NOM : ACHIR
Prénom : Ounissa

Atelier : Utiliser HIVE 2

a- Ouvrir le terminal en ligne de commande de votre VM, et aller lire le fichier de Batting.cvs situé dans le répertoire /formation/ateliers/hive/
après avoir créer la table external Batting

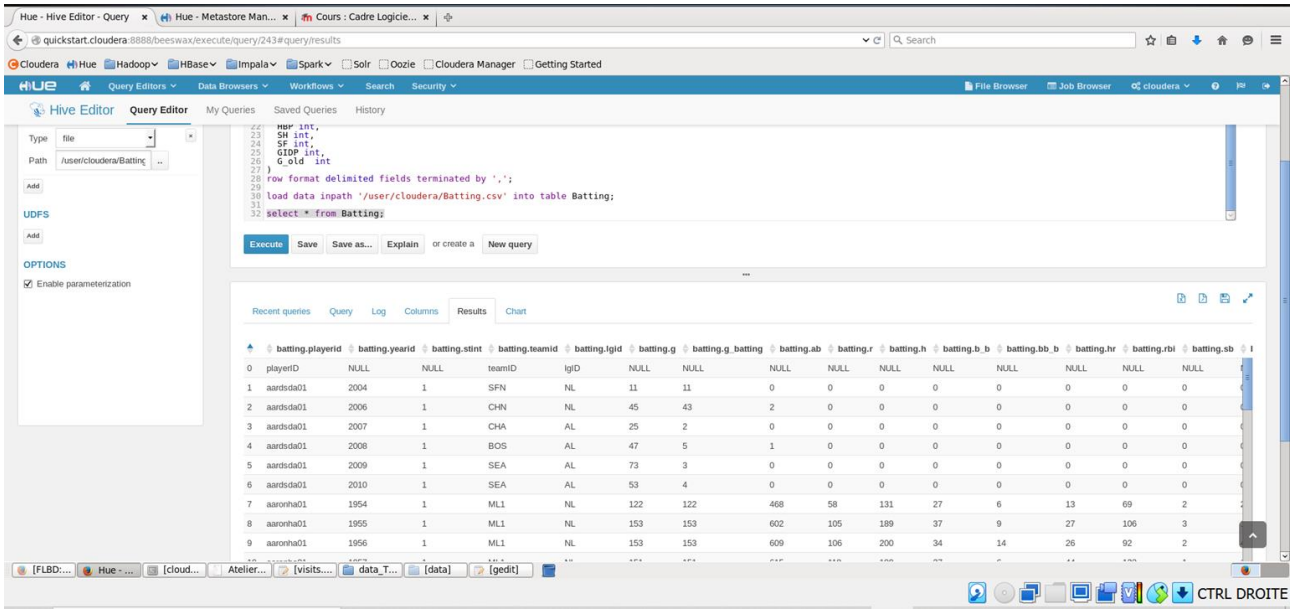
2- Copier ce fichier dans HDFS (soit via IHM soit via PIG)

