数据库应用课程设计

成绩：

学号：2013114120

姓名：杨丽俊

2015年1月20日

三峡大学

**课程设计任务书**

2014 学年秋季学期

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 课题名称 | | 数据库应用课程设计 | | | | | | 指导教师 | 覃兵文 | |
| 学生姓名 | | 杨丽俊 | 学号 | | | 2013114120 | 系别 | 信息管理系 | 专业 | 信息管理 |
| 课题概述：  本课题是数据库原理及应用课程的深化实践教学环节，主要应用已学知识，分析给定的课题，明确系统的需求，提出数据库的设计方案，并以SQL SERVER为DBMS进行实施。通过课程设计，来提高学生对数据库的分析设计及基本应用能力，掌握数据库的相关基本技能。  课题内容：  设计一个五人制足球管理系统，系统的主要要求包括：基本信息管理（含球队管理、球员管理、裁判员管理、球场管理等）、比赛管理（含赛事安排、比赛结果公告、比赛红黄牌处罚公告、球队积分榜等）、系统管理（用户管理、角色管理、用户权限控制等），设计该系统的数据库。 | | | | | | | | | | |
| 设计要求：   1. 认真、独立的完成设计，设计方案详细具体； 2. 根据报告模板，完成设计报告； 3. 基于SQL SERVER创建数据库，并编写调试所有代码。 | | | | | | | | | | |
| 主要参考资料：   1. 数据库系统概论.萨师煊 陈红主编.高等教育出版社，第四版 2. UML统一建模实用教程.王先国、王易.清华大学出版社 3. SQL Server数据库设计与系统开发教程.唐红亮.清华大学出版社 | | | | | | | | | | |
| 设计成果要求：   1. 设计报告书 2. 数据库实施代码 | | | | | | | | | | |
| 设计内容及进度计划安排 | 起止日期 | | | | 要求完成的内容 | | | | | |
| 18周星期一、二 | | | | 分析课题、完成需求分析 | | | | | |
| 18周星期三、四 | | | | 概念结构设计，完成类图和用例图 | | | | | |
| 18周星期五 | | | | 逻辑结构设计，确定数据库模式及外模式 | | | | | |
| 19周星期一 | | | | 物理结构设计，确定存储结构及存取方法 | | | | | |
| 19周星期二 | | | | 数据库实施 | | | | | |
| 19周星期三 | | | | 编写设计报告 | | | | | |
| 审核（教研室主任） | | | |  | | | | | | |

数据库应用课程设计

**摘要**：五人制足球是新兴起的一种足球运动，随着此运动的兴起，如何管理五人制足球的基本信息，成为五人制足球的一个大问题。在这种情况下，开发一个五人制足球管理系统显得尤为重要。这个五人制足球管理系统具有以下功能：基本信息管理、比赛管理和系统管理，以满足需要。

**关键词**：五人制足球;管理系统;数据库设计

# 课题概述

根据数据库设计报告书中的课题设计步骤。首先进行需求分析，了解用户对五人制数据库管理系统的要求，以及这个系统需要实现的功能，初步完成需求分析。根据需要的数据类型做出数据字典。在以上基础上利用powerdesigner工具画出用例图，形象直观的反映出需求分析，同时画出类图，反映更关系之间的联系。完成类图后，可以根据此产生关系模式，进行模式优化。为了满足用户需求，因此需要建立视图，更加符合用户习惯的别名。完成上述步骤后，可以设计存储结构和存取方法设计，提高时间和空间效率。安全性设计中主要是为了对用户权限和数据可见性进行分配，最终完成这个五人制足球管理系统设计。

# 需求分析

# 2.1功能分析

1. 具有的功能

* 五人制足球联赛报名管理：增加，修改，检索球队信息

1. 球队信息（球队名，领队名，教练名，联系电话）
2. 球员信息（球员名，号码，出生年月，性别，身份证号）

* 比赛日程管理：安排，调整比赛信息

1. 比赛日程安排（轮次，比赛时间，场序，主场队伍，客场队伍，球场名）
2. 裁判工作安排（比赛时间，比赛队，裁判名，计时员）

* 比赛结果管理，增加，公示，查询比赛结果

1. 比赛成绩公告
2. 射手榜（球队名，号码，球员名，总进球数）
3. 联赛积分榜（球队名，场次，球队得分，球队失分，球队净胜数，球队积分，球队排名）
4. 比赛红黄牌公告（场序，球队名，号码，球员名，球员红牌数，球员黄牌数，停赛期）

