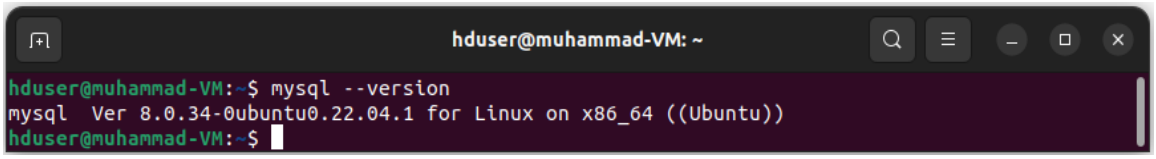
Tutorial 9 YCSB (MongoDB and MySQL)

Database Benchmarking

For benchmarking, we will use two databases (MySQL and MongoDB) in this tutorial and use a benchmarking tool (YCSB) to check their performance.

1) You can check installed version of MySQL by using the following command as

```
$mysql --version
```

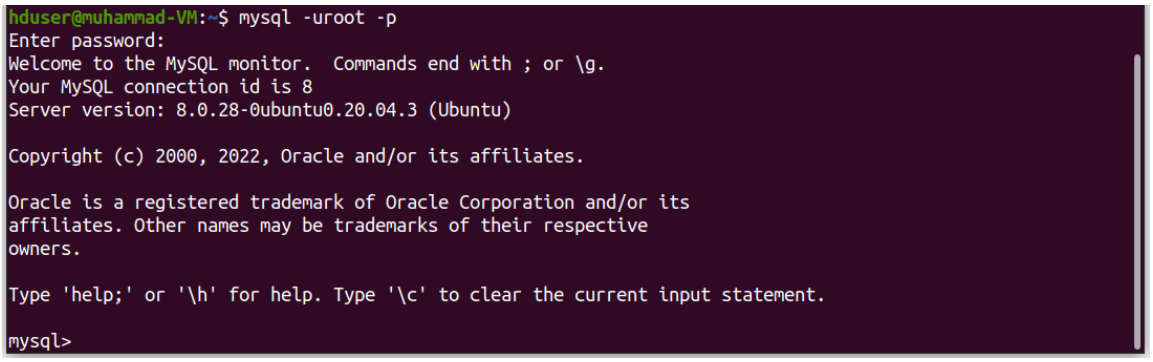


If you could not find MySQL on your VM, then complete Tutorial 5 (part I) before step 2 starting. Otherwise, you can start from step 2.

2) Use password as **password** as we set in Tutorial 5 (part I).

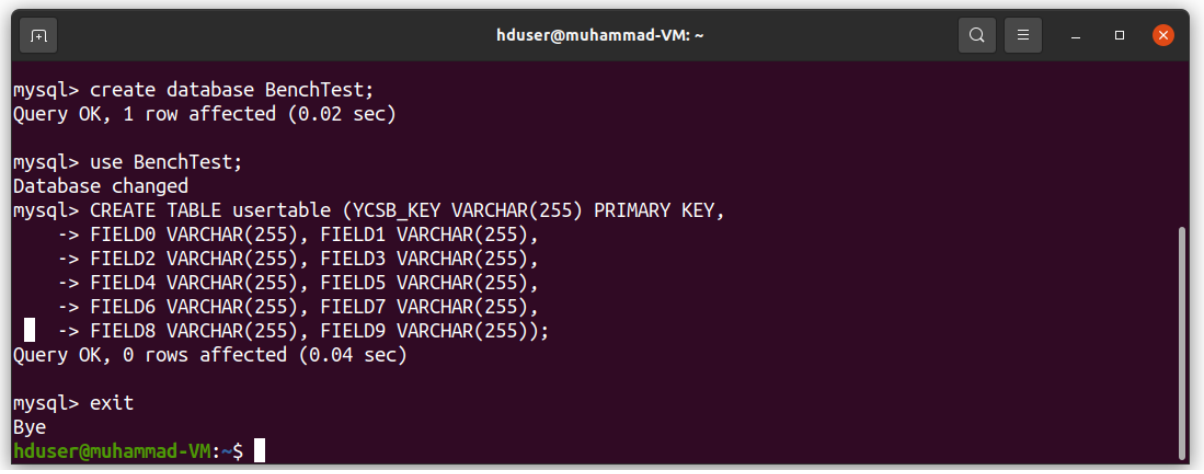
```
$mysql -uroot -p
```

3) Create a database **"BenchTest"** and **"usertable"** table in MySQL. Create the database named as "BenchTest". Then create the table named as **"usertable"**.



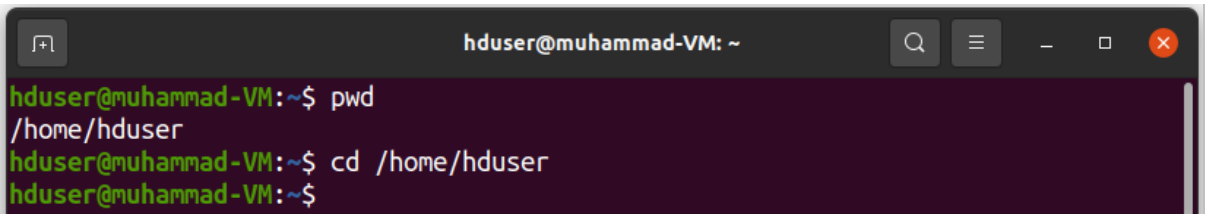
At the `mysql>` command prompt, enter the following command as mentioned as well as in the screen shot below

```
mysql>create database BenchTest;
mysql>use BenchTest;
mysql>CREATE TABLE usertable (YCSB_KEY VARCHAR(255) PRIMARY KEY,
              FIELD0 VARCHAR(255), FIELD1 VARCHAR(255),
              FIELD2 VARCHAR(255), FIELD3 VARCHAR(255),
              FIELD4 VARCHAR(255), FIELD5 VARCHAR(255),
              FIELD6 VARCHAR(255), FIELD7 VARCHAR(255),
              FIELD8 VARCHAR(255), FIELD9 VARCHAR(255));
```

