ZeyoBron Data Enginnering (HDFS, Sqoop, Hive)

Applications Download Links:

7z (Windows Users) --- Download

==========

https://www.7-zip.org/a/7z2201-x64.exe

==========

Oracle Virtual Box --- Download

==========

Window Users

https://download.virtualbox.org/virtualbox/6.1.18/VirtualBox-6.1.18-142142-Win.exe

Ubuntu Users

16 Version- https://download.virtualbox.org/virtualbox/6.1.20/virtualbox-6.1_6.1.20-143896~Ubuntu~xenial_amd64.deb

18 Version- https://download.virtualbox.org/virtualbox/6.1.20/virtualbox-6.1_6.1.20-143896~Ubuntu~bionic_amd64.deb

18 Version- https://download.virtualbox.org/virtualbox/6.1.20/virtualbox-6.1_6.1.20-143896~Ubuntu~eoan_amd64.deb

Mac Users

https://download.virtualbox.org/virtualbox/7.0.4/VirtualBox-7.0.4-154605-OSX.dmg

Mac m1 Chip

https://download.virtualbox.org/virtualbox/7.0.6/VirtualBox-7.0.6_BETA4-155176-macOSArm64.dmg

==========

Putty (Windows users)

==========

https://the.earth.li/~sgtatham/putty/latest/w64/putty.exe

==========

Mobaxterm Download(Windows users)

==========

Windows

https://download.mobatek.net/2212022060563542/MobaXterm_Installer_v22.1.zip

==========

Cloudera --- Any one link to download (MAC, Window, Ubuntu)

==========

Use Any One

https://drive.google.com/file/d/11VyzQsPt5xIwGVHqbDwOcjMj63WiGfKa/view?usp=sharing
https://drive.google.com/file/d/1VYNYI6Cmj0_Ko4BPGcztMj7PRAWL7XH6/view?usp=sharing
https://drive.google.com/file/d/1d538E823H4GuTVWcCLUJC6xpkYWsW_WO/view?usp=sharing
https://drive.google.com/file/d/1ebMjbwzC4Thrb-LnmpihHIpa4TBFL_qH/view?usp=sharing
https://zeyobucketathena.s3.amazonaws.com/Sai_Zeyobron_Powered_Cloudera.zip
https://drive.google.com/file/d/1ZVZgJnmCyyHmM2AoEHjlAUx8H5ResG8V/view?usp=sharing
https://drive.google.com/file/d/1Hx5fAeiAXd2aRe046yK1755zh2csdq9G/view?usp=sharing
https://drive.google.com/file/d/11sEiI7f5jM4v7-O8QZBw6sSC6yJDn1Nh/view?usp=sharing
https://drive.google.com/file/d/1u7F9hLdnWAk0DiflDnymjEDFB21vA2kR/view?usp=sharing
https://drive.google.com/file/d/1T3y3qXXEcnSDcg5WFxZElTc8n5vhJcEQ/view?usp=sharing
https://drive.google.com/file/d/1ycm76Y3Pji5XUBIsdQquR2ToBZPghrzO/view?usp=sharing
https://drive.google.com/file/d/1yjNZD4m_yLlHmfkk0XjaQVSazWqdQP5f/view?usp=sharing
https://drive.google.com/file/d/1DPSfwelNTK4vFqccipeSmNGWCuCSEAIe/view?usp=sharing
https://drive.google.com/file/d/1-4xh-vo2kQyy_koPXesVB8E22NT7kASF/view?usp=sharing
https://drive.google.com/file/d/1TJLcu1o1ymKz-2Dz8ZtiBQxUXaDfayEW/view?usp=sharing
https://drive.google.com/file/d/1mTlcm71Vtm9WoIcTYRPSrwP0OSxVCtyy/view?usp=sharing
https://drive.google.com/file/d/1RxCwKrV1cLeiIAr95T1d2VH-qoJULLBB/view?usp=sharing
https://drive.google.com/file/d/19jUP51V_95mQ1aAZVCa8dGFTP6p1t8nc/view?usp=sharing
https://drive.google.com/file/d/1Cfow1iNyy_Nveo72SWyR9RVQLFkXRgHF/view?usp=sharing
https://drive.google.com/file/d/15RNIYynf_gSSC2v1nvCMR6OxE-AA3xR5/view?usp=sharing

