Sort

1- order by

emp表：通过员工的部门编号升序排列，默认为升序（asc）,降序(desc)

0: jdbc:hive2://hadoop2:10000> select *from emp order by deptno asc;

```
+------------+------------+------------+----------+--------------+----------+-----------+-------------+
| emp.empno  | emp.ename  |  emp.job   | emp.mgr  | emp.hiredate | emp.sal  | emp.comm  | emp.deptno  |
+------------+------------+------------+----------+--------------+----------+-----------+-------------+
| 7934       | MILLER     | CLERK      | 7782     | 1982-1-23    | 1300.0   | NULL      | 10          |
| 7839       | KING       | PRESIDENT  | NULL     | 1981-11-17   | 5000.0   | NULL      | 10          |
| 7782       | CLARK      | MANAGER    | 7839     | 1981-6-9     | 2450.0   | NULL      | 10          |
| 7876       | ADAMS      | CLERK      | 7788     | 1987-5-23    | 1100.0   | NULL      | 20          |
| 7788       | SCOTT      | ANALYST    | 7566     | 1987-4-19    | 3000.0   | NULL      | 20          |
| 7369       | SMITH      | CLERK      | 7902     | 1980-12-17   | 800.0    | NULL      | 20          |
| 7566       | JONES      | MANAGER    | 7839     | 1981-4-2     | 2975.0   | NULL      | 20          |
| 7902       | FORD       | ANALYST    | 7566     | 1981-12-3    | 3000.0   | NULL      | 20          |
| 7844       | TURNER     | SALESMAN   | 7698     | 1981-9-8     | 1500.0   | 0.0       | 30          |
| 7499       | ALLEN      | SALESMAN   | 7698     | 1981-2-20    | 1600.0   | 300.0     | 30          |
| 7698       | BLAKE      | MANAGER    | 7839     | 1981-5-1     | 2850.0   | NULL      | 30          |
| 7654       | MARTIN     | SALESMAN   | 7698     | 1981-9-28    | 1250.0   | 1400.0    | 30          |
| 7521       | WARD       | SALESMAN   | 7698     | 1981-2-22    | 1250.0   | 500.0     | 30          |
| 7900       | JAMES      | CLERK      | 7698     | 1981-12-3    | 950.0    | NULL      | 30          |
+------------+------------+------------+----------+--------------+----------+-----------+-------------+
14 rows selected (47.753 seconds)
```

2- set reducetask

手动设置reduces的个数默认为-1：

0: jdbc:hive2://hadoop2:10000> set mapreduce.job.reduces=3;

No rows affected (0.141 seconds)

0: jdbc:hive2://hadoop2:10000> set mapreduce.job.reduces;

```
+---------------------------+
|            set            |
+---------------------------+
| mapreduce.job.reduces=3   |
+---------------------------+
1 row selected (0.036 seconds)
```

3- sort by

order by是全局排序，需要对整个table进行排序时，当有多个reducetask任务时，可以通过sort by只在各个reducetask内进行排序

0: jdbc:hive2://hadoop2:10000> select *from emp sort by deptno;

```
INFO  : Concurrency mode is disabled, not creating a lock manager
+------------+------------+------------+----------+--------------+----------+-----------+-------------+
| emp.empno  | emp.ename  |  emp.job   | emp.mgr  | emp.hiredate | emp.sal  | emp.comm  | emp.deptno  |
+------------+------------+------------+----------+--------------+----------+-----------+-------------+
| 7782       | CLARK      | MANAGER    | 7839     | 1981-6-9     | 2450.0   | NULL      | 10          |
| 7839       | KING       | PRESIDENT  | NULL     | 1981-11-17   | 5000.0   | NULL      | 10          |
| 7788       | SCOTT      | ANALYST    | 7566     | 1987-4-19    | 3000.0   | NULL      | 20          |
| 7654       | MARTIN     | SALESMAN   | 7698     | 1981-9-28    | 1250.0   | 1400.0    | 30          |
| 7698       | BLAKE      | MANAGER    | 7839     | 1981-5-1     | 2850.0   | NULL      | 30          |
| 7844       | TURNER     | SALESMAN   | 7698     | 1981-9-8     | 1500.0   | 0.0       | 30          |
| 7934       | MILLER     | CLERK      | 7782     | 1982-1-23    | 1300.0   | NULL      | 10          |
| 7876       | ADAMS      | CLERK      | 7788     | 1987-5-23    | 1100.0   | NULL      | 20          |
| 7566       | JONES      | MANAGER    | 7839     | 1981-4-2     | 2975.0   | NULL      | 20          |
| 7900       | JAMES      | CLERK      | 7698     | 1981-12-3    | 950.0    | NULL      | 30          |
| 7521       | WARD       | SALESMAN   | 7698     | 1981-2-22    | 1250.0   | 500.0     | 30          |
| 7499       | ALLEN      | SALESMAN   | 7698     | 1981-2-20    | 1600.0   | 300.0     | 30          |
| 7902       | FORD       | ANALYST    | 7566     | 1981-12-3    | 3000.0   | NULL      | 20          |
| 7369       | SMITH      | CLERK      | 7902     | 1980-12-17   | 800.0    | NULL      | 20          |
+------------+------------+------------+----------+--------------+----------+-----------+-------------+
14 rows selected (60.83 seconds)
```