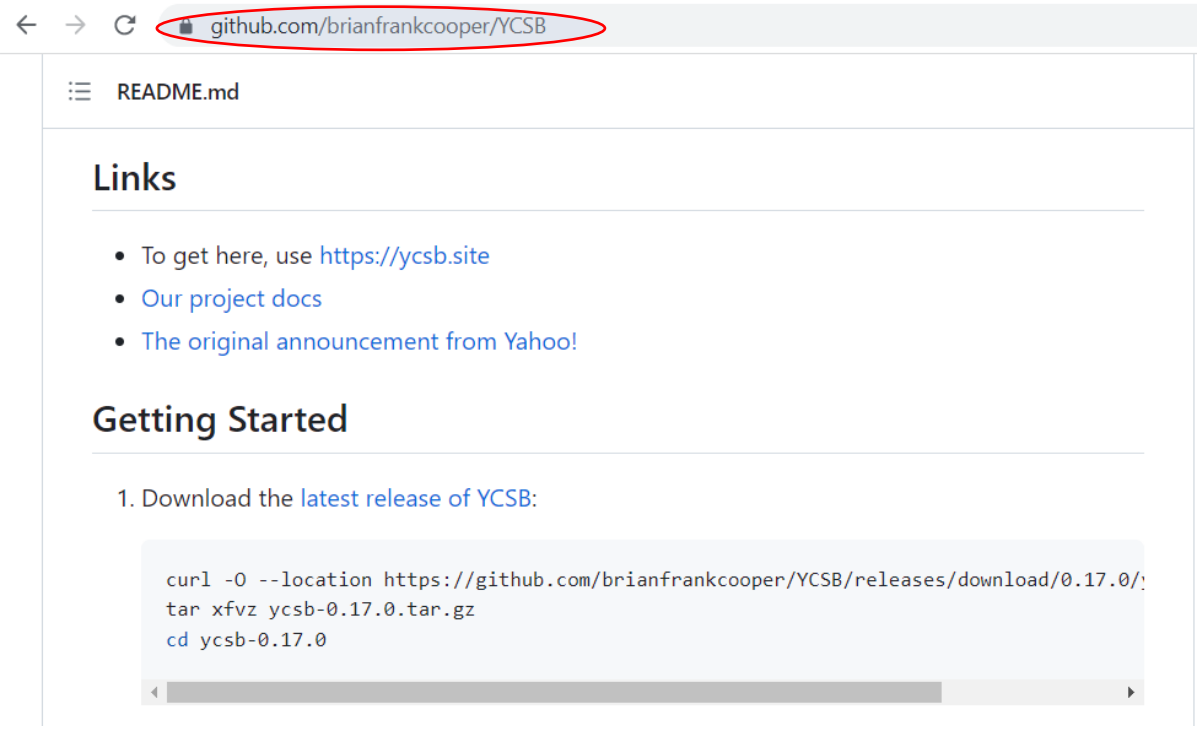


4) Download and install YCSB as mentioned below using the commands on Ubuntu shell as shown.

```
$pwd
$cd /home/hduser
```



Download the ycsb-0.17.0.tar.gz from the below mentioned link by using the commands on the Ubuntu terminal. Scroll down the webpage and you will find the command to download the YCSB.



Unzip the folder using **tar** command on the Ubuntu terminal and write the following commands in sequence. (Install curl command if OS showed the error).

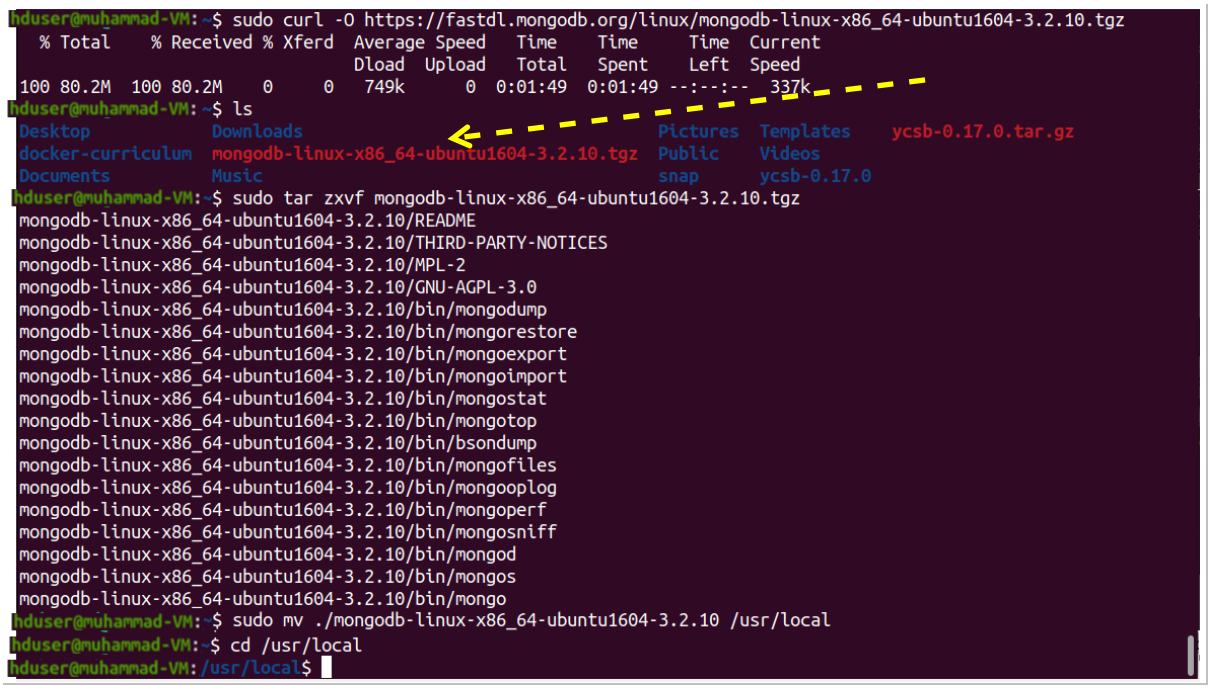
```
$curl -O --location https://github.com/brianfrankcooper/YCSB/releases/download/0.17.0/ycsb-0.17.0.tar.gz
$tar xfvz ycsb-0.17.0.tar.gz
$cd ycsb-0.17.0
```

```
hduser@muhammad-VM:~$ curl -O --location https://github.com/brianfrankcooper/YCSB/releases/download/0.17.0
/ycsb-0.17.0.tar.gz
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
  0     0    0     0    0     0      0      0 --:--:-- --:--:-- --:--:--     0
100  675M  100  675M    0     0  2116k      0  0:05:26  0:05:26 --:--:-- 1398k
hduser@muhammad-VM:~$ tar xfvz ycsb-0.17.0.tar.gz
ycsb-0.17.0/NOTICE.txt
ycsb-0.17.0/LICENSE.txt
ycsb-0.17.0/bin/ycsb.sh
ycsb-0.17.0/bin/ycsb.bat
ycsb-0.17.0/bin/ycsb
ycsb-0.17.0/bin/bindings.properties
ycsb-0.17.0/workloads/
ycsb-0.17.0/workloads/workloade
ycsb-0.17.0/workloads/workloadb
ycsb-0.17.0/workloads/workloadc
```

Now we will use YCSB later to check the performance of SQL and NoSQL databases.

5) Open a new Ubuntu terminal and download MongoDB by using the following commands.

```
$sudo wget https://fastdl.mongodb.org/linux/mongodb-linux-x86_64-ubuntu1604-3.2.10.tgz
$sudo tar zxvf mongodb-linux-x86_64-ubuntu1604-3.2.10.tgz
$sudo mv ./mongodb-linux-x86_64-ubuntu1604-3.2.10 /usr/local
```




```
$cd /usr/local
$sudo chown -R hduser:hadoopgroup  mongodb-linux-x86_64-ubuntu1604-3.2.10
$sudo ln -s  ./mongodb-linux-x86_64-ubuntu1604-3.2.10  ./mongodb
```

```
hduser@muhammad-VM:/usr/local$ sudo chown -R hduser:hadoopgroup mongodb-linux-x86_64-ubuntu1604-3.
2.10
hduser@muhammad-VM:/usr/local$ sudo ln -s mongodb-linux-x86_64-ubuntu1604-3.2.10 mongodb
hduser@muhammad-VM:/usr/local$ ls -l
lrwxrwxrwx  1 root   root              38 Apr 13 00:37 mongodb -> mongodb-linux-x86_64-ubuntu1604
-3.2.10
drwxr-xr-x  3 hduser hadoopgroup     4096 Apr 13 00:35 mongodb-linux-x86_64-ubuntu1604-3.2.10
```

Change the privileges for mongodb folder to hduser

$sudo chown -R hduser:hadoopgroup ./mongodb*

```
hduser@muhammad-VM:/usr/local$ sudo chown -R hduser:hadoopgroup mongodb*
```