Cloudera installation Video

==========

Installation Video

https://youtu.be/xsTbkZ8r1qo

Windows 10 UEFI

https://www.youtube.com/watch?v=MOuTxfzCvMY

Windows 10 -- Legacy Bios Settings

https://www.youtube.com/watch?v=wlfS0UEMUqc

Windows 11 --

https://www.youtube.com/watch?v=UMo-is3fjPI

https://www.youtube.com/watch?v=0WuFGCn036E

https://www.youtube.com/watch?v=t8f-zw_wcWM

Putty Installation video.

https://youtu.be/QACCTS9ioTQ

Task 1 ---- Linux Basics

==========

pwd

ls –al

ls –ll

cd

mkdir /home/cloudera/zeyo1

mkdir /home/cloudera/zeyo1/zeyo11

mkdir /home/cloudera/zeyo2

mkdir /home/cloudera/zeyo2/zeyo22

touch /home/cloudera/zeyo1/zeyo11/zeyofile

ls /home/cloudera/zeyo1/zeyo11/

Task 2

==========

echo zeyobron>zeyofile

cat zeyofile

==========

Hadoop Basics

==========

```
hadoop dfsadmin -safemode leave
hadoop fs -ls /user/cloudera/
hadoop fs -mkdir /user/cloudera/zhdir
hadoop fs -ls /user/cloudera/
hadoop fs -rmdir /user/cloudera/zhdir
hadoop fs -ls /user/cloudera/

cd
echo zeyobron > /home/cloudera/zeyofile
hadoop fs -mkdir /user/cloudera/zh
hadoop fs -ls /user/cloudera/
hadoop fs -put /home/cloudera/zeyofile /user/cloudera/zh/
hadoop fs -ls /user/cloudera/zh
hadoop fs -cat /user/cloudera/zh/zeyofile
rm /home/cloudera/zeyofile   (give y if pop up)
hadoop fs -get /user/cloudera/zh/zeyofile /home/cloudera/
ll
cat zeyofile

hadoop fs -put /home/cloudera/data.avro /user/cloudera/
hadoop fs -text /user/cloudera/data.avro
```

==========

SQOOP

==========

```
sqoop import --connect jdbc:mysql://zeyodb.cveqgaujeiwd.ap-south-1.rds.amazonaws.com/zeyodb --username root --password Aditya908 --table ztab --m 1 --delete-target-dir --target-dir /user/cloudera/firstimport

(Same import statement)
sqoop import
--connect jdbc:mysql://zeyodb.cveqgaujeiwd.ap-south-1.rds.amazonaws.com/zeyodb
--username root
--password Aditya908
--table ztab
--m 1
--delete-target-dir
--target-dir /user/cloudera/firstimport

hadoop fs -ls /user/cloudera/firstimport
hadoop fs -cat /user/cloudera/firstimport/part-m-00000
-----------------------------------------------------------------------

hadoop dfsadmin -safemode leave
cd
echo zeyobron>/home/cloudera/zeyofile
hadoop fs -touchz /user/cloudera/hifile  (create a file)
hadoop fs -appendToFile /home/cloudera/zeyofile /user/cloudera/hifile
hadoop fs -cat /user/cloudera/hifile
-----------------------------------------------------------------------

hadoop dfsadmin -safemode leave
mysql -uroot -pcloudera
create database if not exists zeyodb;
use zeyodb;
drop table if exists zeyotab;
quit
```

