《数据库系统》课程设计

体育项目比赛管理系统设计与开发

班级：

学号：

姓名：

目录

[一、系统需求分析 1](#_Toc25406)

[（一）需求概述 1](#_Toc13625)

[（二）业务流分析 1](#_Toc10146)

[（三）数据流分析 3](#_Toc27915)

[（四）数据字典 4](#_Toc1493)

[二、数据库概念结构设计 5](#_Toc914)

[（一）实体分析 5](#_Toc26748)

[（二）属性分析 5](#_Toc15374)

[（三）联系分析 7](#_Toc9299)

[（四）概念模型分析（.PDM图） 8](#_Toc17605)

[三、数据库逻辑结构设计 9](#_Toc4385)

[（一）概念模型转化为逻辑模型 9](#_Toc13992)

[1.一对一关系的转化 9](#_Toc11055)

[2.一对多关系的转化 9](#_Toc20346)

[3.多对多关系的转化 10](#_Toc2406)

[（二）逻辑模型设计（.PDM图） 10](#_Toc8678)

[四、数据库物理实现 10](#_Toc5517)

[（一）表设计 10](#_Toc4948)

[（二）创建表和完整性约束代码设计 11](#_Toc7992)

[（三）创建视图、索引、存储过程和触发器 12](#_Toc32145)

[五、数据库功能调试 14](#_Toc26469)

[（一）运动员管理模块 14](#_Toc14975)

[（二）比赛项目管理模块 15](#_Toc29152)

[（三）系统管理员管理模块 16](#_Toc21088)

[六、设计系统前台软件 21](#_Toc4834)

[（一）开发软件选择 21](#_Toc2440)

[（二）软件功能要求与设计 21](#_Toc8837)

[（三）软件功能实现 22](#_Toc12397)

[（四）系统测试 22](#_Toc19720)

[七、设计总结 24](#_Toc29042)

# 一、系统需求分析

（一）需求概述

本次课程设计的题目是体育项目比赛管理系统，系统内容要求是：数据库有锦标赛各个代表团、运动员、比赛项目、比赛情况等实体。每一位运动员均来自一个代表团,一个比赛项目属于一个比赛类别，每一位运动员会参加一项或多项比赛项目。

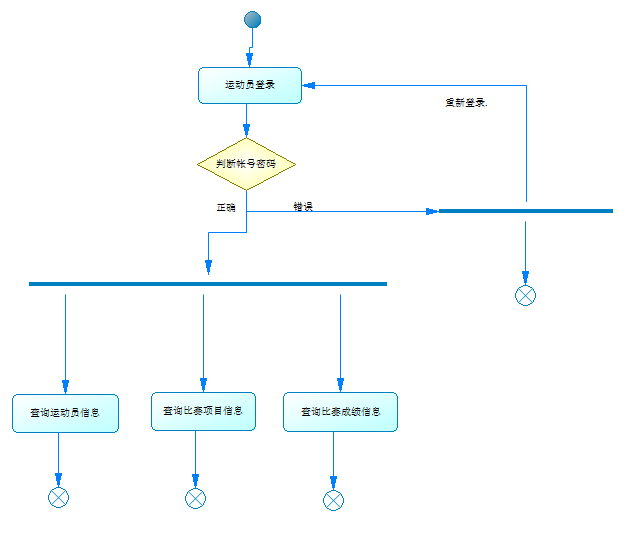
系统的使用者分为管理人员和普通用户。他们对该数据库系统有不同的操作：

管理人员：可以对运动员的个人信息进行增加、删除、更新、查询操作，也可以对代表团的信息进行增加、删除、更新、查询操作，可以对比赛类别的信息进行增加、删除、更新、查询操作，可以对比赛项目的信息进行增加、删除、更新、查询操作。总之，管理人员可以对数据库系统进行增加、删除、更新、查询。

