

**软件学院实验任务书**

题 目： 足球联赛评分系统

专 业： 软件工程

班 级： 软件工程1807班

姓 名： 翟翊君

学 号： 8002118184

完成人数： 1人

起讫日期： 2019. 10. 21 — 2019.11. 18

完成时间： 2019年11月16日

**目录**

[一、引言 3](#_Toc24784)

[1. 编写目的 3](#_Toc16707)

[2. 术语表 3](#_Toc23347)

[二、需求规约 3](#_Toc22911)

[1. 业务描述 3](#_Toc16503)

[1.1数据库创建的背景 3](#_Toc21353)

[1.2 数据库系统要解决的问题 3](#_Toc15028)

[1.3 数据库系统要完成的业务流程 3](#_Toc32095)

[2. 需求分析 4](#_Toc234)

[2.1 业务的内在联系 4](#_Toc13329)

[2.2 创建数据库系统的目的、目标 4](#_Toc16373)

[三、数据库环境说明 4](#_Toc20528)

[四、数据库的命名规则 4](#_Toc22523)

[1. 数据库对象命名规则 4](#_Toc23194)

[2. 数据项编码规则 5](#_Toc31335)

[五、概念结构设计 5](#_Toc753)

[六、逻辑结构设计 7](#_Toc21685)

[七、物理结构设计 8](#_Toc29617)

[1. 表汇总 8](#_Toc5782)

[2. 表 9](#_Toc14410)

[3. 存储过程、函数及触发器的设计 13](#_Toc22068)

[八、 参考文献 15](#_Toc19663)

[九、 附录 16](#_Toc11171)

## 一、引言

### 1. 编写目的

本文档是概要设计文档的组成部分，编写数据库设计文档的目的是：明确数据库的表名、字段名等数据信息，用来指导后期数据库脚本的开发。本文档的读者对象是需求人员、系统设计人员、开发人员、测试人员。

### 术语表

|  |  |  |
| --- | --- | --- |
| **序号** | **术语或缩略语** | **说明性定义** |
| 1 | team | 球队 |
| 2 | player | 队员 |
| 3 | schedule | 赛程表 |
| 4 | scoreofteam | 球队积分榜 |
| 5 | goalofpalyer | 队员进球信息 |
| 6 | judge | 裁判处罚信息 |
| 7 | shooterlist | 队员射手榜 |

## 二、需求规约

### 1. 业务描述

**1.1数据库创建的背景**

足球比赛往往会产生许多的比赛数据，如球队的比分，进球数、失球数、球员的罚单信息等。除此之外，足球比赛往往还会涉及球队的基本信息以及球员的基本信息。这些信息少则几千条，多则几万条。为了对比赛数据进行有效地分类管理以及对比赛数据的后期计算，可以考虑使用数据库进行数据处理。

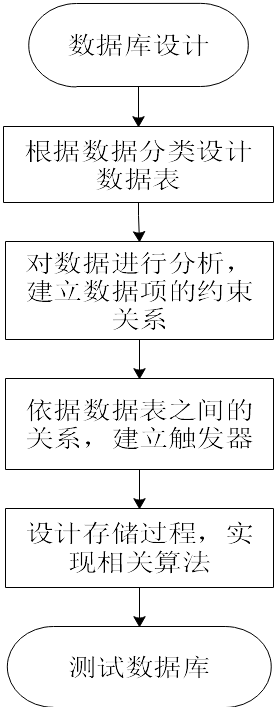
**1.2 数据库系统要解决的问题**

(1) 对比赛数据进行分类管理。

(2) 建立数据集之间的联系。

(3) 设计算法，完成数据的计算。

**1.3 数据库系统要完成的业务流程**



### 2. 需求分析

**2.1 业务的内在联系**

足球比赛的数据表之间的关系的实现依赖于触发器的建立，触发器根据数据表数据的插入、修改、删除，触发不同的事件。触发器所依赖的算法，由存储过程单独来实现。

**2.2 创建数据库系统的目的、目标**

创建数据的目的是为了更好的存储和管理数据，又由于数据库具有非常方便插入、修改、删除、查询等功能，所以建立数据库系统是处理数据的较好选择。

## 三、数据库环境说明

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 数据库  实例 | 数据库  系统 | 数据库部署环境 | 数据库设计工具 | 数据库存放位置 | 说明 |
|  | Mysql,版本8.0 | 软硬件、网络环境 | PowerDesigner |  |  |