==========
Where imports
==========

```
sqoop import --connect jdbc:mysql://localhost/zeyodb --username root --password cloudera --table zeyotab --m 1 --where "city='chennai'" --delete-target-dir --target-dir /user/cloudera/chimport

hadoop fs -ls /user/cloudera/chimport
hadoop fs -cat /user/cloudera/chimport/part-m-00000
```

==========
query imports
==========

```
hadoop dfsadmin -safemode leave
mysql -uroot -pcloudera
create database if not exists zeyodb;
use zeyodb;
drop table if exists z1;
create table z1 (id int,name varchar(100),city varchar(100),mode varchar(100));
insert into z1 values(1,'zeyo','chennai','cash');
insert into z1 values(2,'hema','hyderabad','credit');
insert into z1 values(3,'raji','chennai','cash');
insert into z1 values(4,'viru','delhi','credit');
select * from z1;

drop table if exists z2;
create table z2 (id int,product varchar(100));
insert into z2 values(1,'cookies');
insert into z2 values(2,'mobile');
insert into z2 values(3,'laptop');
insert into z2 values(4,'mouse');
select * from z2;
```

```
select a.*,b.product from z1 a join z2 b on a.id=b.id;
quit

sqoop import --connect jdbc:mysql://localhost/zeyodb --username root --password
cloudera --m 1 --query "select a.*,b.product from z1 a join z2 b on a.id=b.id where
\$CONDITIONS" --delete-target-dir --target-dir /user/cloudera/qimport

hadoop fs -ls /user/cloudera/qimport
hadoop fs -cat /user/cloudera/qimport/part-m-00000
-----------------------------------------------------------------------

======================
*Incremental Imports*
======================

hadoop dfsadmin -safemode leave
mysql -uroot -pcloudera
create database if not exists zeyodb;
use zeyodb;
drop table if exists zeyotab;
create table zeyotab (id int,name varchar(100),city varchar(100),mode varchar(100));
insert into zeyotab values(1,'zeyo','chennai','cash');
insert into zeyotab values(2,'hema','hyderabad','credit');
insert into zeyotab values(3,'raji','chennai','cash');
insert into zeyotab values(4,'viru','delhi','credit');
select * from zeyotab;
quit

=============
*Sqoop Import*
=============

sqoop import --connect jdbc:mysql://localhost/zeyodb --username root --password
cloudera --table zeyotab --m 1 --delete-target-dir --target-dir
/user/cloudera/inimport
```

==========
First Import data
==========
hadoop fs -ls /user/cloudera/inimport
hadoop fs -cat /user/cloudera/inimport/part-m-00000

==========
Add 2 more records in SQL
==========

mysql -uroot -pcloudera
use zeyodb;
insert into zeyotab values(5,'ramu','chennai','cash');
insert into zeyotab values(6,'vasu','delhi','credit');
select * from zeyotab;
quit

==========
SQOOP Incremental Import
==========

sqoop import --connect jdbc:mysql://localhost/zeyodb --username root --password cloudera --table zeyotab --m 1 --target-dir /user/cloudera/inimport --incremental append --check-column id --last-value 4

==========
NEW File with only 2 records
==========

hadoop fs -ls /user/cloudera/inimport
hadoop fs -cat /user/cloudera/inimport/part-m-00001

==========

Sqoop Incremental Jobs

==========

```
hadoop dfsadmin -safemode leave
mysql -uroot -pcloudera
create database if not exists zeyodb;
use zeyodb;
drop table if exists zeyojob;
create table zeyojob (id int,name varchar(100),city varchar(100),mode varchar(100));
insert into zeyojob values(1,'zeyo','chennai','cash');
insert into zeyojob values(2,'hema','hyderabad','credit');
insert into zeyojob values(3,'raji','chennai','cash');
insert into zeyojob values(4,'viru','delhi','credit');
select * from zeyojob;
quit
```

