Task 1:

Explain the below concepts with an example in brief.

- Nosql Databases

Answer:

NoSQL is an approach to database design that can accommodate a wide variety of data models, including key-value, document, columnar and graph formats. NoSQL, which stand for "not only SQL," is an alternative to traditional relational databases in which data is placed in tables and data schema is carefully designed before the database is built. NoSQL databases are especially useful for working with large sets of distributed data.

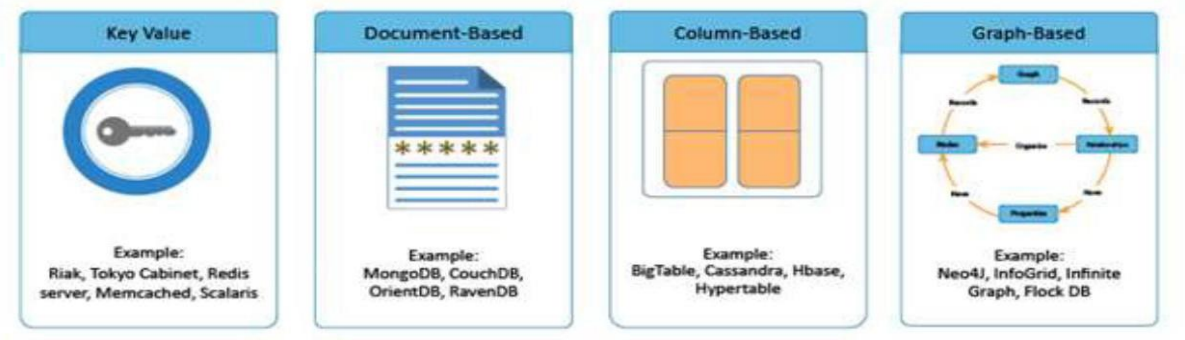
Motivations for this approach include:

- Simplicity of design
- Simpler horizontal scaling to clusters of machines
- Finer control over availability.

NoSQL Database Examples:

There are 4 basic types of NoSQL databases:

- **Document databases** pair each key with a complex data structure known as a document. Documents can contain many different key-value pairs, or key-array pairs, or even nested documents. It stores documents made up of tagged elements. {Example- CouchDB}
- **Graph stores** are used to store information about networks of data, such as social connections. A network database that uses edges and nodes to represent and store data. {Example- Neo4J}
- **Key-value stores** are the simplest NoSQL databases. Every single item in the database is stored as an attribute name (or 'key'), together with its value. It has a Big Hash Table of keys & values {Example- Riak, Amazon S3 (Dynamo)}
- **Wide-column** stores such as HBase are optimized for queries over large datasets, and store columns of data together, instead of rows. Each storage block contains data from only one column, {Example- HBase, Cassandra}



● Types of Nosql Databases

1 Document Store No SQL Database

The central concept of a document store is the notion of a "document". While each document oriented
database implementation differs on the details of this definition, in general, they all assume that documents encapsulate and encode data (or information) in some standard formats or encodings.

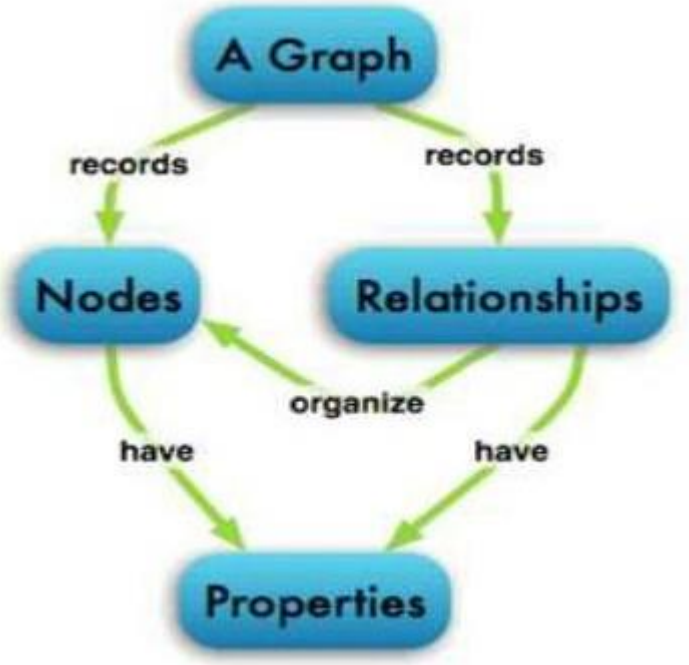
Example:

```
{
    '_id' : 1,
    'artistName' : { 'Iron Maiden' },
    'albums' : [
        {
            'albumname' : 'The Book of Souls',
            'datereleased' : 2015,
            'genre' : 'Hard Rock'
        }, {
            'albumname' : 'Killers',
            'datereleased' : 1981,
            'genre' : 'Hard Rock'
        }, {
            'albumname' : 'Powerslave',
            'datereleased' : 1984,
            'genre' : 'Hard Rock'
        }, {
            'albumname' : 'Somewhere in Time',
            'datereleased' : 1986,
            'genre' : 'Hard Rock'
        }
    ]
}
```

Notice that I decided to add an _id field in the second example. This may or may not be required by the DBMS, however, some DBMSs will automatically insert a unique ID field if one isn't supplied.

2 Graph Store NoSQL Database

In a Graph Base NoSQL Database, you will not find the rigid format of SQL or the tables and columns representation, a flexible graphical representation is instead used which is perfect to address scalability concerns. Graph structures are used with edges, nodes and properties which provides index-free adjacency. Data can be easily transformed from one model to the other using a Graph Base NoSQL database.



- These databases that uses edges and nodes to represent and store data.

- These nodes are organised by some relationships with one another, which is represented by edges between the nodes.
- Both the nodes and the relationships have some defined properties.

3 Key – Value Store NoSQL Database

The schema-less format of a key value database is just about what you need for your storage needs. The key can be synthetic or auto-generated while the value can be String, JSON, BLOB etc.

The key value type basically, uses a hash table in which there exists a unique key and a pointer to a particular item of data. A bucket is a logical group of keys – but they don't physically group the data. There can be identical keys in different buckets. Performance is enhanced to a great degree because of the cache mechanisms that accompany the mappings. To read a value you need to know both the key and the bucket because the real key is a hash (Bucket+ Key).

It is not an ideal method if you are only looking to just update part of a value or query the database.

Phone Directory

Key	Value
Bob	(123) 456-7890
Jane	(234) 567-8901
Tara	(345) 678-9012
Tiara	(456) 789-0123

Disadvantages:

a. The model will not provide any kind of traditional database capabilities (such as atomicity of transactions, or consistency when multiple transactions are executed simultaneously).
If volume of the data increases, maintaining unique values as keys may

b. If volume of the data increases, maintaining unique values as keys may become more difficult; addressing this issue requires the introduction of some complexity in generating character strings that will remain unique among an extremely large set of keys.

4 Wide-Column store Database

In column-oriented NoSQL database, data is stored in cells grouped in columns of data rather than as rows of data. Columns are logically grouped into column families. Column families can contain a virtually unlimited number of columns that can be created at runtime or the definition of the schema. Read and write is done using columns rather than rows.
In comparison, most relational DBMS store data in rows, the benefit of storing data in columns, is fast search/ access and data aggregation. Columnar databases store all the cells corresponding to a column as a continuous disk entry thus makes the search/access faster.
For example: To query the titles from articles is just one disk access, title of all the items can be obtained.