* 在比赛中被裁判员出示红牌或同一场被出示两次黄牌者，会停止该场比赛资格
* 净胜球=本队进球个数-对方进球个数，此处为球队得分，减去球队失分
* 球队积分：赢一场得三分，平一场得1分，输一场得零分。
* 球队排名：循环赛结束后，以积分高低排定名次，取前三名给予奖励。积分相同时，依次比较球队净胜数，成绩占优者排名靠前。
* 基本信息管理

1. 上述录入的球队球员信息
2. 裁判员，计时员管理
3. 裁判员（裁判名，裁判员序号）
4. 计时员（计时员名，计时员序号）
5. 球场管理
6. 球场（球场名，球场地点）
7. 用户角色权限分配

* 一般用户：查询功能

1. 比赛日程信息
2. 比赛成绩公告

* 球员和教练：查询功能

1. 个人详细资料（球员名，号码，出生年月，性别，身份证号）
2. 球队信息
3. 比赛日程信息
4. 比赛成绩公告

* 领队

1. 球队报名
2. 查询球员信息
3. 查询比赛日程和比赛成绩

* 裁判员和计时员

1. 查询和修改个人资料
2. 查询工作安排

* 主办方

1. 球队管理
2. 安排，查询，删除，修改比赛日程，裁判员信息
3. 登记，查询，公示比赛结果

射手榜、联赛积分榜、联赛成绩公告

1. 每轮比赛后公示红黄牌处罚公告，以及监督球员是否按照处罚规定

* 系统管理员

1. 管理用户基本信息（球队，球员，裁判员，比赛信息，主办方等）
2. 用户权限控制：为每一种用户分配所具有的权限
3. 用户登录管理

# 2.2数据字典

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 数据项名 | 别名 | 数据类型 | 长度 | 描述 |
| 球员编号 | Playerorder | Smallint |  | 球员唯一标识 |
| 球员名 | Playername | nvarchar | 20 |  |
| 号码 | Playernumber | Smallint |  |  |
| 出生年月 | Datebirth | Date |  |  |
| 性别 | Gender | Bit |  | 用0和1表示性别 |
| 身份证号 | IDnumber | char | 18 |  |
| 总进球数 | Playergoals | Smallint | 2 | 球员进球数 |
| 球队名 | Teamname | nvarchar | 20 | 每个球队的名称 |
| 领队名 | Leadername | Nvarchar | 10 |  |
| 教练名 | Teamcoach | Nvarchar | 10 |  |
| 联系电话 | Telephone | varchar | 12 |  |
| 轮次 | Turn | nvarchar | 10 | 第几轮比赛 |
| 场次 | FieldOrder | Smallint |  | 如：第一轮的第一场比赛 |
| 比赛时间 | fixture | Datetime |  |  |
| 比赛队伍 | VSteam | Varchar | 60 | 三大地龙vs河南建业 |
| 主场队伍 | Hometeam | nvarchar | 20 | 区分两支参赛队伍 |
| 客场队伍 | Visitingteam | nvarchar | 20 |
| 球员黄牌数 | Playeryellowcard | Smallint |  |  |
| 球员红牌数 | Playerredcard | Smallint |  |  |
| 停赛期 | Speriod | nvarchar | 10 |  |
| 球队得分 | Teamscore | Smallint |  | 纪录一个球队得分数 |
| 球队失分 | Teamlose | Smallint |  | 一个球队所失分数 |
| 球队净胜数 | GoalDifference | Smallint |  | 一个球队净胜得分 |
| 球队积分 | Score | Smallint |  | 计算一个球队的积分 |
| 球队排名 | Teamrangking | Smallint |  |  |
| 裁判名 | Judgername | nvarchar | 10 |  |
| 裁判员序号 | Judgernumber | SMALLINT |  | 裁判员区分的唯一标识 |
| 计时员名 | Timekeepername | Nvarchar | 10 |  |
| 计时员序号 | Timekeepernumber | SMALLINT |  |  |
| 球场名 | Courtname | Nvarchar | 10 |  |
| 球场地点 | Courtplace | nVarchar | 20 |  |
| 用户名 | Username | Nvarchar | 20 |  |
| 类别 | userType | Nvarchar | 10 |  |
| 登录密码 | UserPassword | Char | 10 |  |