4- distribute by

分区排序，是将数据进行按照reducetask的个数分成部分文件，对各个部分文件进行文件内部排序。与sort by结合使用。

0: jdbc:hive2://hadoop2:10000> insert overwrite local directory '/opt/data/sort' row format delimited fields terminated by '\t' select *from emp

. . . . . .> distribute by deptno sort by sal desc;

查看/opt/data/sort目录下的文件

```
[tay@hadoop2 sort]$ ll
总用量 12
-rw-r--r--. 1 tay tay 293 6月   3 09:43 000000_0
-rw-r--r--. 1 tay tay 139 6月   3 09:43 000001_0
-rw-r--r--. 1 tay tay 229 6月   3 09:43 000002_0
[tay@hadoop2 sort]$ cat 000002_0
7788    SCOTT   ANALYST 7566    1987-4-19       3000.0  \N      20
7902    FORD    ANALYST 7566    1981-12-3       3000.0  \N      20
7566    JONES   MANAGER 7839    1981-4-2        2975.0  \N      20
7876    ADAMS   CLERK   7788    1987-5-23       1100.0  \N      20
7369    SMITH   CLERK   7902    1980-12-17      800.0   \N      20
[tay@hadoop2 sort]$
```

5- cluster by

当distribute by 与sort by 的字段一样时，可以直接用cluster by。具有一定的局限性。

0: jdbc:hive2://hadoop2:10000> select *from emp cluster by deptno;

```
INFO  : Concurrency mode is disabled, not creating a lock manager
+------------+------------+------------+----------+--------------+----------+-----------+-------------+
| emp.empno  | emp.ename  |  emp.job   | emp.mgr  | emp.hiredate | emp.sal  | emp.comm  | emp.deptno  |
+------------+------------+------------+----------+--------------+----------+-----------+-------------+
| 7654       | MARTIN     | SALESMAN   | 7698     | 1981-9-28    | 1250.0   | 1400.0    | 30          |
| 7900       | JAMES      | CLERK      | 7698     | 1981-12-3    | 950.0    | NULL      | 30          |
| 7698       | BLAKE      | MANAGER    | 7839     | 1981-5-1     | 2850.0   | NULL      | 30          |
| 7521       | WARD       | SALESMAN   | 7698     | 1981-2-22    | 1250.0   | 500.0     | 30          |
| 7844       | TURNER     | SALESMAN   | 7698     | 1981-9-8     | 1500.0   | 0.0       | 30          |
| 7499       | ALLEN      | SALESMAN   | 7698     | 1981-2-20    | 1600.0   | 300.0     | 30          |
| 7934       | MILLER     | CLERK      | 7782     | 1982-1-23    | 1300.0   | NULL      | 10          |
| 7839       | KING       | PRESIDENT  | NULL     | 1981-11-17   | 5000.0   | NULL      | 10          |
| 7782       | CLARK      | MANAGER    | 7839     | 1981-6-9     | 2450.0   | NULL      | 10          |
| 7788       | SCOTT      | ANALYST    | 7566     | 1987-4-19    | 3000.0   | NULL      | 20          |
| 7566       | JONES      | MANAGER    | 7839     | 1981-4-2     | 2975.0   | NULL      | 20          |
| 7876       | ADAMS      | CLERK      | 7788     | 1987-5-23    | 1100.0   | NULL      | 20          |
| 7902       | FORD       | ANALYST    | 7566     | 1981-12-3    | 3000.0   | NULL      | 20          |
| 7369       | SMITH      | CLERK      | 7902     | 1980-12-17   | 800.0    | NULL      | 20          |
+------------+------------+------------+----------+--------------+----------+-----------+-------------+
14 rows selected (30.387 seconds)
```

Partition

1-创建分区表

0: jdbc:hive2://hadoop2:10000> create table dept_partition(

. . . . . .> dept_no int ,dept_name string,loc string)

. . . . . .> partitioned by (day string)

. . . . . .> row format delimited fields terminated by '\t';

分区表：本质就是添加列，对数据进行过滤。

查看分区表的具体信息。

0: jdbc:hive2://hadoop2:10000> desc dept_partition;

```
+---------------------------+------------+----------+
|         col_name          | data_type  | comment  |
+---------------------------+------------+----------+
| dept_no                   | int        |          |
| dept_name                 | string     |          |
| loc                       | string     |          |
| day                       | string     |          |
|                           | NULL       | NULL     |
| # Partition Information   | NULL       | NULL     |
| # col_name                | data_type  | comment  |
| day                       | string     |          |
+---------------------------+------------+----------+
```

可以看到table是按day进行分区

2-加载数据

通过加载数据对table进行插入数据，按days生成三个partitions.

