

itpub SQL 大赛 第二届

参考：<http://www.itpub.net/thread-1820859-1-4.html>

第2届“盛拓传媒杯”SQL数据库编程大赛：考核基本的算法能力，对SQL功能的掌握能力，性能优化能力，并附带考核文档编写能力。

本次大赛将分**2期进行**，每轮为答题期为**2周**，对每期竞赛的参与者，评委将会公正的给予打分，**累积2期大赛的得分最高的TOP10将赠予礼品一份、Oracle技术嘉年华大会参会门票一张及授予“ITPUB SQL数据库编程大师”的称号。**

具体奖项和礼品说明如下：

一等奖-TOP1：500元京东礼品卡，Oracle技术嘉年华大会B票一张、并授予“ITPUB SQL数据库编程大师”的称号，奖励专属徽章以标识身份；
二等奖-TOP2-3：300元京东礼品卡，Oracle技术嘉年华大会B票一张、并授予“ITPUB SQL数据库编程大师”的称号，奖励专属徽章以标识身份；
三等奖-TOP4-6：100元京东礼品卡，Oracle技术嘉年华大会B票一张、并授予“ITPUB SQL数据库编程大师”的称号，奖励专属徽章以标识身份；
四等奖-TOP7-10：ITPUB社区纪念保温壶一个，Oracle技术嘉年华大会B票一张、并授予“ITPUB SQL数据库编程大师”的称号，奖励专属徽章以标识身份；
参与奖：每位参与的会员均可获赠有盛拓传媒的纪念品一份，ITPUB礼券一张、所有参与的会员均可得到一枚参与徽章。

现在开始第2届“盛拓传媒杯”SQL大赛第1期答题，答题时间为：2013年10月10日-10月24日24：00结束，请大家在此时间内答题。

本期题目：徽章交换(100分)

某人有若干个徽章，是否可能在20次交换以内凑齐一套12个生肖章？列出交换次数最少的一种交换方法。

题目要求：用一句SQL实现

交换规则：

本人一次和同一个用户按照清单（badges_exchange表）一个换一个，如本人给用户id=1的用户一个德国，并从该用户处得到一个牛，这算一次交换。

输入和输出格式：

比如本人现在有如下徽章：

鼠,虎,兔,龙,蛇,羊,鸡,狗,猪,德国,法国,巴西,荷兰,足球,篮球,排球,田径,游泳,跳水,钻石（2个）,红宝石,蓝宝石,蔷薇,郁金香
生肖章缺牛马猴。

输入变量是一个用半角逗号隔开的字符串，如果同一徽章有多个，则徽章名称在字符串中重复出现，但位置不一定相邻。如：钻石,红宝石,蓝宝石,蔷薇,郁金香,钻石

本例的输入变量是：

```
VAR v_badges VARCHAR2(1000);
```

```
EXEC :v_badges := '鼠,虎,兔,龙,蛇,羊,鸡,狗,猪,德国,法国,巴西,荷兰,足球,篮球,排球,田径,游泳,跳水,钻石,红宝石,蓝宝石,蔷薇,郁金香,钻石';
```

一徽章交换清单表，保存着人名，愿意的类型（出/入）、愿意出的徽章和愿意入的徽章名称及个数。

/*每个用户都有一个需要的徽章清单，标记为IN，和现有愿意换出的徽章清单，标记为OUT，

并都愿意用自己OUT清单中任意一个徽章与本人交换IN清单中存在的任意一个徽章

清单中可能存在多个同样的徽章，同一个徽章的数量用QUANTITY表示

*/

CREATE TABLE badges_exchange (---- 注：本人不在其中，这些用户都可与本人交换。本人的所有徽章都视为可换出。

```
    user_id      NUMBER
```

```
    , badge_name  VARCHAR2(20)
```

```
    , flag        VARCHAR2(4)      ---- IN 表示愿意入，OUT表示愿意出
```

```
    , quantity    NUMBER          ---- 如果flag是IN，表示最多可以入几个，满了就不再换；如果flag是OUT，表示最多愿意出几个，出完了就没有了
```

```
    , PRIMARY KEY (user_id, badge_name)
```

```
);
```

```
INSERT INTO badges_exchange VALUES (1, '牛', 'OUT', 1);
```

```
INSERT INTO badges_exchange VALUES (1, '德国', 'IN', 1);
```

```
INSERT INTO badges_exchange VALUES (2, '马', 'OUT', 1);
```

```
INSERT INTO badges_exchange VALUES (2, '玫瑰', 'IN', 1);
```

```

INSERT INTO badges_exchange VALUES (3,'玫瑰','OUT',1);
INSERT INTO badges_exchange VALUES (3,'篮球','IN',1);
INSERT INTO badges_exchange VALUES (4,'猴','OUT',1);
INSERT INTO badges_exchange VALUES (4,'射击','IN',1);
INSERT INTO badges_exchange VALUES (5,'射击','OUT',1);
INSERT INTO badges_exchange VALUES (5,'排球','IN',1);