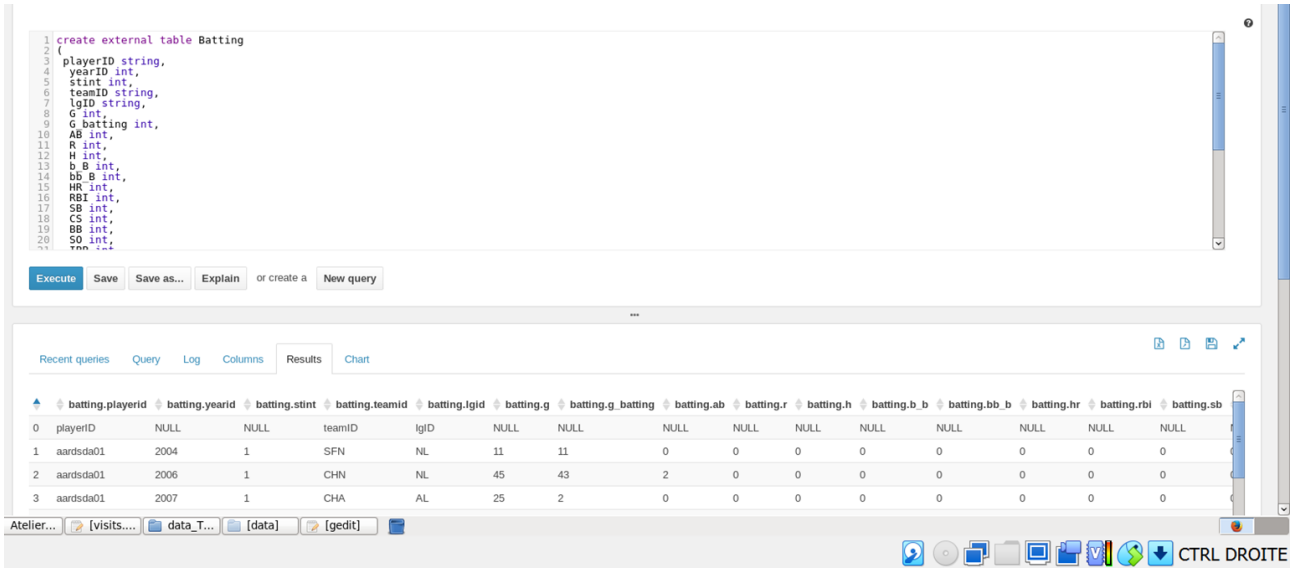
```
[cloudera@quickstart ~]$ hdfs dfs -ls  /user/cloudera/Batting.csv
-rw-r--r--   1 cloudera cloudera    6398990 2023-12-04 07:31 /user/cloudera/Batt
ing.csv
[cloudera@quickstart ~]$ hdfs dfs -ls  /user/cloudera
Found 4 items
-rw-r--r--   1 cloudera cloudera    6398990 2023-12-04 07:31 /user/cloudera/Batt
ing.csv
-rw-r--r--   1 cloudera cloudera         78 2023-11-23 02:05 /user/cloudera/name
s.txt
drwxr-xr-x   - cloudera cloudera          0 2023-10-20 10:20 /user/cloudera/test
drwxr-xr-x   - cloudera cloudera          0 2023-10-20 11:45 /user/cloudera/tmp
[cloudera@quickstart ~]$
```

hdfs dfs -copyFromLocal /home/cloudera/Desktop/data_TP2/Batting.csv /Tphive/Batting.csv

a- Créer une table externe pour visualiser le fichier des batteurs

```
[cloudera@quickstart ~]$ hdfs dfs -copyFromLocal /home/cloudera/Desktop/data_TP2/Batting.csv
/TPhive/Batting.csv
[cloudera@quickstart ~]$ hdfs dfs -ls /TPhive
Found 1 items
-rw-r--r--   1 cloudera supergroup    6398993 2023-12-04 08:24 /TPhive/Batting.csv
[cloudera@quickstart ~]$
```





b- Que s'est-il passé sous hdfs /user/hive/warehouse/
hdfs dfs -ls /user/hive/warehouse