● CAP Theorem

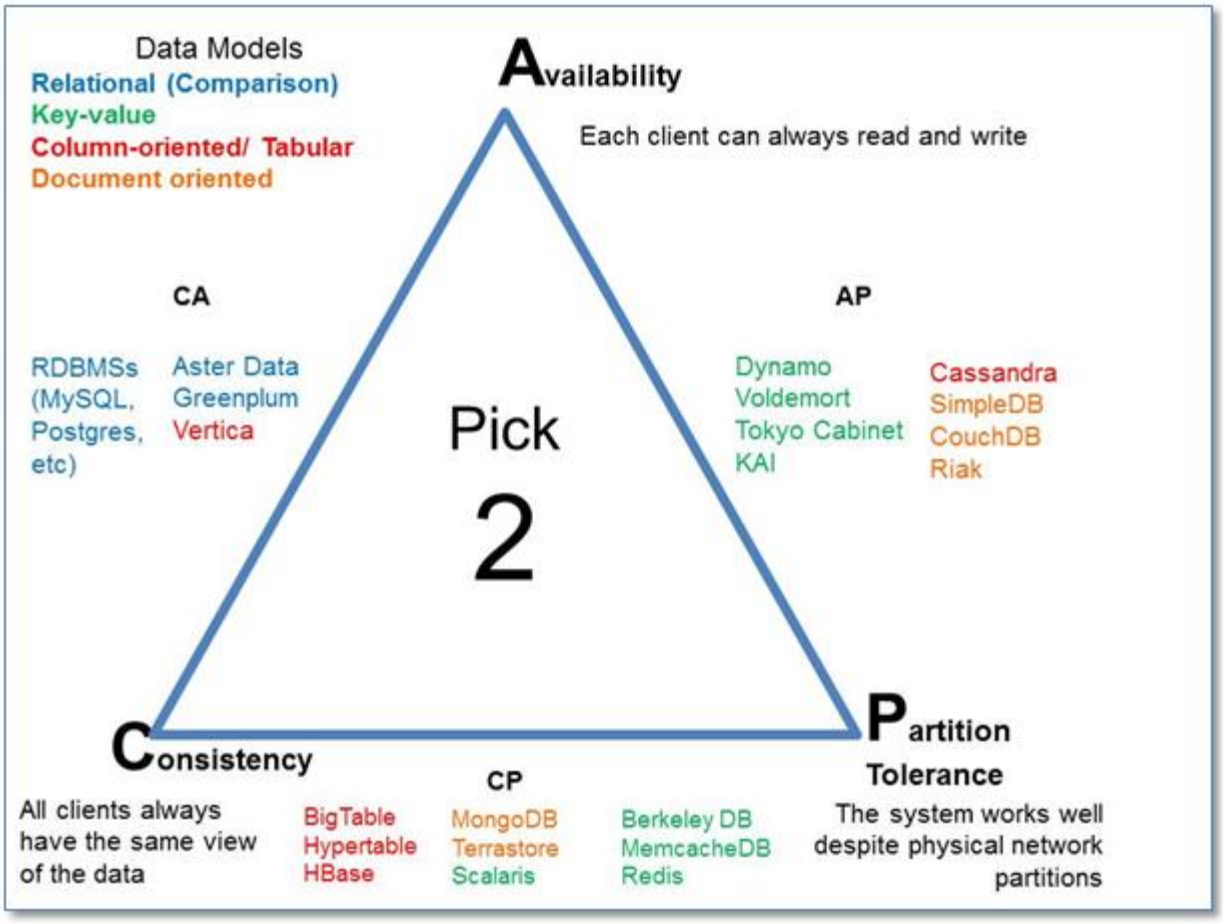
In a distributed system, the following three properties are important.

- **Consistency** - This means that the data in the database remains consistent after the execution of an operation. For example after an update operation, all clients see the same data.
- **Availability** - This means that the system is always on (service guarantee availability), no downtime.
- **Partition Tolerance** - This means that the system continues to function even if the communication among the servers is unreliable, i.e. the servers may be partitioned into multiple groups that cannot communicate with one another.

The CAP theorem was proposed by Eric Brewer.

According to this theorem, in any distributed system, you can use only two of the three properties—consistency, availability, or partition tolerance simultaneously.
Many NoSQL databases provide options for a developer to choose to adjust the database as per requirement. For this, understanding the following requirements is important:

- How the data is consumed by the system?
- Whether the data is read or write heavy.
- If there is a need to query data with random parameters.
- If the system is capable of handling inconsistent data.



Consistency

- Consistency in CAP theorem refers to atomicity and isolation. Consistency means consistent read and write operations for the same sets of data so that concurrent operations see the same valid and consistent data state, without any stale data.

- Consistency in ACID means if the data does not satisfy predefined constraints, it is not persisted. Consistency in CAP theorem is different. In a single-machine database, consistency is achieved using the ACID semantics. However, in the case of NoSQL databases which are scaled out and distributed providing consistency gets complicated.

Availability

According to the CAP theorem, availability means:

- The database system must be available to operate when required. This means that a system that is busy, uncommunicative, unresponsive, or inaccessible is not available.
- If a system is not available to serve a request at a time it is needed, it is unavailable.

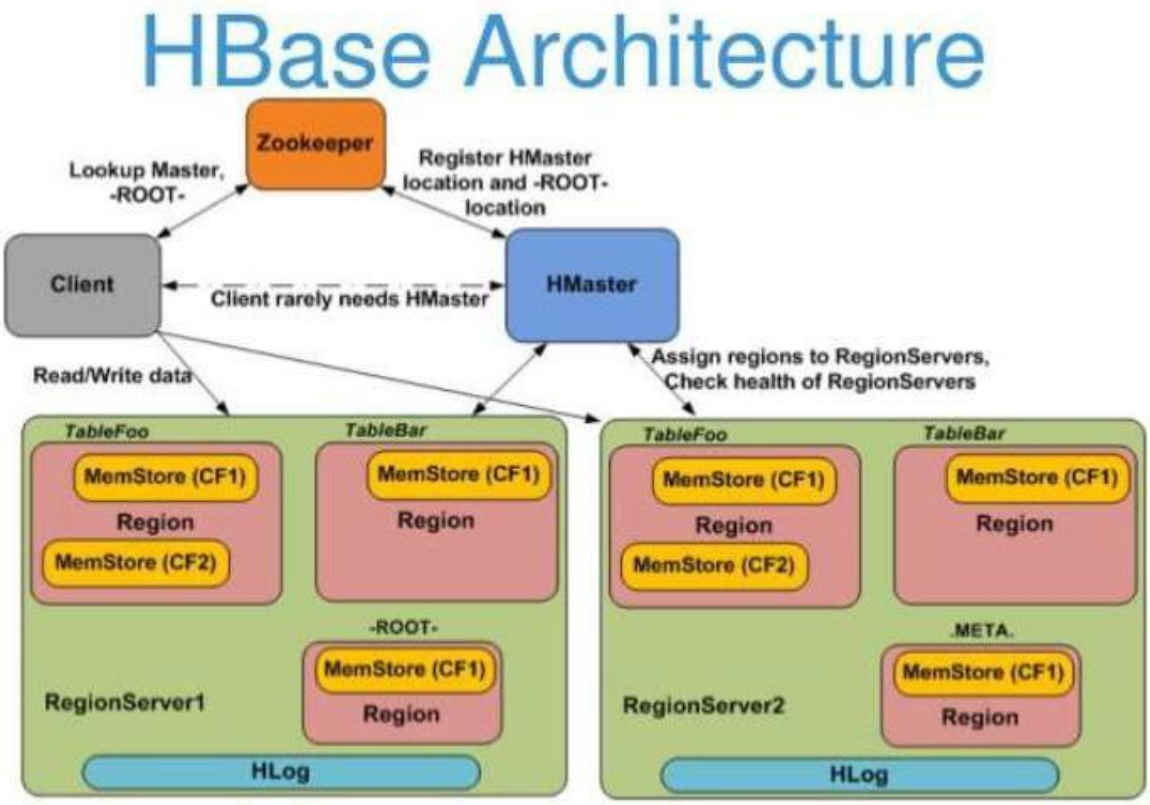
Partition Tolerance

Partition tolerance or fault-tolerance is the third element of the CAP theorem. Partition tolerance measures the ability of a system to continue its service when some of its clusters become unavailable.

● HBase Architecture

In HBase, tables are split into regions and are served by the region servers. Regions are vertically divided by column families into “Stores”. Stores are saved as files in HDFS. Shown below is the architecture of HBase.

Components of Apache HBase Architecture



HBase architecture has 3 important components-

- HMaster (Master Server)
- Region Servers
- ZooKeeper
-

Master Server (HMaster) :

1. Assigns regions to the region servers and takes the help of Apache ZooKeeper for this task.
2. Handles load balancing of the regions across region servers. It unloads the busy servers and shifts the regions to less occupied servers.
3. Maintains the state of the cluster by negotiating the load balancing.
4. Is responsible for schema changes and other metadata operations such as creation of tables and column families.