```

--badges表保存本人及其他用户拥有的各种徽章及其类别

```

CREATE TABLE badges (badge_name VARCHAR2(20) PRIMARY KEY,category VARCHAR2(20));
INSERT INTO badges VALUES ('鼠','生肖');
INSERT INTO badges VALUES ('牛','生肖');
INSERT INTO badges VALUES ('虎','生肖');
INSERT INTO badges VALUES ('兔','生肖');
INSERT INTO badges VALUES ('龙','生肖');
INSERT INTO badges VALUES ('蛇','生肖');
INSERT INTO badges VALUES ('马','生肖');
INSERT INTO badges VALUES ('羊','生肖');
INSERT INTO badges VALUES ('猴','生肖');
INSERT INTO badges VALUES ('鸡','生肖');
INSERT INTO badges VALUES ('狗','生肖');
INSERT INTO badges VALUES ('猪','生肖');
INSERT INTO badges VALUES ('德国','世界杯');
INSERT INTO badges VALUES ('法国','世界杯');
INSERT INTO badges VALUES ('巴西','世界杯');
INSERT INTO badges VALUES ('荷兰','世界杯');
INSERT INTO badges VALUES ('意大利','世界杯');
INSERT INTO badges VALUES ('足球','奥运');
INSERT INTO badges VALUES ('篮球','奥运');
INSERT INTO badges VALUES ('排球','奥运');
INSERT INTO badges VALUES ('射击','奥运');
INSERT INTO badges VALUES ('田径','奥运');
INSERT INTO badges VALUES ('游泳','奥运');
INSERT INTO badges VALUES ('跳水','奥运');
INSERT INTO badges VALUES ('钻石','财富');
INSERT INTO badges VALUES ('红宝石','财富');
INSERT INTO badges VALUES ('蓝宝石','财富');
INSERT INTO badges VALUES ('玫瑰','花卉');
INSERT INTO badges VALUES ('蔷薇','花卉');
INSERT INTO badges VALUES ('郁金香','花卉');
commit;

```

注意：不需要考虑错误处理（包括交换的章不在badges表中以及不合理情况，比如本人既出又入同样的徽章，或者没有可出的徽章却要入徽章，或要出的徽章和要入的徽章总数不等，等）

输出格式：

包含3列，各列含义见示例的注释。如果不能在20次内完成交换，输出0行。

若有多种交换方法的次数相同，则输出任意一个交换路径，如本人的徽章输出交换结果如下：

BADGES_OWNED --表示所有交换结束后，最终楼主拥有的所有徽章的清单，用逗号隔开，如有重复徽章则重复出现，徽章的排列顺序不作要求

EXCHANGE_PATH --表示交换过程，每次交换用“用户id:本人换出-本人换入”格式表示，各次交换彼此用半角逗号隔开

EXCHANGE_CNT --表示交换次数

鼠,虎,兔,龙,蛇,羊,鸡,狗,猪,牛,法国,巴西,荷兰,足球,马,猴,田径,游泳,跳水,钻石,红宝石,蓝宝石,蔷薇,郁金香,钻石

1:德国-牛,3:篮球-玫瑰,2:玫瑰-马,5:排球-射击,4:射击-猴

第一期活动评委：newkid, lastwinner ,〇〇

参赛规则：

每个参赛者以附件方式贴出自己的源代码，并将阅读权限设为11~100之间的数值，答案贴出后不允许编辑帖子。编辑过的帖子做无效处理。需要修改答案请发表新的回复，并以同样的方式上传新的答案。

附件名称以SQL大赛第2期(ID).txt来命名，通过rar压缩加密，推荐采取select sys_guid() from dual的方式产生密码，答题者需要在截止时间前将密码以论坛PM给论坛助理童馨

pm链接：<http://www.itpub.net/pm.php?action=send&uid=21267884>

活动流程：

- 1、发布本期活动题目及答题时间范围，大家可以在此期间根据以上参与规则上传自己的参赛程序文档。
- 2、结束答题，进入评审阶段，评审将在7天内对所有参赛程序进行评审，并给予点评。
- 3、公布评审结果及部分作品点评，并开始下一期竞赛。