## 四、数据库的命名规则

### 数据库对象命名规则

所有的数据库命名都是以模块的缩写加上具体表的英文词汇组成，这样能够统一数据库表的命名，也能够更好的规范数据库表命名。

### 数据项编码规则

组别采用大写英文字母A、B、C、D进行编码。

比赛轮数采用数字1-15进行编码。

组别+比赛轮数可以唯一确定一场比赛。

## 五、概念结构设计

足球联赛评分系统的实体-属性图。

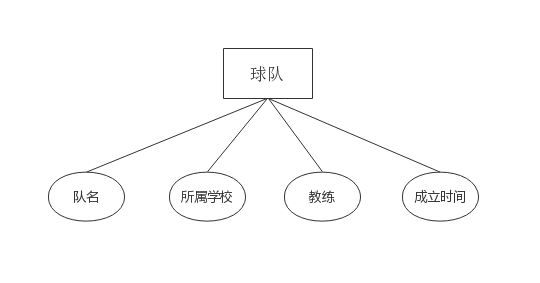


图1 球队实体-属性图

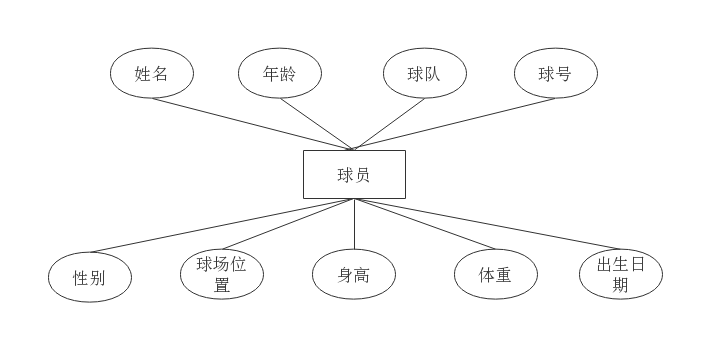


图2 球员实体-属性图

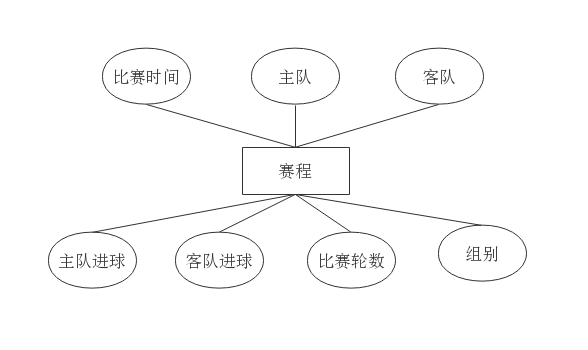


图3 赛程实体-属性图

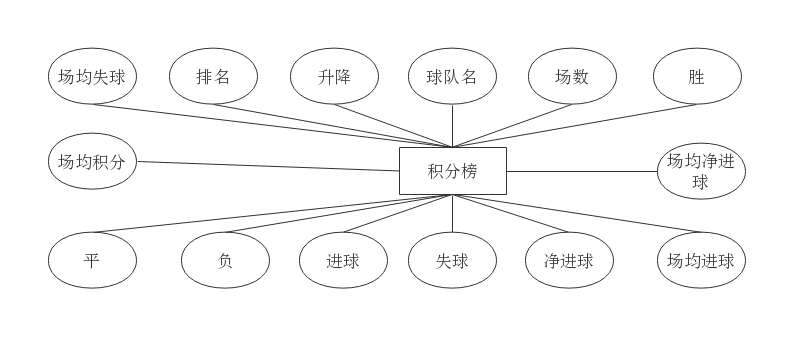


图4 积分榜实体-属性图

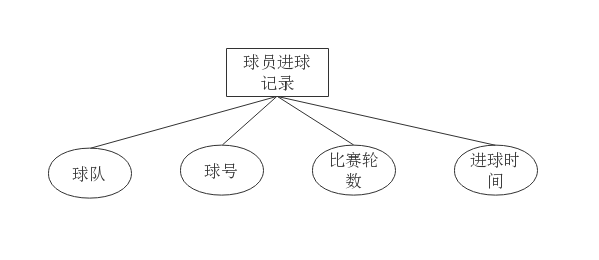


图5 球员进球实体-属性图

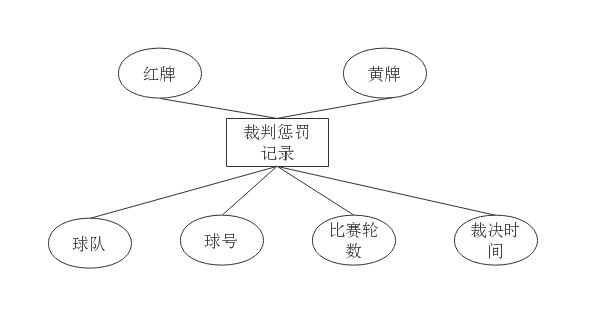


图6 裁判惩罚实体-属性图

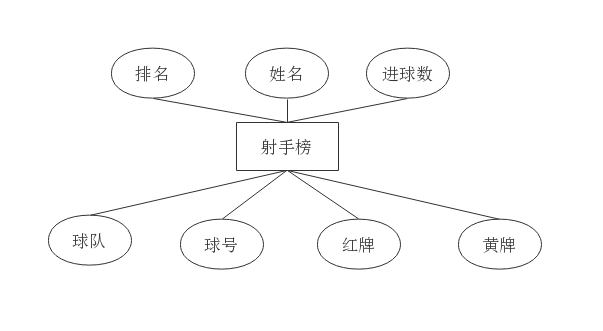


图7 射手榜实体-属性图

## 六、逻辑结构设计

足球联赛评分系统的E-R图。

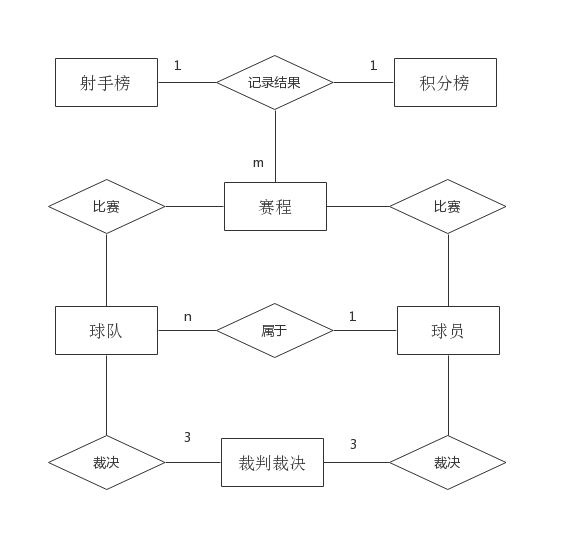


图8 足球联赛评分系统的E-R图

## 七、物理结构设计

### 1. 表汇总

|  |  |
| --- | --- |
| **表名** | **功能说明** |
| team | 记录队名、所属学院或大学、教练、成立时间 |
| player | 记录姓名、年龄、球队、球号、性别、位置、身高、体重、出生日期 |
| schedule | 记录比赛时间、主队、客队、主队进球数、客队进球数、轮数、组别 |
| scoreofteam | 记录排名、升降、队名、场数、胜、平、负、进球数、失球数、净球数、场均进球、场均失球、场均净、场均积分 |
| goalofplayer | 记录队名、球号、轮数、进球时间 |
| judue | 记录队名、球号、红牌、黄牌、轮数、惩罚发布时间 |
| shooterlist | 记录排名、姓名、队名、球号、进球数、红牌、黄牌 |

### 2. 表

**表 1**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **表名** | | team | | | | | |
| **数据库用户** | | 数据库管理人员 | | | | | |
| **主键** | | name 队名 | | | | | |
| **其他排序字段** | | college、coach、seTime | | | | | |
| **序号** | **字段名称** | **数据类型(精度范围)** | **非空约束Y/N** | **唯一约束Y/N** | **区别度** | **默认值** | **约束条件/说明** |
| 1 | Name | varchar(20) | N | N | 0 | 无 | 主键 |
