VIT®

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

ITA-6008 Big Data Analytics

Assignment - 1

By :

Rajat Singh 22MCA0139

Submitted to :

Prof. POUNAMBAL M

1. Procedure to install the Hadoop in your system.

Prerequisite to Hadoop Installation

1. You have installed Ubuntu 22 Desktop version in your Virtual Machine

2. You have installed Java (jdk 8) in your Ubuntu system.

3. Check your hostname is Ubuntu

```
$ hostname --should output Ubuntu
```

Linux Configuration Before Hadoop Installation

We will setup single node Hadoop cluster using a dedicated Hadoop user.

1. Login as Root

2. Adding a dedicated user called hduser

2. Create a Group called Hadoop

```
sudo addgroup Hadoop
```

4. Create an User hduser

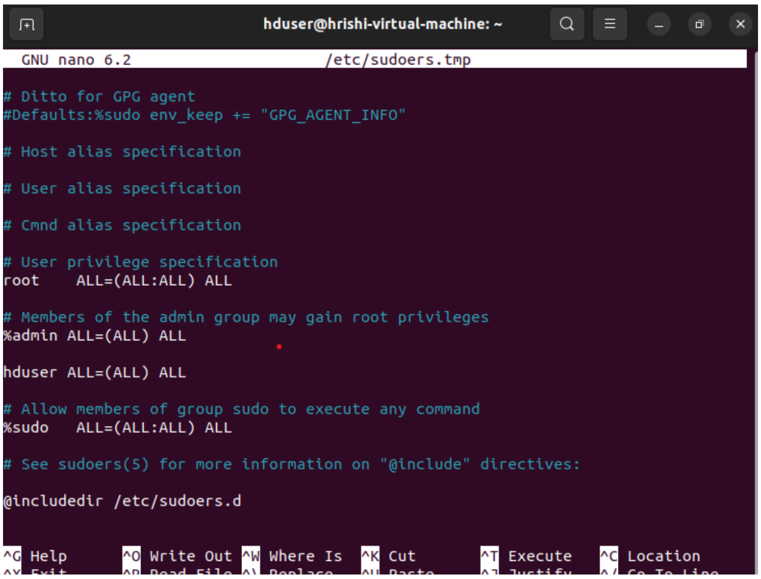
```
sudo adduser hduser
```

5. Add houser to hadoop group

```
sudo adduser houser Hadoop
```

6. Add the 'hduser' to sudoers list so that hduser can do admin tasks.

```
sudo visudo
houser ALL=(ALL) ALL
```



```
  GNU nano 6.2                    /etc/sudoers.tmp

# Ditto for GPG agent
#Defaults:%sudo env_keep += "GPG_AGENT_INFO"

# Host alias specification

# User alias specification

# Cmnd alias specification

# User privilege specification
root    ALL=(ALL:ALL) ALL

# Members of the admin group may gain root privileges
%admin ALL=(ALL) ALL

hduser ALL=(ALL) ALL

# Allow members of group sudo to execute any command
%sudo   ALL=(ALL:ALL) ALL

# See sudoers(5) for more information on "@include" directives:

@includedir /etc/sudoers.d

^G Help      ^O Write Out ^W Where Is  ^K Cut       ^T Execute   ^C Location
^X Exit      ^R Read File ^\ Replace   ^U Paste     ^J Justify   ^/ Go To Line
```

7. Logout Your System and login again as hduser.

8. Configuring SSH

```
sudo apt-get install openssh-server
```

9. Generate SSH for communication

```
hduser@ubuntu:~$ ssh-keygen
```

10.Copy Public Key to Authorized key file & edit the permission

```
hduser@ubuntu:~Scat~/.ssh/id_rsa.pub >> ~/.ssh/authorized_keys
```

```
ho&ser@ubuntu:~$chmod 700 ~/.ssh/auth
```

11.Start SSH

If ssh is not running, then run it by giving the below command

```
hduser@ubuntu:~$ sudo /etc/init.d/ssh restart
```

Enter your Password(hadoop)

