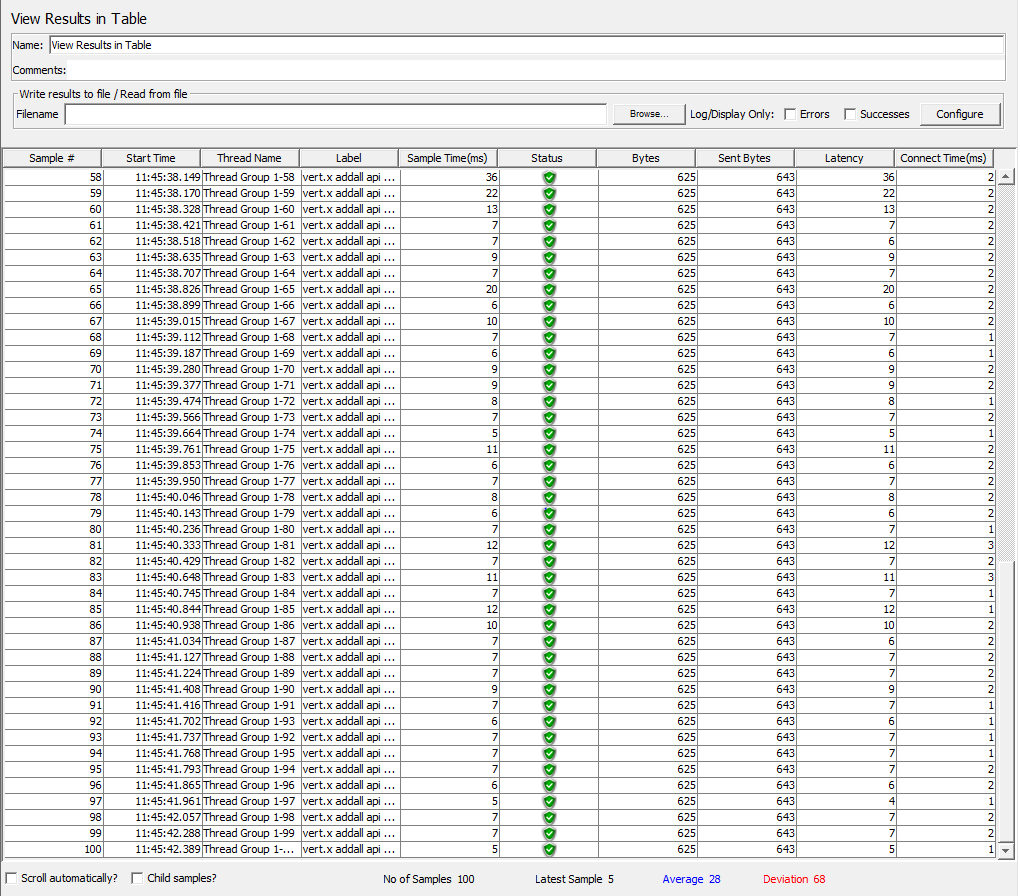
Configuration:

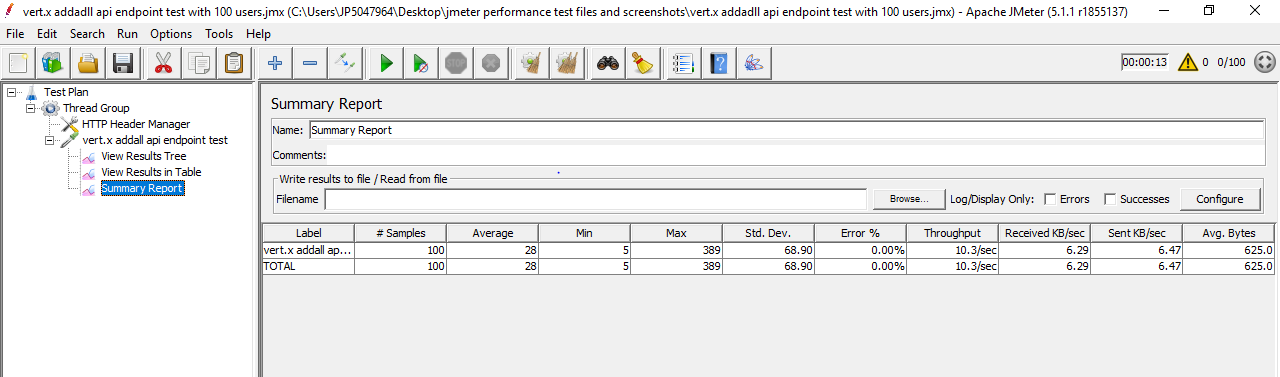
No. of threads:100

Ramp-up period:10

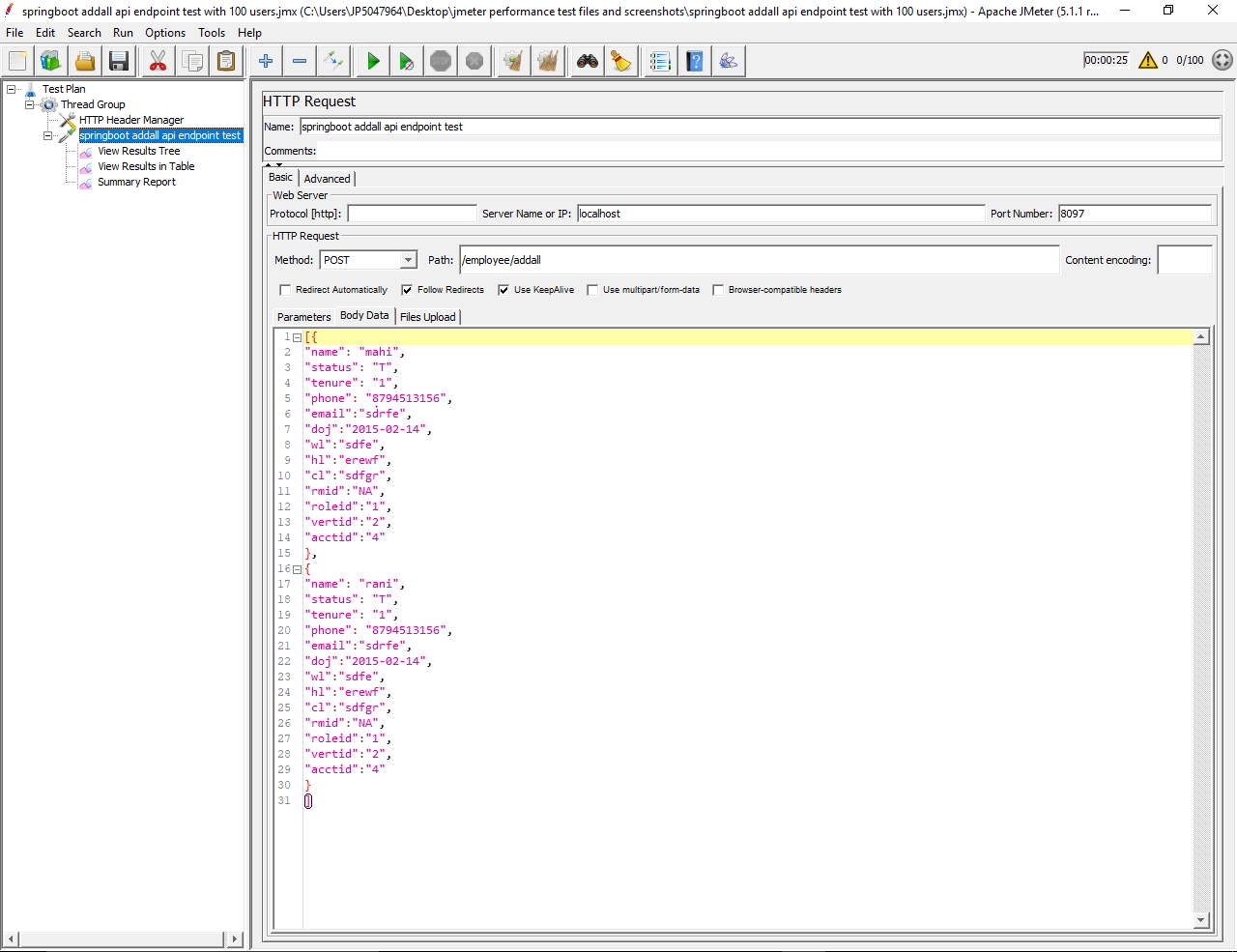
Vert.x result:

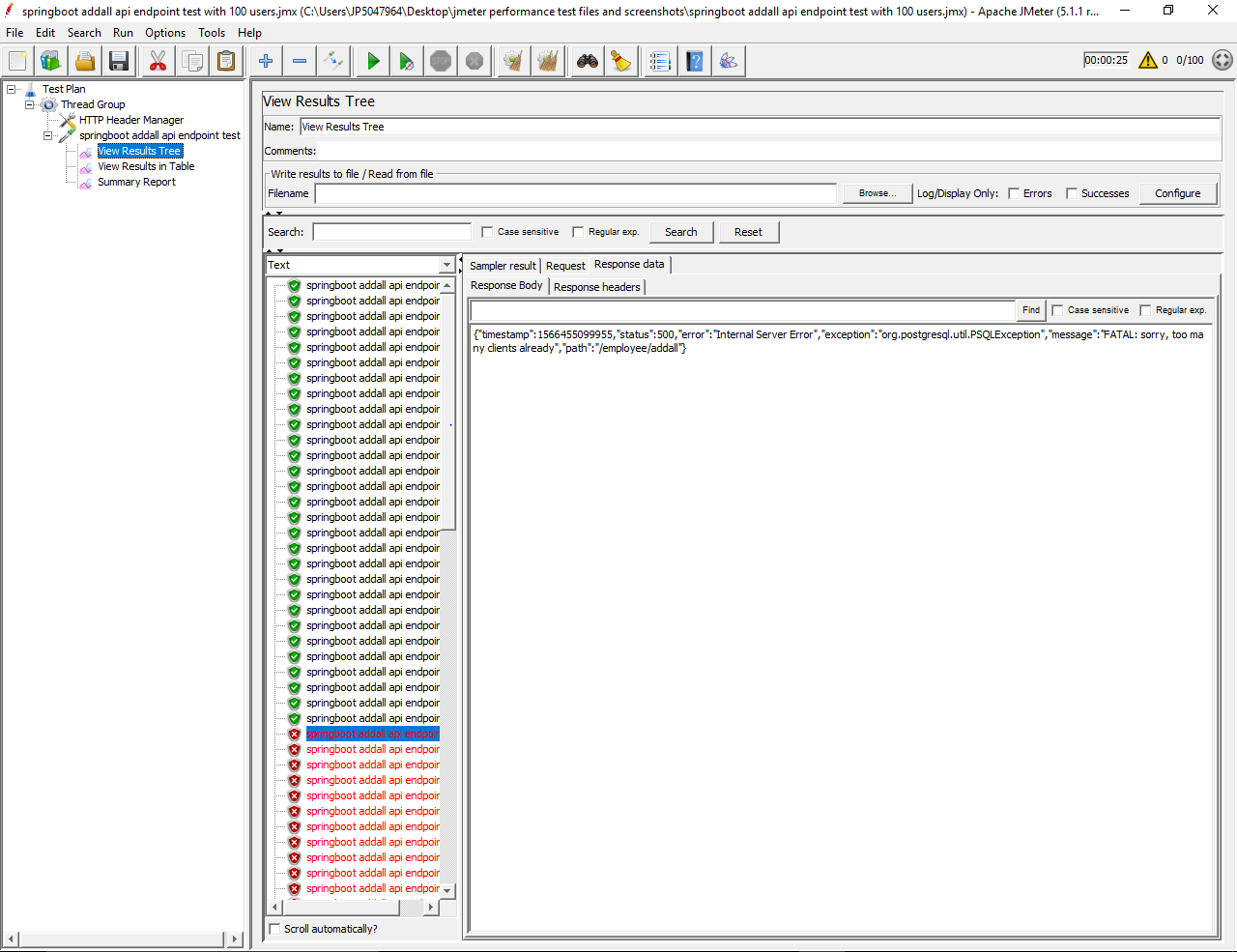


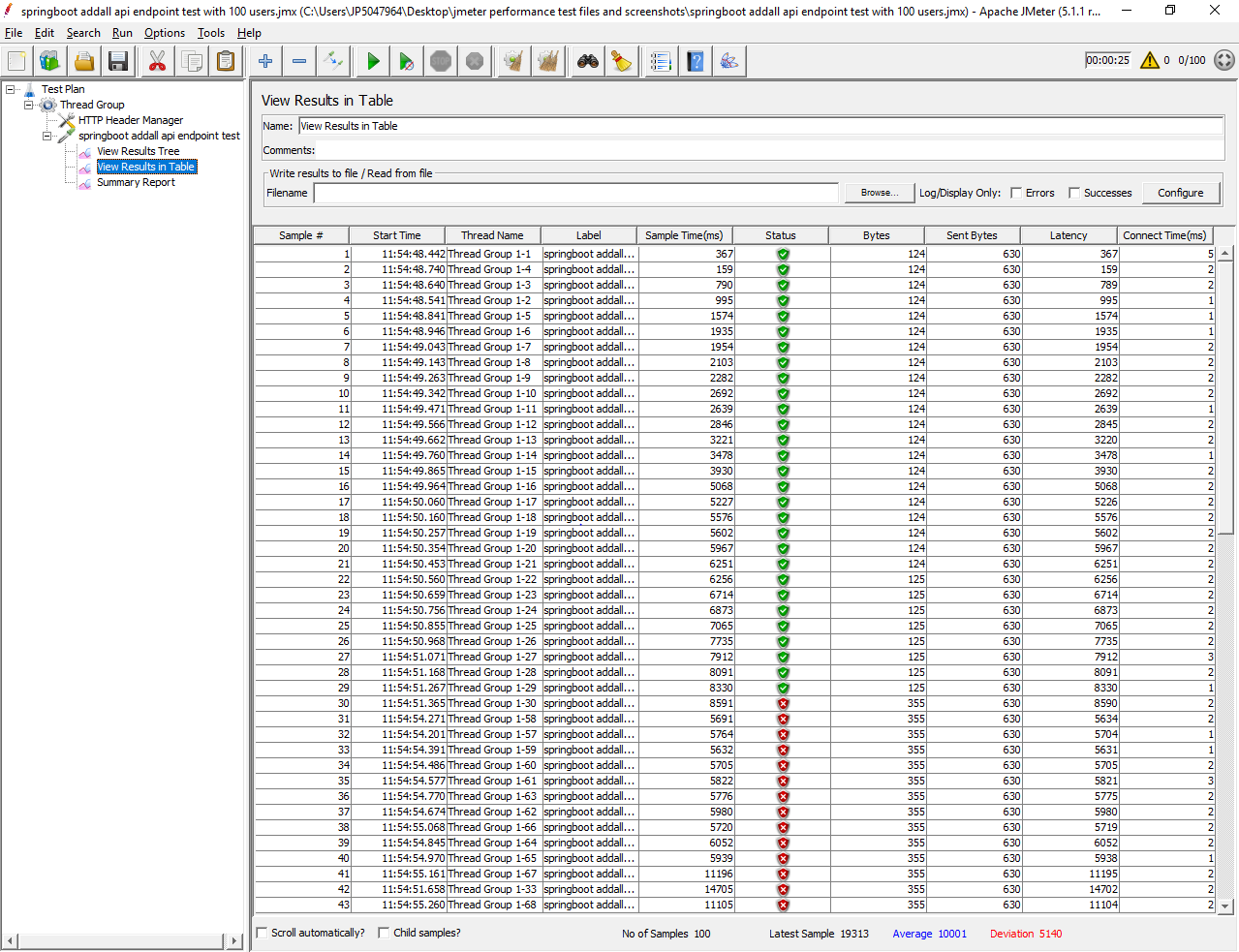


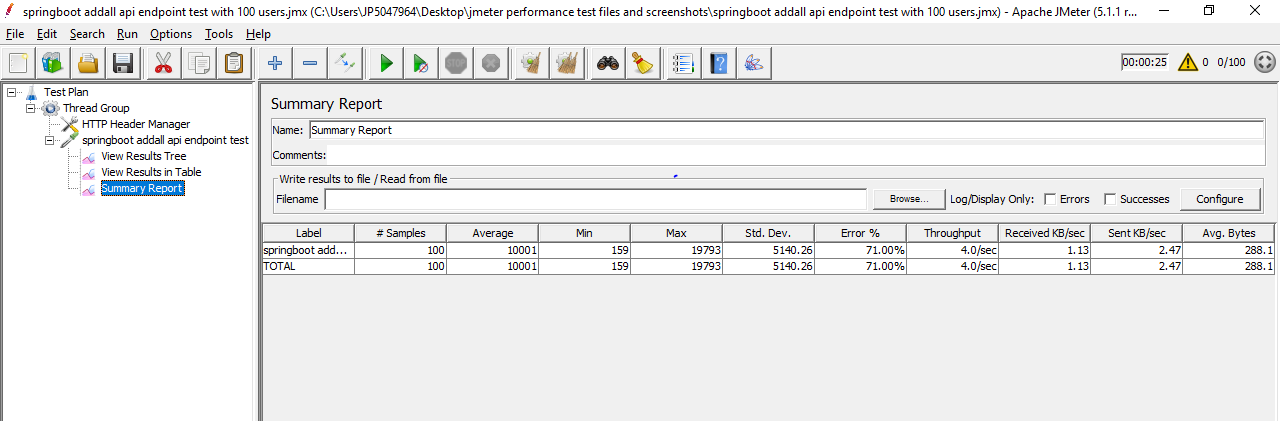


Springboot:









|  |  |  |
| --- | --- | --- |
|  | **vert.x** | **springboot** |
|  |  |  |
| HTTP Request Samples | 100 | 100 |
| Average | 28 | 10001 |
| Min | 5 | 159 |
| Max | 389 | 19793 |
| Standard Deviation | 68.9 | 5140.26 |
| Error % | 0.00% | 71.00% |
| Throughput | 10.3/sec | 4.0/sec |
| Test Execution time | 13sec | 25sec |

In case of Springboot, error occurred when it connected with many users to post the data into database. i.e: org.postgresql.util.PSQLException: FATAL: sorry, too many clients already