| 2 | College | varchar(30) | N | N | 0 | Null | 无 |
| 3 | Coach | varchar(10) | N | N | 0 | 无 | 无 |
| 4 | setTime | timestamp | N | N | 0 | 无 | 无 |
| **SQL脚本** | | create table team(  name varchar(20) primary key,  college varchar(30) default null,  coach varchar(10),  setTime timestamp  ); | | | | | |

**表 2**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **表名** | | player | | | | | |
| **数据库用户** | | 数据库管理人员 | | | | | |
| **主键** | | team、numbers | | | | | |
| **其他排序字段** | | name、age、gender、position、height、weight、birthday | | | | | |
| **序号** | **字段名称** | **数据类型(精度范围)** | **非空约束Y/N** | **唯一约束Y/N** | **区别度** | **默认值** | **约束条件/说明** |
| 1 | Name | varchar(20) | N | N | 0 | 无 | 无 |
| 2 | Age | int | N | N | 0 | 无 | 无 |
| 3 | Team | varchar(20) | N | N | 0 | 无 | 主键 |
| 4 | Numbers | varchar(20) | N | N | 0 | 无 | 主键 |
| 5 | Gender | varchar(20) | N | N | 0 | 无 | 无 |
| 6 | Position | varchar(20) | N | N | 0 | 无 | 无 |
| 7 | Height | varchar(20) | N | N | 0 | 无 | 无 |
| 8 | Weight | varchar(20) | N | N | 0 | 无 | 无 |
| 9 | birthday | timestamp | N | N | 0 | 无 | 无 |
| **SQL脚本** | | create table player(  name varchar(20),  age int,  team varchar(20),  numbers varchar(20),  gender varchar(20),  position varchar(20) check(position in('守门员','后卫','前卫','前锋')),  height varchar(20),  weight varchar(20),  birthdate timestamp,  primary key(team,numbers)  ); | | | | | |