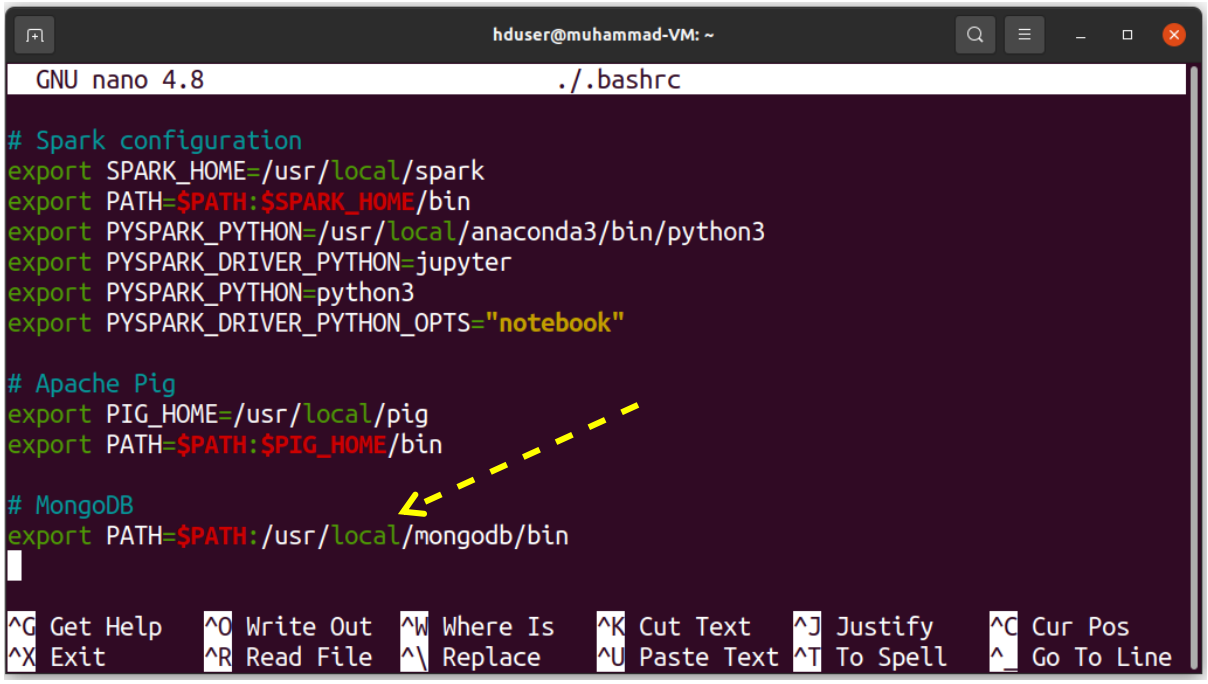
```
$cd ..
$sudo mkdir mongodbdata
$sudo chown -R hduser:hadoopgroup mongodbdata
$cd /
$sudo ln -s /usr/mongodbdata data
$cd data
$mkdir -p db
```

```
hduser@muhammad-VM:/usr/local$ cd ..
hduser@muhammad-VM:/usr$ sudo mkdir mongodbdata
hduser@muhammad-VM:/usr$ sudo chown -R hduser:hadoopgroup mongodbdata
hduser@muhammad-VM:/usr$ ls -l
total 136
drwxr-xr-x   2 root   root        49152 Apr 12 23:19 bin
drwxr-xr-x   2 root   root         4096 Aug 19  2021 games
drwxr-xr-x  41 root   root        16384 Apr  2 21:56 include
drwxr-xr-x 124 root   root         4096 Feb 26 01:00 lib
drwxr-xr-x   2 root   root         4096 Aug 19  2021 lib32
drwxr-xr-x   2 root   root         4096 Mar  1 21:17 lib64
drwxr-xr-x  12 root   root         4096 Jan 27 21:50 libexec
drwxr-xr-x   2 root   root         4096 Aug 19  2021 libx32
drwxr-xr-x  18 root   root         4096 Apr 13 00:37 local
drwxr-xr-x   2 hduser hadoopgroup  4096 Apr 13 00:37 mongodbdata
drwxr-xr-x   2 root   root        20480 Apr 12 23:19 sbin
drwxr-xr-x 264 root   root        12288 Apr  3 14:06 share
drwxr-xr-x   6 root   root         4096 Apr  3 14:01 src
hduser@muhammad-VM:/usr$ cd /
hduser@muhammad-VM:/$ sudo ln -s /usr/mongodbdata data
hduser@muhammad-VM:/$ ls
bin   cdrom  dev  home  lib32  libx32      media  opt   root  sbin  srv       sys  usr
boot  data   etc  lib   lib64  lost+found  mnt    proc  run   snap  swapfile  tmp  var
hduser@muhammad-VM:/$ cd data
hduser@muhammad-VM:/data$ mkdir -p db
hduser@muhammad-VM:/data$ cd
hduser@muhammad-VM:~$
```

6) Edit the **./.bashrc** file to update environment variables

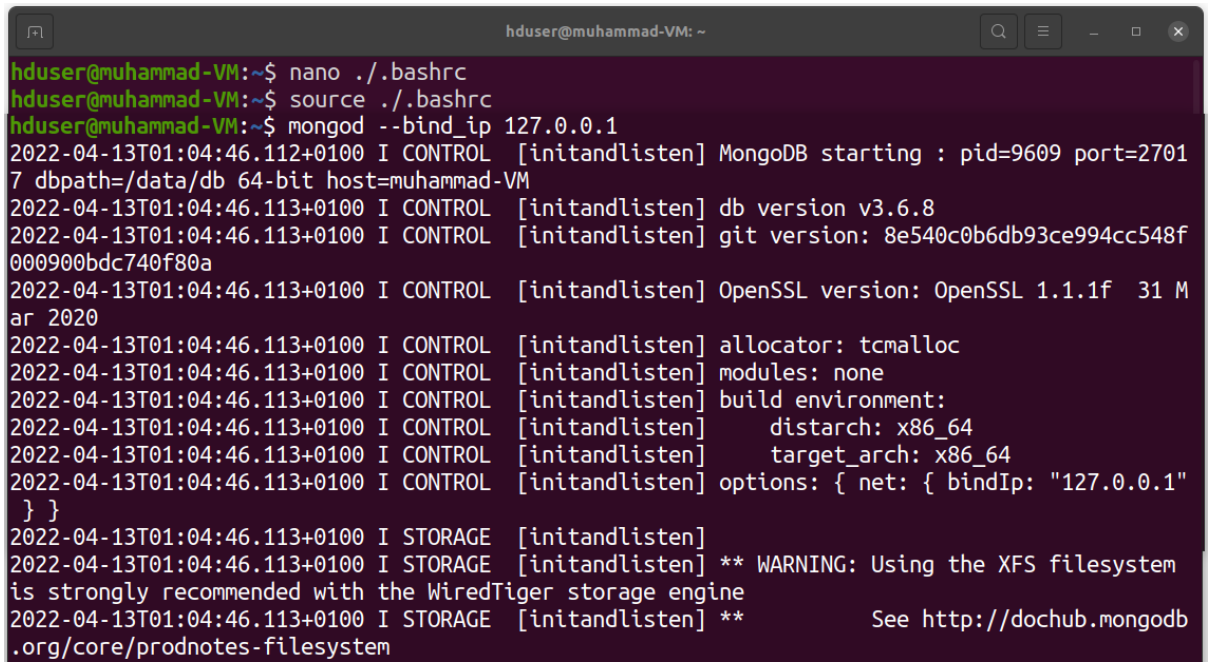
```
hduser@muhammad-VM:~$ pwd
/home/hduser
hduser@muhammad-VM:~$ cd /home/hduser
hduser@muhammad-VM:~$ nano ./.bashrc
```

Update this file and add the line for the path of mongodb at the end of the file as mentioned below



Save this file using nano editor (Ctrl + x, type y and hit the Enter key). If you want to allow the OS to see the updates, then use the following command.

```
$source ./.bashrc
```



Allow this session to run the **mongod** process. The mongoDB database is required in this mode on the terminal to execute the commands. Open a new terminal for the remaining steps of this tutorial. In case of any errors follow the step No. 7.