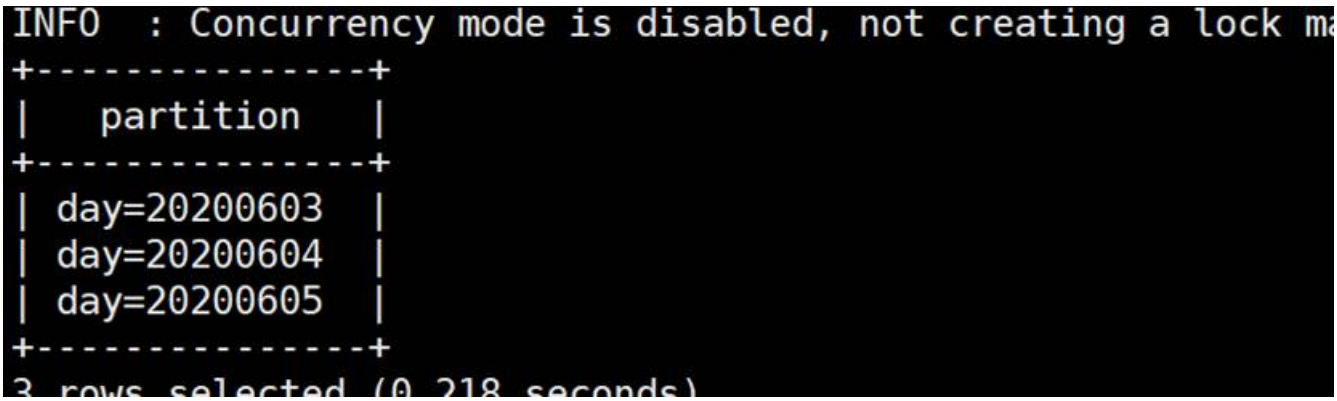
load data local inpath '/opt/data/logs/dept_20200603.log' into table dept_partition partition(day='20200603');

load data local inpath '/opt/data/logs/dept_20200604.log' into table dept_partition partition(day='20200604');

load data local inpath '/opt/data/logs/dept_20200605.log' into table dept_partition partition(day='20200605');

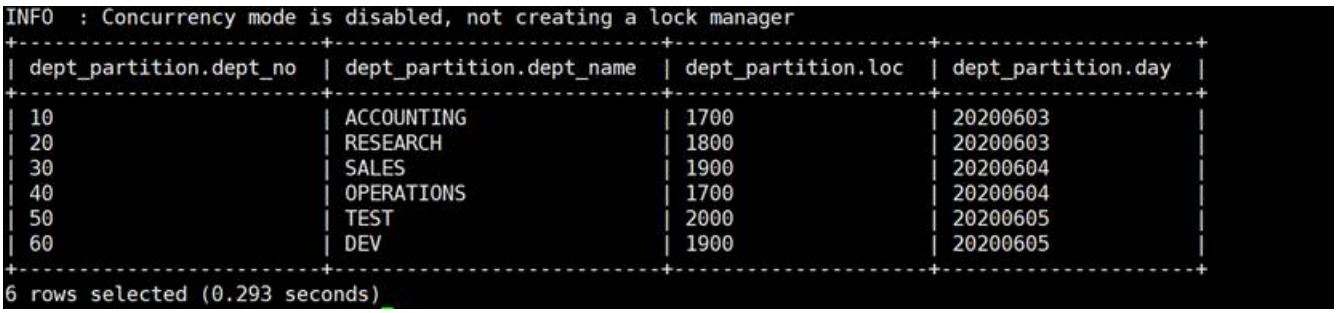
查看table dept_partition的分区。

0: jdbc:hive2://hadoop2:10000> show partitions dept_partition;