5-

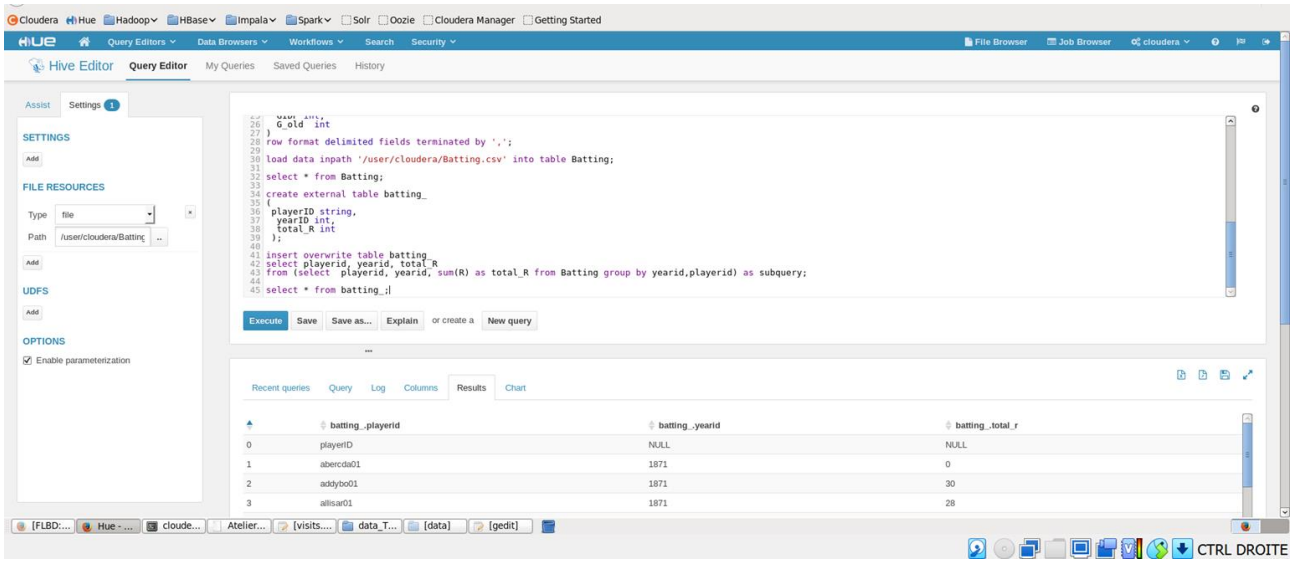
```
[cloudera@quickstart ~]$ hdfs dfs -ls /user/hive/
Found 1 items
drwxrwxrwx   - hive hive          0 2023-12-03 10:18 /user/hive/warehouse
[cloudera@quickstart ~]$ hdfs dfs -ls /user/hive/warehouse
Found 7 items
drwxrwxrwx   - cloudera hive          0 2023-11-30 02:19 /user/hive/warehouse/batting
drwxrwxrwx   - cloudera hive          0 2023-11-23 02:08 /user/hive/warehouse/hivee.db
drwxrwxrwx   - cloudera hive          0 2023-11-23 02:56 /user/hive/warehouse/tphive.db
drwxrwxrwx   - cloudera hive          0 2023-12-04 07:28 /user/hive/warehouse/tphive2.db
drwxrwxrwx   - cloudera hive          0 2023-11-23 02:35 /user/hive/warehouse/visit
drwxrwxrwx   - cloudera hive          0 2023-12-03 10:49 /user/hive/warehouse/visites
drwxrwxrwx   - cloudera hive          0 2023-11-23 02:31 /user/hive/warehouse/visits
[cloudera@quickstart ~]$
```

Identifier pour chaque année le batteur qui a réalisé le plus de Run et insérer ces informations dans la table batting:

```
create external table batting_
(playerid int,
 yearid int ,
total_R int
);

INSERT OVERWRITE TABLE batting_
SELECT playerid, yearid, total_R
FROM (
    SELECT playerid, yearid, SUM(R) as total_R
    FROM Batting
    GROUP BY yearid, playerid
) AS subquery;

select * from batting_;
```



6- Récupérer le nom du fichier de stockage, l'id du joueur dont les runs sont supérieur à 150.

```
select input_file_name(),playerid
from batting_
where total_R > 150;
```

7- Supprimer la table temp_batting:

```
drop table batting_;
```

8- Vérifier le dossier /user/hive/warehouse/.

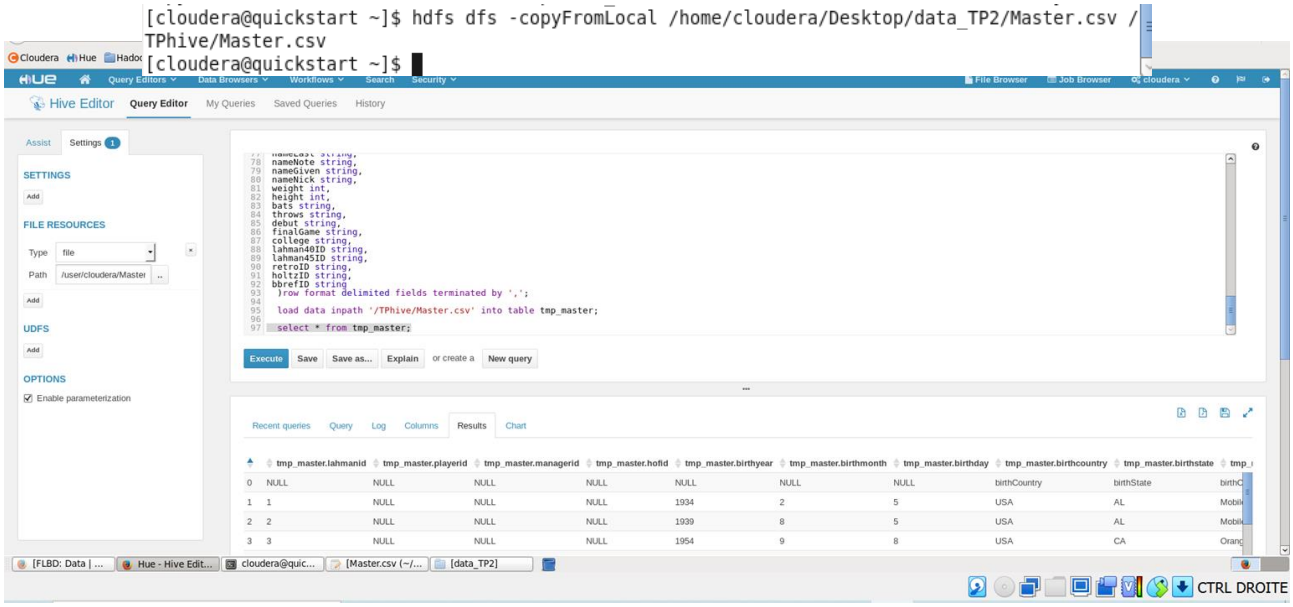
```
hdfs dfs -ls /user/hive/warehouse
```