# 3.概念结构设计

## 3.1用例设计





## 3.2类图设计



# 4.逻辑结构设计

## 4.1类图转换为关系模式

1. 球队（球队名，领队名，教练名，联系电话）

主键：球队名

1. 球员（球员编号，球员名，号码，出生年月，性别，身份证号，球队名）

主键：球员编号，外键：球队名，

1. 裁判员（裁判员序号，裁判名）

主键：裁判员序号

1. 计时员（计时员序号，计时员名）

主键：计时员序号

1. 球场（球场名，球场地点）

主键：球场名

1. 比赛信息（轮次，场序，比赛时间，主场队伍，客场队伍，球场名，裁判名，计时员名）

主键：轮次和场序，外键：主场队伍，客场队伍，

1. 射手榜（球员编号，总进球数）

主键：球员编号，外键：球员编号

1. 联赛积分榜（球队名，场次，球队得分，球队失分，球队净胜数，球队积分）

主键：球队名， 外键：球队名

1. 红黄牌公告（球员编号，轮次，球员黄牌数，球员红牌数，停赛期）

主键：球员编号，外键：球员编号

1. 用户登录（登录名，类别，登录密码）

主键：登录名和类别

## 4.2模式优化

球队

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 列名 | 数据类型 | 数据长度 | 允许NULL值 | 备注 |
| 球队名 | nvarchar | 20 | 否 | 主键 |
| 领队名 | Nvarchar | 10 |  |  |
| 教练名 | Nvarchar | 10 |  |  |
| 联系电话 | Varchar | 12 |  |  |

球员

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 列名 | 数据类型 | 数据长度 | 允许NULL值 | 备注 |
| 球员编号 | Smallint |  | 否 | 主键 |
| 球队名 | nvarchar | 20 | 否 | 外键 |
| 号码 | Smallint |  | 否 |  |
| 球员名 | nvarchar | 20 | 否 | 唯一值 |
| 出生年月 | date |  |  | 进而计算年龄 |
| 性别 | Bit |  |  | 用0或1表示男或女 |
| 身份证号 | char | 18 |  |  |

裁判员

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 列名 | 数据类型 | 数据长度 | 允许NULL值 | 备注 |
| 裁判员序号 | Smallint |  | 否 | 主键 |
| 裁判名 | nvarchar | 10 |  |  |

计时员

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 列名 | 数据类型 | 数据长度 | 允许NULL值 | 备注 |
| 计时员序号 | Smallint |  | 否 | 主键 |
| 计时员名 | Nvarchar | 10 |  |  |

球场

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 列名 | 数据类型 | 数据长度 | 允许NULL值 | 备注 |
| 球场名 | Nvarchar | 10 | 否 | 主键 |
| 球场地点 | nVarchar | 20 |  |  |

比赛信息

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 列名 | 数据类型 | 数据长度 | 允许NULL值 | 备注 |
| 轮次 | nvarchar | 10 | 否 | 主键 |
| 场序 | Smallint |  | 否 | 主键 |
| 比赛时间 | datetime |  | 否 |  |
| 主场队伍 | Nvarchar | 20 | 否 | 外键 |
| 客场队伍 | Nvarchar | 20 | 否 | 外键 |
| 球场名 | nVarchar | 10 | 否 |  |
| 裁判名 | nVarchar | 10 | 否 |  |
| 计时员名 | nVarchar | 10 | 否 |  |

射手榜

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 列名 | 数据类型 | 数据长度 | 允许NULL值 | 备注 |
| 球员编号 | Smallint |  | 否 | 主键 |
| 总进球数 | Smallint |  |  |  |

联赛积分榜

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 列名 | 数据类型 | 数据长度 | 允许NULL值 | 备注 |
| 球队名 | nvarchar | 20 | 否 | 主键，外键 |
| 场次 | Smallint |  | 否 | 自增长形式 |
| 球队得分 | Smallint |  |  |  |
| 球队失分 | Smallint |  |  |  |
| 球队净胜数 | SMALLINT |  |  |  |
| 球队积分 | Smallint |  |  |  |

红黄牌公告

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 列名 | 数据类型 | 数据长度 | 允许NULL值 | 备注 |
| 球员编号 | Smallint |  | 否 | 主键 |
| 轮次 | nvarchar | 10 | 否 |  |
| 球员黄牌数 | Smallint |  |  |  |
| 球员红牌数 | Smallint |  |  |  |
| 停赛期 | nvarchar | 10 |  |  |