table的数据

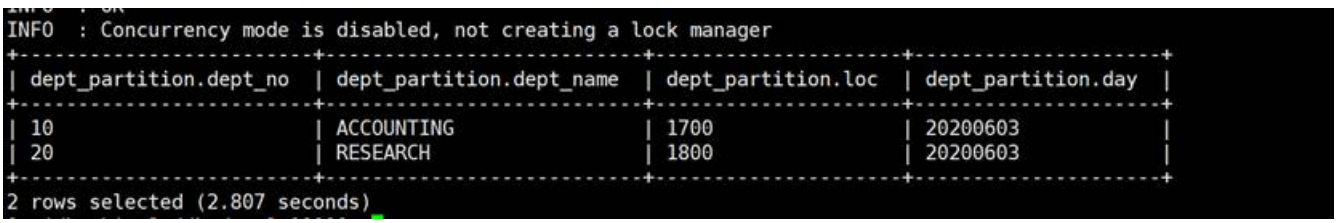
0: jdbc:hive2://hadoop2:10000> select *from dept_partition;



3-按区查找数据

通过day=20200603的条件进行查询（本质就是又加了一个列，对数据进行过滤）

0: jdbc:hive2://hadoop2:10000> select *from dept_partition where day=20200603;



4-在原表的基础上添加分区

1.添加多个分区（各个分区之间不用”,”）：

0: jdbc:hive2://hadoop2:10000> alter table dept_partition add partition(day='20200606') partition(day='20200607');

2.查看目前的分区情况：

0: jdbc:hive2://hadoop2:10000> show partitions dept_partition;

```
INFO  : Concurrency mode is disabled, not creating a lock manager
+---------------+
|   partition   |
+---------------+
| day=20200603  |
| day=20200604  |
| day=20200605  |
| day=20200606  |
| day=20200607  |
+---------------+
5 rows selected (0.145 seconds)
```

5-删除分区

删除分区（注意：各个区之间需要有”,”）

alter table dept_partition drop partition(day='20200605'), partition(day='20200607');

查看分区情况：

0: jdbc:hive2://hadoop2:10000> show partitions dept_partition;

```
INFO  : Concurrency mode is disabled, not creating a lock manager
+---------------+
|   partition   |
+---------------+
| day=20200603  |
| day=20200604  |
| day=20200606  |
+---------------+
3 rows selected (0.133 seconds)
```

查看表的具体信息：

0: jdbc:hive2://hadoop2:10000> desc formatted dept_partition;

```
+------------------------------+----------------------------------------------------+-----------------------+
|          col_name            |                     data_type                      |        comment        |
+------------------------------+----------------------------------------------------+-----------------------+
| # col_name                   | data_type                                          | comment               |
| dept_no                      | int                                                |                       |
| dept_name                    | string                                             |                       |
| loc                          | string                                             |                       |
|                              | NULL                                               | NULL                  |
| # Partition Information      | NULL                                               | NULL                  |
| # col_name                   | data_type                                          | comment               |
| day                          | string                                             |                       |
|                              | NULL                                               | NULL                  |
| # Detailed Table Information | NULL                                               | NULL                  |
| Database:                    | mydb                                               | NULL                  |
| OwnerType:                   | USER                                               | NULL                  |
| Owner:                       | tay                                                | NULL                  |
| CreateTime:                  | Wed Jun 03 11:18:09 CST 2020                       | NULL                  |
| LastAccessTime:              | UNKNOWN                                            | NULL                  |
| Retention:                   | 0                                                  | NULL                  |
| Location:                    | hdfs://hadoop2:9820/user/hive/warehouse/mydb.db/dept_partition | NULL                  |
| Table Type:                  | MANAGED_TABLE                                      | NULL                  |
| Table Parameters:            | NULL                                               | NULL                  |
|                              | bucketing_version                                  | 2                     |
|                              | numFiles                                           | 2                     |
|                              | numPartitions                                      | 3                     |
|                              | numRows                                            | 0                     |
|                              | rawDataSize                                        | 0                     |
|                              | totalSize                                          | 69                    |
|                              | transient_lastDdlTime                              | 1591154289            |
```

6-多级分区

1.多级分区：只需要在partitioned by（）添加多个字段。

0: jdbc:hive2://hadoop2:10000> create table dept_partition1(

. . . . . .> dept_no int ,dept_name string,loc string)

. . . . . .> partitioned by (day string,hour string)

. . . . . .> row format delimited fields terminated by '\t';

load data local inpath '/opt/data/logs/dept_20200603.log' into table dept_partition1 partition(day='20200603',hour=1);