9- Que se passe si nous recréons la table ?

10- Créer une table « local » temp_master
11- Alimenter la table temp_master avec le fichier /user/cloudera/demo/Master.csv

```
create  table tmp_master
(
lahmanID int,
playerID int,
managerID int ,
hofID int,
birthYear int,
birthMonth int,
birthDay int,
birthCountry string,
birthState string,
birthCity string,
deathYear int,
deathMonth int,
deathDay int,
deathCountry string,
```

deathState string,
deathCity string,
nameFirst string,
nameLast string,
nameNote string,
nameGiven string,
nameNick string,
weight int,
height int,
bats string,
throws string,
debut string,
finalGame string,
college string,
lahman40ID string,
lahman45ID string,
retroID string,
holtzID string,
bbrefID string
)row format delimited fields terminated by ‘,’;



12- Que se passe-t-il si nous supprimons la table temp master.csv ? Partitions

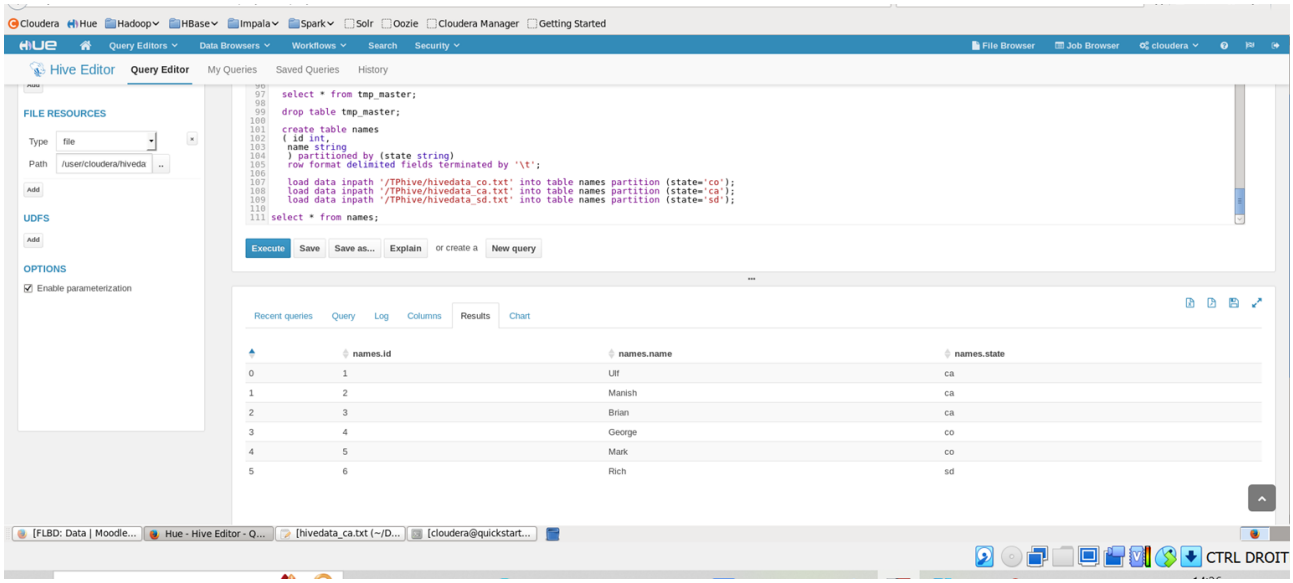
drop table tmp_master ;

13- Créer une table names qui contient une colonne id (entier) et une colonne name (text) et une colonne state (texte). Cette table sera partitionnée par la colonne state.