**表 3**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **表名** | | Schedule | | | | | |
| **数据库用户** | | 数据库管理人员 | | | | | |
| **主键** | | 无 | | | | | |
| **其他排序字段** | | time、hometeam、visitingteam、homescore、visitingscore、turnone、 turntwo | | | | | |
| **序号** | **字段名称** | **数据类型(精度范围)** | **非空约束Y/N** | **唯一约束Y/N** | **区别度** | **默认值** | **约束条件/说明** |
| 1 | time | timestamp | N | N | 0 | 无 | 无 |
| 2 | hometeam | varchar(20) | N | Y | 0 | 无 | 唯一 |
| 3 | visitingteam | varchar(20) | N | Y | 0 | 无 | 唯一 |
| 4 | homescore | int | Y | N | 0 | 无 | 非空 |
| 5 | visitingscore | int | Y | N | 0 | 无 | 非空 |
| 6 | turnone | int | Y | N | 0 | 无 | 非空 |
| 7 | turntwo | varchar(1) | Y | N | 0 | 无 | 非空 |
| **SQL脚本** | | create table schedule(  time timestamp,  hometeam varchar(20),  visitingteam varchar(20),  homescore int not null,  visitingscore int not null,  turnone int not null,  turntwo varchar(1) not null,  unique(hometeam,visitingteam)  ); | | | | | |

**表 4**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **表名** | | scoreofteam | | | | | |
| **数据库用户** | | 数据库管理人员 | | | | | |
| **主键** | | Teamname | | | | | |
| **其他排序字段** | | Ranking、changes、matchs、win、even、beaten、goal、lost、net、avergoal、averlost、avernet、averpoint | | | | | |
| **序号** | **字段名称** | **数据类型(精度范围)** | **非空约束Y/N** | **唯一约束Y/N** | **区别度** | **默认值** | **约束条件/说明** |
| 1 | ranking | int | Y | N | 0 | 0 | 非空 |
| 2 | changes | int | Y | N | 0 | 0 | 非空 |
| 3 | teamname | varchar(20) | Y | N | 0 | 无 | 主键 |
| 4 | matchs | int | Y | N | 0 | 无 | 非空 |
| 5 | win | int | Y | N | 0 | 无 | 非空 |
| 6 | even | int | Y | N | 0 | 无 | 非空 |
| 7 | beaten | int | Y | N | 0 | 无 | 非空 |
| 8 | goal | int | Y | N | 0 | 无 | 非空 |
| 9 | lost | int | Y | N | 0 | 无 | 非空 |
| 10 | net | int | Y | N | 0 | 无 | 非空 |
| 11 | avergoal | float | Y | N | 0 | 无 | 非空 |
| 12 | averlost | float | Y | N | 0 | 无 | 非空 |
| 13 | avernet | float | Y | N | 0 | 无 | 非空 |
| 14 | averpoint | float | Y | N | 0 | 无 | 非空 |
| **SQL脚本** | | create table scoreofteam(  ranking int not null default 0, #排名  changes int not null default 0, #排名升降  teamname varchar(20) primary key, #队名  matchs int not null, #赛  win int not null, # 胜  even int not null, # 平  beaten int not null,#负  goal int not null,#得  lost int not null,#失  net int not null,#净  avergoal float not null,#均得  averlost float not null,#均失  avernet float not null,#均净  averpoint float not null# 均积分  ); | | | | | |

**表 5**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **表名** | | Goalofplayer | | | | | |
| **数据库用户** | | 数据库管理人员 | | | | | |
| **主键** | | Teamname、numbers、goaltime、turn | | | | | |
| **其他排序字段** | | 无 | | | | | |
| **序号** | **字段名称** | **数据类型(精度范围)** | **非空约束Y/N** | **唯一约束Y/N** | **区别度** | **默认值** | **约束条件/说明** |
| 1 | teamname | varchar(20) | Y | N | 0 | 无 | 非空，主键 |
| 2 | numbers | varchar(20) | Y | N | 0 | 无 | 非空、主键 |
| 3 | turn | int | Y | N | 0 | 无 | 非空、主键 |
| 4 | goaltime | timestamp | Y | N | 0 | 无 | 非空、主键 |
| **SQL脚本** | | create table goalofplayer(  teamname varchar(20) not null,  numbers varchar(20) not null,  turn int not null,  goaltime timestamp not null,  primary key (teamname,numbers,turn,goaltime)  ); | | | | | |