7) If you observe some kind error after the execution of `$mongod --bind_ip 127.0.0.1` command, then the following two commands should be executed in Ubuntu 22.04.* versions. Otherwise you can ignore the step no. 7.

(i)

```
$wget http://snapshot.debian.org/archive/debian/20190501T215844Z/pool/main/g/glibc/multiarch-support_2.28-10_amd64.deb
```

Type the above command and check the screenshot for further understanding. Then execute the below command

```
$sudo dpkg -i multiarch-support*.deb
```

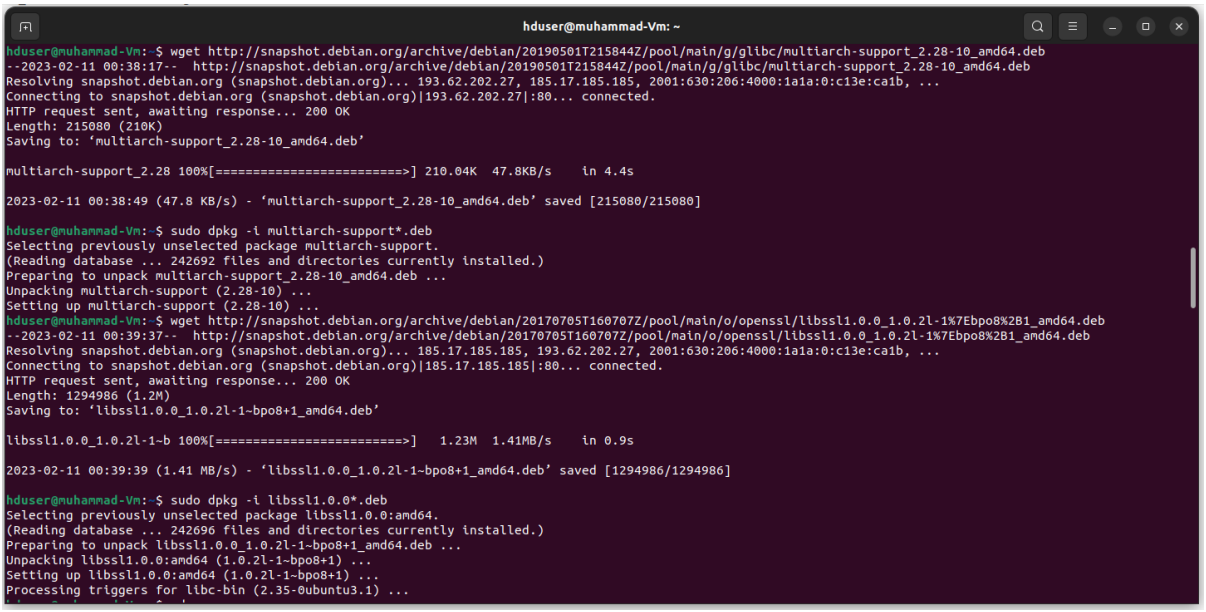
(ii)

```
$wget http://snapshot.debian.org/archive/debian/20170705T160707Z/pool/main/o/openssl/libssl1.0.0_1.0.2l-1%7Ebpo8%2B1_amd64.deb
```

Then execute the below command

```
$sudo dpkg -i libssl1.0.0*.deb
```

You can check the execution of the commands in the screenshot as mentioned below



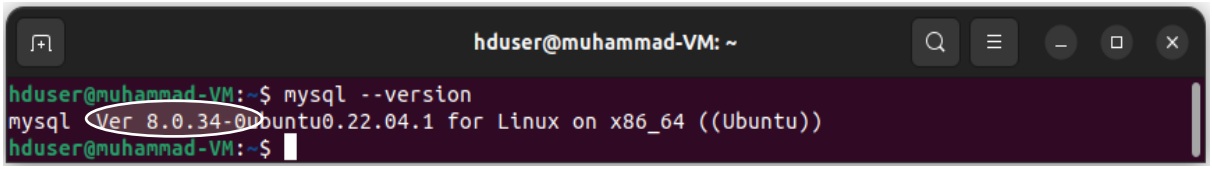
Restart your VM after the installation of the above commands. Execute the command as mentioned below

```
$mongod --bind_ip 127.0.0.1
```

Allow this session to run the **mongod** process. The mongoDB database is required in this mode on the terminal to execute the commands.

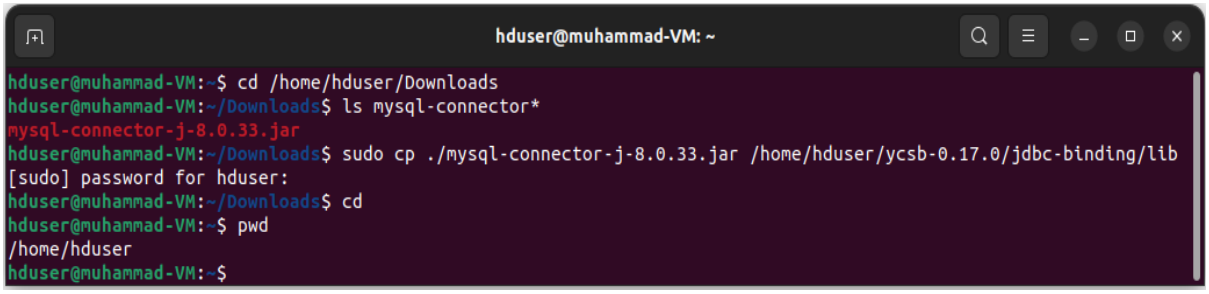
8) **Use YCSB with mySQL and mongoDB**

(i) Open up a new terminal session and check the version of mySQL as we did in step 1.

