

**Ecole Nationale d’Electronique et des**

**Télécommunications de Sfax**

|  |  |
| --- | --- |
| AU | 2021-2022 |
| Matière | Big Data |

# DDL-DML Hive Tutore Guide

1. **Télécharger le fichier Emp.txt et placez le dans Dossier Personnel, puis Lancer hive**
2. **Afficher la liste des BD existantes**

show databases;

1. **Vérifier la BD active et afficher ses tables**

Pour vérifier la BD active, exécuter :

select current\_database();

Pour voir ses tables, exécuter :

show tables;

1. **Créer une BD nommée EmpDB**

create database EmpDB;

Dans le navigateur, allez vers localhost:50070/ menu utilities/Browse the file system.

Suivez user/hive/warehouse. Un répertoire est créé nommé : ***empdb.db***

Ce répertoire est actuellement vide.

1. **Utiliser la base EmpDB pour la création de tables**

Pour utiliser la BD EmpDB, exécuter :

use EmpDB;

1. **Créer une table interne EmpTest ayant le schéma suivant :** EmpTest(EMPNO int, ENAME varchar(15), JOB varchar(15),MGR int, SAL int, COMM int, DEPTNO int).

**Insérer des données avec insert into**

**Charger les données avec load à partir du fichier Emp.txt**

**Supprimer la table**

Exécuter :

use EmpDB;

Puis, créer une table interne :

create table EmpTest(EMPNO int, ENAME varchar(15),

JOB varchar(15),MGR int, SAL int, COMM int, DEPTNO int);

Le répertoire ***empdb.db*** n'est plus vide. Il contient un répertoire qui porte le même nom que la table ***empTest***. Ce répertoire est actuellement vide.

**Insérer des données avec insert into**

insert into Emptest values(7499,'ALLEN','SALESMAN',7698,1600,300,30);

***emptest*** n'est plus vide. À chaque INSERT un fichier est ajouté.

Remarquez la taille 37 et pourtant un fichier de taille 128 est réservé.

**Charger les données avec load à partir du fichier Emp.txt**

load data local inpath '/home/u1/Emp.txt' into table emptest

exécuter select \* from emptest;

null est affiché,

- car le séparateur par défaut est control+A alors que dans le fichier le séparateur est « , »

- les valeurs nulles sont représentées par \N, alors que dans le fichier les null sont représentés par chaine vide

**Supprimer la table**

drop table emptest;

Tout le répertoire est supprimé ainsi que les données

1. **Créer une table externe EmpLocation en indiquant son emplacement et en plaçant des commentaires pour les colonnes et pour la table. Insérer des données**

**Supprimer les données**

**Supprimer la table**

**Utilisation de la clause LOCATION**

create external table EmpLocation

(EMPNO int,ENAME varchar(15),JOB varchar(15),MGR int,SAL int,

COMM int, deptno int comment 'département correspondant')

comment 'liste des emp'

location '/user/hive/EmpLocation';

Pour afficher une description de la table, exécuter :

desc EmpLocation

Pour afficher une description plus détaillée de la table, exécuter :

desc formatted EmpLocation

**Insérer des données**

insert into Emplocation values(1000,'ALLEN','SALESMAN',7698,1600,300,30),(2000,'ALLEN','SALESMAN',7698,1600,300,30);

**Supprimer les données**

Essayer : truncate table emplocation

Un message d’erreur sera affiché pour dire que la table n’est pas gérée par hive. C’est une table externe

**Supprimer la table**

Essayer :

drop table emplocation;

L’instruction est exécutée

essayer : show tables;

La table n’existe plus

Allez vers /user/hive/emplocation

Vous trouvez le répertoire ainsi que le fichier des données.

1. **Créer une table interne EmpDelimited permettant de charger les données correctement du fichier Emp.txt**

create table EmpDelimited (EMPNO int, ENAME varchar(15),

JOB varchar(15), MGR int, SAL int, COMM int, DEPTNO int)

row format delimited fields terminated by ','

tblproperties('skip.header.line.count'='1', 'serialization.null.format'='');

Le chargement des données peut se faire à partir d’un fichier. Cette instruction indique à Hive que :

- le délimiteur entre les colonnes utilisé dans le fichier est le caractère ","

- la première ligne représente les titres des colonnes

- les nulls sont représentés par chaine vide

**Chargement de données**

load data local inpath '/home/u1/Emp.txt' into table empdelimited

1. **Créer une table EmpCTAS en utilisant create … AS SELECT. Remplir la table des employés du département20 à partir de empdelimited**

create table empCTAS as select \* from empdelimited where deptno=20;

Exécutez :

select \* from empctas

Elle affiche les employés du dept 20

Exécutez :

desc empctas

la table *empCTAS* a la même structure que la table *empdelimited*.