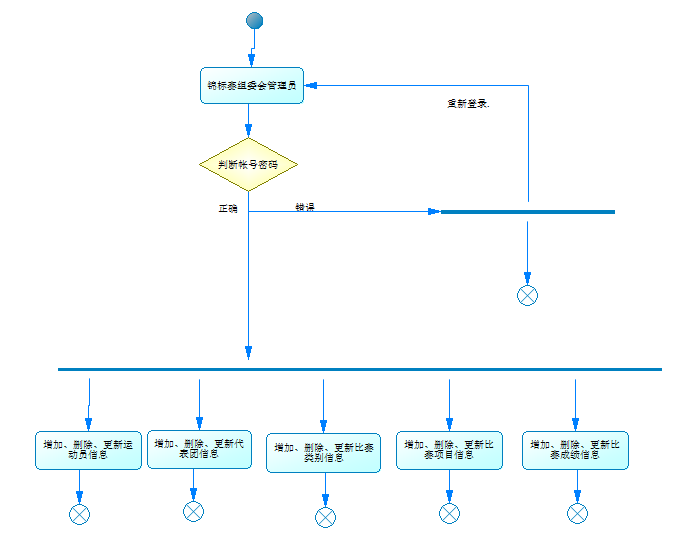
普通用户：普通用户可以根据运动员编号来找到该运动员所对应的姓名、年龄、性别和所属代表团；可以通过查询项目编号得到该比赛项目所对应的项目名、级别和所属比赛类别的类别编号、类别名和主管信息。还可以根据运动员编号查询到所参加的比赛项目的项目名、比赛时间和得分。总之，普通用户可以对数据库系统进行查询操作。

（二）业务流分析

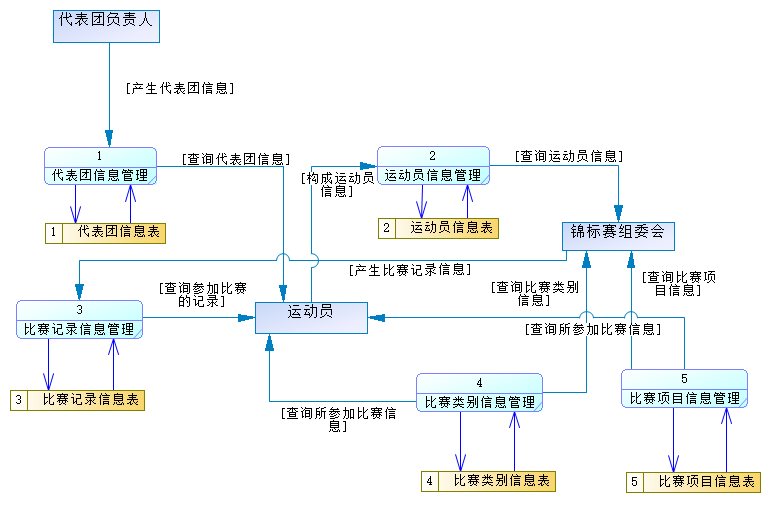
从运动员角度：



从锦标赛组委会管理员角度：



（三）数据流分析



（四）数据字典

数据项：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 数据项名 | 含义说明 | 数据类型 | 长度 | 取值范围 | 数据项联系 |
| 团编号 | 区别代表团 | char | 4 | 0001-9999 |  |
| 地区 | 代表团地区 | char | 20 |  |  |
| 住所 | 代表团住所 | Char | 20 |  |  |
| 类别编号 | 标注比赛类别 | Char | 4 | 0001-9999 | 一个类别编号对应一个类别名 |
| 类别名 | 标注比赛类别 | char | 20 |  | 一个类别名对应一个类别编号 |
| 主管 | 标注比赛类别 | char | 20 |  |  |
| 编号 | 标注运动员 | char | 4 |  | 一个编号对应一个姓名 |
| 姓名 | 标注运动员 | Char | 20 |  | 一个姓名对应一个编号 |
| 年龄 | 标注运动员 | Int | 2 |  |  |
| 性别 | 区分男女 | char | 4 | “男”或“女” |  |
| 项目编号 | 标注比赛项目 | char | 4 |  | 一个项目编号对应一个项目名 |
| 项目名 | 标注比赛项目 | char | 20 |  | 一个项目名对应一个项目编号 |
| 级别 | 标注比赛项目 | char | 2 | “男”或“女” |  |
| 比赛时间 |  | Int | 4 |  |  |
| 得分 |  | Int | 4 | 0001-9999 |  |