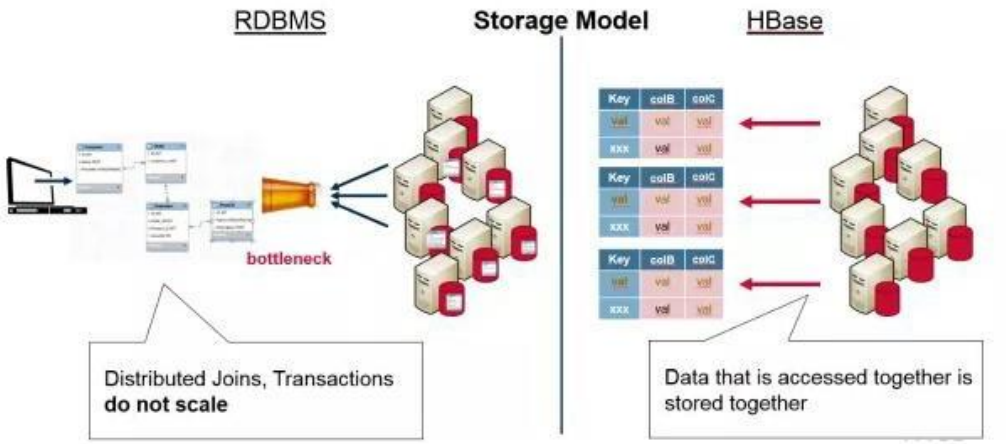
Region servers

5. Communicate with the client and handle data-related operations.
6. Handle read and write requests for all the regions under it.
7. Decide the size of the region by following the region size thresholds.
8. Region Server runs on HDFS DataNode and consists of the following components
    a. Block Cache – This is the read cache. Most frequently reads the data stored in the read cache and whenever the block cache is full, recently used data is evicted.
    b. MemStore- This is the write cache and stores new data that is not yet written to the disk. Every column family in a region has a MemStore.
    c. Write Ahead Log (WAL) is a file that stores new data that is not persisted to permanent storage.
    d. HFile- It is the actual storage file that stores the rows as sorted key values on a disk.

Zookeeper

9. Zookeeper is an open-source project that provides services like maintaining
10. configuration information, naming, providing distributed synchronization, etc.
11. Zookeeper has ephemeral nodes representing different region servers. Master
12. servers use these nodes to discover available servers.
13. In addition to availability, the nodes are also used to track server failures or network
14. partitions.
15. Clients communicate with region servers via zookeeper.
16. In pseudo and standalone modes, HBase itself will take care of zookeeper.

● HBase vs RDBMS



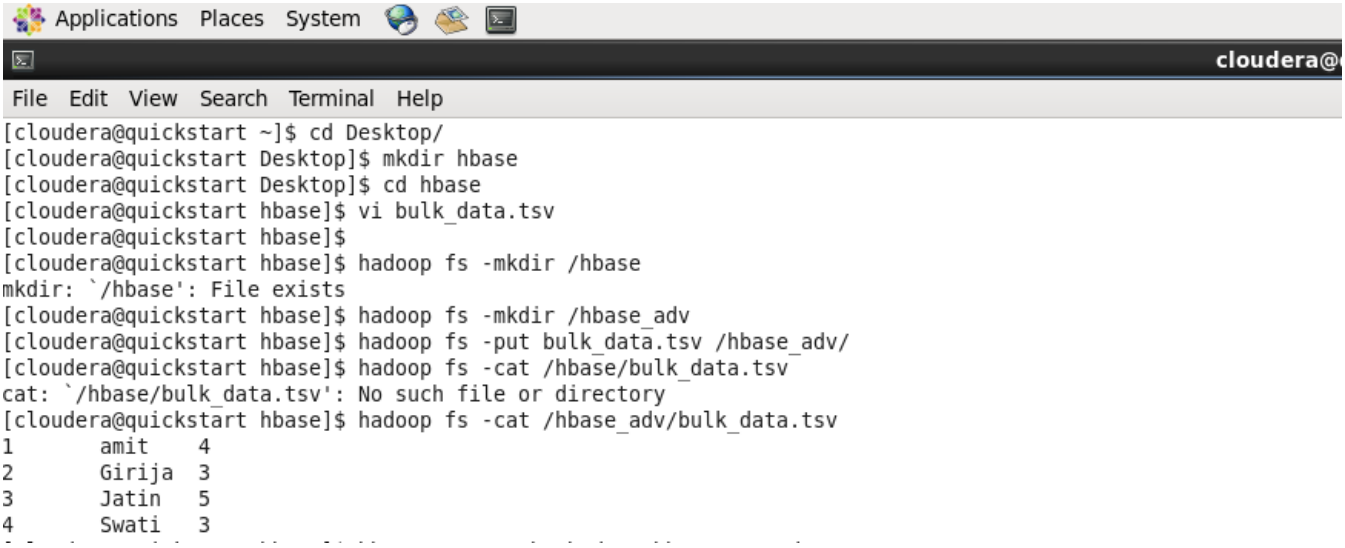
HBase	RDBMS
✓ 1. Column-oriented	✓ 1. Row-oriented(mostly)
✓ 2. Flexible schema, add columns on the	✓ 2. Fixed schema
✓ 3. Good with sparse tables.	✓ 3. Not optimized for sparse tables.
✓ 4. No query language	✓ 4. SQL
✓ 5. Wide tables	✓ 5. Narrow tables
✓ 6. Joins using MR – not ✓ optimized	✓ 6. optimized for Joins(small, fast ✓ ones)
✓ 7. Tight – Integration with MR	✓ 7. Not really
✓ 8. De-normalize your data.	✓ 8. Normalize as you can
✓ 9. Horizontal scalability-just add hard war.	✓ 9. Hard to share and scale.
✓ 10. Consistent	✓ 10. Consistent
✓ 11. No transactions.	✓ 11. transactional
✓ 12. Good for semi-structured data as well as structured data.	✓ 12. Good for structured data.

Task 2

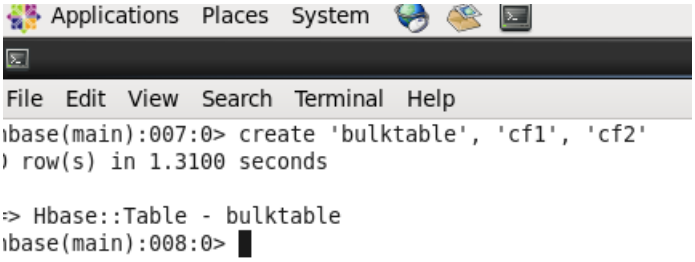
Execute blog present in below link

https://acadgild.com/blog/importtsv-data-from-hdfs-into-hbase/

- created file bulk_data.tsv in local file system
- inserted data given in blog(tab delimited)
- using -put command moved data from local to Hadoop
- using cat command viewed the data