==========

Password file creation

==========

```
cd
echo -n cloudera>passfile
```

(-n for no spaces and no new lines after cloudera)

```
sqoop job --delete zjob
sqoop job --create zjob -- import --connect jdbc:mysql://localhost/zeyodb --
username root --password-file file:///home/cloudera/passfile --table zeyojob --m 1
--target-dir /user/cloudera/jobimport --incremental append --check-column id
--last-value 0

sqoop job --exec zjob

hadoop fs -ls /user/cloudera/jobimport
hadoop fs -cat /user/cloudera/jobimport/part-m-00000
```

```
=========================
*Add 2 more records in SQL*
=========================

mysql -uroot -pcloudera
use zeyodb;
insert into zeyojob values(5,'ramu','chennai','cash');
insert into zeyojob values(6,'vasu','delhi','credit');
select * from zeyojob;
quit

===============
Execute job
===============

sqoop job --exec zjob

hadoop fs -ls /user/cloudera/jobimport
hadoop fs -cat /user/cloudera/jobimport/part-m-00001



======================
Cloud Import
======================

mysql -uroot -pcloudera
create database if not exists zdb;
use zdb;

drop table cust;
create table cust(id int,name varchar(100));
insert into cust value(1,'zeyo');
insert into cust value(2,'analytics');
select * from cust;
quit
```

```
sqoop import -Dfs.s3a.access.key=AKIA2TITMOYY2U3PUBBW
-Dfs.s3a.secret.key=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
-Dfs.s3a.endpoint=s3.ap-south-1.amazonaws.com
--connect jdbc:mysql://localhost/zdb --username root --password cloudera
--table cust --m 1  --target-dir  s3a://zeyo36buck/<URNAME>
-----------------------------------------------------------------


====================
Cloudera File formats
====================
Task 1 --

mysql -uroot -pcloudera
create database if not exists ad;
use ad;
drop table ttab;
create table ttab(id int,name varchar(100),amount int);
insert into ttab values(1,'zeyo',40);
insert into ttab values(2,'vasu',50);
insert into ttab values(3,'rani',70);
select * from ttab;
quit

Task 1 – Parquetfile format

sqoop import --connect jdbc:mysql://localhost/ad --username root --password
cloudera --table ttab --m 1 --delete-target-dir --target-dir /user/cloudera/pdata
--as-parquetfile

hadoop fs -ls /user/cloudera/pdata
hadoop fs -cat /user/cloudera/pdata/*

press ctrl+c in keyboard
-----------------------------------------------------------------
```

Task 2 – Sequence file format

```
sqoop import --connect jdbc:mysql://localhost/ad --username root --password
cloudera --table ttab --m 1 --delete-target-dir --target-dir /user/cloudera/sdata
--as-sequencefile

hadoop fs -ls /user/cloudera/sdata
hadoop fs -cat /user/cloudera/sdata/*

press ctrl+c in keyboard
```

Task 3 – Avrodata file format

```
sqoop import --connect jdbc:mysql://localhost/map --username root --password
cloudera --table mtab --m 1 --delete-target-dir --target-dir /user/cloudera/adir
--as-avrodatafile

hadoop fs -ls /user/cloudera/sdata
hadoop fs -cat /user/cloudera/sdata/*

press ctrl+c in keyboard
```

```
====================
Cloudera Multi mappers
====================

mysql -uroot -pcloudera

drop database map;
create database map;
use map;
drop table mtab;
create table mtab(id int,name varchar(100),amount int);
insert into mtab values(1,'zeyo',40);
insert into mtab values(2,'vasu',50);
insert into mtab values(3,'rani',70);
insert into mtab values(4,'raji',40);
insert into mtab values(5,'viru',50);
insert into mtab values(6,'raj',70);
insert into mtab values(7,'vinu',40);
insert into mtab values(8,'ajit',50);
insert into mtab values(9,'raki',70);
insert into mtab values(10,'rinu',40);
insert into mtab values(11,'dini',50);
insert into mtab values(12,'div',70);

select * from mtab;
quit

====================
1 Mappers
====================
sqoop import --connect jdbc:mysql://localhost/map --username root --password
cloudera --table mtab --m 1 --delete-target-dir --target-dir /user/cloudera/mtab

hadoop fs -ls /user/cloudera/mtab
hadoop fs -cat /user/cloudera/mtab/part-m-00000
```