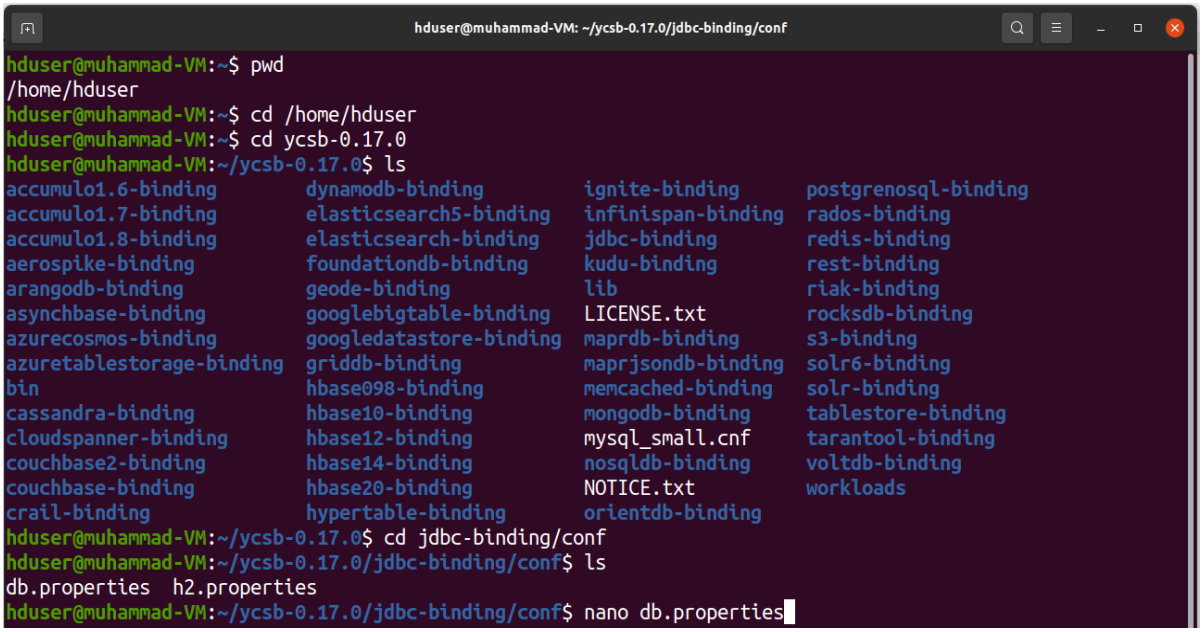


We are using 8.0.34 mySQL and if you are using some other version, then download the relevant connector from mySQL resources (https://dev.mysql.com/downloads/connector/j/?os=26).

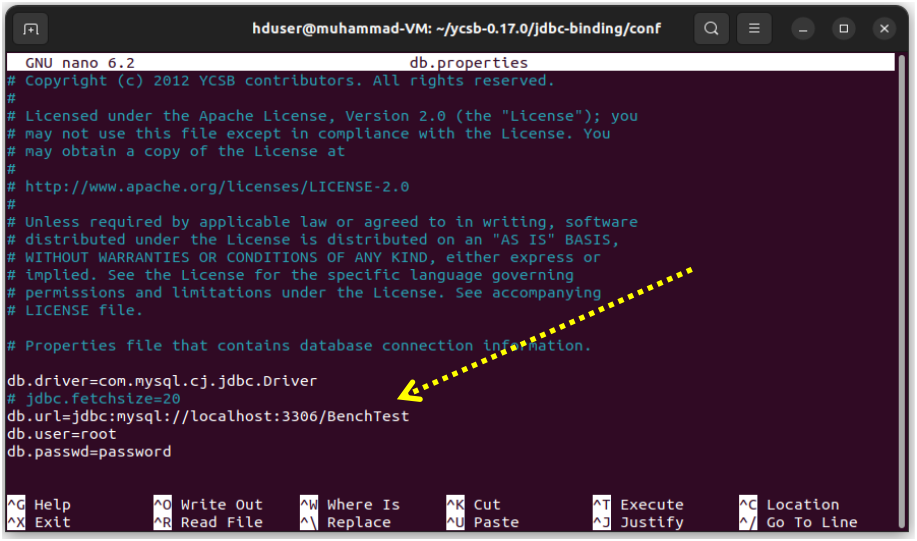
(ii) Download mysql connector from Moodle. Copy the **mysql-connector-java-8.0.33.jar** file into the **/home/hduser/ycsb-0.17.0/jdbc-binding/lib** environment and shared folder (Download connector file from Moodle) as mentioned below in the screenshot.



9) Edit the file named as **db.properties** in the **ycsb-0.17.0/jbc-binding/conf** directory as follows:



Update the contents of the file as mentioned below



Save this file using **nano** editor as you did in the previous tutorials and use the following command to move to ycsb-0.0.17.0 folder and moved to the folder ycsb-0.17.0 by using
$cd ../..

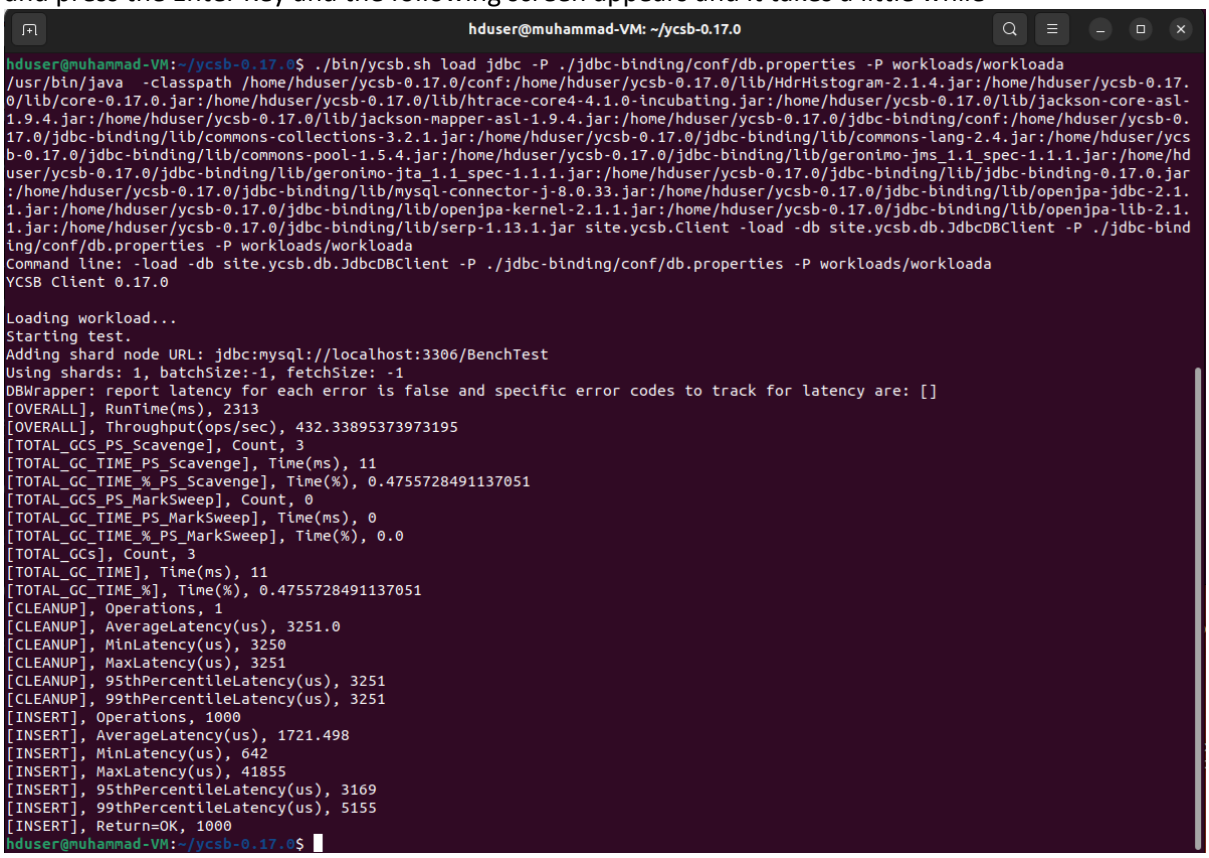
```
hduser@muhammad-VM:~/ycsb-0.17.0/jdbc-binding/conf$ cd ../..
hduser@muhammad-VM:~/ycsb-0.17.0$
```

10) The environment is set for **mongodb** and **mysql** and now we execute the workload with **ycsb-0.17.0** by using the following commands as