评审规则：

- 1、由ITPUB论坛助理下载参赛者附件并解压后删除参赛者ID, 将各参赛文档重编序号发布给当期评委。
 - 2、评委抽签获得序号并评审相应序号的答案。运行程序，将运行出正确答案的参赛文档选出。
 - 3、所有评委独立对运行出正确答案的程序打分，把相应分数告知论坛助理。论坛助理取平均数为该答案最后得分。
- 注：出题者负责向评委讲解思路，并配合评委书写点评。

评审依据：

1. 用一句SQL实现，通俗易懂，可读性(10分)（禁止在代码中包含参赛者ID、昵称等可能标识参赛者身份的信息）
 2. 解决问题的思路(10分)
 3. 解决问题的正确性和完善程度(40分)
 4. 性能要求(20分)
 5. 通用性(10分)比如输入变量
 6. 可扩展性(10分)比如扩大数据量、增加数据类别
- 具体标准可根据当期赛题进行适当调整。

特别声明：

出题者和评委需严格保守题目秘密，禁止外泄。禁止评委及同一会员用多个马甲同时参与此次活动，禁止其他一切有违比赛公平的事项。违者一经查实，将取消相关奖励以及比赛参与资格。

本次大赛将分**2期进行，每轮为答题期为2周**，对每期竞赛的参与者，评委将会公正的给予打分，**累积2期大赛的得分最高的TOP10将赠予礼品一份、Oracle技术嘉年华大会参会门票一张及授予“ITPUB SQL数据库编程大师”的称号。**

具体奖项和礼品说明如下：

一等奖-TOP1：500元京东礼品卡，Oracle技术嘉年华大会B票一张、并授予“ITPUB SQL数据库编程大师”的称号，奖励专属徽章以标识身份；
二等奖-TOP2-3：300元京东礼品卡，Oracle技术嘉年华大会B票一张、并授予“ITPUB SQL数据库编程大师”的称号，奖励专属徽章以标识身份；
三等奖-TOP4-6：100元京东礼品卡，Oracle技术嘉年华大会B票一张、并授予“ITPUB SQL数据库编程大师”的称号，奖励专属徽章以标识身份；
四等奖-TOP7-10：ITPUB社区纪念保温壶一个，Oracle技术嘉年华大会B票一张、并授予“ITPUB SQL数据库编程大师”的称号，奖励专属徽章以标识身份；
参与奖：每位参与的会员均可获赠有盛拓传媒的纪念品一份，ITPUB礼券一张、所有参与的会员均可得到一枚参与徽章。

第二期开始设置附加题，除了上述奖励，附加题奖励说明如下：

附加题SQL和PL/SQL各取前3名会员给予如下奖励

SQL第1名： 200元京东礼品卡+背包+2014年数据库技术大会门票B票一张

其余5名： 背包+2014年数据库技术大会门票B票一张

注： 所有做对本题的会员均可得生肖徽章一套，此外，如果是用SQL做对本附加题的会员还可获赠绝版SQL极客徽章一枚
(SQL和PL/SQL都做了的，按SQL的得分参与排名，分数相同的选手，按提交时间先后排名)

现在开始第2届“盛拓传媒杯”SQL大赛第2期答题，答题时间为：2013年10月29日-11月12日24：00结束，请大家在此时间内答题。

本期题目详情：（正题和附加题公用）

在M*N的矩阵中，单元格中的数字表示该单元格周围地雷的数目，有数字的单元格肯定不是地雷。其余的单元格要么是地雷，要么是空位而且四周都没有地雷。

周围的定义为紧挨着的单元格，例如：
若单元格在矩阵的内部，则周围有8个单元格，如图a所示
若单元格在矩阵的四边，则周围有5个单元格，如图b所示
若单元格在矩阵的四角，则周围有3个单元格，如图c所示

```
x x x
x c x
x x x
```

图a

```
x x x
x c x
-----
```

图b

```
| x x
| c x
-----
```

图c

输入输出格式：
用3个变量v_height、v_width、v_cnt表示雷区的长度、宽度和地雷个数，其中v_height、v_width均为大于0且小于32的整数，v_cnt为大于0且小于或等于v_height*v_width的整数。

```
var v_width NUMBER;
EXEC :v_width := 4;
```

```
var v_height NUMBER;
EXEC :v_height := 4;
```

```
var v_cnt NUMBER;
EXEC :v_cnt := 3;
```

用1个字符串变量表示从矩阵第一行至最后一行，每行从第一列到最后一列依次输出
如矩阵（为明显起见，用下划线表示空格，实际做题的输入输出仍用空格）

```
1 1 1 _
2 * 1 _
* 3 2 1
1 2 * 1
```