1. **Créer une table temporaire EmpTemporary, ayant la même structure que la table EmpDelimited**

create temporary table EmpTemporary like EmpDelimited;

Exécutez :

desc emptemporary

*emptemporary* a la même structure que *empdelimited*

Exécutez :

select \* from emptemporary

*emptemporary* est vide

1. **Créer une table EmpParquet en utilisant STORED AS PARQUET. Essayer insert into … values et load**

Exécutez :

create table EmpParquet(EMPNO int,ENAME varchar(15),JOB varchar(15),MGR int,SAL int,COMM int,DEPTNO int) row format delimited fields terminated by ',' stored as parquet

**insert into**

insert into EmpParquet values(7499,'ALLEN','SALESMAN',7698,1600,300,30);

Vérifier :

select \* from EmpParquet;

Les données sont affichées

Pour afficher son contenu :

hdfs dfs -cat /user/hive/warehouse/empdb.db/empparquet/000000\_0

Le contenu n'est pas visible

**load empparquet**

Exécutez :

Truncate table EmpParquet;

alter table EmpParquet set tblproperties('skip.header.line.count'='1');

alter table EmpParquet set tblproperties('serialization.null.format'='');

load data local inpath '/home/u1/Emp.txt' into table EmpParquet

Dans le navigateur, suivez user/hive/warehouse/empdb.db/EmpParquet. Un fichier Emp.txt est ajouté. Il est non visible.

select \* from EmpParquet

Échoue car il faut un fichier de type parquet

***Remarque :*** Avec le format Parquet, nous pouvons utiliser INSERT INTO. Les données seront converties automatiquement en format parquet, mais ce n'est pas le cas avec load. Avec load, il faut que le fichier soit de type parquet.

C’est le même cas avec ***avro*** et ***orc***

Comment peut-on charger les données à partir d’un fichier source txt ?

Solution :

* Créer temporary table

create temporary table EmpTemp(EMPNO int,ENAME varchar(15),JOB varchar(15),MGR int,SAL int,COMM int,DEPTNO int) row format delimited fields terminated by ',' tblproperties('skip.header.line.count'='1','serialization.null.format'='');

load data local inpath '/home/u1/Emp.txt' into table EmpTemp;

* create table parquet et la remplir à partir de temporary table

create table EmpParquet stored as parquet as select \* from emptemp

La table est créée et chargée.

\*\*\*\*\*\*\*\* ou bien

create temporary table EmpParquet (EMPNO int,ENAME varchar(15),JOB varchar(15),MGR int,SAL int,COMM int,DEPTNO int) *STORED AS parquet;*

Copy to parquet table:

*INSERT INTO TABLE* EmpParquet *SELECT \* FROM EmpTemp;*

1. **Créer une table EmpPartitioned en considérant la colonne deptno comme colonne de partition**

create table EmpPartitioned(EMPNO int,ENAME varchar(15),JOB varchar(15),MGR int,SAL int,COMM int) partitioned by (DEPTNO int) row format delimited fields terminated by ',' stored as textfile

***Remarque*** : la colonne de partition figure seulement dans la clause PARTITIONED BY et non pas dans la liste des colonnes

En exécutant desc, la colonne deptno s'affiche comme les autres colonnes

1. **Charger la table EmpPartitioned en mode partitionnement dynamique**

***Avec insert into … values***

Exécutez :

set hive.exec.dynamic.partition.mode=nonstrict;

Exécutez :

insert into EmpPartitioned partition(deptno=10) values(1000,'ALLEN','SALESMAN',7698,1600,300),(2000,'ALLEN','SALESMAN',7698,1600,300);

insert into EmpPartitioned partition(deptno=20) values(1000,'ALLEN','SALESMAN',7698,1600,300),(2000,'ALLEN','SALESMAN',7698,1600,300);

Exécutez :

select \* from EmpPartitioned

Toutes les données sont affichées, même les numéros de deptno

Exécutez :

hdfs dfs -cat /user/hive/warehouse/empdb.db/emppartitioned/deptno=10/000000\_0

Toutes les données sont affichées à l’exception de la colonne deptno

***Avec load***

alter table EmpPartitioned set tblproperties('skip.header.line.count'='1');

alter table EmpPartitioned set tblproperties('serialization.null.format'='');

load data local inpath '/home/u1/Emp.txt' into table EmpPartitioned partition(deptno=40)

***Problème*** : ici le partitionnement est dynamique mais on ne peut pas filtrer les données du fichier

***Solution*** : charger à partir d'une table et une requête select.

* Créer une table partitionnée.
* Créer une table temporaire
* Charger la table temporaire à partir du fichier
* Charger la table partitionnée à partir de la table temporaire

create temporary table EmpTemp(EMPNO int,ENAME varchar(15),JOB varchar(15),MGR int,SAL int,COMM int,DEPTNO int) row format delimited fields terminated by ',' tblproperties('skip.header.line.count'='1','serialization.null.format'='');

load data local inpath '/home/u1/Emp.txt' into table EmpTemp

insert overwrite table EmpPartitioned partition(deptno) select EMPNO,ENAME,JOB,MGR,SAL,COMM,DEPTNO from emptemp

Ici automatiquement trois partitions sont créées, car la colonne deptno contient trois différentes valeurs.