Execute a sample workload against mySQL as mentioned below

```
hduser@muhammad-vm:~/ycsb-0.17.0$ ./bin/ycsb.sh load jdbc -P ./jdbc-binding/conf/db.properties -P workloads/workloada
```

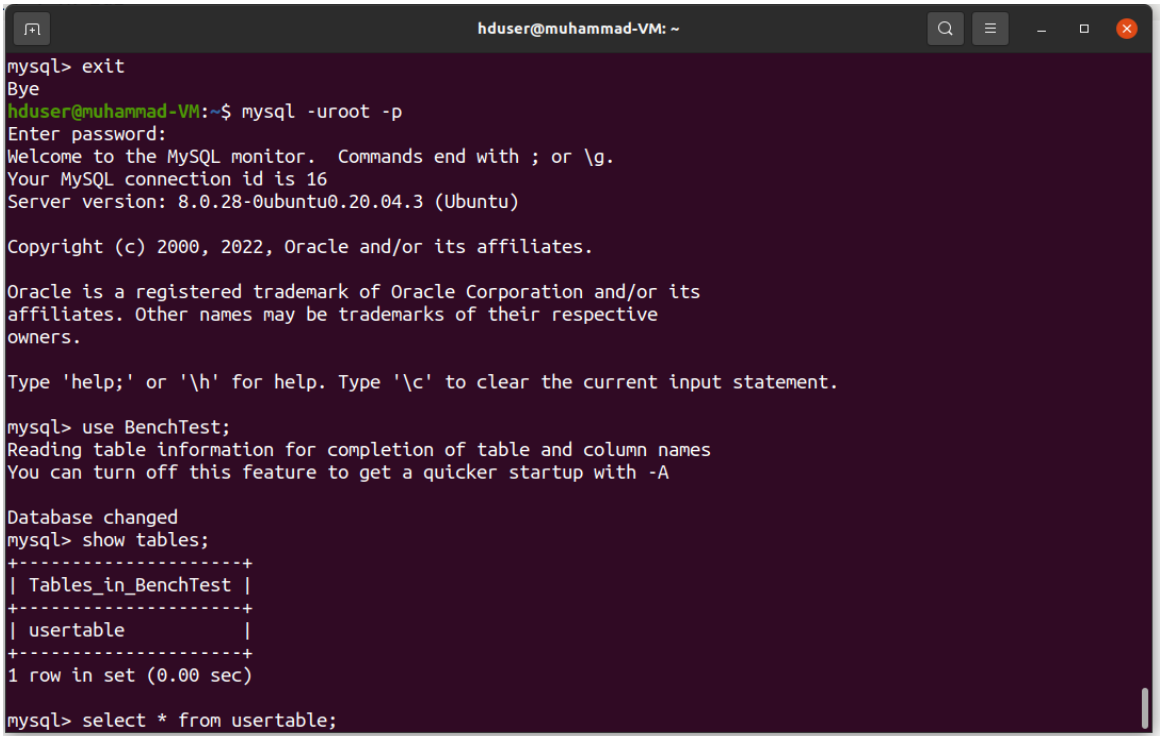
and press the Enter Key and the following screen appears and it takes a little while



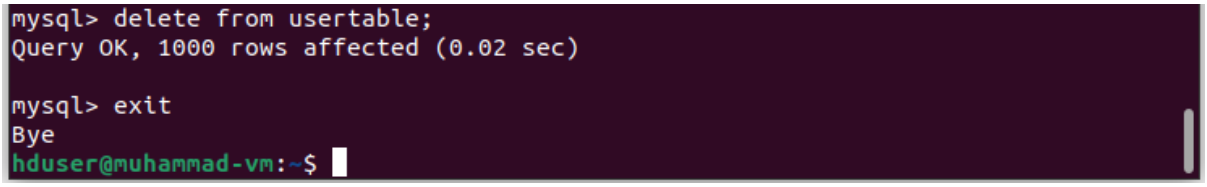
Create an excel sheet and record the parameters (AverageLatency, MinLatency, ..) as you obtained after the execution of the above command for workloada for mySQL. Execute the above command for

different workloads as mentioned in the lecture notes and ycsb web resource (Check for conceptual understanding). This command will store 1000 records in the usertable in MySQL database named as "BenchTest". You can check by using "`mysql>select * from usertable;`"

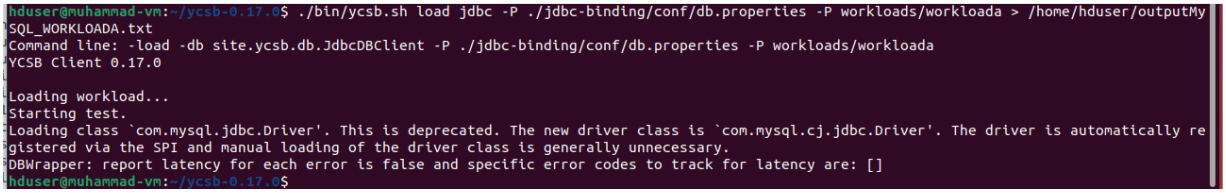
You can start a new session of mySQL on the new terminal. Perform the query in the **BenchTest** database's **usertable** to see if records have been successfully inserted as shown below in the screenshots.
Note: Password for mysql is "**password**" as we set in the Tutorial 5 (part I).



Note: If you would like to execute the above command again, remove the records stored in MySQL table (usertable) by using the command. (`mysql>delete from usertable;`).



If you would like to store the YCSB generated output in the text file, then use the following command. (**Make sure the records should be deleted from mySQL table before the execution of below mentioned command**)