数据结构：

|  |  |  |
| --- | --- | --- |
| 数据结构名 | 含义名 | 组成 |
| 代表团 | 代表团信息 | 团编号、地区、住所 |
| 比赛类别 | 比赛类别信息 | 类别编号、类别名、主管 |
| 运动员 | 运动员信息 | 编号、团编号、姓名、年龄、性别 |
| 比赛项目 | 比赛项目信息 | 项目编号、项目名、级别 |
| 比赛记录 | 运动参加比赛及比赛结果信息 | 编号、项目编号、比赛时间、得分 |

数据流：

|  |  |  |  |
| --- | --- | --- | --- |
| 数据流名 | 数据流来源 | 数据流去向 | 组成 |
| 代表团信息管理 | 代表团 | 代表团信息表 | 代表团信息 |
| 比赛类别信息管理 | 锦标赛组委会 | 比赛类别信息表 | 比赛类别信息 |
| 运动员信息管理 | 运动员 | 运动员信息表 | 运动员信息 |
| 比赛项目信息管理 | 锦标赛组委会 | 比赛项目信息表 | 比赛项目信息 |
| 比赛记录信息管理 | 锦标赛组委会 | 比赛记录信息表 | 比赛记录信息 |

数据存储：

|  |  |  |  |
| --- | --- | --- | --- |
| 数据存储名 | 输入的数据流 | 输出的数据流 | 组成 |
| 代表团信息表 | 代表团信息 | 代表团信息  运动员信息 | 代表团信息  运动员信息 |
| 比赛类别信息表 | 比赛类别信息 | 比赛类别信息  比赛项目信息 | 比赛类别信息  比赛项目信息 |
| 运动员信息表 | 运动员信息 | 运动员信息  代表团信息 | 运动员信息  代表团信息 |
| 比赛项目信息表 | 比赛项目信息 | 比赛类别信息  比赛项目信息 | 比赛类别信息  比赛项目信息 |
| 比赛记录信息表 | 运动员信息  比赛项目信息 | 运动员信息  比赛项目信息  比赛记录信息 | 运动员信息  比赛项目信息  比赛记录信息 |

# 二、数据库概念结构设计

（一）实体分析

经需求分析，本次课程设计中包含四个实体，他们分别是：代表团实体、比赛类别实体、运动员实体、比赛项目实体。

（二）属性分析

代表团实体属性：团编号、地区、住所。如下：

代表团

地区

团编号

住所

比赛类别实体属性：类别编号、类别名、主管。如下：

比赛类别

类别名

类别编号

主管

运动员实体属性：编号、姓名、年龄、性别。如下：

运动员

姓名

编号

年龄

性别

比赛项目实体属性：项目编号、项目名、级别。如下：

比赛项目

项目名

项目编号

级别

（三）联系分析

一个运动员只属于一个代表团，但一个代表团有多个运动员，运动员与代表团之间的联系是多对一的联系；一个运动员能参加多个比赛项目，一个比赛项目需要多个运动员参与，运动员与比赛项目的关系是多对多的关系；一个比赛项目只属于一个比赛类别，但一个比赛类别有多个比赛项目，比赛类别与比赛项目之间的联系是一对多的联系