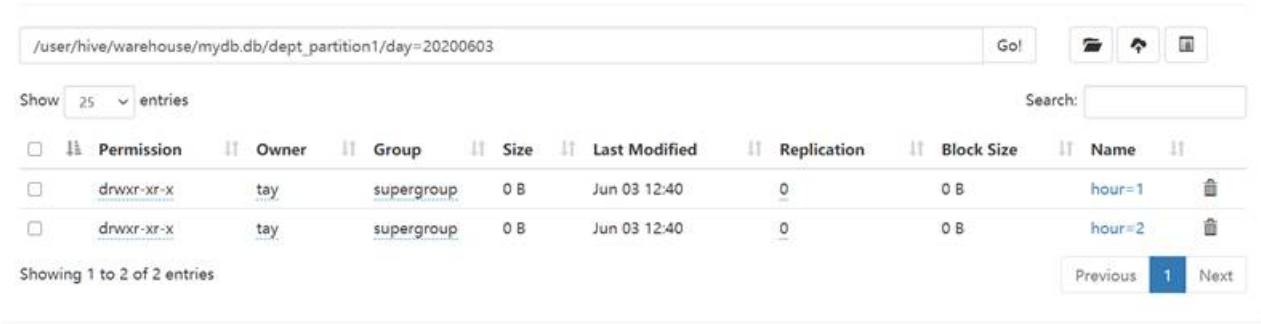
load data local inpath '/opt/data/logs/dept_20200604.log' into table dept_partition1 partition(day='20200603',hour=2);

load data local inpath '/opt/data/logs/dept_20200604.log' into table dept_partition1 partition(day='20200604',hour=1);

load data local inpath '/opt/data/logs/dept_20200605.log' into table dept_partition1 partition(day='20200604',hour=2);

2.查看web端的元数据默认目录

/user/hive/warehouse/mydb.db/dept_partition1/day=20200603/其中day=20200603是一级目录（分区），在他day=20200603下面有两个二级目录（二级分区）。



3.查看数据：

0: jdbc:hive2://hadoop2:10000> select *from dept_partition1;

```
INFO  : Concurrency mode is disabled, not creating a lock manager
+--------------------------+----------------------------+---------------------+----------------------+-----------------------+
| dept_partition1.dept_no  | dept_partition1.dept_name  | dept_partition1.loc | dept_partition1.day  | dept_partition1.hour  |
+--------------------------+----------------------------+---------------------+----------------------+-----------------------+
| 10                       | ACCOUNTING                 | 1700                | 20200603             | 1                     |
| 20                       | RESEARCH                   | 1800                | 20200603             | 1                     |
| 30                       | SALES                      | 1900                | 20200603             | 2                     |
| 40                       | OPERATIONS                 | 1700                | 20200603             | 2                     |
| 30                       | SALES                      | 1900                | 20200604             | 1                     |
| 40                       | OPERATIONS                 | 1700                | 20200604             | 1                     |
| 50                       | TEST                       | 2000                | 20200604             | 2                     |
| 60                       | DEV                        | 1900                | 20200604             | 2                     |
+--------------------------+----------------------------+---------------------+----------------------+-----------------------+
8 rows selected (0.282 seconds)
```

7-分区表与数据的三种关联方式

1.先在HDFS上建立数据分区,并上传到HDFS数据,再通过msck repair进行修复

0: jdbc:hive2://hadoop2:10000>

dfs -mkdir -p /user/hive/warehouse/mydb.db/dept_partition1/day=20200603/hour=3;

0:jdbc:hive2://hadoop2:10000>dfs -put /opt/data/logs/dept_20200603.log /user/hive/warehouse/mydb.db/dept_partition1/day=20200603/hour=3;

0:jdbc:hive2://hadoop2:10000>msck repair table dept_partition2

0: jdbc:hive2://hadoop2:10000> select *from dept_partition1;

```
INFO  : Concurrency mode is disabled, not creating a lock manager
+--------------------------+----------------------------+-----------------------+-----------------------+------------------------+
| dept_partition1.dept_no  | dept_partition1.dept_name  | dept_partition1.loc   | dept_partition1.day   | dept_partition1.hour   |
+--------------------------+----------------------------+-----------------------+-----------------------+------------------------+
| 10                       | ACCOUNTING                 | 1700                  | 20200603              | 1                      |
| 20                       | RESEARCH                   | 1800                  | 20200603              | 1                      |
| 30                       | SALES                      | 1900                  | 20200603              | 2                      |
| 40                       | OPERATIONS                 | 1700                  | 20200603              | 2                      |
| 10                       | ACCOUNTING                 | 1700                  | 20200603              | 3                      |
| 20                       | RESEARCH                   | 1800                  | 20200603              | 3                      |
| 30                       | SALES                      | 1900                  | 20200604              | 1                      |
| 40                       | OPERATIONS                 | 1700                  | 20200604              | 1                      |
| 50                       | TEST                       | 2000                  | 20200604              | 2                      |
| 60                       | DEV                        | 1900                  | 20200604              | 2                      |
+--------------------------+----------------------------+-----------------------+-----------------------+------------------------+
10 rows selected (0.244 seconds)
```