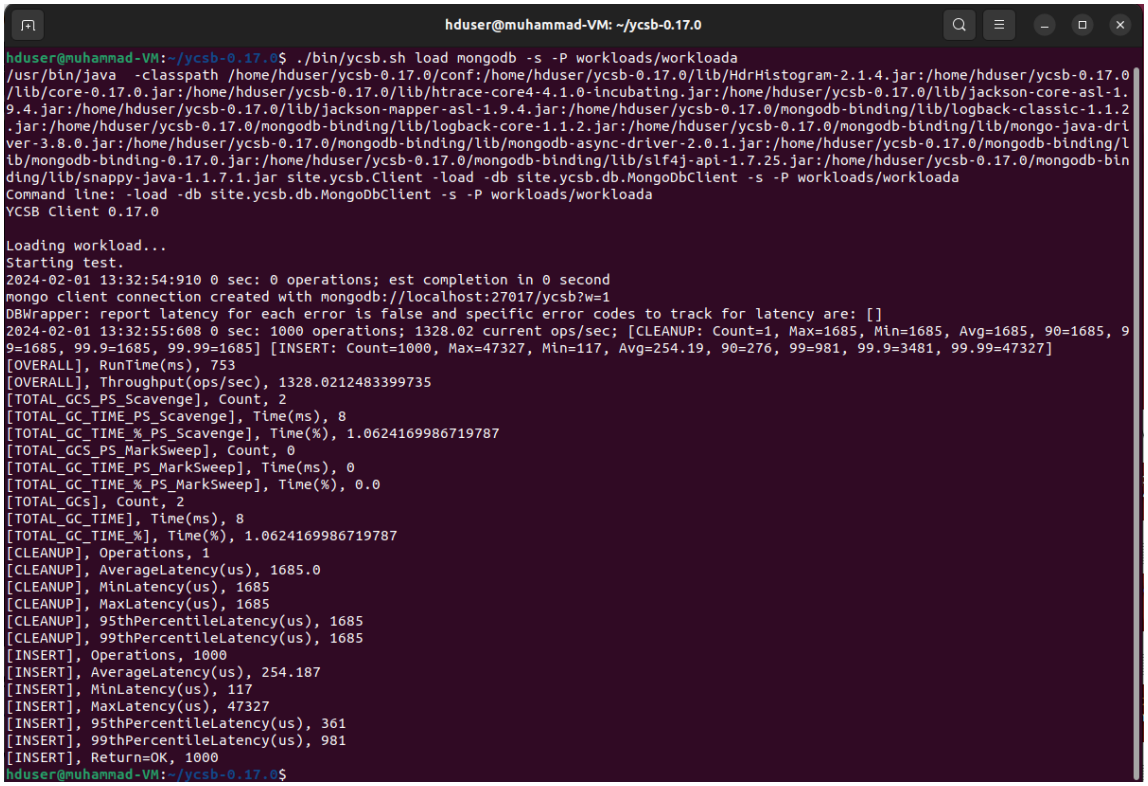


Check the folder /home/hduser/ in ubuntu and you will find the file named as "outputMySQL_WORKLOADA.txt"

11) For mongoDB, you can use the following command to execute a sample workload against mongoDB.

```
hduser@muhammad-vm:~/ycsb-0.17.0$ ./bin/ycsb.sh load mongodb -s -P workloads/workloada
```

The entire output after the execution of above command in this step is mentioned below.



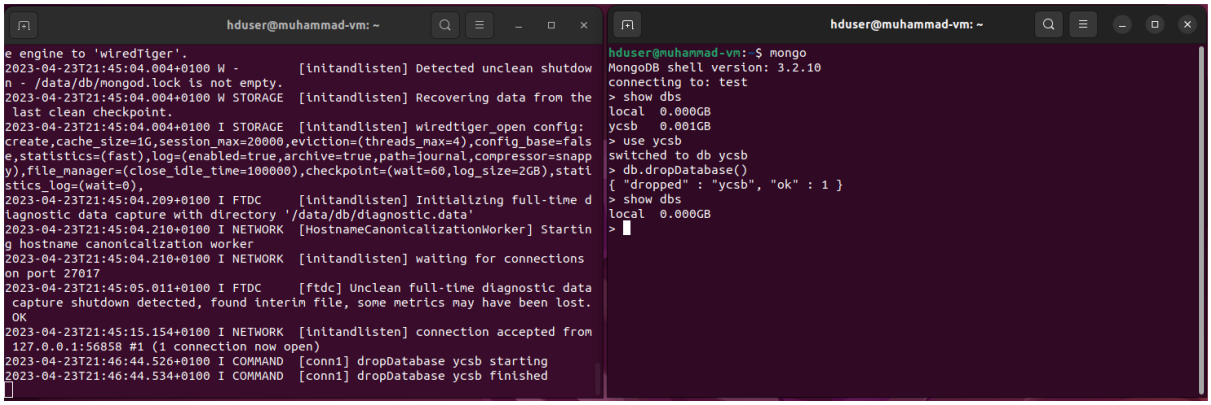
```
hduser@muhammad-VM: ~/ycsb-0.17.0
hduser@muhammad-VM:~/ycsb-0.17.0$ ./bin/ycsb.sh load mongodb -s -P workloads/workloada
/usr/bin/java  -classpath /home/hduser/ycsb-0.17.0/conf:/home/hduser/ycsb-0.17.0/lib/HdrHistogram-2.1.4.jar:/home/hduser/ycsb-0.17.0
/lib/core-0.17.0.jar:/home/hduser/ycsb-0.17.0/lib/htrace-core4-4.1.0-incubating.jar:/home/hduser/ycsb-0.17.0/lib/jackson-core-asl-1.
9.4.jar:/home/hduser/ycsb-0.17.0/lib/jackson-mapper-asl-1.9.4.jar:/home/hduser/ycsb-0.17.0/mongodb-binding/lib/logback-classic-1.1.2
.jar:/home/hduser/ycsb-0.17.0/mongodb-binding/lib/logback-core-1.1.2.jar:/home/hduser/ycsb-0.17.0/mongodb-binding/lib/mongo-java-dri
ver-3.8.0.jar:/home/hduser/ycsb-0.17.0/mongodb-binding/lib/mongodb-async-driver-2.0.1.jar:/home/hduser/ycsb-0.17.0/mongodb-binding/l
ib/mongodb-binding-0.17.0.jar:/home/hduser/ycsb-0.17.0/mongodb-binding/lib/slf4j-api-1.7.25.jar:/home/hduser/ycsb-0.17.0/mongodb-bin
ding/lib/snappy-java-1.1.7.1.jar site.ycsb.Client -load -db site.ycsb.db.MongoDbClient -s -P workloads/workloada
Command line: -load -db site.ycsb.db.MongoDbClient -s -P workloads/workloada
YCSB Client 0.17.0

Loading workload...
Starting test.
2024-02-01 13:32:54:910 0 sec: 0 operations; est completion in 0 second
mongo client connection created with mongodb://localhost:27017/ycsb?w=1
DBWrapper: report latency for each error is false and specific error codes to track for latency are: []
2024-02-01 13:32:55:608 0 sec: 1000 operations; 1328.02 current ops/sec; [CLEANUP: Count=1, Max=1685, Min=1685, Avg=1685, 90=1685, 9
9=1685, 99.9=1685, 99.99=1685] [INSERT: Count=1000, Max=47327, Min=117, Avg=254.19, 90=276, 99=981, 99.9=3481, 99.99=47327]
[OVERALL], RunTime(ms), 753
[OVERALL], Throughput(ops/sec), 1328.0212483399735
[TOTAL_GCS_PS_Scavenge], Count, 2
[TOTAL_GC_TIME_PS_Scavenge], Time(ms), 8
[TOTAL_GC_TIME_%_PS_Scavenge], Time(%), 1.0624169986719787
[TOTAL_GCS_PS_MarkSweep], Count, 0
[TOTAL_GC_TIME_PS_MarkSweep], Time(ms), 0
[TOTAL_GC_TIME_%_PS_MarkSweep], Time(%), 0.0
[TOTAL_GCs], Count, 2
[TOTAL_GC_TIME], Time(ms), 8
[TOTAL_GC_TIME_%], Time(%), 1.0624169986719787
[CLEANUP], Operations, 1
[CLEANUP], AverageLatency(us), 1685.0
[CLEANUP], MinLatency(us), 1685
[CLEANUP], MaxLatency(us), 1685
[CLEANUP], 95thPercentileLatency(us), 1685
[CLEANUP], 99thPercentileLatency(us), 1685
[INSERT], Operations, 1000
[INSERT], AverageLatency(us), 254.187
[INSERT], MinLatency(us), 117
[INSERT], MaxLatency(us), 47327
[INSERT], 95thPercentileLatency(us), 361
[INSERT], 99thPercentileLatency(us), 981
[INSERT], Return=OK, 1000
hduser@muhammad-VM:~/ycsb-0.17.0$
```

You can repeat the exercise for other workloads also and record the data in the excel sheet for comparison as mentioned for mySQL.