的字符串表示为：

```
VAR v_str VARCHAR2(1000);
exec :v_str := '111 2 1 32112 1'
```

本期正题：挖地雷之标出有地雷的格子（100分）
题目要求：用一句SQL实现

有若干地雷分布在图中，它们都有*标记，请把矩阵中的数字标出来。
如输入字符串为：

```
VAR v_str VARCHAR2(1000);
exec :v_str := '      * *      * '
```

输出格式：在输入字符串中有地雷的位置保留'*',同时对它周围的单元格标上地雷数，若单元格周围没有地雷，则保持空格，对上述输入，则输出：

```
111 2*1 *32112*1
```

本期附加题：挖地雷之标出有地雷的格子
注：用SQL的满分为100分，用PL/SQL的满分为90分。PL/SQL的按原评分标准或等价评分标准得出总分后再乘以0.9计算
题目要求：用一句SQL实现或用一个PL/SQL函数实现。

(如果用PL/SQL实现,则函数必须为

```
create or replace function winmine(p_str varchar2,p_width NUMBER,p_height NUMBER,p_cnt NUMBER)
return varchar2
as
begin
...
return ...;
end;
/
```

结果在sqlplus用select winmine(:v_str,:v_width,:v_height,:v_cnt) from dual;输出

如果有相应授权需要把grant语句一并给出,用户名为scott

)

有C个地雷分布在图中,根据输入字符串提供的格子周围地雷数,把所有埋了地雷的格子标出来。

如输入字符串为:

```
VAR v_str VARCHAR2(1000);
```

```
exec :v_str := '111 2 1 32112 1'
```

输入地雷数为:

```
var v_cnt NUMBER;
```

```
EXEC :v_cnt := 3;
```

输出格式:在输入字符串中有地雷的位置标上'*',同时保留它周围的单元格标的地雷数,若单元格周围没有地雷,则保持空格,对上述输入,则输出:

```
111 2*1 *32112*1
```

本题不需要考虑错误处理,如果输入错误(比如地雷数输入变量和实际不符、雷区不是矩形、字符串中标的地雷数字错误),就允许任何输出。

数据库平台: Oracle 11g R2 版本(不能用12c,因为它有在sql语句中编写自定义函数功能)

第二期活动评委: newkid, lastwinner ,〇〇

参赛规则:

每个参赛者以附件方式贴出自己的源代码,并将阅读权限设为11~100之间的数值,答案贴出后不允许编辑帖子。编辑过的帖子做无效处理。需要修改答案请发表新的回复,并以同样的方式上传新的答案。

附件名称以SQL大赛第2期(ID).txt来命名,通过rar压缩加密,推荐采取select sys_guid() from dual的方式产生密码,答题者需要在截止时间前将密码以论坛PM给论坛助理童馨pm链接: <http://www.itpub.net/pm.php?action=send&uid=21267884>

活动流程:

- 1、发布本期活动题目及答题时间范围,大家可以在此期间根据以上参与规则上传自己的参赛程序文档。
- 2、结束答题,进入评审阶段,评审将在7天内对所有参赛程序进行评审,并给予点评。
- 3、公布评审结果及部分作品点评。

评审规则:

- 1、由ITPUB论坛助理下载参赛者附件并解压后删除参赛者ID,将各参赛文档重编序号发布给当期评委。
- 2、评委抽签获得序号并评审相应序号的答案。运行程序,将运行出正确答案的参赛文档选出。
- 3、所有评委独立对运行出正确答案的程序打分,把相应分数告知论坛助理。论坛助理取平均数为该答案最后得分。

注:出题者负责向评委讲解思路,并配合评委书写点评。

评审依据:

1. 用一句SQL实现(附加题也可以用一个PL/SQL函数实现),通俗易懂,可读性(10分)(禁止在代码中包含参赛者ID、昵称等可能标识参赛者身份的信息)
2. 解决问题的思路(10分)
3. 解决问题的正确性和完善程度(40分)
4. 性能要求(20分)
5. 通用性(10分)比如输入变量
6. 可扩展性(10分)比如扩大数据量、增加数据类别具体标准可根据当期赛题进行适当调整。

特别声明：

出题者和评委需严格保守题目秘密，禁止外泄。禁止评委及同一会员用多个马甲同时参与此次活动，禁止其他一切有违比赛公平的事项。违者一经查实，将取消相关奖励以及比赛参与资格。

注意：答题时间到禁止诸位会员在本贴贴出代码进行讨论，以免影响评委进行公正打分！