E-R图分析如下：

参加

代表团

住所

团编号

地区

成员

运动员

编号

年龄

姓名

性别

1

N

比赛类别

主管

类别编号

类别名

属于

比赛项目

项目编号

级别

项目名

1

N

比赛时间

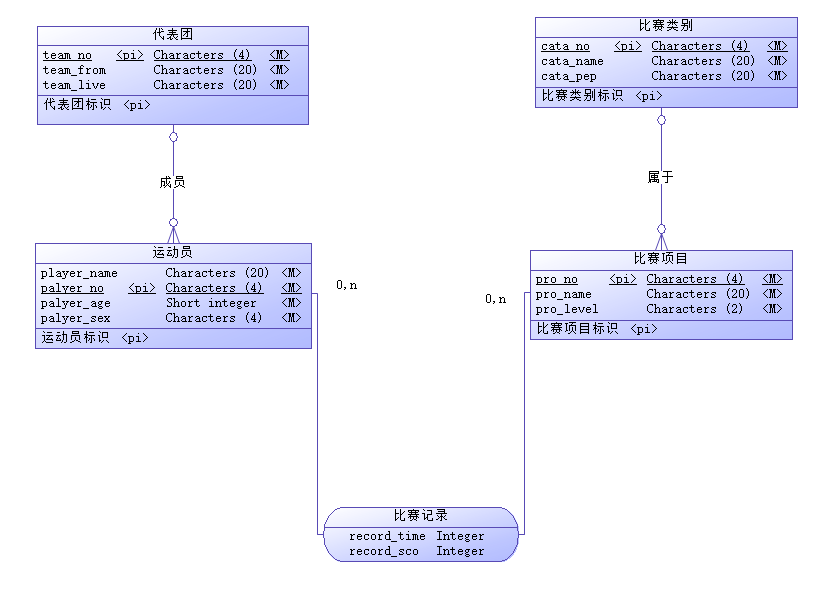
得分

N

M

（四）概念模型分析（.PDM图）

在经过需求分析和实体属性的分析，以及各实体之间的关系，最终得到概念模型如下：



# 三、数据库逻辑结构设计

（一）概念模型转化为逻辑模型

1.一对一关系的转化

在体育项目比赛管理管理系统中没有一对一关系的转化

2.一对多关系的转化

一个运动员只属于一个代表团，但一个代表团有多个运动员。

一个比赛项目只属于一个比赛类别，但一个比赛类别有多个比赛项目，比赛类别与比赛项目之间的联系是一对多的联系。

运动员（运动员编号（主键）、姓名、年龄、性别、团编号（外键）、地区、住所）

代表团（团编号（主键）、地区、住所）

比赛类别（类别编号（主键）、类别名、主管）

比赛项目（项目编号（主键）、项目名、级别、类别编号（主键）、类别名、主管）

3.多对多关系的转化

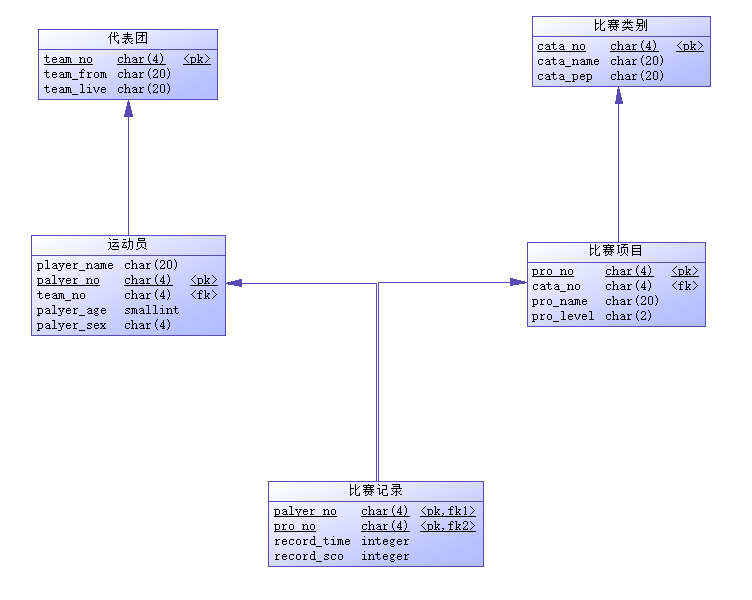
一个运动员能参加多个比赛项目，一个比赛项目需要多个运动员参与，运动员与比赛项目的关系是多对多的关系。