Conclusion:

Vert.x is performed better than Springboot

**Test 10:** To test update api endpoint.

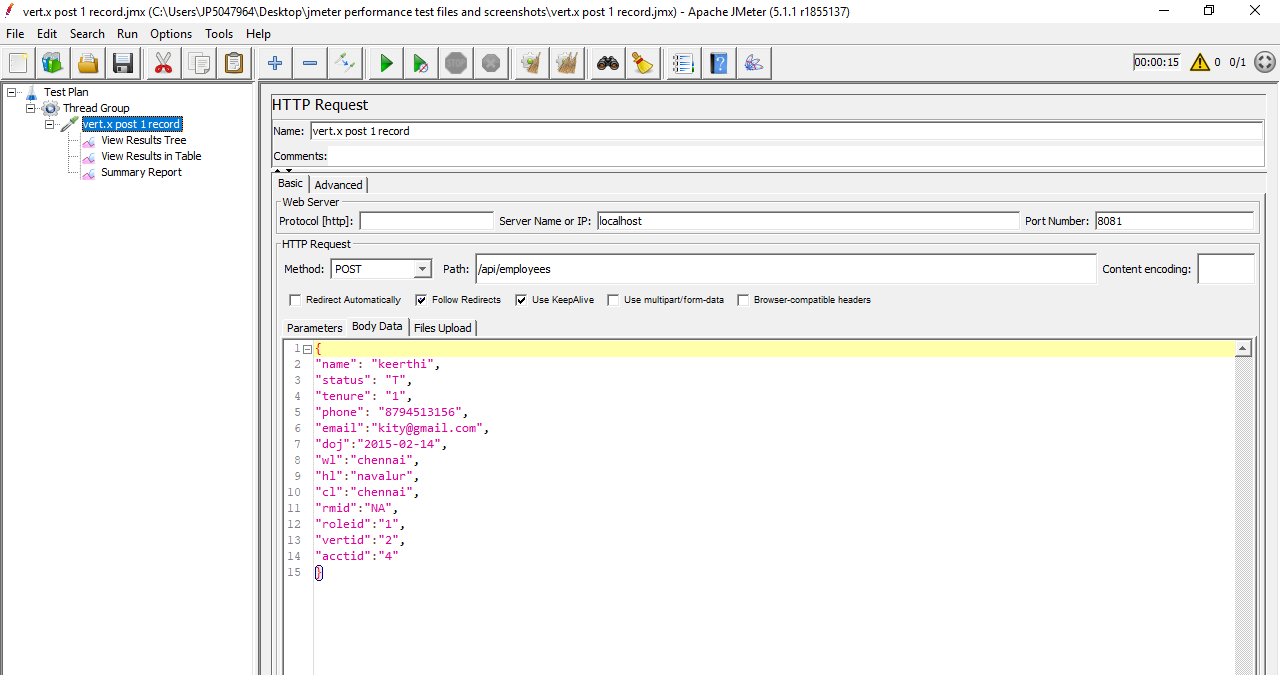
Configuration:

No. of threads:1

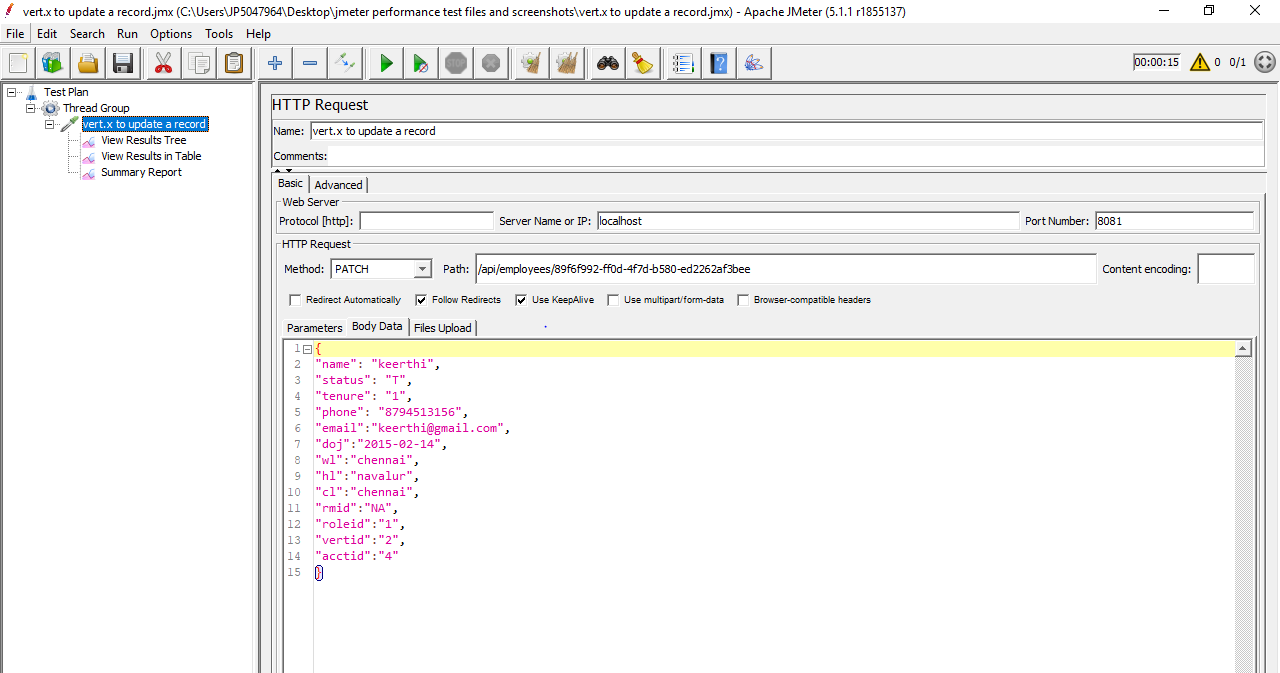
Ramp-up period:1

Vert.x

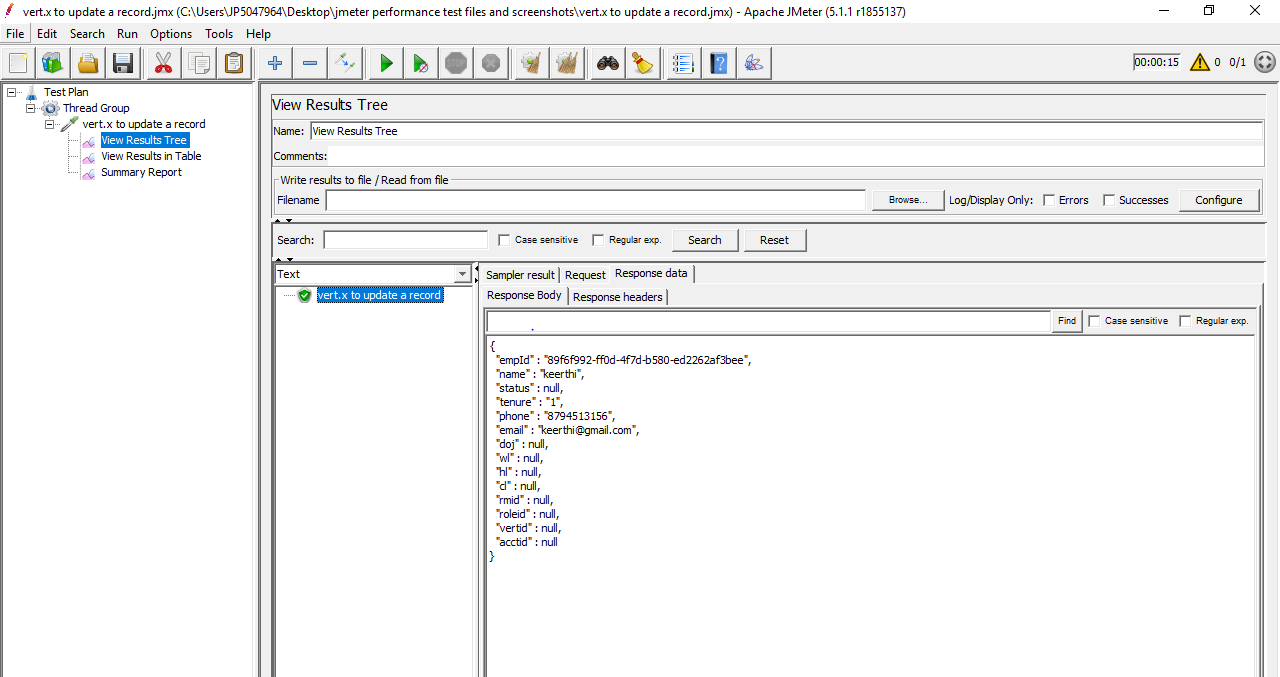
Below are the details inserted before updating (Email:kity@gmail.com)