**表 6**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **表名** | | shooterlist | | | | | |
| **数据库用户** | | 数据库管理人员 | | | | | |
| **主键** | | Teamname、numbers | | | | | |
| **其他排序字段** | | Ranking、name、goals、redcard、yellowcard | | | | | |
| **序号** | **字段名称** | **数据类型(精度范围)** | **非空约束Y/N** | **唯一约束Y/N** | **区别度** | **默认值** | **约束条件/说明** |
| 1 | Ranking | int | Y | N | 0 | 0 | 非空 |
| 2 | name | varchar(20) | Y | N | 0 | 无 | 非空 |
| 3 | teamname | varchar(20) | N | N | 0 | 无 | 主键 |
| 4 | numbers | varchar(20) | N | N | 0 | 无 | 主键 |
| 5 | goals | int | Y | N | 0 | 无 | 非空 |
| 6 | redcard | int | Y | N | 0 | 无 | 非空 |
| 7 | yellowcard | int | Y | N | 0 | 无 | 非空 |
| **SQL脚本** | | create table shooterlist(  ranking int not null default 0, #排名  name varchar(20) not null,  teamname varchar(20),#队名  numbers varchar(20),  goals int not null,#得  redcard int not null,  yellowcard int not null,  primary key(teamname,numbers)  ); | | | | | |

**表 7**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **表名** | | Judge | | | | | |
| **数据库用户** | | 数据库管理人员 | | | | | |
| **主键** | | Teamname、numbers、redcard、yellowcard、turn、judgetime | | | | | |
| **其他排序字段** | | 无 | | | | | |
| **序号** | **字段名称** | **数据类型(精度范围)** | **非空约束Y/N** | **唯一约束Y/N** | **区别度** | **默认值** | **约束条件/说明** |
| 1 | teamname | varchar(20) | Y | N | 0 | 无 | 主键 |
| 2 | numbers | varchar(20) | Y | N | 0 | 无 | 主键 |
| 3 | redcard | int | Y | N | 0 | 无 | 主键 |
| 4 | yellowcard | int | Y | N | 0 | 无 | 主键 |
| 5 | turn | int | Y | N | 0 | 无 | 主键 |
| 6 | judgetime | timestamp | Y | N | 0 | 无 | 主键 |
| **SQL脚本** | | create table judge(  teamname varchar(20) not null,  numbers varchar(20) not null,  redcard int not null,  yellowcard int not null,  turn int not null,  judgetime timestamp not null,  primary key(teamname,numbers,redcard,yellowcard,turn,judgetime)  ); | | | | | |

### 3. 存储过程、函数及触发器的设计

**3. 1触发器的设计**

**触发器一:** 插入球队触发插入积分榜。当向球队表Team中插入一行数据时，会自动向积分榜scoreofteam中插入一条对应的数据。

1. **create** **trigger** teaminsert
2. **after** **insert** **on** team
3. **for** each row
4. **insert** **into** scoreofteam(ranking,changes,teamname,matchs,win,even,bea
5. ten,goal,lost,net,avergoal,averlost,avernet,averpoint) **values**(0,0,ne
6. w.**name**,0,0,0,0,0,0,0,0,0,0,0);

**触发器二:** 删除球队触发删除射手榜、球员、比赛进程、积分榜。

1. delimiter $
2. **create** **trigger** teamdel
3. before **delete** **on** team
4. **for** each row
5. **begin**
6. **delete** **from** shooterlist **where** teamname=old.**name**;
7. **delete** **from** player **where** team=old.**name**;
8. **delete** **from** schedule **where** hometeam=old.**name** or visitingteam=old.**name**;
9. **delete** **from** scoreofteam **where** teamname=old.**name**;
10. **end**$
11. delimiter ;

**触发器三:** 插入比赛进程时调整积分榜。见附录

**触发器四:** 插入进球时调整射手榜(进球信息，更新排名)。见附录

**触发器五:** 修改球队名称(或球队信息)时自动调整相关表格中的球队名称

1. delimiter $
2. **create** **trigger** TeamUpdate **after** **update** **on** team **for** each row
3. **begin**
4. if old.**name**!= new.**name** **then**
5. **delete** **from** scoreofteam **where** teamname=new.**name**;
6. **update** goalofplayer **set** teamname=new.**name** **where** teamname=old.**name**;
7. **update** player **set** team=new.**name** **where** team=old.**name**;
8. **update** schedule **set** hometeam=new.**name** **where** hometeam=old.**name**;
9. **update** schedule **set** visitingteam=new.**name** **where** visitingteam=old.**name**;
10. **update** scoreofteam **set** teamname=new.**name** **where** teamname=old.**name**;
11. **update** shooterlist **set** teamname=new.**name** **where** teamname=old.**name**;
12. setTime=new.setTime **where** **name**=old.**name**;
13. **end** if;
14. **end**$
15. delimiter ;