运动员（运动员编号（主键）、团编号（外键）、姓名、年龄、性别）

比赛项目（项目编号（主键）、项目名、级别）

比赛记录（运动员编号（主键/外键）、项目编号（主键/外键）、比赛时间、得分）

（二）逻辑模型设计（.PDM图）



# 四、数据库物理实现

（一）表设计

**代表团信息表设计**

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名称** | **字段解释** | **数据类型** | **备注** |
| **tno** | **团编号** | **char(4)** | **主键** |
| **tfrom** | **地区** | **char(20)** | **非空** |
| **tlive** | **住所** | **char(20)** | **非空** |

**运动员信息表设计：**

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名称** | **字段解释** | **数据类型** | **备注** |
| **pno** | **运动员编号** | **char(4)** | **主键** |
| **pname** | **姓名** | **char(20)** | **非空** |
| **tno** | **团编号** | **Char(4)** | **外键** |
| **page** | **年龄** | **Smallint** | **非空** |
| **psex** | **性别** | **char(4)** | **从“男”“女”中选择** |

**比赛类别信息表设计：**

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名称** | **字段解释** | **数据类型** | **备注** |
| **cno** | **类别编号** | **char(4)** | **主键** |
| **cname** | **类别名** | **char(20)** | **非空** |
| **cpep** | **主管** | **char(20)** | **非空** |

**比赛项目信息表设计：**

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名称** | **字段解释** | **数据类型** | **备注** |
| **eno** | **项目编号** | **char(20)** | **主键** |
| **cno** | **类别编号** | **char(20)** | **外键，非空** |
| **ename** | **项目名** | **char(20)** | **非空** |
| **elevel** | **级别** | **char(2)** | **非空** |

**比赛记录信息表设计：**

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名称** | **字段解释** | **数据类型** | **备注** |
| **eno** | **项目编号** | **char(4)** | **外键** |
| **pno** | **运动员编号** | **char(4)** | **外键** |
| **rtime** | **比赛时间** | **int** |  |
| **rsco** | **得分** | **int** |  |

（二）创建表和完整性约束代码设计

1. 创建代表团表及完整性约束：

CREATE TABLE TEAM(

tno char(4) primary key,

tfrom char(20) NOT NULL,

tlive char(20) Not NULL)

1. 创建运动员表及完整性约束：

CREATE TABLE PLAYER(

pno char(4) primary key,

pname char(20) NOT NULL,

tno char(4) Not NULL,

page smallint Not NULL,

psex char(4) check (psex in ('男','女'))

foreign key (tno) references TEAM(tno))

1. 创建比赛类别表及完整性约束：

CREATE TABLE CATA(

cno char(4) primary key,

cname char(20) NOT NULL,

cpep char(20) NOT NULL)

1. 创建比赛项目表及完整性约束：

CREATE TABLE EVE(

eno char(20) primary key,

cno char(4) NOT NULL,

ename char(20) NOT NULL,

elevel char(2) NOT NULL,

foreign key (cno) references CATA(cno))

1. 创建比赛记录表及完整性约束:

CREATE TABLE RECORD(

eno char(20),

pno char(4) ,

rtime int,

rsco int,

foreign key (eno) references EVE(eno),

foreign key (pno) references PLAYER(pno))