You can check a database called "ycsb" and a collection named "usertable" in the MongoDB database to see if the MongoDB workloada was successfully executed or not. Utilizing the MongoDB client, you can examine them.

```
hduser@muhammad-vm: ~
hduser@muhammad-vm:~$ mongo
MongoDB shell version: 3.2.10
connecting to: test
> show dbs
local  0.000GB
ycsb   0.001GB
> use ycsb
switched to db ycsb
> show collections
usertable
> db.usertable.find()
{ "_id" : "user6284781860667377211", "field1" : BinData(0,"Kj12KyM2My9kL01lJFcrNUU/LkFjOz4yJF5vJlwxOjl4LiQ2K0J5LicgJkV/LkglLCMuJT
kqNEgxJDA6PUQvIDNyKjN4M0wpLURtIihoKF9lPzVsPlclMSZgMDMyLDcyO1ApIQ=="), "field0" : BinData(0,"IkhlNl5jKVQlMCQyLFh3P0JpJDRoMih0IlMzI
Vo/PEFxJlYxNFlvLiw2LFR5I0VrIF19O14tISUqMCokPEUxJER3OC4gMCI0MTd8L1BjJDd0P1Z1JVg3JEZ3IEghIDUoNSFoJg=="), "field7" : BinData(0,"LjVq
MlYuJDYuPSMiLlU9MEs1PEA9NTtkMVExLEMjLTQsJEx9IzR8JkJhLERlKkB3NCR+OydiPjg4MEJhLjcmLFclOUhtKitiL0ptPiV+KEs9OSxqODV2M09tM0prJiJ8OTskI
A=="), "field6" : BinData(0,"MytoO1gvLz1oL0NpJVglLE9hLU5hP0Z/JkttJycwOFgvKSgqJS5+Lk9vLUgnLTJsP0FzLkh5KDI4OVJnK0onJFpnL0M/NEojJkV5
JyV8Jig4NDwkPCggNDZ0JyBqNiMqMlh9Ow=="), "field9" : BinData(0,"KVE1MSA6N1ptLl1hIDowKUklJEl/PktlPipsL0ttMy8sOEkhLFFxMzpwPFE/KSNwN10
lLEwjNyc4NUonJ187KE4jLjxsNlNvN1RxLkgtPyYuMS9uOTMmID54Oi58IkoxOEMnIA=="), "field8" : BinData(0,"KS42MEk7OiFmKlwjNDpwJzZwPy52Jj1mKC
```

In order to execute the workload command again, remove the mondoDB database before the execution of the workload command for second workload as shown in the screenshot.



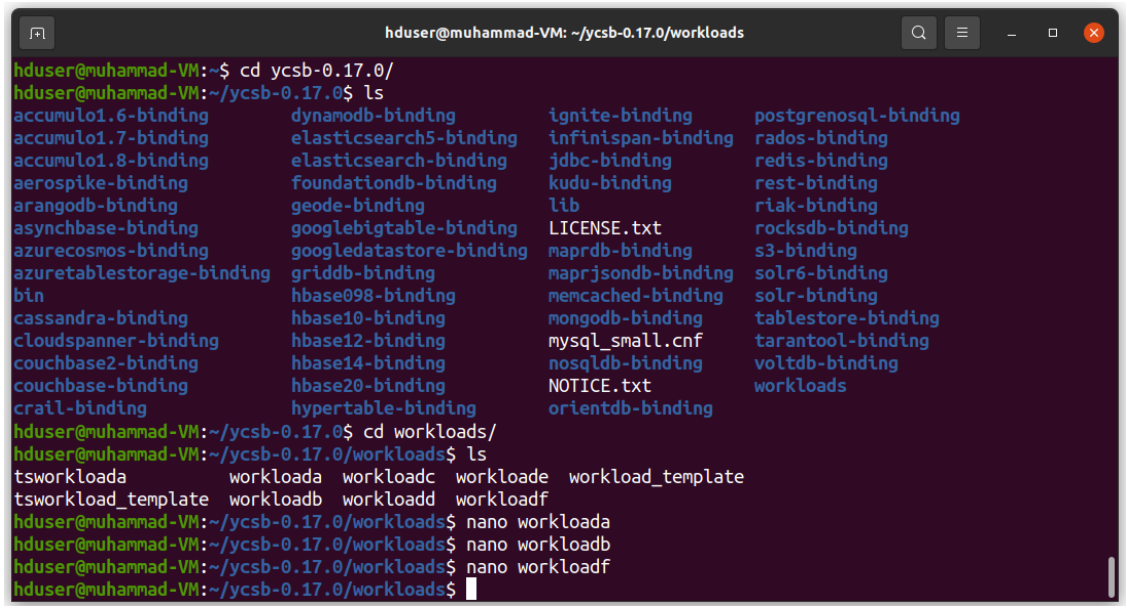
You can execute for second workload after the deletion of the database in mongoDB or you can change the name of database for different workloads. Please explore yourself.

12) If you would like to store output generated by YCSB in a file, then use the following commands for the database (mongoDB) as mentioned below

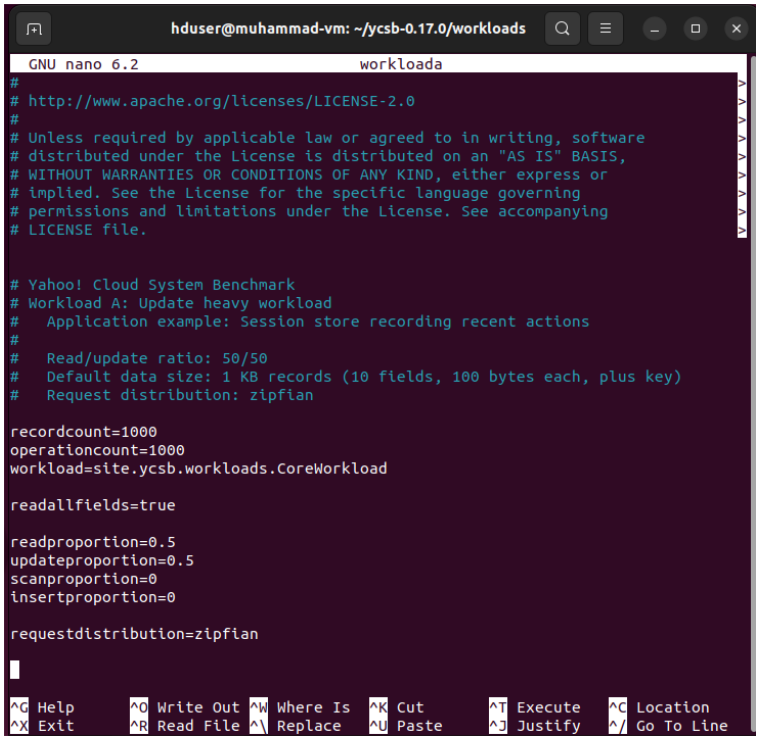
```
$cd /home/hduser/ycsb-0.17.0
$./bin/ycsb load mongodb -s -P workloads/workloada > /home/hduser/output-Load.txt
```

13) Further exploration of workloads files is available in the below-mentioned folders. Different workloads are present in the folder

```
$cd ycsb-0.17.0/workloads
```



You can apply different workloads for **mysql** and **mongodb** to perform a comparative analysis for read, write and update operations. You can modify the recordcount also in these files if you are interested to explore further.



If you need more explanation for any above-mentioned command, please check the below mentioned references or ycsb/mongodb/SQL documentation.

References:

- https://github.com/brianfrankcooper/YCSB/tree/master/mongodb
- https://github.com/brianfrankcooper/YCSB/tree/master/jdbc
- https://kiran4t4t.blogspot.com/2016/05/quick-start-guide-to-run-ycsb-on-mongodb.html
- https://learning.oreilly.com/library/view/mastering-mongodb-6-x/9781803243863/