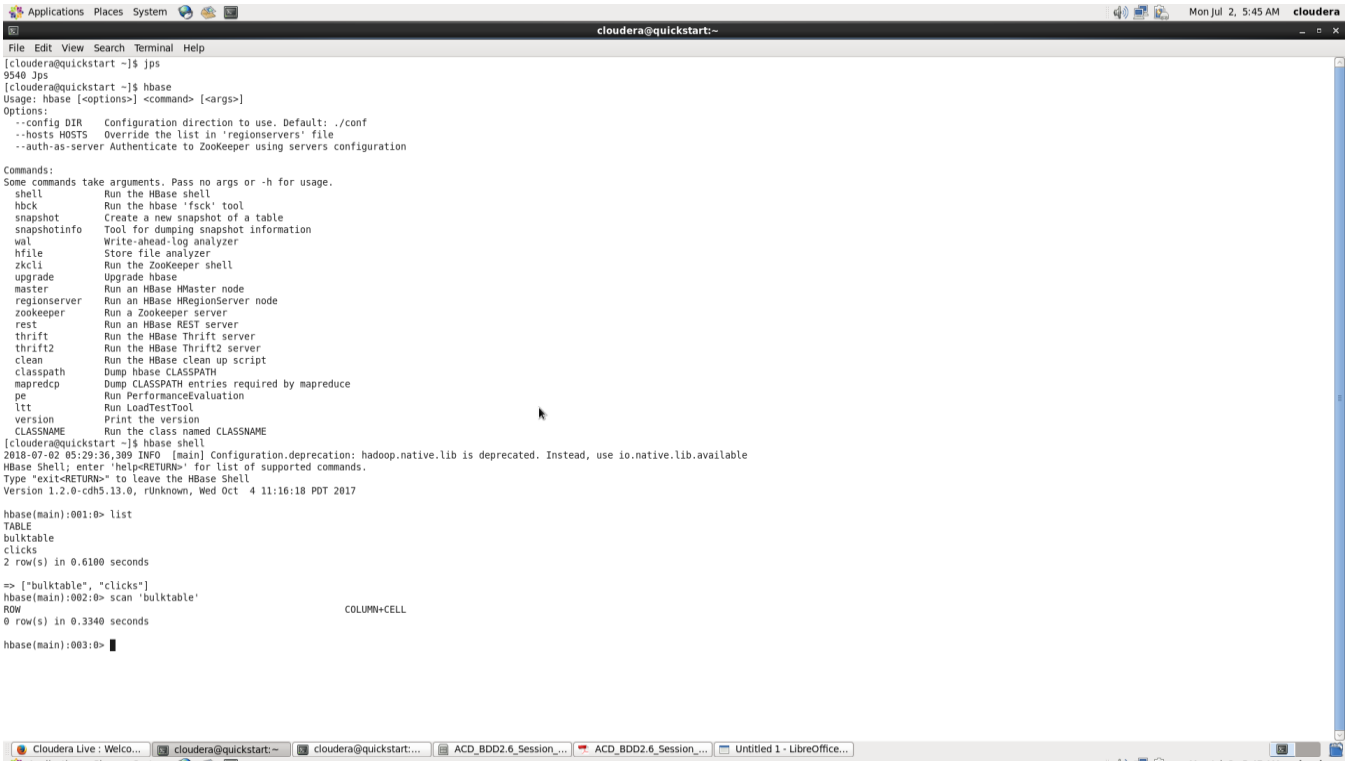
- **Created HBASE table 'bulktable '**

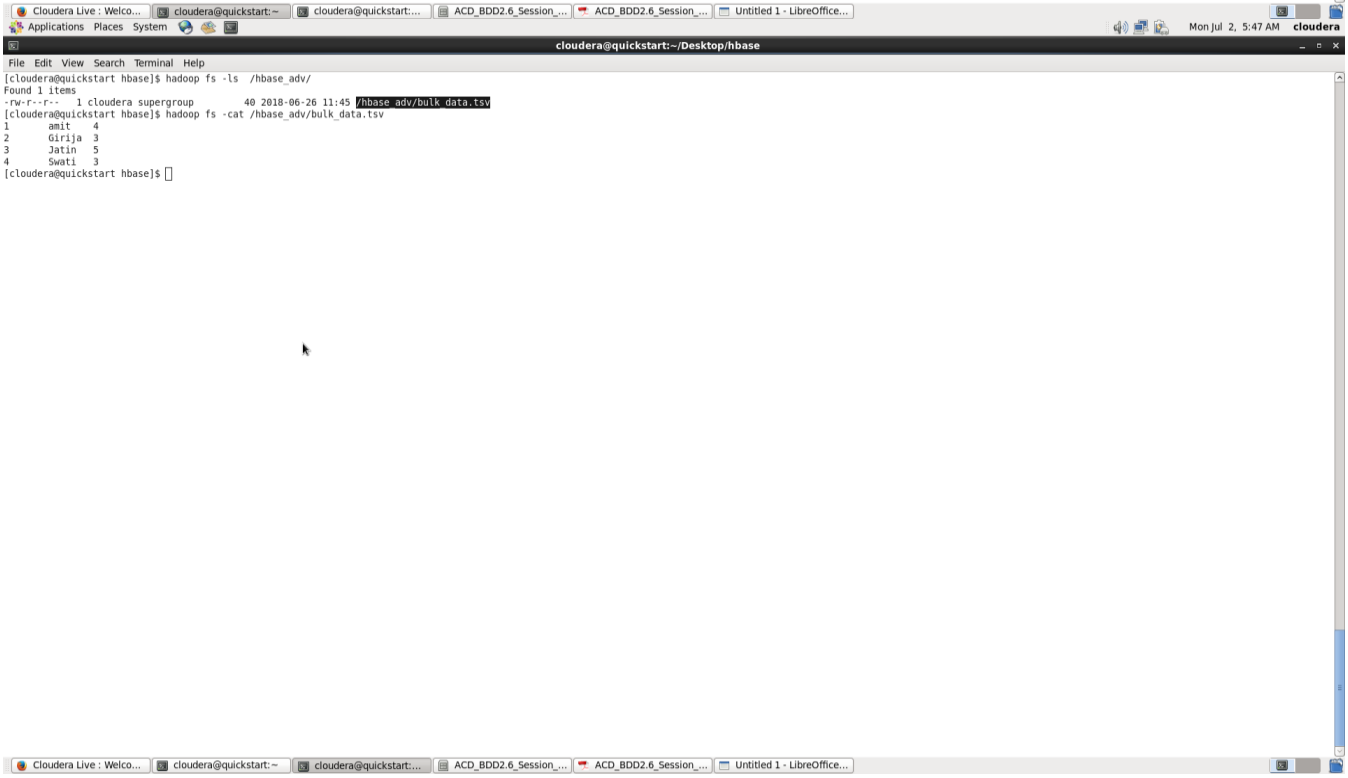


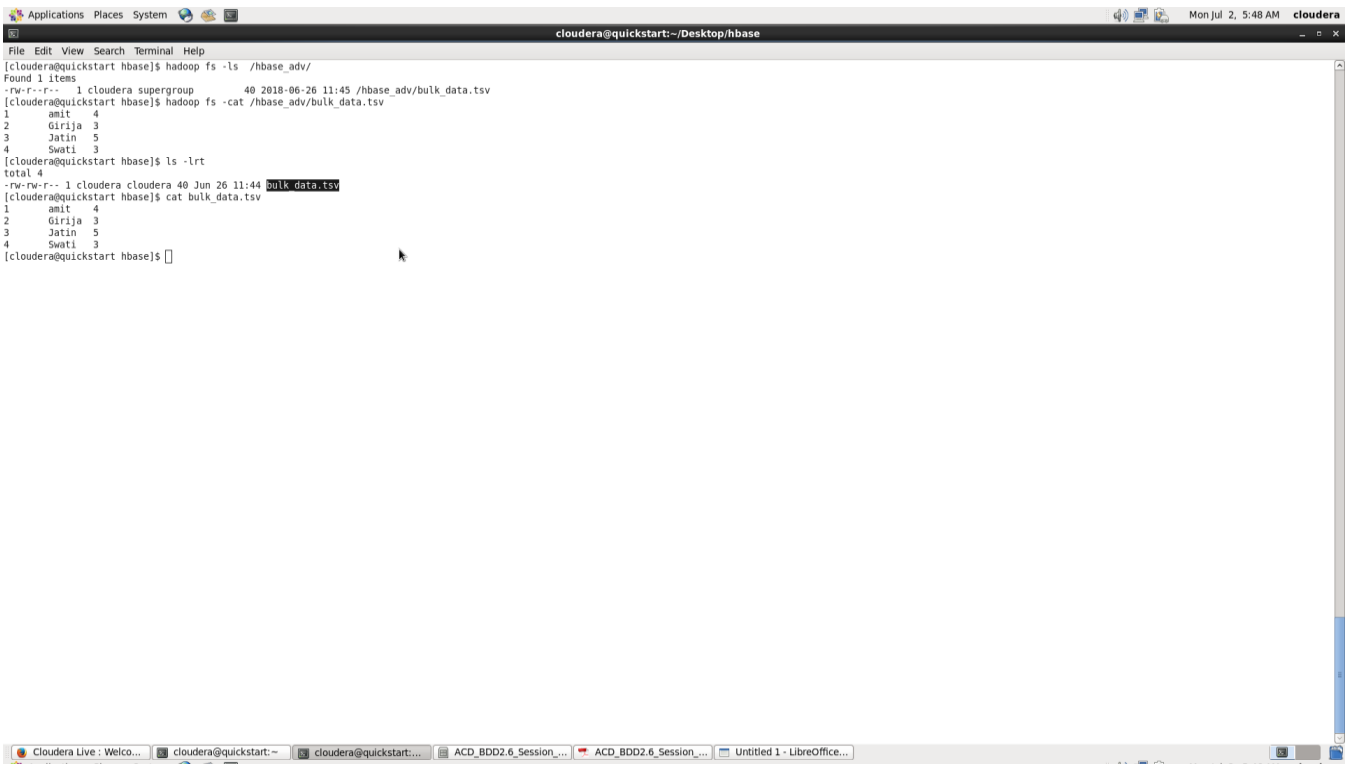
- Inserted data from hdfs to hbase using query in blog