2. 先在HDFS上建立数据分区,并上传到HDFS数据,再通过alter add进行添加刚才建立的分区。

dfs -mkdir -p /user/hive/warehouse/mydb.db/dept_partition1/day=20200603/hour=4;

0:jdbc:hive2://hadoop2:10000>dfs -put /opt/data/logs/dept_20200604.log /user/hive/warehouse/mydb.db/dept_partition1/day=20200603/hour=4;

0: jdbc:hive2://hadoop2:10000> alter table dept_partition1 add partition(day='20200603',hour=4);

```
INFO  : Concurrency mode is disabled, not creating a lock manager
+--------------------------+----------------------------+-----------------------+-----------------------+------------------------+
| dept_partition1.dept_no  | dept_partition1.dept_name  | dept_partition1.loc   | dept_partition1.day   | dept_partition1.hour   |
+--------------------------+----------------------------+-----------------------+-----------------------+------------------------+
| 10                       | ACCOUNTING                 | 1700                  | 20200603              | 1                      |
| 20                       | RESEARCH                   | 1800                  | 20200603              | 1                      |
| 30                       | SALES                      | 1900                  | 20200603              | 2                      |
| 40                       | OPERATIONS                 | 1700                  | 20200603              | 2                      |
| 10                       | ACCOUNTING                 | 1700                  | 20200603              | 3                      |
| 20                       | RESEARCH                   | 1800                  | 20200603              | 3                      |
| 30                       | SALES                      | 1900                  | 20200603              | 4                      |
| 40                       | OPERATIONS                 | 1700                  | 20200603              | 4                      |
| 30                       | SALES                      | 1900                  | 20200604              | 1                      |
| 40                       | OPERATIONS                 | 1700                  | 20200604              | 1                      |
| 50                       | TEST                       | 2000                  | 20200604              | 2                      |
| 60                       | DEV                        | 1900                  | 20200604              | 2                      |
+--------------------------+----------------------------+-----------------------+-----------------------+------------------------+
12 rows selected (0.324 seconds)
```

3．先在HDFS上建立数据分区，通过加载时指定分区。

0: jdbc:hive2://hadoop2:10000> dfs -mkdir -p /user/hive/warehouse/mydb.db/dept_partition1/day=20200603/hour=5;

0: jdbc:hive2://hadoop2:10000> load data local inpath '/opt/data/logs/dept_20200605.log' into table dept_partition1 partition(day='20200603',hour=5);

8-动态分区

1.创建表

0: jdbc:hive2://hadoop2:10000> create table dept_dy_partition(

. . . . . .> dept_no int ,dept_name string)

. . . . . .> partitioned by (loc string)

. . . . . .> row format delimited fields terminated by '\t';

0: jdbc:hive2://hadoop2:10000> set hive.exec.dynamic.partition.mode;

```
0: jdbc:hive2://hadoop2:10000> set hive.exec.dynamic.partition.mode;
+-----------------------------------------+
|                   set                   |
+-----------------------------------------+
| hive.exec.dynamic.partition.mode=strict  |
+-----------------------------------------+
```

2.设置动态分区非严格模式：

0: jdbc:hive2://hadoop2:10000> set hive.exec.dynamic.partition.mode=nostrict;

```
0: jdbc:hive2://hadoop2:10000> set hive.exec.dynamic.partition.mode;
+-----------------------------------------+
|                   set                   |
+-----------------------------------------+
| hive.exec.dynamic.partition.mode=nostrict  |
+-----------------------------------------+
1 row selected (0.022 seconds)
```

3.插入一条数据，按loc=10000分区。

0: jdbc:hive2://hadoop2:10000> insert into table dept_dy_partition values(11,'ALI','10000');

0: jdbc:hive2://hadoop2:10000> show partitions dept_dy_partition;

```
INFO  : Concurrency mode is disabled, not creating a lock manager
+------------+
| partition  |
+------------+
| loc=10000  |
+------------+
1 row selected (0.179 seconds)
```

4.也可以从其他表（列的形式相同）插入数据

0: jdbc:hive2://hadoop2:10000> insert into table dept_dy_partition partition(loc) select *from dept;

5.查看分区：

0: jdbc:hive2://hadoop2:10000> select *from dept_dy_partition;

```
INFO  : Concurrency mode is disabled, not creating a lock manager
+---------------------------+-----------------------------+-----------------------+
| dept_dy_partition.dept_no  | dept_dy_partition.dept_name  | dept_dy_partition.loc  |
+---------------------------+-----------------------------+-----------------------+
| 11                         | ALI                         | 10000                 |
| 10                         | ACCOUNTING                  | 1700                  |
| 40                         | OPERATIONS                  | 1700                  |
| 20                         | RESEARCH                    | 1800                  |
| 30                         | SALES                       | 1900                  |
+---------------------------+-----------------------------+-----------------------+
5 rows selected (0.672 seconds)
```