（三）创建视图、索引、存储过程和触发器

1. 创建视图

（1）创建运动员编号为1的参赛记录信息的视图。

go

create view V1

as

Select pname,eno,rtime,rsco

from PLAYER,RECORD

WHERE PLAYER.pno = 1

（2）创建比赛项目编号为1的比赛信息的视图。

go

create view V2

as

select ename,elevel,CATA.cno,cname,cpep

from EVE,CATA

WHERE eno = 1

（3）创建代表团编号为1的运动员信息的视图。

go

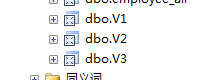
create view V3

as

select pno,pname,page,psex

from TEAM,PLAYER

WHERE TEAM.tno = 1



1. 创建索引

为改数据库中的RECORD表建立索引。其中比赛时间按升序建立唯一索引，得分按降序建立唯一索引。

CREATE UNIQUE INDEX RECORDTIME on RECORD(rtime ASC,rsco DESC);

1. 创建存储过程

建立存储过程，输入运动员编号得到该运动员的所有信息。

create procedure player\_info

@pno char (4)

as

select pname,team.tno,EVE.eno ,EVE.ename,rsco

from PLAYER,EVE ,TEAM,RECORD

where PLAYER.pno = @pno and team.tno=PLAYER .pno AND RECORD.pno=@pno AND EVE.eno=RECORD.eno

建立存储过程，输入项目编号得到该项目的相关信息。

create procedure eve\_info

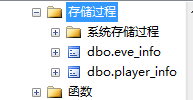
@eno char (20)