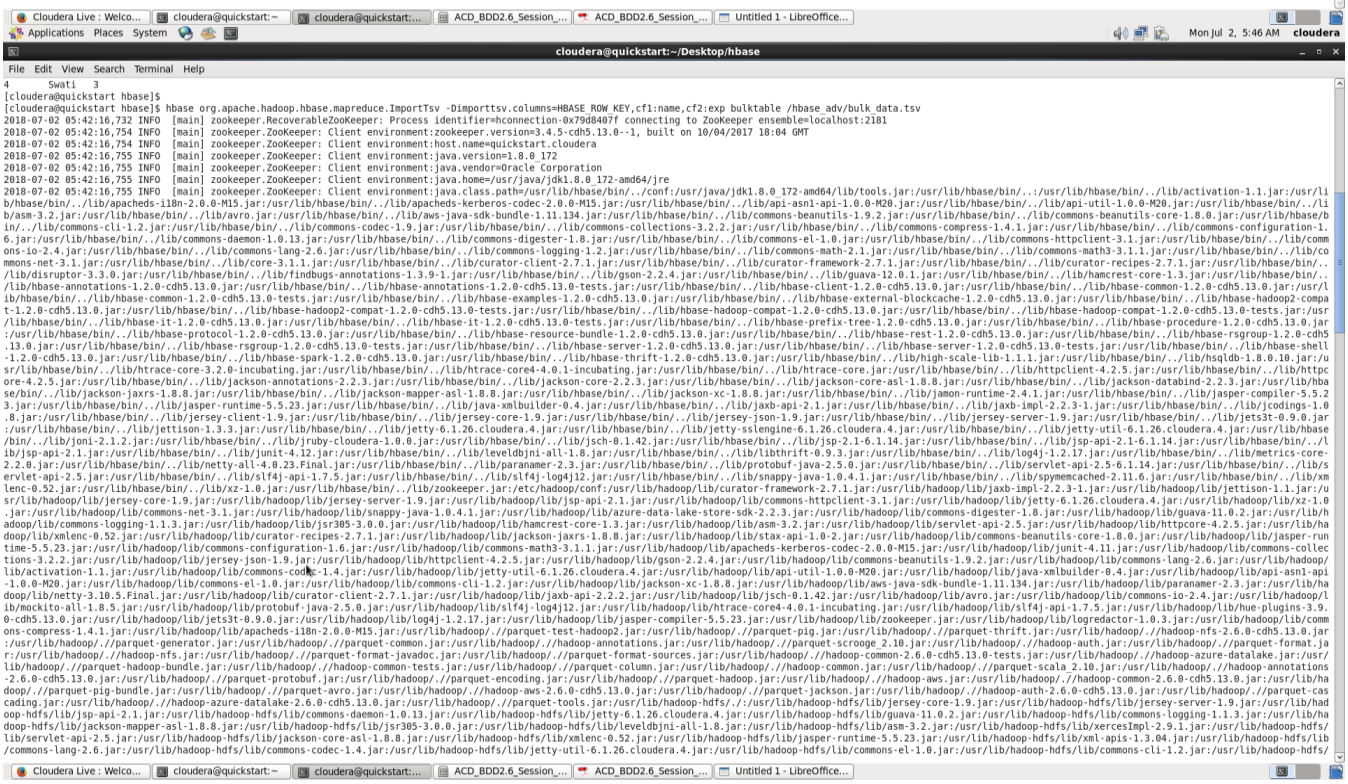
hbase org.apache.hadoop.hbase.mapreduce.ImportTsv –Dimporttsv.columns=HBASE_ROW_KEY,cf1:name,cf2:exp bulktable /hbase/bulk_data.tsv

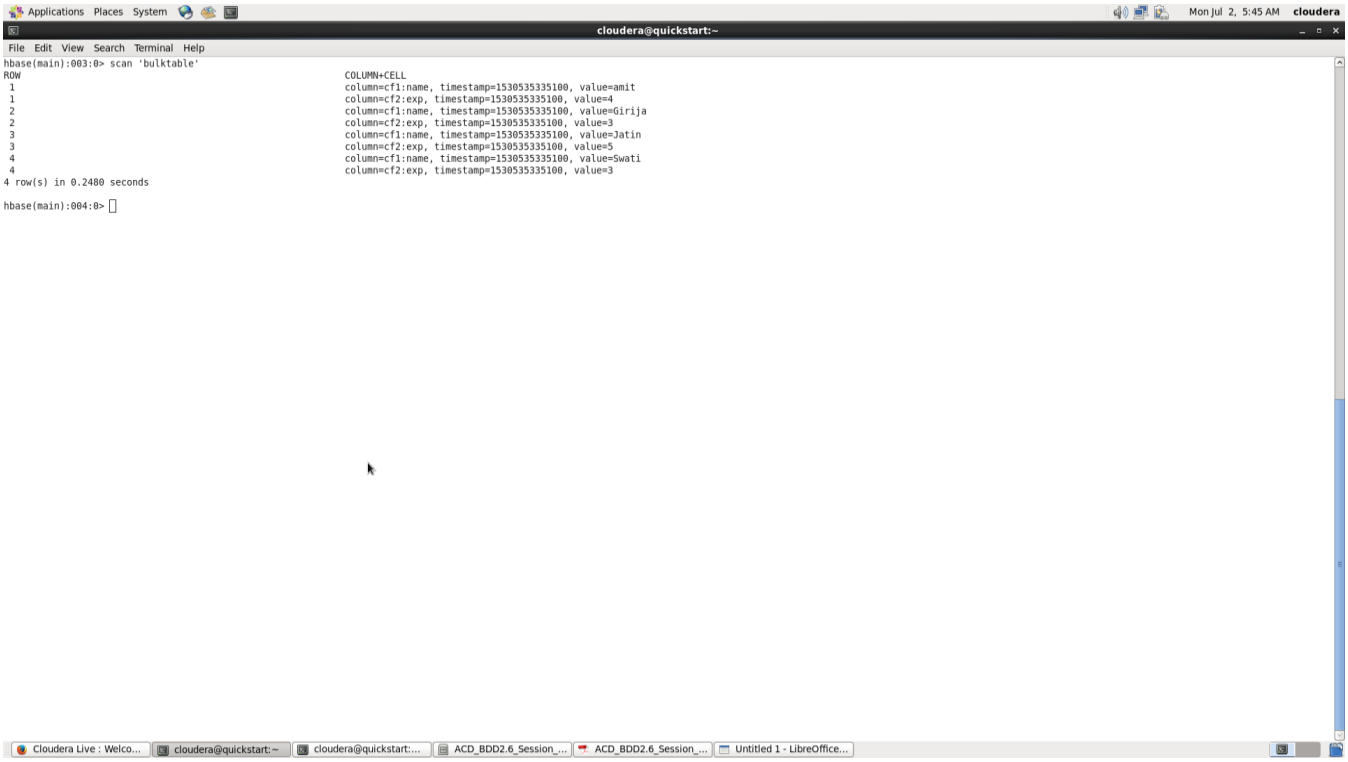
- Please find the screenshot's given & scanned table after insert


Applications Places System
Mon Jul 2, 5:45 AM cloudera
cloudera@quickstart:~
File Edit View Search Terminal Help
[cloudera@quickstart ~]$ jps
9540 Jps
[cloudera@quickstart ~]$ hbase
Usage: hbase [<options>] <command> [<args>]
Options:
--config DIR Configuration direction to use. Default: ./conf
--hosts HOSTS Override the list in 'regionservers' file
--auth-as-server Authenticate to ZooKeeper using servers configuration
Commands:
Some commands take arguments. Pass no args or -h for usage.
shell Run the HBase shell
hbck Run the hbase 'fsck' tool
snapshot Create a new snapshot of a table
snapshotinfo Tool for dumping snapshot information
wal Write-ahead-log analyzer
hfile Store file analyzer
zkcli Run the ZooKeeper shell
upgrade Upgrade hbase
master Run an HBase HMaster node
regionserver Run an HBase HRegionServer node
zookeeper Run a Zookeeper server
rest Run an HBase REST server
thrift Run the HBase Thrift server
thrift2 Run the HBase Thrift2 server
clean Run the HBase clean up script
classpath Dump hbase CLASSPATH
mapredcp Dump CLASSPATH entries required by mapreduce
pe Run PerformanceEvaluation
ltt Run LoadTestTool
version Print the version
CLASSNAME Run the class named CLASSNAME
[cloudera@quickstart ~]$ hbase shell
2018-07-02 05:29:36,309 INFO [main] Configuration.deprecation: hadoop.native.lib is deprecated. Instead, use io.native.lib.available
HBase Shell; enter 'help<RETURN>' for list of supported commands.
Type "exit<RETURN>" to leave the HBase Shell
Version 1.2.0-cdh5.13.0, rUnknown, Wed Oct 4 11:16:18 PDT 2017
hbase(main):001:0> list
TABLE
bulktable
clicks
2 row(s) in 0.6100 seconds
=> ["bulktable", "clicks"]
hbase(main):002:0> scan 'bulktable'
ROW COLUMN+CELL
0 row(s) in 0.3340 seconds
hbase(main):003:0>
Cloudera Live : Welco...
cloudera@quickstart:~
cloudera@quickstart:...
ACD_BDD2.6_Session_...
ACD_BDD2.6_Session_...
Untitled 1 - LibreOffice...