用户

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 列名 | 数据类型 | 数据长度 | 允许NULL值 | 备注 |
| 用户名 | nvarchar | 20 | 否 | 主键 |
| 类别 | Nvarchar | 10 | 否 | 主键 |
| 登录密码 | char | 10 | 否 |  |

## 4.3外模式设计

建立各种视图：

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 视图名 | 需要的信息 | | | | | | | | 可见人员 |
| 球队基本信息 | 球员名 | 号码 | 年龄 | 性别 | 身份证号 | 领队名 | 教练名 | 联系电话 | 球队人员 |
| 全部球队基本信息 | 球员名 | 号码 | 年龄 | 性别 | 身份证号 | 领队名 | 教练名 | 联系电话 | 主办方 |
| 球队信息 | 球队名 | 领队名 | 教练名 | 球员名 | 号码 |  |  |  | 一般用户 |
| 工作安排 | 比赛时间 | 球场名 | 裁判名 | 裁判员序号 | 计时员名 | 计时员序号 | 球场地点 |  | 主办方，工作人员 |
| 比赛安排 | 轮次 | 场序 | 比赛时间 | 主场队伍 | 客场队伍 | 球场名 |  |  | 所有人员 |
| 射手榜公告 | 球队名 | 号码 | 球员名 | 总进球数 |  |  |  |  | 所有人员 |
| 联赛积分查询 | 球队名 | 场次 | 球队得分 | 球队失分 | 球队净胜数 | 球队积分 |  |  | 所有人员 |
| 处罚公告 | 球队名 | 号码 | 轮次 | 球员名 | 球员黄牌数 | 球员红牌数 | 停赛期 |  | 所有人员 |

# 5.物理结构设计

## 5.1参数分析

数据查询事务

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 事务说明 | 查询的关系  关系名 | 查询条件所涉及的属性 | 连接条件所涉及的属性 | 查询的投影属性 |
| 球队基本信息 | 球队，球员 | 球队名 | 球队名 | 球队名，号码，球员名，领队名，教练名， |
| 工作安排 | 裁判员，计时员，比赛信息 | 裁判员名，计时员名，球场名 |  | 比赛时间，球场名，裁判名，裁判员序号，计时员名，计时员序号 |
| 比赛相关信息 | 比赛信息 |  |  | 轮次，场序，比赛时间，主场队伍，客场队伍，球场名 |
| 射手信息 | 射手榜，球员 | 球队名，号码 | 球队名，号码 | 球队名，球员名，总进球数 |
| 比赛结果信息 | 联赛积分榜 |  |  | 球队名，场次，球队得分，球队失分，球队净胜数，球队积分，球队排名 |
| 处罚公告 | 红黄牌公告，球员 | 球队名，号码 |  | 球队名，号码，轮次，球员名，球员红牌数，球员黄牌数，停赛期 |

数据更新事务

|  |  |  |
| --- | --- | --- |
| 被更新的关系 | 每个关系上的更新操作条件所涉及的属性 | 修改操作要改变的属性 |
| 比赛信息 | 轮次，场序 | 比赛时间，主场队伍，客场队伍，球场名，裁判名，计时员名 |
| 射手榜 | 球队名，号码 | 总进球数 |
| 联赛积分榜 | 球队名 | 场次，球队得分，球队失分，球队净胜数，球队积分，球队排名 |
| 红黄牌公告 | 轮次 | 球队名，号码，球员名，球员黄牌数，球员红牌数，停赛期 |
| 用户登录 | 登录名，类别 | 登录密码 |

## 5.2存取方法设计

DBMS常用的存储方法有索引方法，聚簇方法和HASH方法。

在表“球队”中，球队名经常出现在查询条件中，在作为主键的同时，已经建立了唯一性索引，在表“球员”中，属性球队名和号码在连接条件上经常出现，将两者设为主键的同时已经对其建立了索引，同时可以选择建立聚簇存取，将同一支球队的球员集中存放，减少访问磁盘的次数；对“比赛信息”中的“轮次”建立聚簇存取，将每一轮的比赛队伍集中存放；对“联赛积分榜”中的球队名建立唯一性索引。对“红黄牌公告”中的“轮次”，选择建立聚簇索引，将每一轮结束后的的处罚球员信息集中存放，方便进行处罚公告。