==========

2 Mappers

==========

```
sqoop import --connect jdbc:mysql://localhost/map --username root --password cloudera --table mtab --m 2 --split-by id --delete-target-dir --target-dir /user/cloudera/mtab2

hadoop fs -ls /user/cloudera/mtab2
hadoop fs -cat /user/cloudera/mtab2/part-m-00000
hadoop fs -cat /user/cloudera/mtab2/part-m-00001
```

==========

No Mappers (Default is 4 Mappers)

==========

```
sqoop import --connect jdbc:mysql://localhost/map --username root --password cloudera --split-by id --table mtab --delete-target-dir --target-dir /user/cloudera/mtab4

hadoop fs -ls /user/cloudera/mtab4
hadoop fs -cat /user/cloudera/mtab4/part-m-00000
hadoop fs -cat /user/cloudera/mtab4/part-m-00001
hadoop fs -cat /user/cloudera/mtab4/part-m-00002
hadoop fs -cat /user/cloudera/mtab4/part-m-00003
```

```
======================
Cloudera staging Exports
======================
mysql -uroot -pcloudera
create database if not exists exp;
use exp;
drop table if exists ttab;
drop table if exists stab;
create table ttab(id int,name varchar(100),amount int);
create table st_ttab(id int,name varchar(100),amount int);
quit

cd
echo 1,zeyo,40>zfile
echo 2,ravi,70>>zfile
echo 3,rani,70>>zfile
hadoop fs -mkdir /user/cloudera/exdir
hadoop fs -put zfile /user/cloudera/exdir

sqoop export --connect jdbc:mysql://localhost/exp --username root --password
cloudera --table ttab --staging-table st_ttab --m 1  --export-dir
/user/cloudera/exdir

mysql -uroot -pcloudera
use exp;
select * from ttab;
select * from st_ttab;
quit
```

==========
AVRO Task Cloudera
==========

mysql -uroot -pcloudera

```
drop database if exists map;
create database  if not exists map;
use map;
drop table if exists mtab;
create table mtab(id int,name varchar(100),amount int);
insert into mtab values(1,'zeyo',40);
insert into mtab values(2,'vasu',50);
insert into mtab values(3,'rani',70);
select * from mtab;
quit
```

```
sqoop import --connect jdbc:mysql://localhost/map --username root --password
cloudera --table mtab --m 1 --delete-target-dir --target-dir /user/cloudera/adir
--as-avrodatafile
```

```
hadoop fs -ls /user/cloudera/adir
hadoop fs -cat /user/cloudera/adir/*
```

click ctrl+c in keyboard

=========
HIVE
=========

Create hive avro table on top of avro imported data
==========

hadoop dfsadmin -safemode leave

hive ---> go inside hive
drop database if exists testdb;
create database testdb;
!hadoop fs -ls /user/hive/warehouse/;
use testdb;
create table testtab(id int);
!hadoop fs -ls /user/hive/warehouse/testdb.db;

hadoop dfsadmin -safemode leave

cd
hadoop fs -mkdir /user/cloudera/tdir
hive

create database if not exists odb;
use odb;
create table o1(id int);
describe formatted o1;
!hadoop fs -ls /user/hive/warehouse/odb.db; --- u will see o1 directory

create table o2(id int);
describe formatted o2;
!hadoop fs -ls /user/hive/warehouse/odb.db; --- u will see o2 directory