Applications Places System
Mon Jul 2, 5:47 AM cloudera
cloudera@quickstart:~/Desktop/hbase
File Edit View Search Terminal Help
[cloudera@quickstart hbase]$ hadoop fs -ls /hbase_adv/
Found 1 items
-rw-r--r-- 1 cloudera supergroup 40 2018-06-26 11:45 /hbase_adv/bulk_data.tsv
[cloudera@quickstart hbase]$ hadoop fs -cat /hbase_adv/bulk_data.tsv
1 amit 4
2 Girija 3
3 Jatin 5
4 Swati 3
[cloudera@quickstart hbase]$
Cloudera Live : Welco...
cloudera@quickstart:~
cloudera@quickstart:...
ACD_BDD2.6_Session_...
ACD_BDD2.6_Session_...
Untitled 1 - LibreOffice...


Applications Places System
Mon Jul 2, 5:48 AM cloudera
cloudera@quickstart:~/Desktop/hbase
File Edit View Search Terminal Help
[cloudera@quickstart hbase]$ hadoop fs -ls /hbase_adv/
Found 1 items
-rw-r--r-- 1 cloudera supergroup 40 2018-06-26 11:45 /hbase_adv/bulk_data.tsv
[cloudera@quickstart hbase]$ hadoop fs -cat /hbase_adv/bulk_data.tsv
1 amit 4
2 Girija 3
3 Jatin 5
4 Swati 3
[cloudera@quickstart hbase]$ ls -lrt
total 4
-rw-rw-r-- 1 cloudera cloudera 40 Jun 26 11:44 bulk_data.tsv
[cloudera@quickstart hbase]$ cat bulk_data.tsv
1 amit 4
2 Girija 3
3 Jatin 5
4 Swati 3
[cloudera@quickstart hbase]$