Bucket

分桶的注意事项：当指定的分桶个数大于设置的reduce个数就会出错，会出现空指针异常。

需要将reduce的个数设置为大于bucket的个数，即reduce>=bucket

当然也可以把reduce设置为-1.

set mapreduce.job.reduces=-1;

1-创建分桶表

1.按id进行分到4个桶。

0: jdbc:hive2://hadoop2:10000> create table stu_bucket(

. . . . . .> id int,name string)

. . . . . .> clustered by(id) into 4 buckets

. . . . . .> row format delimited fields terminated by '\t';

2-加载数据到table的2种方法：

1．直接加载文件的数据

0: jdbc:hive2://hadoop2:10000> load data local inpath '/opt/data/bucket.txt' into table stu_bucket;

0: jdbc:hive2://hadoop2:10000> truncate table stu_bucket;

2．从其他表直接插入

0: jdbc:hive2://hadoop2:10000> insert into table stu_bucket select *from student;

0: jdbc:hive2://hadoop2:10000> select *from stu_bucket;

```
+----------------+------------------+
| stu_bucket.id  | stu_bucket.name  |
+----------------+------------------+
| 1016           | ss16             |
| 1012           | ss12             |
| 1008           | ss8              |
| 1004           | ss4              |
| 1009           | ss9              |
| 1005           | ss5              |
| 1001           | ss1              |
| 1013           | ss13             |
| 1010           | ss10             |
| 1002           | ss2              |
| 1006           | ss6              |
| 1014           | ss14             |
| 1003           | ss3              |
| 1011           | ss11             |
| 1007           | ss7              |
| 1015           | ss15             |
+----------------+------------------+
16 rows selected (0.846 seconds)
```

3-抽样调查

0: jdbc:hive2://hadoop2:10000> select *from stu_bucket tablesample(bucket 2 out of 4 on id);

0: jdbc:hive2://hadoop2:10000> select *from stu_bucket tablesample(0.5 percent);

函数

1- nvl

对空的字段进行赋值

0: jdbc:hive2://hadoop2:10000> select comm,nvl(comm,0) as new_comm from emp;

```
INFO  : Concurrency mode is disabled, not creating a lock manager
+---------+-----------+
|  comm   | new_comm  |
+---------+-----------+
| NULL    | 0.0       |
| 300.0   | 300.0     |
| 500.0   | 500.0     |
| NULL    | 0.0       |
| 1400.0  | 1400.0    |
| NULL    | 0.0       |
| NULL    | 0.0       |
| NULL    | 0.0       |
| NULL    | 0.0       |
| 0.0     | 0.0       |
| NULL    | 0.0       |
| NULL    | 0.0       |
| NULL    | 0.0       |
| NULL    | 0.0       |
+---------+-----------+
14 rows selected (0.14 seconds)
```

2-case when then else end

1.创建表：

0: jdbc:hive2://hadoop2:10000> create table emp_sex(

. . . . . .> name string,dept_id string,sex string)

. . . . . .> row format delimited fields terminated by '\t';

0: jdbc:hive2://hadoop2:10000> load data local inpath '/opt/data/emp_sex.txt' into table emp_sex;

2.查看表：

0: jdbc:hive2://hadoop2:10000> select *from emp_sex;

```
INFO  : OK
INFO  : Concurrency mode is disabled, not creating a lock manager
+---------------+------------------+--------------+
| emp_sex.name  | emp_sex.dept_id  | emp_sex.sex  |
+---------------+------------------+--------------+
| 悟空          | A                | 男           |
| 大海          | A                | 男           |
| 宋宋          | B                | 男           |
| 凤姐          | A                | 女           |
| 婷姐          | B                | 女           |
| 婷婷          | B                | 女           |
+---------------+------------------+--------------+
6 rows selected (0.229 seconds)
```

3.练习语法：按部门分组统计各个部门的男女个数

0: jdbc:hive2://hadoop2:10000> select dept_id,

. . . . . .> sum(case sex when '男' then 1 else 0 end) as man,

. . . . . .> sum(case sex when '女' then 1 else 0 end) as woman

. . . . . .> from emp_sex group by dept_id;

```
INFO  : Concurrency mode is disabled, not creating a lock manager
+----------+------+--------+
| dept_id  | man  | woman  |
+----------+------+--------+
| A        | 2    | 1      |
| B        | 1    | 2      |
+----------+------+--------+
2 rows selected (12.299 seconds)
0: jdbc:hive2://hadoop2:10000>
```

