**一、查找每个用户连续登陆的最大天数**

use girls

create table user\_date(

uid int(20),

login\_date date

);

insert into user\_date

VALUES(201,'20170101')

,(201,'20170102')

,(202,'20170102')

,(202,'20170103')

,(203,'20170103')

,(201,'20170104')

,(202,'20170104')

,(201,'20170105')

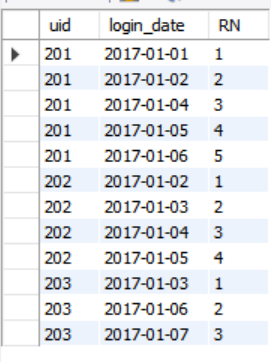
,(202,'20170105')

,(201,'20170106')

,(203,'20170106')

,(203,'20170107');

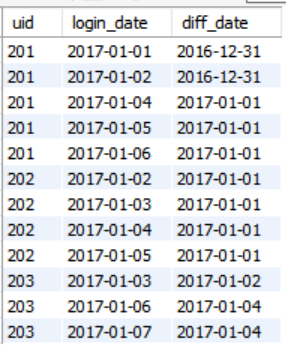
（1）第一步：按照uid分组，login\_date排序，增加一个新字段排名RN

select uid,login\_date,

row\_number() over (partition by uid order by login\_date) RN

from user\_date

（2）第二步：在第一步的基础上增加date\_diff列，即login\_date减去RN



select uid,login\_date,date\_sub(login\_date,interval RN day) as diff\_date

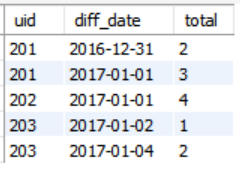
from(

select uid,login\_date,

row\_number() over (partition by uid order by login\_date) RN

from user\_date) t1

（3）得到每个人在每一天的连续登陆天数



select uid,diff\_date,count(1) as total

from(

select uid,login\_date,date\_sub(login\_date,interval RN day) as diff\_date

from(

select uid,login\_date,

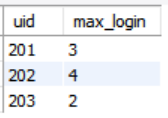
row\_number() over (partition by uid order by login\_date) RN

from user\_date) t1

) t2

group by uid,diff\_date

（4）取出最大值



select uid,max(total) as max\_login

from(

select uid,diff\_date,count(1) as total

from(

select uid,login\_date,date\_sub(login\_date,interval RN day) as diff\_date

from(

select uid,login\_date,

row\_number() over (partition by uid order by login\_date) RN

from user\_date) t1

) t2

group by uid,diff\_date

) t3

group by uid

二、

use girls;

create table student\_test\_info(

id int(20),

student\_id int(20),

student\_name varchar(20),

course\_id int(20),

course\_name varchar(20),

score int(20)

);

insert into student\_test\_info

values(1,201001,"张黛丝",101,"商务统计",73),

(2,201001,"张黛丝",104,"时间序列分析",67),

(3,201001,"张黛丝",105,"随机过程",52),

(4,201001,"李美妮",104,"时间序列分析",61),

(5,201001,"李美妮",102,"应用回归分析",63),

(6,201001,"孙纳德",101,"商务统计",87),

(7,201001,"孙纳德",107,"计量经济学",82);

(1).编写hive sql语句提取“不偏科的学生”名单。

其中“不偏科的学生”的定义是：该学生选的所有课程中，每一门课程的考试分数均高于该课程的平均分。查询的环境为原生hive2.0

Select student\_name,course\_name,score,avg\_score

From(

Select course\_id,avg(score) avg\_score

From student\_test\_info

Group by course\_id

) a

inner join student\_test\_info b on a.course\_id=b.course\_id

Where score>avg\_score;

（2）选出每科排名前2名的同学名字

Select student\_name,course\_name,score,RN

from(

Select \*,dense\_rank() over (partition by course\_id order by score desc) RN

From student\_test\_info) as t1

Where RN=1 or RN=2;