## 5.3存储结构设计

包括确定1：数据的存放位置和存储结构；2：确定系统配置。

确定数据的存放位置和存储结构：五人制足球管理系统由数据文件和日志文件构成，数据文件的大小是5mb，增长速度是15%，而日志文件大小是2mb，增长速度是1mb，两者均存放在D盘上；而数据库数据备份和日志文件备份在五人制足球系统出现故障时才会使用，可以放在E盘上；对于建立的表选择建在D盘上，将上述建立的索引放在E盘上，提高物理I/O读写效率。

确定系统配置：对于数据库大小在确定数据的存放位置和存储结构时已经提到了，这个五人制足球管理系统是在win7系统，sql server 2005条件下运行的。

# 6.安全性设计

数据库安全性控制的常用方法有：用户标识和鉴定，存取控制，视图，审计，密码存储

用户标识：此数据库系统有以下用户：一般用户，球员，教练，领队，裁判员，计时员，主办方，系统管理员；每次用户进入系统时，都需要系统对用户提供的身份标识进行核对，通过鉴定后，才具有使用权

存取控制：先对用户定义存取权限，然后对用户进行的各种操作进行控制，确保执行操作合法，可以授权用户的权利也可以收回，利用GRANT/REVOKE语句实现。有：自主存取控制和强制存取控制。

|  |  |  |  |
| --- | --- | --- | --- |
| 用户名 | 数据对象名 | 允许的操作类型 | 能否转授权 |
| 球队人员（球员，教练，领队） | 关系：球队 | Select | 不能 |
| 关系：球员 | Select | 不能 |
| 关系：比赛信息 | Select | 能 |
| 关系：射手榜 | Select | 能 |
| 关系：联赛积分榜 | Select | 能 |
| 关系：红黄牌公告 | Select | 能 |
| 一般用户 | 关系：球队 | Select | 能 |
| 关系：球员 | Select | 能 |
| 关系：比赛信息 | Select | 能 |
| 关系：射手榜 | Select | 能 |
| 关系：联赛积分榜 | Select | 能 |
| 关系：红黄牌公告 | Select | 能 |
| 裁判员 | 关系：裁判员 | Select | 能 |
| 计时员 | 关系：计时员 | Select | 能 |
| 主办方 | 关系：球队 | All | 不能 |
| 关系：球员 | All | 不能 |
| 关系：裁判员 | All | 不能 |
| 关系：计时员 | All | 不能 |
| 关系：球场 | All | 不能 |
| 关系：比赛信息 | All | 不能 |
| 关系：射手榜 | All | 不能 |
| 关系：联赛积分榜 | All | 不能 |
| 关系：红黄牌公告 | All | 不能 |

视图：为用户建立视图，与授权机制配合，使用视图屏蔽掉一部分保密数据，提供数据独立性，其中在外模式设计中已经建立了视图。

审计：可以将用户对数据库的所有操作记录在上面，利用AUDIT语句来设置审计功能，而NOAUDIT则是取消审计。

# 7.总结

此次课程设计给我的体会有以下几点：

看似简单，其实不然 。以前有过课程设计的经验，也只是数据结构课程设计，写出八皇后问题的代码。和这次的数据库课程设计相比，这次更加复杂。不仅需要代码实现，更重要的是能够将自己思考的东西进行完善和全面。

冷静思考，富有耐心 。需求分析作为课程设计的开头，想要做好这个课程设计，就必须让自己静下心来，全面思考，多查资料。百度是一个很有用的工具，在这个上面可以找到自己想要的许多答案。五人制足球管理系统虽然很复杂，是一个大难题，但是只要将一个个的问题解决了以后，这个大问题也就迎刃而解了。

善于交流、回归书本 。有时出现问题了，觉得思路一片混乱，无意间出现的问题甚至会不知道怎么解决。这是可以选择和同学交流沟通，也许别人不经意间的一句话，便是这个问题的解决突破口。另外，这次的设计步骤也是跟据书本上的数据库设计的顺序来安排的，所以在做课程设计时需要回归书本，找到有解决方法。

总之，虽然说这次数据库课程设计自己设计的东西没有那么全面，但在这个过程中学会了依靠自己，而对于数据库，自己学习的知识对于这次的课程设计完全不够用，所以在必须平时也要多学习有关数据库的知识。