12. Test Your SSH Connectivity

```
hduser@ubuntu:~$ ssh localhost
```

13. Test Your SSH Connectivity

```
hduser@ubuntu:-S ssh localhost
```

Download Hadoop

1. Download hadoop-3.3.1.tar.gz and save it to hduser/Desktop.
https://downloads.apache.org/hadoop/

2. move the above downloaded file to /us/local/
Open Terminal(Ctr|+Alt+T)

```
$ sudo mv ~/Desktop/hadoop-3.3.1.tar.gz /us/local/
cd /usr/local
sudo tar -xvf hadoop-3.3.1.tar.gz
sudo rm hadoop-3.3.1.tar.gz
sudo ln -s hadoop-3.3.1 hadoop
sudo chown -R hduser:hadoop hadoop-3.3.1
sudo chmod 777 hadoop-3.3.1
```

3. Edit hadoop-env.sh and configure Java.

```
$ sudo vim /ust/local/hadoop/etc/hadoop/hadoop-env.sh
```

```
export HADOOP_OPTS=Djava.net.preferiPv4Stack=true
export HADOOP_HOME_WARN_SUPPRESS-"TRUE"
export JAVA_HOME=/us/local/java/jdk
```

3. Update $HOME/.bashrc

```
# Set Hadoop-related environment variables
export HADOOP_HOME=/usr/local/hadoop
export HADOOP_MAPRED_HOME=${HADOOP_HOME}
export HADOOP_COMMON_HOME=${HADOOP_HOME}
export HADOOP_HDFS_HOME=${HADOOP_HOME}
export HADOOP_YARN_HOME=${HADOOP_HOME}
export HADOOP_CONF_DIR=${HADOOP_HOME}/etc/hadoop

# Native Path
export
HADOOP_COMMON_LIB_NATIVE_DIR=${HADOOP_PREFIX}/lib/native
export HADOOP_OPTS="-Djava.library.path=$HADOOP_PREFIX/lib"

# Set JAVA_HOME (we will also configure JAVA_HOME directly for Hadoop later
on)
export JAVA_HOME=/usr/local/java/jdk
# Some convenient aliases and functions for running Hadoop-related commands
unaliasfs&> /dev/null
aliasfs="hadoop fs"
unaliashls&> /dev/null
aliashls="fs -ls"
```

```
export
PATH=$PATH:$HADOOP_HOME/bin:$PATH:$JAVA_HOME/bin:$HADOOP_H
OME/sbin
```

5. Update yarn-site.xml

```
$sudo vim /us/local/hadoop/etc/hadoop/yarn-site.xml
```

```
<property>
                <name>yarn.nodemanager.aux-services</name>
                <value>mapreduce_shuffle</value>
            </property>
            <property>
     <name>yarn.nodemanager.aux-services.mapreduce.shuffle.class</name>
   <value>org.apache.hadoop.mapred.ShuffleHandler</value>
            </property>
   <property>
      <name>yarn.nodemanager.vmem-check-enabled</name>
      <value>false</value>
      <description>Whether virtual memory limits will be enforced for
containers</description>
   </property>

  <property>
     <name>yarn.nodemanager.vmem-pmem-ratio</name>
      <value>4</value>
      <description>Ratio between virtual memory to physical memory when setting
memory limits for containers</description>
  </property>
```

6. Update core-site.xml file

```
$ sudo vim /ust/local/hadoop/etc/hadoop/core-site.xml
```

```
<property>
                <name>hadoop.tmp.dir</name>
                <value>/app/hadoop/tmp</value>
                <description>A base for other temporary directories.</description>

            </property>

 <property>
             <name>fs.default.name</name>
            <value>hdfs://localhost:9000</value>
            <description>default host and port</description>
            </property>
```

```
<property>
   <name>hadoop.proxyuser.hduser.hosts</name>
   <value>*</value>
 </property>

<property>
   <name>hadoop.proxyuser.hduser.groups</name>
   <value>*</value>
</property>
```

7. Create the above temp folder and give appropriate permission

```
sudo mkdir -p /app/hadoop/tmp
sudo chown hduser:hadoop -R /app/hadog
sudo chmod 750 /app/hadoop/tmp
```

8. Edit mapred-site.xml

```
sudo vim /us/local/hadoop/etc/hadoop/mapred-site.xml
```

```
<property>
   <name>mapreduce.framework.name</name>
         <value>yarn</value>
         </property>
  <property>
         <name>mapreduce.jobhistory.address</name>
                  <value>localhost:10020</value>
                  <description>Host and port for Job History Server (default

                  0.0.0.0:10020)</description>
</property>

  <property>
      <name>yarn.app.mapreduce.am.env</name>
      <value>HADOOP_MAPRED_HOME=${HADOOP_HOME}</value>
   </property>
   <property>
      <name>mapreduce.map.env</name>
      <value>HADOOP_MAPRED_HOME=${HADOOP_HOME}</value>
   </property>
   <property>
      <name>mapreduce.reduce.env</name>
      <value>HADOOP_MAPRED_HOME=${HADOOP_HOME}</value>
   </property>
```