1. **Créer une table EmpPartStatic en considérant la colonne deptno comme colonne de partition, puis charger la table en mode partitionnement statique**

Si vous êtes en mode de partitionnement dynamique, Exécutez :

set hive.exec.dynamic.partition.mode=strict;

create table EmpPartStatic(EMPNO int,ENAME varchar(15),JOB varchar(15),MGR int,SAL int,COMM int) partitioned by (DEPTNO int) ;

Il faut que la table soit partitionnée

alter table EmpPartStatic add partition (deptno=10) ;

alter table EmpPartStatic add partition (deptno=20) ;

alter table EmpPartStatic add partition (deptno=20) ;

Vérifier dans le navigateur : deux partitions sont créées

Créer une table temporaire à partir de laquelle les données seront chargées

create temporary table EmpTemp(EMPNO int,ENAME varchar(15),JOB varchar(15),MGR int,SAL int,COMM int,DEPTNO int) row format delimited fields terminated by ',' tblproperties('skip.header.line.count'='1', 'serialization.null.format'='');

load data local inpath '/home/u1/Emp.txt' into table EmpTemp;

Charger la table EmpPartStatic

insert overwrite table EmpPartStatic partition(deptno=10) select EMPNO,ENAME,JOB,MGR,SAL,COMM from emptemp where deptno=10;

***Remarque*** : Dans select, il ne faut pas mettre deptno

Vérifier dans le navigateur, dans la partition du deptno=10, in fichier est ajouté

Répéter l’instruction pour les deux autres départements :

insert overwrite table EmpPartStatic partition(deptno=20) select EMPNO,ENAME,JOB,MGR,SAL,COMM from emptemp where deptno=20;

insert overwrite table EmpPartStatic partition(deptno=30) select EMPNO,ENAME,JOB,MGR,SAL,COMM from emptemp where deptno=30;

1. **Créer une table EmpPartBuck et diviser les données par deptno en 3 groupes**

create table EmpBucket(EMPNO int,ENAME varchar(15),JOB varchar(15),MGR int,SAL int,COMM int,DEPTNO int) clustered by(deptno) into 3 buckets tblproperties('skip.header.line.count'='1','serialization.null.format'='');

Essayer de charger la table avec load:

load data local inpath '/home/u1/Emp.txt' into table Empbucket;

\*\*\*\*\*non possible. Le message d'erreur indique qu'il faut utiliser une table intermédiaire puis loader par un insert...select

***Solution*** : charger la table à partir d’une autre table.

insert into table EmpBucket select EMPNO,ENAME,JOB,MGR,SAL,COMM,deptno from emptemp;

Dans le navigateur, remarquer la création de 3 fichiers (et non répertoire). Chacun de taille différente. Chacun concerne un département

Essayez

insert into EmpBucket values(1000,'ALLEN','SALESMAN',7698,1600,null,30),(2000,'ALLEN','SALESMAN',7698,1600,null,30);

Trois autres fichiers sont ajoutés (car c’est indiqué lors de la création : 3 buckets). Un seul contenant des données et les autres de taille 0, car les deux insertions concernent le même département 30

Essayez

insert into EmpBucket values(1000,'ALLEN','SALESMAN',7698,1600,null,10),(2000,'ALLEN','SALESMAN',7698,1600,null,20),(2000,'ALLEN','SALESMAN',7698,1600,null,30),(2000,'ALLEN','SALESMAN',7698,1600,null,**40**);

Trois autres fichiers sont ajoutés (car c’est indiqué lors de la création : 3 buckets). Un fichier est de taille double (contenant des données de deux départements).

1. **Créer une table EmpAlter et modifier sa structure**

create table EmpAlter(EMPNO int,ENAME varchar(15),JOB varchar(15),SAL int) row format delimited fields terminated by ',' stored as textfile;

* Modifier le nom de la table vers employee

alter table empalter rename to employee;

Vérifier :

show tables

* Modifier la colonne empno vers id de type int

alter table employee change empno id int;

Vérifier :

desc employee

* Modifier le type de colonne sal vers decimal (4.2)

alter table employee change sal sal decimal(4,2);

Vérifier :

desc employee

* Ajouter une colonne deptno de type int

alter table employee add columns(deptno int); desc employee

Vérifier :

desc employee