**附录代码：**

create database 五人制足球

on primary

(

name='五人制足球\_data',

filename='D:\五人制足球\_data.mdf',

size=5mb,

maxsize=100mb,

filegrowth=15%

)

log on

(

name='五人制足球\_log',

filename='D:\五人制足球\_log.ldf',

size=2mb,

filegrowth=1mb

)

use 五人制足球

create table 球队

(

球队名 nvarchar(20) primary key,

领队名 Nvarchar(10),

教练名 Nvarchar(10),

联系电话 Varchar(12),

)

create table 球员

(

球员编号 smallint,

球队名 nvarchar(20),

号码 SMALLINT,

球员名 nvarchar(20) not null,

出生年月 date ,

性别 bit,

身份证号 char(18),

primary key(球员编号),

foreign key (球队名) references 球队(球队名)

)

alter table 球员 add constraint u\_球员 unique(球员名)

alter table 球员 add constraint u\_球员号码 unique(号码)

create table 裁判员

(

裁判员序号 smallint primary key,

裁判名 nvarchar(10),

)

create table 计时员

(

计时员序号 smallint primary key,

计时员名 nvarchar(10),

)

create table 球场

(

球场名 nvarchar(10)primary key,

球场地点 nvarchar(10),

)

create table 比赛信息

(

轮次 nvarchar(10),

场序 smallint,

比赛时间 datetime,

主场队伍 nvarchar(20),

客场队伍 nvarchar(20),

球场名 nvarchar(10),

裁判名 nvarchar(10),

计时员名 nvarchar(10),

primary key(轮次,场序),

foreign key (主场队伍)references 球队(球队名),

foreign key (客场队伍)references 球队(球队名),

)

create table 射手榜

(

球员编号 smallint,

总进球数 smallint,

primary key(球员编号),

foreign key (球员编号) references 球员(球员编号),

--foreign key(球队名)references 球队(球队名)

)

create table 联赛积分榜

(

球队名 nvarchar(20)primary key,

场次 smallint,

球队得分 smallint,

球队失分 smallint,

球队净胜数 smallint,

球队积分 smallint,

球队排名 smallint,

foreign key (球队名) references 球队(球队名),

)

create table 红黄牌公告

(

球员编号 smallint,

轮次 nvarchar(10),

球员黄牌数 smallint,

球员红牌数 smallint,

停赛期 nvarchar(10),

primary key(球员编号),

foreign key(球员编号)references 球员(球员编号),

)

create table 用户

(

用户名 nvarchar(20),

类别 nvarchar(10),

登录密码 char(10),

primary key (用户名,类别),

)

--球队基本信息，给球队人员看

create view 球队基本信息

as

select 球队.球队名 ,教练名,领队名,联系电话,球员名,号码,性别,出生年月,身份证号

from 球员,球队

where 球员.球队名=球队 .球队名

--球队信息,给一般用户看

create view 球队信息

as

select 球队.球队名 ,教练名,领队名,球员名,号码

from 球员,球队

where 球员.球队名=球队 .球队名

--工作安排,给工作人员和主办方看

create view 工作安排

as

select 比赛时间,比赛信息 .球场名,球场地点 ,比赛信息 .裁判名 ,裁判员序号 ,比赛信息.计时员名,计时员序号

from 比赛信息 ,裁判员 ,计时员 ,球场

where 比赛信息 .裁判名=裁判员 .裁判名 and 比赛信息 .计时员名 =计时员 .计时员名 and 球场 .球场名 =比赛信息 .球场名

--比赛安排,给所有人看

create view 比赛安排

as

select 轮次,场序,比赛时间,主场队伍,客场队伍,球场名

from 比赛信息

--射手榜公告,所有人可见

create view 射手榜公告

as

select 球队名 ,号码,球员名,总进球数

from 射手榜 ,球员

where 射手榜 .球员编号 =球员.球员编号

--联赛积分公告,所有人可见

create view 联赛积分公告

as

select top 100 percent 球队名,场次,球队得分,球队失分,球队净胜数,球队积分

from 联赛积分榜

order by 球队积分 ,球队净胜数,球队名,场次,球队得分,球队失分

--处罚公告,所有人可见

create view 处罚公告

as

select 球队名 ,号码 ,轮次,球员名,球员黄牌数,球员红牌数,停赛期

from 红黄牌公告,球员

where 红黄牌公告 .球员编号 =球员 .球员编号

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/ 建立索引

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

create unique index qiuyuanming on 球员(球员名)

create unique index lunci on 红黄牌公告(轮次)