**Hive常用命令**

创建表  
hive> CREATE TABLE pokes (foo INT, bar STRING);   
创建表并创建索引字段ds  
hive> CREATE TABLE invites (foo INT, bar STRING) PARTITIONED BY (ds STRING);   
显示所有表  
hive> SHOW TABLES;  
按正条件（[正则](http://www.linuxso.com/book/11723.html" \t "_blank)表达式）显示表，  
hive> SHOW TABLES '.\*s';  
表添加一列   
hive> ALTER TABLE pokes ADD COLUMNS (new\_[col](http://www.linuxso.com/command/col.html" \t "_blank) INT);  
添加一列并增加列字段注释  
hive> ALTER TABLE invites ADD COLUMNS (new\_col2 INT COMMENT 'a comment');  
更改表名  
hive> ALTER TABLE events RENAME TO 3koobecaf;  
删除列  
hive> DROP TABLE pokes;  
元数据存储  
将文件中的数据加载到表中  
hive> LOAD DATA LOCAL INPATH './[ex](http://www.linuxso.com/command/ex.html" \t "_blank)amples/[file](http://www.linuxso.com/command/file.html" \t "_blank)s/kv1.txt' OVERWRITE INTO TABLE pokes;   
加载本地数据，同时给定分区信息  
hive> LOAD DATA LOCAL INPATH './examples/files/kv2.txt' OVERWRITE INTO TABLE invites PARTITION (ds='2008-08-15');  
加载DFS数据 ，同时给定分区信息  
hive> LOAD DATA INPATH '/user/myname/kv2.txt' OVERWRITE INTO TABLE invites PARTITION (ds='2008-08-15');  
The above command will load data from an HDFS file/directory to the table. Note that loading data from HDFS will re[su](http://www.linuxso.com/command/su.html" \t "_blank)lt in moving the file/directory. As a result, the operation is almost instantaneous.   
SQL 操作  
按先件查询  
hive> SELECT a.foo FROM invites a WHERE a.ds='<DATE>';  
将查询数据输出至目录  
hive> INSERT OVERWRITE DIRECTORY '/tmp/h[df](http://www.linuxso.com/command/df.html" \t "_blank)s\_out' SELECT a.\* FROM invites a WHERE a.ds='<DATE>';  
将查询结果输出至本地目录  
hive> INSERT OVERWRITE LOCAL DIRECTORY '/tmp/local\_out' SELECT a.\* FROM pokes a;  
选择所有列到本地目录   
hive> INSERT OVERWRITE TABLE events SELECT a.\* FROM profiles a;  
hive> INSERT OVERWRITE TABLE events SELECT a.\* FROM profiles a WHERE a.key < 100;   
hive> INSERT OVERWRITE LOCAL DIRECTORY '/tmp/reg\_3' SELECT a.\* FROM events a;  
hive> INSERT OVERWRITE DIRECTORY '/tmp/reg\_4' select a.invites, a.pokes FROM profiles a;  
hive> INSERT OVERWRITE DIRECTORY '/tmp/reg\_5' SELECT COUNT(1) FROM invites a WHERE a.ds='<DATE>';  
hive> INSERT OVERWRITE DIRECTORY '/tmp/reg\_5' SELECT a.foo, a.bar FROM invites a;  
hive> INSERT OVERWRITE LOCAL DIRECTORY '/tmp/sum' SELECT SUM(a.pc) FROM pc1 a;  
将一个表的统计结果插入另一个表中  
hive> FROM invites a INSERT OVERWRITE TABLE events SELECT a.bar, count(1) WHERE a.foo > 0 GROUP BY a.bar;  
hive> INSERT OVERWRITE TABLE events SELECT a.bar, count(1) FROM invites a WHERE a.foo > 0 GROUP BY a.bar;  
JOIN  
hive> FROM pokes t1 JOIN invites t2 ON (t1.bar = t2.bar) INSERT OVERWRITE TABLE events SELECT t1.bar, t1.foo, t2.foo;  
将多表数据插入到同一表中  
FROM src  
INSERT OVERWRITE TABLE dest1 SELECT src.\* WHERE src.key < 100  
INSERT OVERWRITE TABLE dest2 SELECT src.key, src.value WHERE src.key >= 100 and src.key < 200  
INSERT OVERWRITE TABLE dest3 PARTITION(ds='2008-04-08', hr='12') SELECT src.key WHERE src.key >= 200 and src.key < 300  
INSERT OVERWRITE LOCAL DIRECTORY '/tmp/dest4.out' SELECT src.value WHERE src.key >= 300;  