**触发器六:** 插入比赛处罚信息时调整射手榜

1. delimiter $
2. **create** **trigger** judge\_insert **after** **insert** **on** judge **for** each row
3. **begin**
4. if not exists(**select** **name** **from** shooterlist **where** teamname=new.teamna me and numbers=new.numbers) **then**
5. **insert** **into** shooterlist **values**(0,(**select** **name** **from** player **where** team=new.teamname and numbers=new.numbers),new.teamname,new.numbers,0,0,0);
6. **end** if;
8. **update** shooterlist **set** redcard=new.redcard+redcard, yellowcard=new.yellowcard+yellowcard **where** teamname=new.teamname and numbers=new.numbers;
9. **end**$
10. delimiter ;

**3.2 存储过程的设计**

**存储过程一:** 利用游标对名次进行处理

1. delimiter $
2. **drop** **procedure** if exists updateRanking\_score;
3. **create** **procedure** updateRanking\_score\_for\_female()
4. **begin**
5. **declare** temp\_teamname **varchar**(20);
6. **declare** temp\_ranking **int**;
7. **declare** done **int** **default** **false**;
8. **declare** scoreofteam\_cursor **cursor** **for** **select** teamname **from** scoreofteam **where** teamname in(**select** **distinct** team **from** player **where** gender='女')  **order** **by** averpoint **desc**,net **desc**,goal **desc**;
9. **declare** **continue** HANDLER **for** not found **set** done = **true**;
10. **set** temp\_ranking=0;
11. **open** scoreofteam\_cursor;
12. **fetch** scoreofteam\_cursor **into** temp\_teamname;
13. while(not done) do
14. **set** temp\_ranking=temp\_ranking+1;
15. **update** scoreofteam **set** changes = ranking-temp\_ranking, ranking=temp\_ranking **where** teamname=temp\_teamname;
16. **fetch** scoreofteam\_cursor **into** temp\_teamname;
17. **end** while;
18. **close** scoreofteam\_cursor;
19. **end**$
20. delimiter ;