9. Create a temporary directory which will be used as base location for

```
sudo  mkdir  -p  /us/local/hadoop  tmp/hdfs/namenode
sudo,mkdir  -p  /ust/local/hadoop_tmp/hdfs/datanode
sudo chown hduser:hadoop -R /us/local/hadoop tmp/
```

10. Update hdfs-site.xmI file

```
$ sudo vim /us/local/hadoop/etc/hadoop/hdfs-site.xml
```

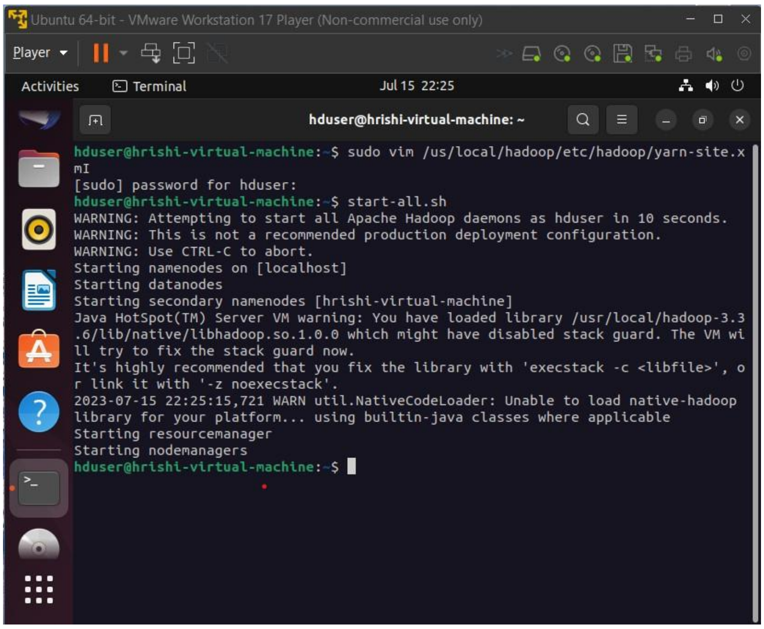
```
            <property>
              <name>dfs.replication</name>
              <value>1</value>
             </property>
             <property>
              <name>dfs.namenode.name.dir</name>
              <value>file:/usr/local/hadoop_tmp/hdfs/namenode</value>
             </property>
             <property>
              <name>dfs.datanode.data.dir</name>
              <value>file:/usr/local/hadoop_tmp/hdfs/datanode</value>
             </property>
```

11.Format your namenode

```
$ hadoop namenode -format
```

12.Starting your single-node cluster

```
$ start-all.sh
```


Ubuntu 64-bit - VMware Workstation 17 Player (Non-commercial use only)
Player
Activities
Terminal
Jul 15 22:25
hduser@hrishi-virtual-machine: ~