create table o3(id int) location '/user/cloudera/tdir';
describe formatted o3; ---pointing to /user/cloudera/tdir
!hadoop fs -ls /user/hive/warehouse/odb.db; --- u will not see o3 directory

```
====================
HDFS Loads (Loading data into hive table)
====================
Data creation
====================
hadoop dfsadmin -safemode leave

cd
echo 1,Sai,I,IND>allc
echo 2,zeyo,I,IND>>allc
echo 3,Hema,K,UK>>allc
echo 4,Gomathi,K,UK>>allc
echo 5,Jai,S,US>>allc
echo 6,Swathi,S,US>>allc

====================
*data copy to HDFS*
====================
hadoop fs -put allc /user/cloudera/

hive   -- Go Inside Hive
create database if not exists zdb;
use zdb;
drop table if exists htab;

==================
*TABLE CREATION*
==================
create table htab(id int,name string,chk string,country string) row format delimited
fields terminated by ',';

===============
*hdfs data loads*
===============
load data inpath '/user/cloudera/allc' into table htab;
select * from htab;
```

==========

Types of table (Managed_table, External_table)

==========

Managed_table = when u drop the table, folder get deleted and data also.

External _table = when u drop the table, folder don't get deleted and data will remain in that folder.

```
cd
echo 1,Sai,I,IND>allc
echo 2,zeyo,I,IND>>allc
echo 3,Hema,K,UK>>allc
echo 4,Gomathi,K,UK>>allc
echo 5,Jai,S,US>>allc
echo 6,Swathi,S,US>>allc

hadoop fs -mkdir /user/cloudera/mdir
hadoop fs -mkdir /user/cloudera/edir
hadoop fs -put allc /user/cloudera/mdir
hadoop fs -put allc /user/cloudera/edir

hive --- Go Inside Hive

create database if not exists zdb;
use zdb;

create table mtab(id int,name string,chk string,country string) row format delimited fields terminated by ',' location '/user/cloudera/mdir/';  - Managed Table

select * from mtab;
describe formatted mtab;

create external table etab(id int,name string,chk string,country string) row format delimited fields terminated by ',' location '/user/cloudera/edir/';  - External Table

select * from etab;
describe formatted etab;
```

```
drop table mtab;
drop table etab;
show tables;          ==== mtab,etab tables do not exists

!hadoop fs -ls  /user/cloudera/;          ===> U will see edir not mdir


====================
Static Loads
====================
data creation
====================

hadoop dfsadmin -safemode leave

cd
echo 1,Sai,I>INDTxns
echo 2,zeyo,I>>INDTxns
echo 3,Hema,S>>USTxns
echo 4,Gomathi,S>>USTxns
echo 5,Jai,K>>UKTxns
echo 6,Swathi,K>>UKTxns

hadoop fs -rmr /user/cloudera/partdir
hadoop fs -mkdir /user/cloudera/partdir

hive -- go inside

create database if not exists partdb;
use partdb;
drop table if exists parttab;
create table parttab(id int,name string,chk string) partitioned by (country string) row
format delimited fields terminated by ',' location '/user/cloudera/partdir';

load data local inpath '/home/cloudera/INDTxns' into table parttab partition
(country='INDIA');
```

```
load data local inpath '/home/cloudera/USTxns' into table parttab partition (country='USA');
load data local inpath '/home/cloudera/UKTxns' into table parttab partition (country='UK');

!hadoop fs -ls /user/cloudera/partdir;
!hadoop fs -ls /user/cloudera/partdir/country=INDIA;
!hadoop fs -ls /user/cloudera/partdir/country=USA;



====================
Static Insert and Dynamic
====================
data creation
====================

hadoop dfsadmin -safemode leave

cd
echo 1,Sai,I,IND>allc
echo 2,zeyo,I,IND>>allc
echo 3,Hema,K,UK>>allc
echo 4,Gomathi,K,UK>>allc
echo 5,Jai,S,US>>allc
echo 6,Swathi,S,US>>allc

hadoop fs -rmr /user/cloudera/srcdir
hadoop fs -mkdir /user/cloudera/srcdir
hadoop fs -rmr /user/cloudera/sidir
hadoop fs -mkdir /user/cloudera/sidir

hive -- go inside
```

```
============================
Create database and Source Table
============================

create database if not exists partdb;
use partdb;
drop table if exists srctab;
create table srctab(id int,name string,chk string,country string) row format delimited
fields terminated by ',' location '/user/cloudera/srcdir';

================================
Load data to source table
================================
load data local inpath '/home/cloudera/allc' into table srctab;

================================
Target table for Static Partition and insert the data
================================
drop table if exists sitab;
create table sitab(id int,name string,chk string) partitioned by (country string) row
format delimited fields terminated by ',' location '/user/cloudera/sidir';

insert into sitab partition(country='USA') select id, name, chk from srctab where
country='US';

!hadoop fs -ls /user/cloudera/sidir;
!hadoop fs -ls /user/cloudera/sidir/country=USA;
```