**存储过程二：**见附录

**存储过程三：**见附录

**存储过程四：**见附录

**存储过程五：**见附录

**存储过程六：**见附录

## 参考文献

1. 刘志宏,向东,宿营.Java程序设计教程(修订版)[M].北京.航空工业出版社.2010.7

## 附录

触发器三

1. delimiter $
2. **create** **trigger** schedule\_insert before **insert** **on** schedule **for** each row
3. **begin**
4. if not exists(**select** teamname **from** scoreofteam **where** teamname=new.hometeam) **then**
5. **insert** **into** scoreofteam **values**(0,0,new.hometeam,0,0,0,0,0,0,0,0,0,0,0);
6. **end** if;
7. if not exists(**select** teamname **from** scoreofteam **where** teamname=new.visitingteam) **then**
8. **insert** **into** scoreofteam **values**(0,0,new.visitingteam,0,0,0,0,0,0,0,0,0,0,0);
9. **end** if;
10. if new.homescore>new.visitingscore **then**
11. **update** scoreofteam
12. **set**
13. matchs=matchs+1,
14. win=win+1,
15. goal=goal+new.homescore,
16. lost=lost+new.visitingscore,
17. net=net+(new.homescore-new.visitingscore),
18. avergoal = (goal \* 1.0) / matchs,
19. averlost = (lost \* 1.0)/ matchs,
20. avernet = (net \* 1.0)/ matchs,
21. averpoint = (win \* 3.0 + even)/ matchs
22. **where** teamname=new.hometeam;
23. **update** scoreofteam
24. **set**
25. matchs=matchs+1,
26. beaten = beaten+1,
27. goal = goal+new.visitingscore,
28. lost = lost+new.homescore,
29. net=net+(new.visitingscore-new.homescore),
30. avergoal = (goal \* 1.0) / matchs,
31. averlost = (lost \* 1.0)/ matchs,
32. avernet = (net \* 1.0)/ matchs
33. **where** teamname=new.visitingteam;
34. elseif new.homescore=new.visitingscore **then**
35. **update** scoreofteam
36. **set**
37. matchs=matchs+1,
38. even=even+1,
39. goal=goal+new.homescore,
40. lost=lost+new.visitingscore,
41. avergoal = (goal \* 1.0) / matchs,
42. averlost = (lost \* 1.0)/ matchs,
43. averpoint = (win \* 3.0 + even)/ matchs
44. **where** teamname=new.hometeam;
45. **update** scoreofteam
46. **set**
47. matchs=matchs+1,
48. even=even+1,
49. goal=goal+new.visitingscore,
50. lost=lost+new.homescore,
51. avergoal = (goal \* 1.0) / matchs,
52. averlost = (lost \* 1.0)/ matchs,
53. averpoint = (win \* 3.0 + even)/ matchs
54. **where** teamname=new.visitingteam;
55. **else**
56. **update** scoreofteam
57. **set**
58. matchs=matchs+1,
59. beaten=beaten+1,
60. goal=goal+new.homescore,
61. lost=lost+new.visitingscore,
62. net=net+(new.homescore-new.visitingscore),
63. avergoal = (goal \* 1.0) / matchs,
64. averlost = (lost \* 1.0)/ matchs,
65. avernet = (net \* 1.0)/ matchs
66. **where** teamname=new.hometeam;
67. **update** scoreofteam
68. **set**
69. matchs=matchs+1,
70. win=win+1,
71. goal = goal+new.visitingscore,
72. lost = lost+new.homescore,
73. net=net+(new.visitingscore-new.homescore),
74. avergoal = (goal \* 1.0) / matchs,
75. averlost = (lost \* 1.0)/ matchs,
76. avernet = (net \* 1.0)/ matchs,
77. averpoint = (win \* 3.0 + even)/ matchs
78. **where** teamname=new.visitingteam;
79. **end** if;
80. if new.hometeam in(**select** **distinct** team **from** player **where** gender='女') **then**
81. call updateRanking\_score\_for\_female();
82. elseif new.hometeam in(**select** **name** **from** team **where** college=NULL) **then**
83. call updateRanking\_score\_for\_adult();
84. **else**
85. call updateRanking\_score\_for\_male();
86. **end** if;
87. **end**$
88. delimiter ;

触发器四

1. delimiter $
2. **create** **trigger** goal\_insert **after** **insert** **on** goalofplayer **for** each row
3. **begin**
4. if not exists(**select** **name** **from** shooterlist **where** teamname=new.teamname and numbers=new.numbers) **then**
5. **insert** **into** shooterlist **values**(0,(**select** **name** **from** player **where** team=new.teamname and numbers=new.numbers),new.teamname,new.numbers,0,0,0);
6. **end** if;
8. **update** shooterlist **set** goals=goals+1 **where** teamname=new.teamname and numbers=new.numbers;
9. if new.teamname in(**select** **distinct** team **from** player **where** gender='女') **then**
10. call updateRanking\_shooter\_for\_female();
11. elseif new.teamname in(**select** **name** **from** team **where** college=NULL) **then**
12. call updateRanking\_shooter\_for\_adult();
13. **else**
14. call updateRanking\_shooter\_for\_male();
15. **end** if;
16. **end**$
17. delimiter ;

存储过程二

1. delimiter $
2. **drop** **procedure** if exists updateRanking\_score;
3. **create** **procedure** updateRanking\_score\_for\_male()
4. **begin**
5. **declare** temp\_teamname **varchar**(20);
6. **declare** temp\_ranking **int**;
7. **declare** done **int** **default** **false**;
8. **declare** scoreofteam\_cursor **cursor** **for** **select** teamname **from** scoreofteam
9. **where** teamname not in(**select** **distinct** team **from** player **where** gender='女') and teamname not in(**select** **name** **from** team **where** college='无')
10. **order** **by** averpoint **desc**,net **desc**,goal **desc**;
11. **declare** **continue** HANDLER **for** not found **set** done = **true**;
12. **set** temp\_ranking=0;
13. **open** scoreofteam\_cursor;
14. **fetch** scoreofteam\_cursor **into** temp\_teamname;
15. while(not done) do
16. **set** temp\_ranking=temp\_ranking+1;
17. **update** scoreofteam **set** changes = ranking-temp\_ranking, ranking=temp\_ranking **where** teamname=temp\_teamname;
18. **fetch** scoreofteam\_cursor **into** temp\_teamname;
19. **end** while;
20. **close** scoreofteam\_cursor;
21. **end**$
22. delimiter ;