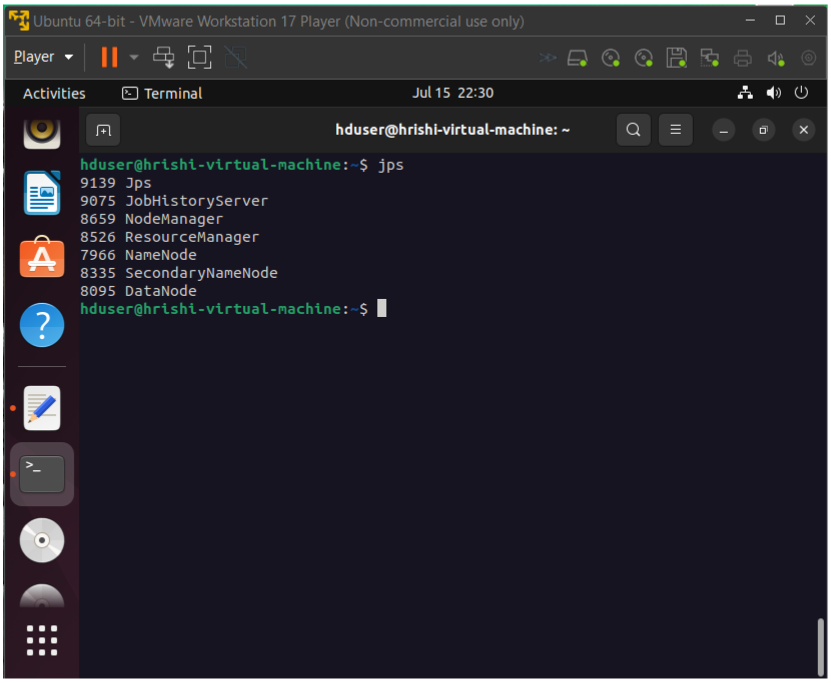
```
hduser@hrishi-virtual-machine:~$ sudo vim /us/local/hadoop/etc/hadoop/yarn-site.x
mI
[sudo] password for hduser:
hduser@hrishi-virtual-machine:~$ start-all.sh
WARNING: Attempting to start all Apache Hadoop daemons as hduser in 10 seconds.
WARNING: This is not a recommended production deployment configuration.
WARNING: Use CTRL-C to abort.
Starting namenodes on [localhost]
Starting datanodes
Starting secondary namenodes [hrishi-virtual-machine]
Java HotSpot(TM) Server VM warning: You have loaded library /usr/local/hadoop-3.3
.6/lib/native/libhadoop.so.1.0.0 which might have disabled stack guard. The VM wi
ll try to fix the stack guard now.
It's highly recommended that you fix the library with 'execstack -c <libfile>', o
r link it with '-z noexecstack'.
2023-07-15 22:25:15,721 WARN util.NativeCodeLoader: Unable to load native-hadoop
library for your platform... using builtin-java classes where applicable
Starting resourcemanager
Starting nodemanagers
hduser@hrishi-virtual-machine:~$ 
```

13. Start your history-server.

```
$ mr-jobhistory-daemon.sh start historyserver
$ mr-jobhistory-daemon.sh stop historyserver
```

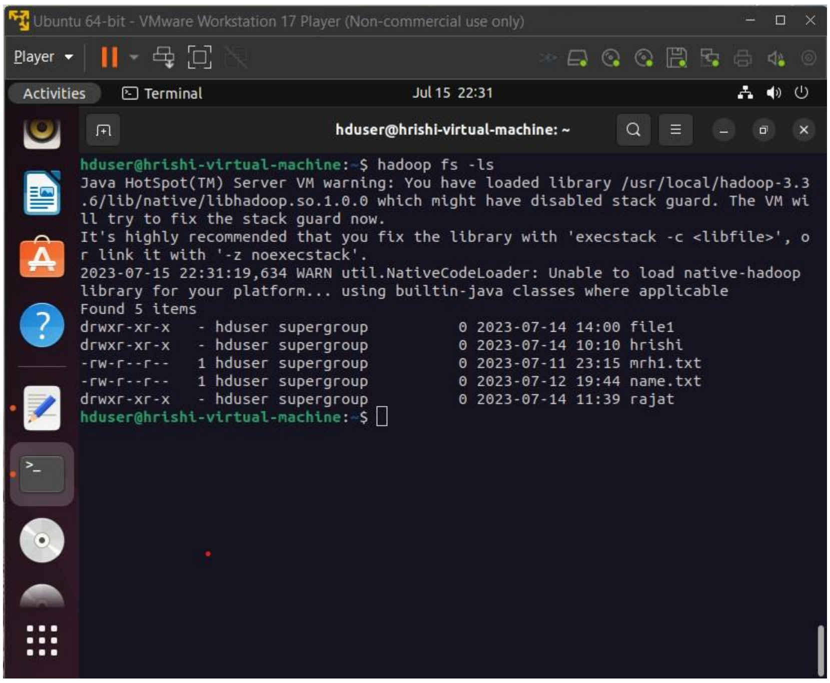
14. Check if all the necessary hadoop daemon is running or not

```
$ jps
```



Ubuntu 64-bit - VMware Workstation 17 Player (Non-commercial use only)
Player
Activities
Terminal
Jul 15 22:30
hduser@hrishi-virtual-machine: ~

```
hduser@hrishi-virtual-machine:~$ jps
9139 Jps
9075 JobHistoryServer
8659 NodeManager
8526 ResourceManager
7966 NameNode
8335 SecondaryNameNode
8095 DataNode
hduser@hrishi-virtual-machine:~$
```