将文件流直接插入文件  
hive> FROM invites a INSERT OVERWRITE TABLE events SELECT TRANSFORM(a.foo, a.bar) AS (oof, rab) USING '/bin/[cat](http://www.linuxso.com/command/cat.html" \t "_blank)' WHERE a.ds > '2008-08-09';  
This s[tr](http://www.linuxso.com/command/tr.html" \t "_blank)eams the data in the map phase through the script /bin/cat (like hadoop streaming). Similarly - streaming can be u[sed](http://www.linuxso.com/command/sed.html" \t "_blank) on the re[du](http://www.linuxso.com/command/du.html" \t "_blank)ce s[id](http://www.linuxso.com/command/id.html" \t "_blank)e (please see the Hive Tutorial or examples)   
实际示例  
创建一个表  
CREATE TABLE u\_data (  
userid INT,  
movieid INT,  
rating INT,  
unix[time](http://www.linuxso.com/command/time.html" \t "_blank) STRING)  
ROW FORMAT DELIMITED  
FIELDS TERMINATED BY '\t'  
STORED AS TEXTFILE;  
下载示例数据文件，并解[压缩](http://www.linuxso.com/linuxpeixun/12737.html" \t "_blank)  
wget http://www.grouplens.org/system/files/ml-data.tar\_\_0.gz  
tar xvzf ml-data.tar\_\_0.gz  
加载数据到表中  
LOAD DATA LOCAL INPATH 'ml-data/u.data'  
OVERWRITE INTO TABLE u\_data;  
统计数据总量  
SELECT COUNT(1) FROM u\_data;  
现在做一些复杂的数据分析  
创建一个 weekday\_mapper.py: 文件，作为数据按周进行分割   
import sys  
import [date](http://www.linuxso.com/command/date.html" \t "_blank)time  
for line in sys.stdin:  
line = line.strip()  
userid, movieid, rating, unixtime = line.[split](http://www.linuxso.com/command/split.html" \t "_blank)('\t')  
生成数据的周信息  
weekday = datetime.datetime.fromtimestamp(float(unixtime)).isoweekday()  
print '\t'.[join](http://www.linuxso.com/command/join.html" \t "_blank)([userid, movieid, rating, str(weekday)])  
使用映射脚本  
//创建表，按分割符分割行中的字段值  
CREATE TABLE u\_data\_new (  
userid INT,  
movieid INT,  
rating INT,  
weekday INT)  
ROW FORMAT DELIMITED  
FIELDS TERMINATED BY '\t';  
//将python文件加载到系统  
a[dd](http://www.linuxso.com/command/dd.html" \t "_blank) FILE weekday\_mapper.py;  
将数据按周进行分割  
INSERT OVERWRITE TABLE u\_data\_new  
SELECT  
TRANSFORM (userid, movieid, rating, unixtime)  
USING 'python weekday\_mapper.py'  
AS (userid, movieid, rating, weekday)  
FROM u\_data;  
SELECT weekday, COUNT(1)  
FROM u\_data\_new  
GROUP BY weekday;  
处理Apache Weblog 数据  
将WEB日志先用[正则表达式](http://www.linuxso.com/book/11723.html" \t "_blank)进行组合，再按需要的条件进行组合输入到表中  
add jar ../build/contrib/hive\_contrib.jar;  
CREATE TABLE apachelog (  
host STRING,  
identity STRING,  
user STRING,  
time STRING,  
request STRING,  
[stat](http://www.linuxso.com/command/stat.html" \t "_blank)us STRING,  
size STRING,  
referer STRING,  
agent STRING)  
ROW FORMAT SERDE 'org.apache.hadoop.hive.contrib.serde2.RegexSerDe'  
WITH SERDEPROPERTIES (  
"input.regex" = "([^ ]\*) ([^ ]\*) ([^ ]\*) (-|\\[[^\\]]\*\\]) ([^ \"]\*|\"[^\"]\*\") (-|[0-9]\*) (-|[0-9]\*)(?: ([^ \"]\*|\"[^\"]\*\") ([^ \"]\*|\"[^\"]\*\"))?",  
"output.fo[rm](http://www.linuxso.com/command/rm.html" \t "_blank)at.string" = "%1$s %2$s %3$s %4$s %5$s %6$s %7$s %8$s %9$s"  
)  
STORED AS TEXTFILE;http://www.linuxso.com/architecture/11357.html