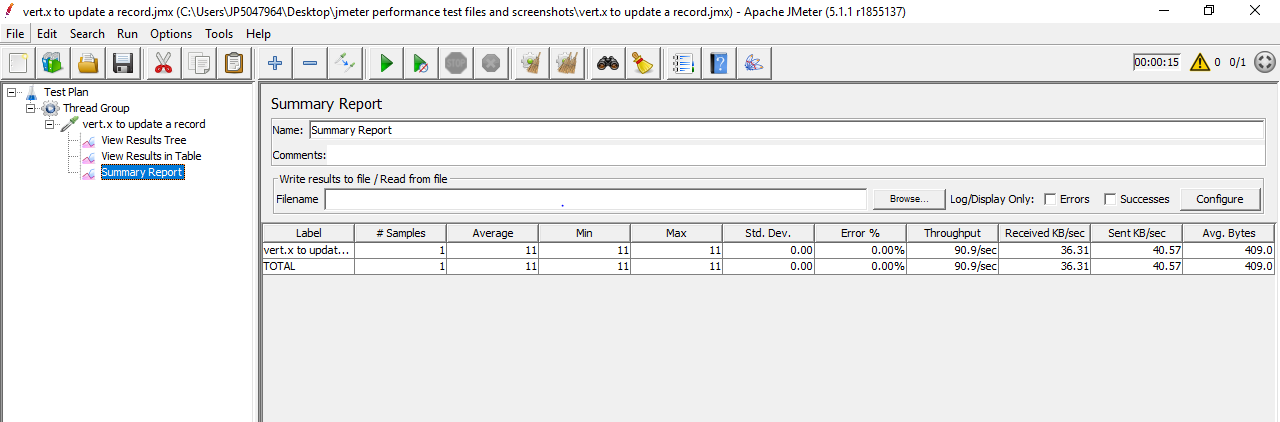


Below is the data sending to update the record using patch method based on id



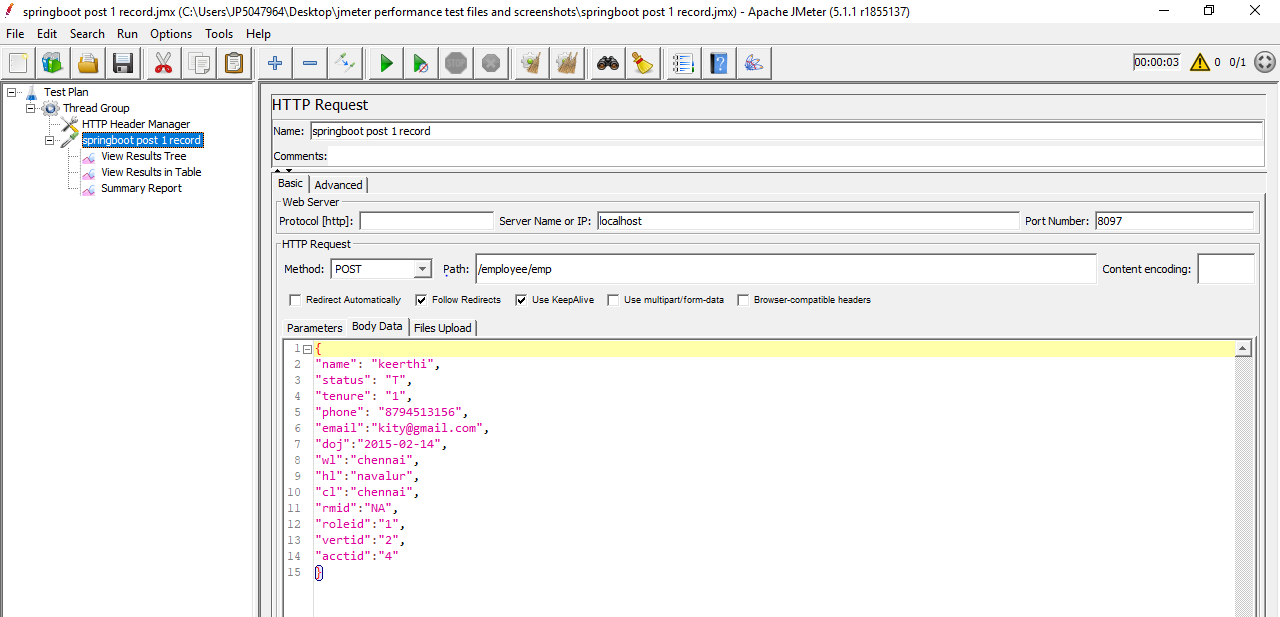
Below are the details after updating value for inserted record based on id.