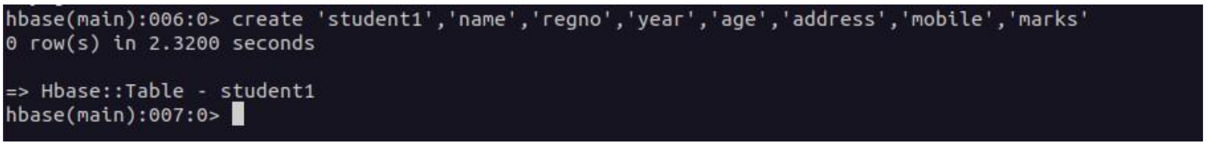
Ubuntu 64-bit - VMware Workstation 17 Player (Non-commercial use only)
Player
Activities
Terminal
Jul 15 22:31
hduser@hrishi-virtual-machine: ~

```
hduser@hrishi-virtual-machine:~$ hadoop fs -ls
Java HotSpot(TM) Server VM warning: You have loaded library /usr/local/hadoop-3.3
.6/lib/native/libhadoop.so.1.0.0 which might have disabled stack guard. The VM wi
ll try to fix the stack guard now.
It's highly recommended that you fix the library with 'execstack -c <libfile>', o
r link it with '-z noexecstack'.
2023-07-15 22:31:19,634 WARN util.NativeCodeLoader: Unable to load native-hadoop
library for your platform... using builtin-java classes where applicable
Found 5 items
drwxr-xr-x   - hduser supergroup          0 2023-07-14 14:00 file1
drwxr-xr-x   - hduser supergroup          0 2023-07-14 10:10 hrishi
-rw-r--r--   1 hduser supergroup          0 2023-07-11 23:15 mrh1.txt
-rw-r--r--   1 hduser supergroup          0 2023-07-12 19:44 name.txt
drwxr-xr-x   - hduser supergroup          0 2023-07-14 11:39 rajat
hduser@hrishi-virtual-machine:~$
```

2. Creation of sample table in HABASE with 7 attributes with 2 sub columns and displaying of valid data.

Table Creation :

create 'student1', 'name', 'regno', 'year'', 'age', 'address', 'mobile', 'marks'



```
hbase(main):006:0> create 'student1','name','regno','year','age','address','mobile','marks'
0 row(s) in 2.3200 seconds

=> Hbase::Table - student1
hbase(main):007:0> 
```

put ‘student1’, 'row1', 'Name', 'John'

put ' student1', 'row1', 'RegNo', '2021001'

put 'student1', 'row1', 'Year', '2023'

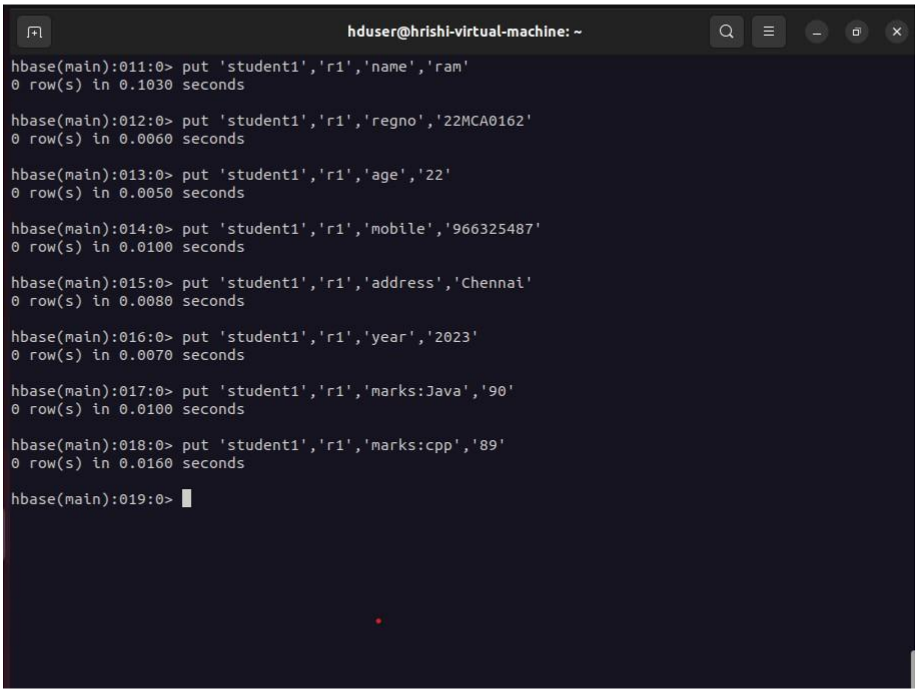
put ‘student1’, 'row1', 'Age', '25'

put ‘student1’, 'row1', 'Mobile', '9876543210'