存储过程三

1. delimiter $
2. **drop** **procedure** if exists updateRanking\_score;
3. **create** **procedure** updateRanking\_score\_for\_adult()
4. **begin**
5. **declare** temp\_teamname **varchar**(20);
6. **declare** temp\_ranking **int**;
7. **declare** done **int** **default** **false**;
8. **declare** scoreofteam\_cursor **cursor** **for** **select** teamname **from** scoreofteam **where** teamname in(**select** **name** **from** team **where** college='无') **order** **by** averpoint **desc**,net **desc**,goal **desc**;
9. **declare** **continue** HANDLER **for** not found **set** done = **true**;
10. **set** temp\_ranking=0;
11. **open** scoreofteam\_cursor;
12. **fetch** scoreofteam\_cursor **into** temp\_teamname;
13. while(not done) do
14. **set** temp\_ranking=temp\_ranking+1;
15. **update** scoreofteam **set** changes = ranking-temp\_ranking, ranking=temp\_ranking **where** teamname=temp\_teamname;
16. **fetch** scoreofteam\_cursor **into** temp\_teamname;
17. **end** while;
18. **close** scoreofteam\_cursor;
19. **end**$
20. delimiter ;

存储过程四

1. delimiter $
2. **drop** **procedure** if exists updateRanking\_shooter;
3. **create** **procedure** updateRanking\_shooter\_for\_female()
4. **begin**
5. **declare** temp\_ranking **int**;
6. **declare** temp\_teamname **varchar**(20);
7. **declare** temp\_numbers **varchar**(20);
8. **declare** done **int** **default** **false**;
9. **declare** shooterlist\_cursor **cursor** **for** **select** teamname,numbers **from** shooterlist **where** teamname in(**select** **distinct** team **from** player **where** gender='女') **order** **by** goals **desc**,**name**;
10. **declare** **continue** HANDLER **for** not found **set** done = **true**;
11. **set** temp\_ranking=0;
12. **open** shooterlist\_cursor;
13. **fetch** shooterlist\_cursor **into** temp\_teamname,temp\_numbers;
14. while(not done) do
15. **set** temp\_ranking=temp\_ranking+1;
16. **update** shooterlist **set** ranking=temp\_ranking **where** teamname=temp\_teamname and numbers=temp\_numbers;
17. **fetch** shooterlist\_cursor **into** temp\_teamname,temp\_numbers;
18. **end** while;
19. **close** shooterlist\_cursor;
20. **end**$
21. delimiter ;

存储过程五

1. delimiter $
2. **drop** **procedure** if exists updateRanking\_shooter;
3. **create** **procedure** updateRanking\_shooter\_for\_adult()
4. **begin**
5. **declare** temp\_ranking **int**;
6. **declare** temp\_teamname **varchar**(20);
7. **declare** temp\_numbers **varchar**(20);
8. **declare** done **int** **default** **false**;
9. **declare** shooterlist\_cursor **cursor** **for** **select** teamname,numbers **from** shooterlist **where** teamname in(**select** **name** **from** team **where** college='无') **order** **by** goals **desc**,**name**;
10. **declare** **continue** HANDLER **for** not found **set** done = **true**;
11. **set** temp\_ranking=0;
12. **open** shooterlist\_cursor;
13. **fetch** shooterlist\_cursor **into** temp\_teamname,temp\_numbers;
14. while(not done) do
15. **set** temp\_ranking=temp\_ranking+1;
16. **update** shooterlist **set** ranking=temp\_ranking **where** teamname=temp\_teamname and numbers=temp\_numbers;
17. **fetch** shooterlist\_cursor **into** temp\_teamname,temp\_numbers;
18. **end** while;
19. **close** shooterlist\_cursor;
20. **end**$
21. delimiter ;

存储过程六

1. delimiter $
2. **drop** **procedure** if exists updateRanking\_shooter;
3. **create** **procedure** updateRanking\_shooter\_for\_male()
4. **begin**
5. **declare** temp\_ranking **int**;
6. **declare** temp\_teamname **varchar**(20);
7. **declare** temp\_numbers **varchar**(20);
8. **declare** done **int** **default** **false**;
9. **declare** shooterlist\_cursor **cursor** **for** **select** teamname,numbers **from** shooterlist **where** teamname not in(**select** **name** **from** team **where** college='无') and teamname not in(**select** **distinct** team **from** player **where** gender='女') **order** **by** goals **desc**,**name**;
10. **declare** **continue** HANDLER **for** not found **set** done = **true**;
11. **set** temp\_ranking=0;
12. **open** shooterlist\_cursor;
13. **fetch** shooterlist\_cursor **into** temp\_teamname,temp\_numbers;
14. while(not done) do
15. **set** temp\_ranking=temp\_ranking+1;
16. **update** shooterlist **set** ranking=temp\_ranking **where** teamname=temp\_teamname and numbers=temp\_numbers;
17. **fetch** shooterlist\_cursor **into** temp\_teamname,temp\_numbers;
18. **end** while;
19. **close** shooterlist\_cursor;
20. **end**$
21. delimiter ;