SQL高阶

**一、TRIGGER**

create TRIGGER xxx

after/before insert on 表xx for each row

begin

xxxxxx;

end;

**二、procedure**

create procedure xxx on 表xx

begin

xxxxxx;

end;

调用时 call xx

**三、case when**

case

when then 表达式/结果值

when then

else

end;

**四、表中新增一列属性**

ALTER TABLE “表名” ADD“字段名” “...属性”AFTER “字段名”

**五、WITH ROLLUP**

是用来在分组统计数据的基础上再进行统计汇总，即用来得到group by的汇总信息；

**六、截取日期中的年份、月份、日期**

「1」year(日期) / month（日期） / day（日期）

「2」DATE\_FORMAT(TXN\_DATE, "%m")

**七、类型转换 cast**

CAST函数语法规则是：Cast(字段名 as 转换的类型 )，其中类型可以为：

CHAR[(N)] 字符型

DATE 日期型

DATETIME 日期和时间型

DECIMAL float型

SIGNED int

TIME 时间型

**八、IF ELSE**

if xx then ;

elseif xx then;

else xx;

endif;

九、函数

create function xxx() returns 类型

begin

end;

调用时 select 函数名

【1】RANK()

    在计算排序时，若存在相同位次，会跳过之后的位次。

    例如，有3条排在第1位时，排序为：1，1，1，4······

【2】DENSE\_RANK()

    这就是题目中所用到的函数，在计算排序时，若存在相同位次，不会跳过之后的位次。

    例如，有3条排在第1位时，排序为：1，1，1，2······

【3】ROW\_NUMBER()

    这个函数赋予唯一的连续位次。

    例如，有3条排在第1位时，排序为：1，2，3，4······

窗口函数用法：

<窗口函数> OVER ( [PARTITION BY <列清单> ]

ORDER BY <排序用列清单> ）

【4】字符函数

[1]char\_length(str)

（1）计算单位：字符

（2）不管汉字还是数字或者是字母都算是一个字符

[2]length(str)

（1）计算单位：字节

（2）utf8编码：一个汉字三个字节，一个数字或字母一个字节。

（3）gbk编码：一个汉字两个字节，一个数字或字母一个字节。

concat group\_concat concat\_was

（1）concat(str1, str2,...)

返回结果为连接参数产生的字符串，如果有任何一个参数为null，则返回值为null。

（2）concat\_ws(separator, str1, str2, ...)

第一个参数指定分隔符。需要注意的是分隔符不能为null，如果为null，则返回结果为null。

（3）group\_concat()函数

在有group by的查询语句中，select指定的字段要么就包含在group by语句的后面，作为分组的依据，要么就包含在聚合函数中。

先group 再 concat

lpad/rpad

lpad用制定字符实现左填充长度

rpad用制定字符实现左填充长度

lpad(“被填充的字符串”,返回被填充后的长度,”用来填充的字符”)

instr

返回子串第一次出现的索引位置

tirm

去除字符串左右两侧的制定字符

trim(“需要被去除的字符”，”字符串”)

replace

replace(“字符串”，“被替换的元素”，“替换后的元素”)

【5】数学函数

（1）round四舍五入

round（数字，保留的小数位数）

（2）ceil向上取整

（3）floor向下取整

（4）truncate截断

truncate（小数，保留位数）

（5）mod取模

【6】日期函数

（1）now（）当前系统日期、时间

（2）curdate（）当前日期

（3）curtime()当前时间

（4）获取指定部分

年year(日期)-月month()-日

返回英文：yearname()

（5）str\_to\_date()字符串转换为日期

（6）date\_to\_str()日期转换为字符串

（7）



【7】流程控制函数

（1）IF

IF（条件，成立返回值，不成立返回值）

（2）CASE

CASE

WHEN 。。。 THEN 。。。

ELSE 。。。

END

（3）

【8】