/\*

\*sql中的聚合函数count()参数

\*/

count(1),其实就是计算一共有多少符合条件的行.

1并不是表示第一个字段,而是表示一个固定值.

其实就可以想成表中有这么一个字段,这个字段就是固定值1,count(1),就是计算一共有多少个1..

同理,count(2),也可以,得到的值完全一样,count('x'),count('y')都是可以的.一样的理解方式.在你这个语句理都可以使用,返回的值完全是一样的.就是计数.

count(\*),执行时会把星号翻译成字段的具体名字,效果也是一样的,不过多了一个翻译的动作,比固定值的方式效率稍微低一些.

/\*

\*mysql中各种整形的数字范围

\*/

MySQL整型：TINYINT, SMALLINT, MEDIUMINT, INT, BIGINT的范围

TINYINT：有符号的范围是-128至127，无符号的范围是0到255，2的8 次方

SMALLINT：有符号的范围是-32768至32767，无符号的范围是0到65535，2的16次方

MEDIUMINT：有符号的范围是-8388608至8388607，无符号的范围是0到16777215，2的24次方

INT(integer)：有符号的范围是-2147483648至2147483647，无符号的范围是0到4294967295，2的32次方

BIGINT：有符号的范围是-9223372036854775808至9223372036854775807，无符号的范围是0到18446744073709551615，2的64次方

/\*

\*mysql中分组连接变换分割符

\*/

select id,group\_concat(name separator ';') from aa group by id;

/\*

\*mysql给用户设置权限

\*/

mysql> GRANT ALL PRIVILEGES ON databasename.\* TO "wordpressusername"@"hostname"

-> IDENTIFIED BY "password";

/\*

\*curl原生写法

\*$param 是传递的参数

\*$data 是返回值

\*/

$backUrl = "http://" . C("DOMAIN\_LOG") . "/log.php/".$url ;

$curl = curl\_init();

curl\_setopt($curl, CURLOPT\_URL, $backUrl);

curl\_setopt($curl, CURLOPT\_RETURNTRANSFER, 1);

curl\_setopt($curl, CURLOPT\_POST, 1);

curl\_setopt($curl, CURLOPT\_POSTFIELDS, http\_build\_query($param));

$data = curl\_exec($curl);

curl\_close($curl);

Mysql的if既可以作为表达式用，也可在存储过程中作为流程控制语句使用，如下是做为表达式使用：

IF表达式

IF(expr1,expr2,expr3)

如果 expr1 是TRUE (expr1 <> 0 and expr1 <> NULL)，则 IF()的返回值为expr2; 否则返回值则为 expr3。IF() 的返回值为数字值或字符串值，具体情况视其所在语境而定。

select \*,if(sva=1,"男","女") as ssva from taname where sva != ""

php以图搜图

http://www.welefen.com/similar-image-search-in-php.html

拉丁字符集

http://blog.csdn.net/hemingwang0902/article/details/3621946

mysql中按in排序

select id fromtablewhere id in(2,1,3,5)order bysubstring\_index('2,1,3,5',id,1);

substring\_index(str,delim,count) 字符串截取函数

str 要截取的字符串 delim 截取的分割符 count 截取的数量

返回从字符串str的第count个出现的分隔符delim之后的子串。如果count是正数，返回最后的分隔符到左边(从左边数) 的所有字符。如果count是负数，返回最后的分隔符到右边的所有字符(从右边数)。

那 order by substring\_index(‘2,1,3,5′,id,1) 这个啥意思呢

这个是在字符串’2,1,3,5′里查找id,如果找不到，就返回整个字符串select id fromtablewhere id in(2,1,3,5)order by find\_in\_set(id,'2,1,3,5')

find\_in\_set(str,strlist)

字符串函数

如果字符串str在由N子串组成的表strlist之中，返回一个1到N的值。如果str不是在strlist里面或如果strlist是空字符串，返回0。

SELECT FIND\_IN\_SET('b','a,b,c,d');

->2

所以 find\_in\_set返回的是一个1到N的值，在order by的话，就是按顺序排了。

此句测试中id为主键,type一直为all，不知是有优化方法？

另一种，没有试过（收藏一下）

SQL: select \* from table where id IN (3,6,9,1,2,5,8,7);

这样的情况取出来后，其实，id还是按1,2,3,4,5,6,7,8,9,排序的，但如果我们真要按IN里面的顺序排序怎么办？SQL能不能完成？是否需要取回来后再foreach一下？

其实可以这样

sql: select \* from table where id IN (3,6,9,1,2,5,8,7) order by field(id,3,6,9,1,2,5,8,7);

出来的顺序就是指定的顺序了

项目　中用到

$order = "order by substring\_index('$ids',g.goods\_id,1)";//按in 排序用

Tp中这么写：

$m\_special = M('special');

$w\_special['special\_id'] = array('in',$special\_id);

$f\_special='special\_id,title,img\_url\_single,preheat\_img\_single,open\_time,close\_time,start\_time';

$ids= implode(',',$special\_id);

$o\_special = 'field(special\_id,'.$ids.')';

$list = $m\_special->where($w\_special)->field($f\_special)->order($o\_special)->select();

遍历控制器文件夹，并去掉.,..

if ( $handle = opendir ( '../sysadmin\_app/library/Action' )) {

while ( false !== ( $file = readdir ( $handle ))) {

if ( $file != "." && $file != ".." ) {

echo $file,'<br>' ;

}

}

closedir ( $handle );

}

参数为控制器的名字即可获取该控制器中所有的方法

$data=get\_class\_methods('WorkloadAction');