Cloudera Live : Welco...
cloudera@quickstart:~
cloudera@quickstart:...
ACD_BDD2.6_Session_...
ACD_BDD2.6_Session_...
Untitled 1 - LibreOffice...
Applications Places System
Mon Jul 2, 5:46 AM cloudera
cloudera@quickstart:~/Desktop/hbase
File Edit View Search Terminal Help
4 Swati 3
[cloudera@quickstart hbase]$
[cloudera@quickstart hbase]$ hbase org.apache.hadoop.hbase.mapreduce.ImportTsv -Dimporttsv.columns=HBASE_ROW_KEY,cf1:name,cf2:exp bulktable /hbase_adv/bulk_data.tsv
2018-07-02 05:42:16,732 INFO [main] zookeeper.RecoverableZooKeeper: Process identifier=hconnection-0x79d8407f connecting to ZooKeeper ensemble=localhost:2181
2018-07-02 05:42:16,754 INFO [main] zookeeper.ZooKeeper: Client environment:zookeeper.version=3.4.5-cdh5.13.0--1, built on 10/04/2017 18:04 GMT
2018-07-02 05:42:16,754 INFO [main] zookeeper.ZooKeeper: Client environment:host.name=quickstart.cloudera
2018-07-02 05:42:16,755 INFO [main] zookeeper.ZooKeeper: Client environment:java.version=1.8.0_172
2018-07-02 05:42:16,755 INFO [main] zookeeper.ZooKeeper: Client environment:java.vendor=Oracle Corporation
2018-07-02 05:42:16,755 INFO [main] zookeeper.ZooKeeper: Client environment:java.home=/usr/java/jdk1.8.0_172-amd64/jre
2018-07-02 05:42:16,755 INFO [main] zookeeper.ZooKeeper: Client environment:java.class.path=/usr/lib/hbase/bin/../conf:/usr/java/jdk1.8.0_172-amd64/lib/tools.jar:/usr/lib/hbase/bin/..:/usr/lib/hbase/bin/../lib/activation-1.1.jar:/usr/lib/hbase/bin/../lib/apacheds-i18n-2.0.0-M15.jar:/usr/lib/hbase/bin/../lib/apacheds-kerberos-codec-2.0.0-M15.jar:/usr/lib/hbase/bin/../lib/api-asn1-api-1.0.0-M20.jar:/usr/lib/hbase/bin/../lib/api-util-1.0.0-M20.jar:/usr/lib/hbase/bin/../lib/asm-3.2.jar:/usr/lib/hbase/bin/../lib/avro.jar:/usr/lib/hbase/bin/../lib/aws-java-sdk-bundle-1.11.134.jar:/usr/lib/hbase/bin/../lib/commons-beanutils-1.9.2.jar:/usr/lib/hbase/bin/../lib/commons-beanutils-core-1.8.0.jar:/usr/lib/hbase/bin/../lib/commons-cli-1.2.jar:/usr/lib/hbase/bin/../lib/commons-codec-1.9.jar:/usr/lib/hbase/bin/../lib/commons-collections-3.2.2.jar:/usr/lib/hbase/bin/../lib/commons-compress-1.4.1.jar:/usr/lib/hbase/bin/../lib/commons-configuration-1.6.jar:/usr/lib/hbase/bin/../lib/commons-daemon-1.0.13.jar:/usr/lib/hbase/bin/../lib/commons-digester-1.8.jar:/usr/lib/hbase/bin/../lib/commons-el-1.0.jar:/usr/lib/hbase/bin/../lib/commons-httpclient-3.1.jar:/usr/lib/hbase/bin/../lib/commons-io-2.4.jar:/usr/lib/hbase/bin/../lib/commons-lang-2.6.jar:/usr/lib/hbase/bin/../lib/commons-logging-1.2.jar:/usr/lib/hbase/bin/../lib/commons-math-2.1.jar:/usr/lib/hbase/bin/../lib/commons-math3-3.1.1.jar:/usr/lib/hbase/bin/../lib/commons-net-3.1.jar:/usr/lib/hbase/bin/../lib/core-3.1.1.jar:/usr/lib/hbase/bin/../lib/curator-client-2.7.1.jar:/usr/lib/hbase/bin/../lib/curator-framework-2.7.1.jar:/usr/lib/hbase/bin/../lib/curator-recipes-2.7.1.jar:/usr/lib/hbase/bin/../lib/disruptor-3.3.0.jar:/usr/lib/hbase/bin/../lib/findbugs-annotations-1.3.9-1.jar:/usr/lib/hbase/bin/../lib/gson-2.2.4.jar:/usr/lib/hbase/bin/../lib/guava-12.0.1.jar:/usr/lib/hbase/bin/../lib/hamcrest-core-1.3.jar:/usr/lib/hbase/bin/../lib/hbase-annotations-1.2.0-cdh5.13.0.jar:/usr/lib/hbase/bin/../lib/hbase-annotations-1.2.0-cdh5.13.0-tests.jar:/usr/lib/hbase/bin/../lib/hbase-client-1.2.0-cdh5.13.0.jar:/usr/lib/hbase/bin/../lib/hbase-common-1.2.0-cdh5.13.0.jar:/usr/lib/hbase/bin/../lib/hbase-common-1.2.0-cdh5.13.0-tests.jar:/usr/lib/hbase/bin/../lib/hbase-examples-1.2.0-cdh5.13.0.jar:/usr/lib/hbase/bin/../lib/hbase-external-blockcache-1.2.0-cdh5.13.0.jar:/usr/lib/hbase/bin/../lib/hbase-hadoop2-compat-1.2.0-cdh5.13.0.jar:/usr/lib/hbase/bin/../lib/hbase-hadoop2-compat-1.2.0-cdh5.13.0-tests.jar:/usr/lib/hbase/bin/../lib/hbase-hadoop-compat-1.2.0-cdh5.13.0.jar:/usr/lib/hbase/bin/../lib/hbase-hadoop-compat-1.2.0-cdh5.13.0-tests.jar:/usr/lib/hbase/bin/../lib/hbase-it-1.2.0-cdh5.13.0.jar:/usr/lib/hbase/bin/../lib/hbase-it-1.2.0-cdh5.13.0-tests.jar:/usr/lib/hbase/bin/../lib/hbase-prefix-tree-1.2.0-cdh5.13.0.jar:/usr/lib/hbase/bin/../lib/hbase-procedure-1.2.0-cdh5.13.0.jar:/usr/lib/hbase/bin/../lib/hbase-protocol-1.2.0-cdh5.13.0.jar:/usr/lib/hbase/bin/../lib/hbase-resource-bundle-1.2.0-cdh5.13.0.jar:/usr/lib/hbase/bin/../lib/hbase-rest-1.2.0-cdh5.13.0.jar:/usr/lib/hbase/bin/../lib/hbase-rsgroup-1.2.0-cdh5.13.0.jar:/usr/lib/hbase/bin/../lib/hbase-rsgroup-1.2.0-cdh5.13.0-tests.jar:/usr/lib/hbase/bin/../lib/hbase-server-1.2.0-cdh5.13.0.jar:/usr/lib/hbase/bin/../lib/hbase-server-1.2.0-cdh5.13.0-tests.jar:/usr/lib/hbase/bin/../lib/hbase-shell-1.2.0-cdh5.13.0.jar:/usr/lib/hbase/bin/../lib/hbase-spark-1.2.0-cdh5.13.0.jar:/usr/lib/hbase/bin/../lib/hbase-thrift-1.2.0-cdh5.13.0.jar:/usr/lib/hbase/bin/../lib/high-scale-lib-1.1.1.jar:/usr/lib/hbase/bin/../lib/hsqldb-1.8.0.10.jar:/usr/lib/hbase/bin/../lib/htrace-core-3.2.0-incubating.jar:/usr/lib/hbase/bin/../lib/htrace-core4-4.0.1-incubating.jar:/usr/lib/hbase/bin/../lib/htrace-core.jar:/usr/lib/hbase/bin/../lib/httpclient-4.2.5.jar:/usr/lib/hbase/bin/../lib/httpcore-4.2.5.jar:/usr/lib/hbase/bin/../lib/jackson-annotations-2.2.3.jar:/usr/lib/hbase/bin/../lib/jackson-core-2.2.3.jar:/usr/lib/hbase/bin/../lib/jackson-core-asl-1.8.8.jar:/usr/lib/hbase/bin/../lib/jackson-databind-2.2.3.jar:/usr/lib/hbase/bin/../lib/jackson-jaxrs-1.8.8.jar:/usr/lib/hbase/bin/../lib/jackson-mapper-asl-1.8.8.jar:/usr/lib/hbase/bin/../lib/jackson-xc-1.8.8.jar:/usr/lib/hbase/bin/../lib/jamon-runtime-2.4.1.jar:/usr/lib/hbase/bin/../lib/jasper-compiler-5.5.23.jar:/usr/lib/hbase/bin/../lib/jasper-runtime-5.5.23.jar:/usr/lib/hbase/bin/../lib/java-xmlbuilder-0.4.jar:/usr/lib/hbase/bin/../lib/jaxb-api-2.1.jar:/usr/lib/hbase/bin/../lib/jaxb-impl-2.2.3-1.jar:/usr/lib/hbase/bin/../lib/jcodings-1.0.8.jar:/usr/lib/hbase/bin/../lib/jersey-client-1.9.jar:/usr/lib/hbase/bin/../lib/jersey-core-1.9.jar:/usr/lib/hbase/bin/../lib/jersey-json-1.9.jar:/usr/lib/hbase/bin/../lib/jersey-server-1.9.jar:/usr/lib/hbase/bin/../lib/jets3t-0.9.0.jar:/usr/lib/hbase/bin/../lib/jettison-1.3.3.jar:/usr/lib/hbase/bin/../lib/jetty-6.1.26.cloudera.4.jar:/usr/lib/hbase/bin/../lib/jetty-sslengine-6.1.26.cloudera.4.jar:/usr/lib/hbase/bin/../lib/jetty-util-6.1.26.cloudera.4.jar:/usr/lib/hbase/bin/../lib/joni-2.1.2.jar:/usr/lib/hbase/bin/../lib/jruby-cloudera-1.0.0.jar:/usr/lib/hbase/bin/../lib/jsch-0.1.42.jar:/usr/lib/hbase/bin/../lib/jsp-2.1-6.1.14.jar:/usr/lib/hbase/bin/../lib/jsp-api-2.1-6.1.14.jar:/usr/lib/hbase/bin/../lib/jsp-api-2.1.jar:/usr/lib/hbase/bin/../lib/junit-4.12.jar:/usr/lib/hbase/bin/../lib/leveldbjni-all-1.8.jar:/usr/lib/hbase/bin/../lib/libthrift-0.9.3.jar:/usr/lib/hbase/bin/../lib/log4j-1.2.17.jar:/usr/lib/hbase/bin/../lib/metrics-core-2.2.0.jar:/usr/lib/hbase/bin/../lib/netty-all-4.0.23.Final.jar:/usr/lib/hbase/bin/../lib/paranamer-