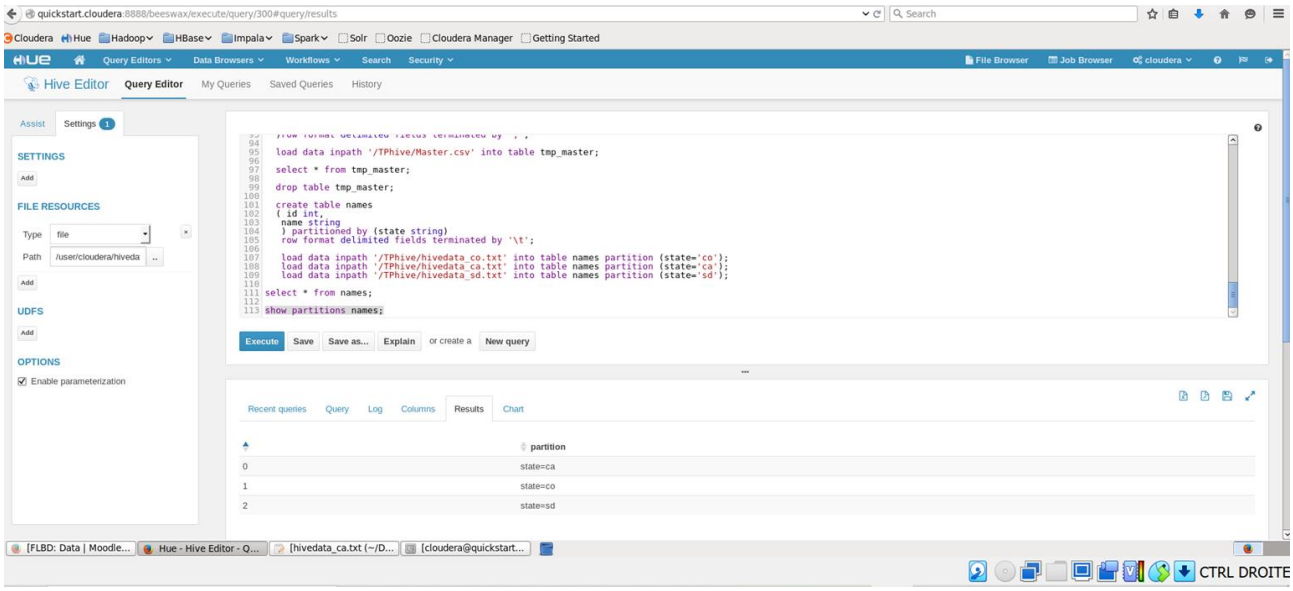
Create tabel names
(id int,
name string

) partitioned by (state string)
row format delimited fields terminated by '\t';

14- Charger les fichiers hivedata_<>.txt dans la table names
15- Vérifier que toutes les données sont bien dans la table



16- Vérifier les partitions avec la commande show partitions
show partitions names;



17- Comment se traduit les partitions sous hdfs :

Dans hdfs les partitions correspondent a des répertoires.

18- Quand dans une requête on spécifie la partition, Hive est beaucoup plus performant et va lire seulement le répertoire correspondant

Le partitionnement dans Hive est conçu pour faciliter la prédiction de l'emplacement des données et améliorer les performances des requêtes qui exploitent cette prédiction.

oui, Hive est généralement beaucoup plus performant lorsqu'on spécifie une partition dans une requête, car cela permet une lecture plus ciblée des données stockées dans le répertoire correspondant à la partition spécifiée.

19- Remarque : il n’y a pas de job MapReduce qui s’est exécuté. Pourquoi ?

la table est suffisamment petite, Hive peut utiliser l'optimisation "Small Table Optimization" et éviter de lancer un job MapReduce complet.

Notre requête inclut une clause de filtrage basée sur une partition spécifique et que cette partition peut être identifiée sans avoir besoin d'un job MapReduce, alors Hive peut effectuer la sélection directement sans déclencher un MapReduce.