Atitit 视图字段重复得到表的所有字段

Per col get a alias..

CREATE DEFINER=`root`@`%` PROCEDURE `group\_concatEX2`(`tab` varchar(255),`tabas` varchar(255))

BEGIN

# for each block

begin

#Routine body goes here...

-- 定义变量

declare timeout int default 15; -- sql执行超时时间

DECLARE exittag int DEFAULT 0;

DECLARE pid int; declare col VARCHAR(255);

DECLARE sec int;DECLARE sec2 int;

DECLARE CURSOR11 CURSOR FOR select column\_name from information\_schema.columns WHERE table\_name=tab and table\_schema='cp'

;

-- select trx\_mysql\_thread\_id,now()-trx\_started as secs from information\_schema.innodb\_trx;

-- this just like derfer...

DECLARE exit HANDLER FOR NOT FOUND

BEGIN

select 'rows finish hadler ';-- leave loop\_name;

close CURSOR11;

-- cant use leave sttmt..

end;

/\*

DECLARE exit HANDLER FOR SQLEXCEPTION BEGIN

GET DIAGNOSTICS CONDITION 1 @p1 = RETURNED\_SQLSTATE, @p2 = MESSAGE\_TEXT;

select @p1,@p2 msg;

end;

\*/

-- DECLARE CONTINUE HANDLER FOR NOT FOUND SET exittag=1;

-- 打开游标

open CURSOR11;

-- fetch CURSOR11 into pid,sec;

set @t='';

loop\_name:loop

begin

/\*

DECLARE continue HANDLER FOR SQLEXCEPTION BEGIN

GET DIAGNOSTICS CONDITION 1 @p1 = RETURNED\_SQLSTATE, @p2 = MESSAGE\_TEXT,@e3=MYSQL\_ERRNO;

select @p1,@p2 msg,@e3;

end;

\*/

fetch CURSOR11 into col;

-- 当s不等于1，也就是未遍历完时，会一直循环

-- SELECT exittag,col;

set @t=CONCAT(@t,',', tabas,'.',col,' as ',tabas,'\_',col);

-- select @t;

-- insert logx set log1=111;

-- 执行业务逻辑

end;

end loop;

end ;

# end for block

select SUBSTR(@t,2) as colsx ;

set @t='';

select 'finish';

-- 关闭游标

END

以下的althor has alias ,,but in view cant use ,,,,col same... must per col hads alias...

CREATE DEFINER=`root`@`%` PROCEDURE `group\_concatEx`(`tab` varchar(255),`tabas` varchar(255))

BEGIN

#Routine body goes here...

DECLARE rt VARCHAR(2000);

DECLARE t VARCHAR(1000);

set t=CONCAT(',',tabas,'.');

-- select t;

select t;

set @sql=CONCAT(' select GROUP\_CONCAT(COLUMN\_NAME SEPARATOR ''', t, ''')' ,' into @c ');

select @sql;

set @sql= CONCAT(@sql,' from information\_schema.columns WHERE table\_name=''',tab,'''');

select @sql;

PREPARE INSERT\_SQL FROM @sql;

EXECUTE INSERT\_SQL;

select CONCAT(tabas,'.',@c) as outx ;

END