( Email: [keerthi@gmail.com](mailto:keerthi@gmail.com))

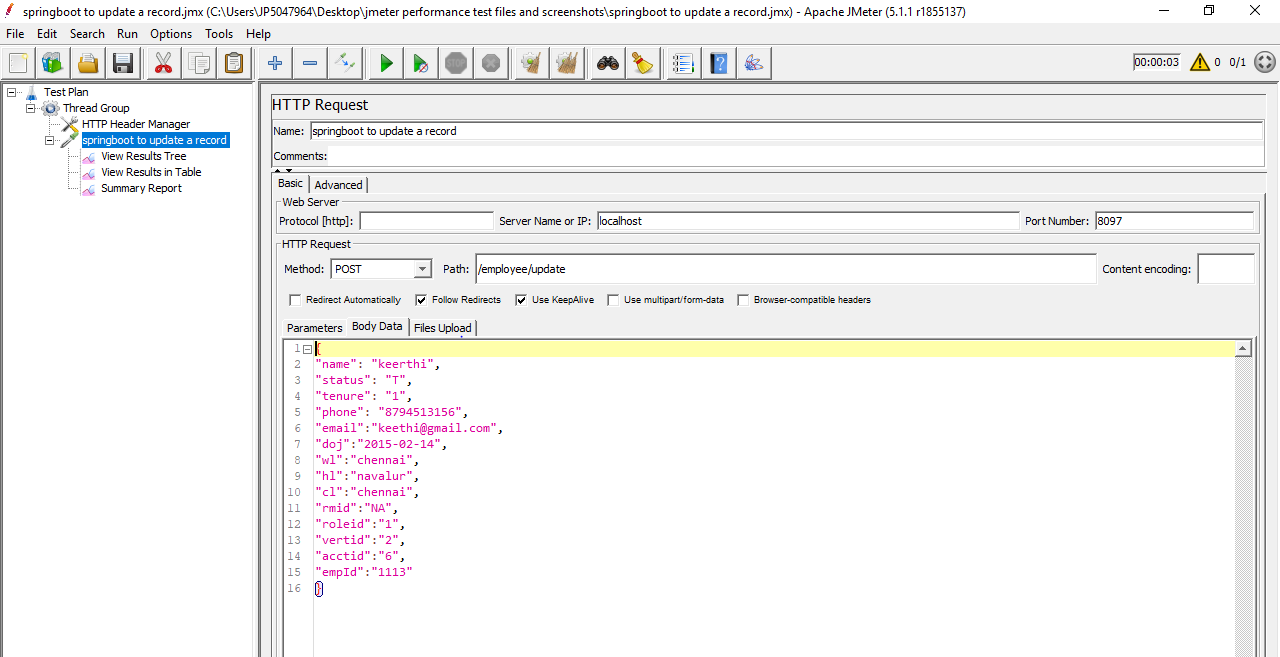


Springboot:

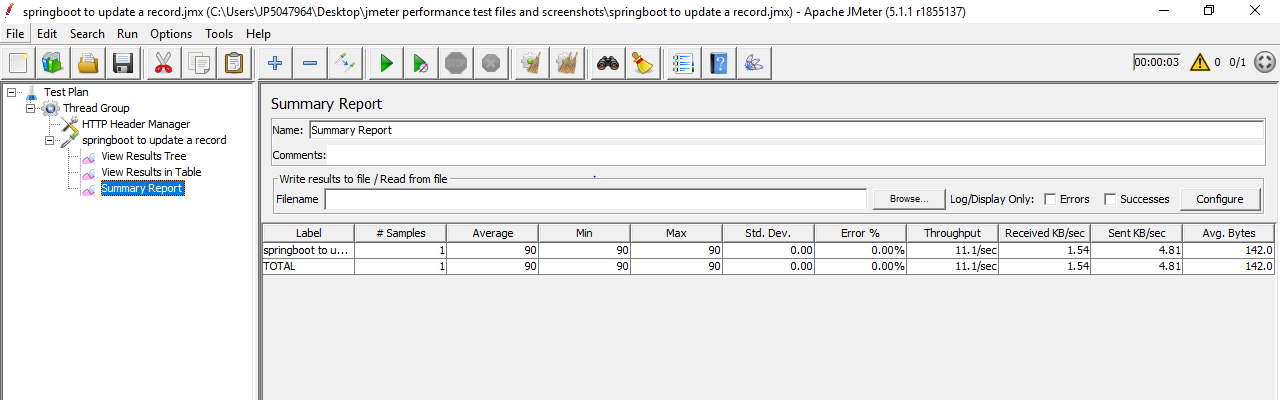
Below are the details inserted before updating (Email:kity@gmail.com)



Below is the data sending to update the record using patch method based on id



Below are the details after updating value for inserted record based on id.

( Email: [keerthi@gmail.com)](mailto:keerthi@gmail.com))

Conclusion:

Since throughput is higher , Vert.x is better than Springboot.

**Final Conclusion:**

After performing test, the conclusion is as follows:

Vert.x is performing better than Springboot in all cases.

Vert.x handles more number of requests faster than Springboot.

Vert.x supports concurrency and multithreading i.e: concurrent number of users/threads processing requests at a time.