3-列转行

1.Concat：将任意类型的字段进行拼接

0: jdbc:hive2://hadoop2:10000> select concat(ename," is ",sal) from emp;

```
INFO  : Concurrency mode is disabled, not creating a lock manager
+------------------+
|       _c0        |
+------------------+
| SMITH is 800.0   |
| ALLEN is 1600.0  |
| WARD is 1250.0   |
| JONES is 2975.0  |
| MARTIN is 1250.0 |
| BLAKE is 2850.0  |
| CLARK is 2450.0  |
| SCOTT is 3000.0  |
| KING is 5000.0   |
| TURNER is 1500.0 |
| ADAMS is 1100.0  |
| JAMES is 950.0   |
| FORD is 3000.0   |
| MILLER is 1300.0 |
+------------------+
14 rows selected (0.798 seconds)
```

2.Concat_ws：按照指定格式进行拼接（字段必须是string or array）

0: jdbc:hive2://hadoop2:10000> select concat_ws(' - ',ename,cast(sal as string)) from emp;

```
INFO  : Concurrency mode is disabled, not creating a lock manager
+------------------+
|       _c0        |
+------------------+
| SMITH - 800.0    |
| ALLEN - 1600.0   |
| WARD - 1250.0    |
| JONES - 2975.0   |
| MARTIN - 1250.0  |
| BLAKE - 2850.0   |
| CLARK - 2450.0   |
| SCOTT - 3000.0   |
| KING - 5000.0    |
| TURNER - 1500.0  |
| ADAMS - 1100.0   |
| JAMES - 950.0    |
| FORD - 3000.0    |
| MILLER - 1300.0  |
+------------------+
14 rows selected (7.938 seconds)
0: jdbc:hive2://hadoop2:10000> select concat_ws(' - ',ename,cast(sal as string)) from emp;
```

3.Collect_set字段去重：

0: jdbc:hive2://hadoop2:10000> select collect_set(ename) from emp;

```
INFO  : Concurrency mode is disabled, not creating a lock manager
+----------------------------------------------------+
|                        _c0                         |
+----------------------------------------------------+
| ["SMITH","ALLEN","WARD","JONES","MARTIN","BLAKE","CLARK","SCOTT","KING","TURNER","ADAMS","JAMES","FORD","MILLER"] |
+----------------------------------------------------+
1 row selected (3.36 seconds)
```

4.语法练习：

需求：对下面的table,列出星座与血型的组合的所有

```
+-------------------+----------------------------+-------------------------+
| person_info.name  | person_info.constellation  | person_info.blood_type  |
+-------------------+----------------------------+-------------------------+
| 孙悟空             | 白羊座                      | A                       |
| 大海               | 射手座                      | A                       |
| 宋宋               | 白羊座                      | B                       |
| 猪八戒             | 白羊座                      | A                       |
| 凤姐               | 射手座                      | A                       |
| 苍老师             | 白羊座                      | B                       |
+-------------------+----------------------------+-------------------------+
6 rows selected (3.783 seconds)
```

```
+--------+-------+
|   cb   | name  |
+--------+-------+
| 白羊座,A  | 孙悟空  |
| 射手座,A  | 大海   |
| 白羊座,B  | 宋宋   |
| 白羊座,A  | 猪八戒  |
| 射手座,A  | 凤姐   |
| 白羊座,B  | 苍老师  |
+--------+-------+
6 rows selected (0.42 seconds)
0: jdbc:hive2://hadoop2:10000> select concat_ws(',',constellation,blood_type) as cb,name from person_info;
```

先求出星座与血型的组合

0: jdbc:hive2://hadoop2:10000> select t1.cb from (select concat_ws(',',constellation,blood_type) as cb,name from person_info) as t1 group by t1.cb;

```
INFO  : OK
INFO  : Concurrency mode is disabled, not creating a lock manager
+---------+
| t1.cb   |
+---------+
| 射手座,A  |
| 白羊座,A  |
| 白羊座,B  |
+---------+
3 rows selected (10.563 seconds)
```

最终答案：

0: jdbc:hive2://hadoop2:10000> select t1.cb, concat_ws('|',collect_set(t1.name)) from (select concat_ws(',',constellation,blood_type) as cb,name from person_info) as t1 group by t1.cb;

```
INFO  : OK
INFO  : Concurrency mode is disabled, not creating a lock manager
+--------+----------+
| t1.cb  |   _c1    |
+--------+----------+
| 射手座,A  | 大海|凤姐     |
| 白羊座,A  | 孙悟空|猪八戒   |
| 白羊座,B  | 宋宋|苍老师    |
+--------+----------+
3 rows selected (42.463 seconds)
```