2.3.jar:/usr/lib/hbase/bin/../lib/protobuf-java-2.5.0.jar:/usr/lib/hbase/bin/../lib/servlet-api-2.5-6.1.14.jar:/usr/lib/hbase/bin/../lib/servlet-api-2.5.jar:/usr/lib/hbase/bin/../lib/slf4j-api-1.7.5.jar:/usr/lib/hbase/bin/../lib/slf4j-log4j12.jar:/usr/lib/hbase/bin/../lib/snappy-java-1.0.4.1.jar:/usr/lib/hbase/bin/../lib/spymemcached-2.11.6.jar:/usr/lib/hbase/bin/../lib/xmlenc-0.52.jar:/usr/lib/hbase/bin/../lib/xz-1.0.jar:/usr/lib/hbase/bin/../lib/zookeeper.jar:/etc/hadoop/conf:/usr/lib/hadoop/lib/curator-framework-2.7.1.jar:/usr/lib/hadoop/lib/jaxb-impl-2.2.3-1.jar:/usr/lib/hadoop/lib/jettison-1.1.jar:/usr/lib/hadoop/lib/jersey-core-1.9.jar:/usr/lib/hadoop/lib/jersey-server-1.9.jar:/usr/lib/hadoop/lib/jsp-api-2.1.jar:/usr/lib/hadoop/lib/commons-httpclient-3.1.jar:/usr/lib/hadoop/lib/jetty-6.1.26.cloudera.4.jar:/usr/lib/hadoop/lib/xz-1.0.jar:/usr/lib/hadoop/lib/commons-net-3.1.jar:/usr/lib/hadoop/lib/snappy-java-1.0.4.1.jar:/usr/lib/hadoop/lib/azure-data-lake-store-sdk-2.2.3.jar:/usr/lib/hadoop/lib/commons-digester-1.8.jar:/usr/lib/hadoop/lib/guava-11.0.2.jar:/usr/lib/hadoop/lib/commons-logging-1.1.3.jar:/usr/lib/hadoop/lib/jsr305-3.0.0.jar:/usr/lib/hadoop/lib/hamcrest-core-1.3.jar:/usr/lib/hadoop/lib/asm-3.2.jar:/usr/lib/hadoop/lib/servlet-api-2.5.jar:/usr/lib/hadoop/lib/httpcore-4.2.5.jar:/usr/lib/hadoop/lib/xmlenc-0.52.jar:/usr/lib/hadoop/lib/curator-recipes-2.7.1.jar:/usr/lib/hadoop/lib/jackson-jaxrs-1.8.8.jar:/usr/lib/hadoop/lib/stax-api-1.0-2.jar:/usr/lib/hadoop/lib/commons-beanutils-core-1.8.0.jar:/usr/lib/hadoop/lib/jasper-runtime-5.5.23.jar:/usr/lib/hadoop/lib/commons-configuration-1.6.jar:/usr/lib/hadoop/lib/commons-math3-3.1.1.jar:/usr/lib/hadoop/lib/apacheds-kerberos-codec-2.0.0-M15.jar:/usr/lib/hadoop/lib/junit-4.11.jar:/usr/lib/hadoop/lib/commons-collections-3.2.2.jar:/usr/lib/hadoop/lib/jersey-json-1.9.jar:/usr/lib/hadoop/lib/httpclient-4.2.5.jar:/usr/lib/hadoop/lib/gson-2.2.4.jar:/usr/lib/hadoop/lib/commons-beanutils-1.9.2.jar:/usr/lib/hadoop/lib/commons-lang-2.6.jar:/usr/lib/hadoop/lib/activation-1.1.jar:/usr/lib/hadoop/lib/commons-codec-1.4.jar:/usr/lib/hadoop/lib/jetty-util-6.1.26.cloudera.4.jar:/usr/lib/hadoop/lib/api-util-1.0.0-M20.jar:/usr/lib/hadoop/lib/java-xmlbuilder-0.4.jar:/usr/lib/hadoop/lib/api-asn1-api-1.0.0-M20.jar:/usr/lib/hadoop/lib/commons-el-1.0.jar:/usr/lib/hadoop/lib/commons-cli-1.2.jar:/usr/lib/hadoop/lib/jackson-xc-1.8.8.jar:/usr/lib/hadoop/lib/aws-java-sdk-bundle-1.11.134.jar:/usr/lib/hadoop/lib/paranamer-2.3.jar:/usr/lib/hadoop/lib/netty-3.10.5.Final.jar:/usr/lib/hadoop/lib/curator-client-2.7.1.jar:/usr/lib/hadoop/lib/jaxb-api-2.2.2.jar:/usr/lib/hadoop/lib/jsch-0.1.42.jar:/usr/lib/hadoop/lib/avro.jar:/usr/lib/hadoop/lib/commons-io-2.4.jar:/usr/lib/hadoop/lib/mockito-all-1.8.5.jar:/usr/lib/hadoop/lib/protobuf-java-2.5.0.jar:/usr/lib/hadoop/lib/slf4j-log4j12.jar:/usr/lib/hadoop/lib/htrace-core4-4.0.1-incubating.jar:/usr/lib/hadoop/lib/slf4j-api-1.7.5.jar:/usr/lib/hadoop/lib/hue-plugins-3.9.0-cdh5.13.0.jar:/usr/lib/hadoop/lib/jets3t-0.9.0.jar:/usr/lib/hadoop/lib/log4j-1.2.17.jar:/usr/lib/hadoop/lib/jasper-compiler-5.5.23.jar:/usr/lib/hadoop/lib/zookeeper.jar:/usr/lib/hadoop/lib/logredactor-1.0.3.jar:/usr/lib/hadoop/lib/commons-compress-1.4.1.jar:/usr/lib/hadoop/lib/apacheds-i18n-2.0.0-M15.jar:/usr/lib/hadoop/.//parquet-test-hadoop2.jar:/usr/lib/hadoop/.//parquet-pig.jar:/usr/lib/hadoop/.//parquet-thrift.jar:/usr/lib/hadoop/.//hadoop-nfs-2.6.0-cdh5.13.0.jar:/usr/lib/hadoop/.//parquet-generator.jar:/usr/lib/hadoop/.//parquet-common.jar:/usr/lib/hadoop/.//hadoop-annotations.jar:/usr/lib/hadoop/.//parquet-scrooge_2.10.jar:/usr/lib/hadoop/.//hadoop-auth.jar:/usr/lib/hadoop/.//parquet-format.jar:/usr/lib/hadoop/.//hadoop-nfs.jar:/usr/lib/hadoop/.//parquet-format-javadoc.jar:/usr/lib/hadoop/.//parquet-format-sources.jar:/usr/lib/hadoop/.//hadoop-common-2.6.0-cdh5.13.0-tests.jar:/usr/lib/hadoop/.//hadoop-azure-datalake.jar:/usr/lib/hadoop/.//parquet-hadoop-bundle.jar:/usr/lib/hadoop/.//hadoop-common-tests.jar:/usr/lib/hadoop/.//parquet-column.jar:/usr/lib/hadoop/.//hadoop-common.jar:/usr/lib/hadoop/.//parquet-scala_2.10.jar:/usr/lib/hadoop/.//hadoop-annotations-2.6.0-cdh5.13.0.jar:/usr/lib/hadoop/.//parquet-protobuf.jar:/usr/lib/hadoop/.//parquet-encoding.jar:/usr/lib/hadoop/.//parquet-hadoop.jar:/usr/lib/hadoop/.//hadoop-aws.jar:/usr/lib/hadoop/.//hadoop-common-2.6.0-cdh5.13.0.jar:/usr/lib/hadoop/.//parquet-pig-bundle.jar:/usr/lib/hadoop/.//parquet-avro.jar:/usr/lib/hadoop/.//hadoop-aws-2.6.0-cdh5.13.0.jar:/usr/lib/hadoop/.//parquet-jackson.jar:/usr/lib/hadoop/.//hadoop-auth-2.6.0-cdh5.13.0.jar:/usr/lib/hadoop/.//parquet-cascading.jar:/usr/lib/hadoop/.//hadoop-azure-datalake-2.6.0-cdh5.13.0.jar:/usr/lib/hadoop/.//parquet-tools.jar:/usr/lib/hadoop-hdfs/./:/usr/lib/hadoop-hdfs/lib/jersey-core-1.9.jar:/usr/lib/hadoop-hdfs/lib/jersey-server-1.9.jar:/usr/lib/hadoop-hdfs/lib/jsp-api-2.1.jar:/usr/lib/hadoop-hdfs/lib/commons-daemon-1.0.13.jar:/usr/lib/hadoop-hdfs/lib/jetty-6.1.26.cloudera.4.jar:/usr/lib/hadoop-hdfs/lib/guava-11.0.2.jar:/usr/lib/hadoop-hdfs/lib/commons-logging-1.1.3.jar:/usr/lib/hadoop-hdfs/lib/jackson-mapper-asl-1.8.8.jar:/usr/lib/hadoop-hdfs/lib/jsr305-3.0.0.jar:/usr/lib/hadoop-hdfs/lib/leveldbjni-all-1.8.jar:/usr/lib/hadoop-hdfs/lib/asm-3.2.jar:/usr/lib/hadoop-hdfs/lib/xercesImpl-2.9.1.jar:/usr/lib/hadoop-hdfs/lib/servlet-api-2.5.jar:/usr/lib/hadoop-hdfs/lib/jackson-core-asl-1.8.8.jar:/usr/lib/hadoop-hdfs/lib/xmlenc-0.52.jar:/usr/lib/hadoop-hdfs/lib/jasper-runtime-5.5.23.jar:/usr/lib/hadoop-hdfs/lib/xml-apis-1.3.04.jar:/usr/lib/hadoop-hdfs/lib/commons-lang-2.6.jar:/usr/lib/hadoop-hdfs/lib/commons-codec-1.4.jar:/usr/lib/hadoop-hdfs/lib/jetty-util-6.1.26.cloudera.4.jar:/usr/lib/hadoop-hdfs/lib/commons-el-1.0.jar:/usr/lib/hadoop-hdfs/lib/commons-cli-1.2.jar:/usr/lib/hadoop-hdfs/
Cloudera Live : Welco...
cloudera@quickstart:~
cloudera@quickstart:...
ACD_BDD2.6_Session_...
ACD_BDD2.6_Session_...
Untitled 1 - LibreOffice...


Applications Places System
Mon Jul 2, 5:45 AM cloudera
cloudera@quickstart:~
File Edit View Search Terminal Help
hbase(main):003:0> scan 'bulktable'
ROW COLUMN+CELL
1 column=cf1:name, timestamp=1530535335100, value=amit
1 column=cf2:exp, timestamp=1530535335100, value=4
2 column=cf1:name, timestamp=1530535335100, value=Girija
2 column=cf2:exp, timestamp=1530535335100, value=3
3 column=cf1:name, timestamp=1530535335100, value=Jatin
3 column=cf2:exp, timestamp=1530535335100, value=5
4 column=cf1:name, timestamp=1530535335100, value=Swati
4 column=cf2:exp, timestamp=1530535335100, value=3
4 row(s) in 0.2480 seconds
hbase(main):004:0>
Cloudera Live : Welco...
cloudera@quickstart:~
cloudera@quickstart:...
ACD_BDD2.6_Session_...
ACD_BDD2.6_Session_...
Untitled 1 - LibreOffice...