==========

Dynamic partitions and insert

==========

```
drop table if exists dyntab;
create table dyntab(id int,name string,chk string) partitioned by (country string) row format delimited fields terminated by ',' location '/user/cloudera/dyndir';  -- dynamic table

set hive.exec.dynamic.partition.mode=nonstrict;
insert into dyntab partition(country) select id,name,chk,country from srctab;  -- dynamic insert into table,
```

*** Partitioned column should be in last while selecting columns, while inserting data into partition tables. Eg: so country is in last in select statement in above query.

```
!hadoop fs -ls /user/cloudera/dyndir;
```

==========

Static Loads --- Repair Table Task (if only error occurs)

==========

data creation

==========

```
hadoop dfsadmin -safemode leave

cd
echo 1,Sai,I>INDTxns
echo 2,zeyo,I>>INDTxns
echo 3,Hema,K>UKTxns
echo 4,Gomathi,K>>UKTxns
echo 5,Jai,S>USTxns
echo 6,Swathi,S>>USTxns

hadoop fs -rmr /user/cloudera/partdir
hadoop fs -mkdir /user/cloudera/partdir

hive -- go inside

create database if not exists partdb;
use partdb;
drop table if exists parttab;
create table parttab(id int,name string,chk string) partitioned by (country string) row
format delimited fields terminated by ',' location '/user/cloudera/partdir';

load data local inpath '/home/cloudera/INDTxns' into table parttab partition
(country='INDIA');
load data local inpath '/home/cloudera/USTxns' into table parttab partition
(country='USA');
load data local inpath '/home/cloudera/UKTxns' into table parttab partition
(country='UK');

select * from parttab;
drop table if exists parttab2;
```

```
create table parttab2(id int,name string,chk string) partitioned by (country string)
row format delimited fields terminated by ',' location '/user/cloudera/partdir';

select * from parttab2;
msck repair table parttab2;
select * from parttab2;

=====================================================================

=============
Cloudera avro (creating to insert data into hive table as avro file format)
=============
mysql -uroot -pcloudera

create database if not exists dataa;
use dataa;

drop table if exists atab;
create table atab(id int,name varchar(100),amount int);
insert into atab values(1,'rajesh',40);
insert into atab values(2,'vishnu',10);
insert into atab values(3,'rani',60);
select * from atab;

quit;

=============
Edge Node
=============

sqoop import --connect jdbc:mysql://localhost/dataa --username root --password
cloudera --table atab --m 1  --delete-target-dir --target-dir /user/cloudera/adir
--as-avrodatafile
```

========
Hive (how to make table to Avro format table)
========

```
create database if not exists adb;
use adb;
drop table atab;
create table atab(id int,name string,amount int) stored as avro location
'/user/cloudera/adir';
select * from atab;
```

====================

```
mysql -uroot -pcloudera
use dataa;
alter table atab drop column name;
insert into atab values(4,90);
insert into atab values(5,20);
select * from atab;

quit
```

========
Edge Node

```
sqoop import --connect jdbc:mysql://localhost/dataa --username root --password
cloudera --table atab --m 1 --target-dir /user/cloudera/adir --as-avrodatafile --
incremental append --check-column id --last-value 3
```

========

```
hive
select * from adb.atab;
```

==========

Phase 2: Scala and Spark (RDD, DataFrames ..etc)
========