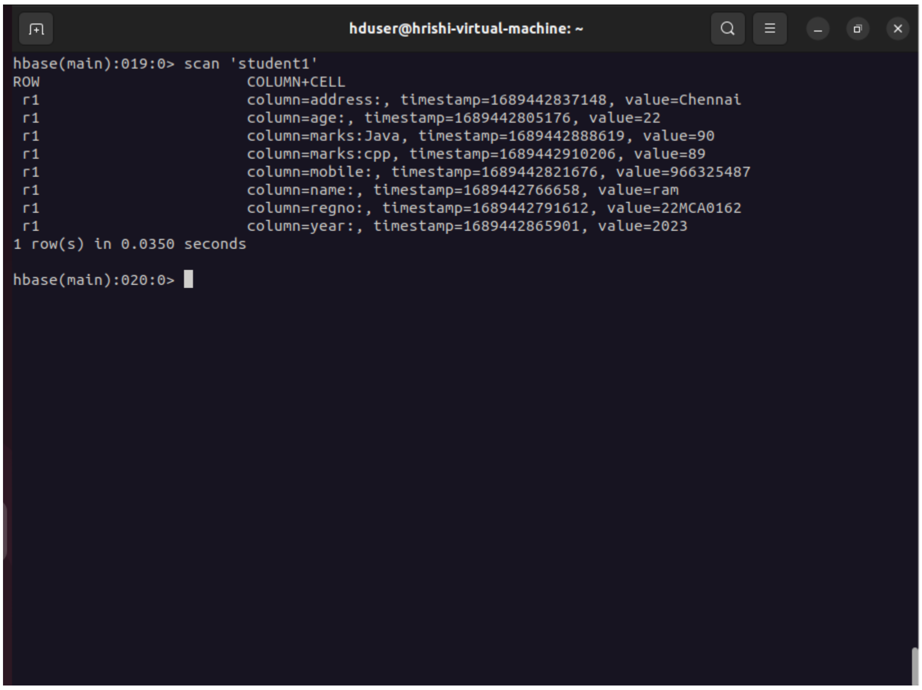
put ‘student1’, 'row1', 'Address', '123 Main Street'

put ‘student1’, 'row1', 'Marks:Marks1', '80'

put ‘student1’, 'row1', 'Marks:Marks2', '75'



```
hbase(main):011:0> put 'student1','r1','name','ram'
0 row(s) in 0.1030 seconds

hbase(main):012:0> put 'student1','r1','regno','22MCA0162'
0 row(s) in 0.0060 seconds

hbase(main):013:0> put 'student1','r1','age','22'
0 row(s) in 0.0050 seconds

hbase(main):014:0> put 'student1','r1','mobile','966325487'
0 row(s) in 0.0100 seconds

hbase(main):015:0> put 'student1','r1','address','Chennai'
0 row(s) in 0.0080 seconds

hbase(main):016:0> put 'student1','r1','year','2023'
0 row(s) in 0.0070 seconds

hbase(main):017:0> put 'student1','r1','marks:Java','90'
0 row(s) in 0.0100 seconds

hbase(main):018:0> put 'student1','r1','marks:cpp','89'
0 row(s) in 0.0160 seconds

hbase(main):019:0> 
```



```
hbase(main):019:0> scan 'student1'
ROW                       COLUMN+CELL
 r1                       column=address:, timestamp=1689442837148, value=Chennai
 r1                       column=age:, timestamp=1689442805176, value=22
 r1                       column=marks:Java, timestamp=1689442888619, value=90
 r1                       column=marks:cpp, timestamp=1689442910206, value=89
 r1                       column=mobile:, timestamp=1689442821676, value=966325487
 r1                       column=name:, timestamp=1689442766658, value=ram
 r1                       column=regno:, timestamp=1689442791612, value=22MCA0162
 r1                       column=year:, timestamp=1689442865901, value=2023
1 row(s) in 0.0350 seconds

hbase(main):020:0> 
```