as

select ename ,elevel,CATA.cno,CATA.cname ,cpep

from EVE ,CATA

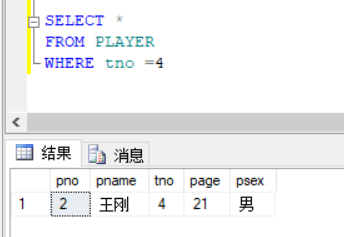
where EVE.cno=CATA.cno AND EVE.eno=@eno



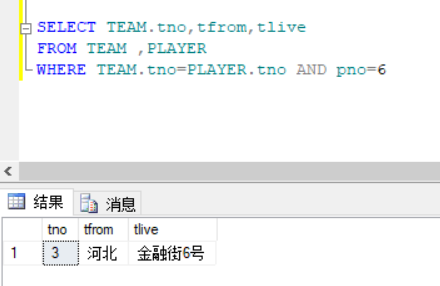
# 五、数据库功能调试

（一）运动员管理模块

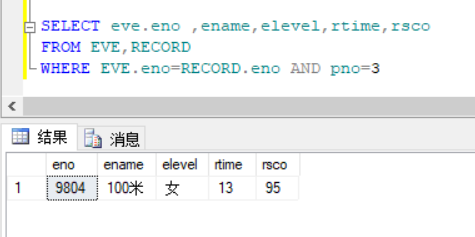
1. 查询运动员信息



1. 查询运动员所在代表团信息

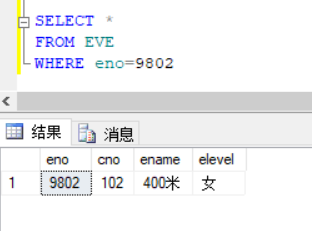


1. 查询运动员参加比赛信息

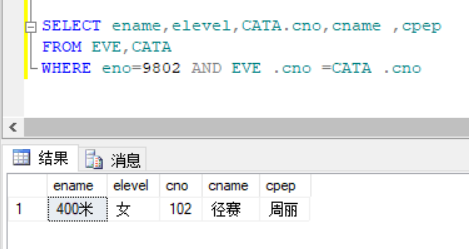


（二）比赛项目管理模块

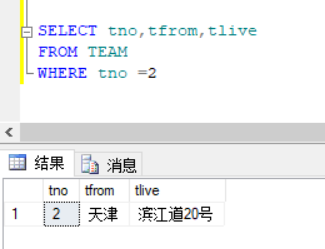
1. 查询比赛项目信息



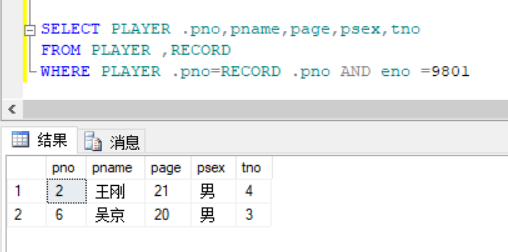
1. 查询比赛项目所属比赛类别信息



1. 查询代表团信息



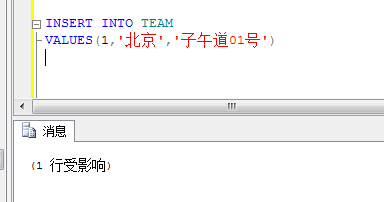
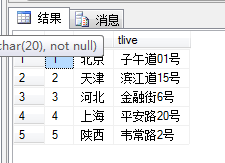
1. 查询参与比赛的运动员信息



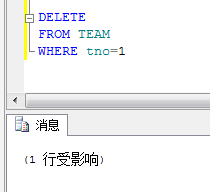
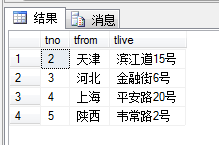
（三）系统管理员管理模块

1. 对代表团的信息管理

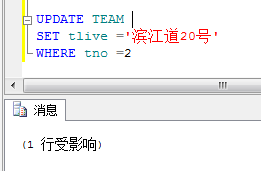
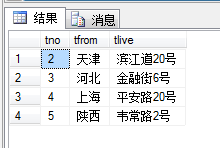
1.1插入代表团信息

1.2删除代表团信息

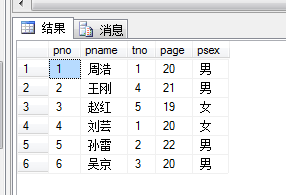
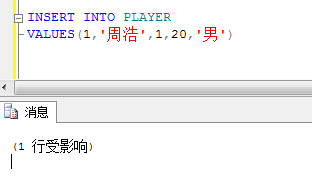
 

1.3修改代表团信息

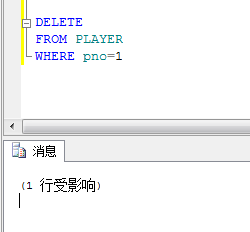
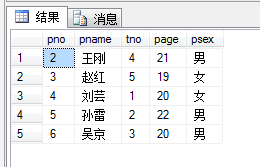
 

1. 对运动员的信息管理

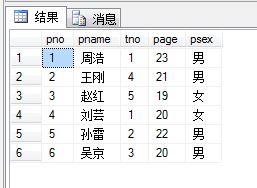
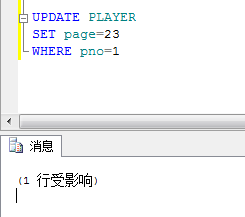
2.1插入运动员信息



2.2删除运动员信息

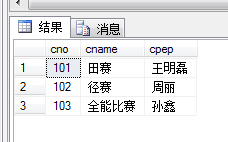
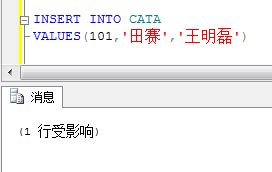
 

2.3修改运动员信息

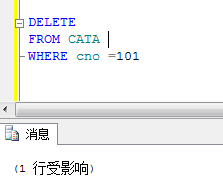
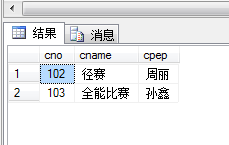


1. 对比赛类别的信息管理

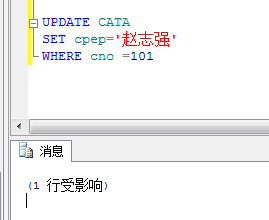
3.1插入比赛类别信息



3.2删除比赛类别信息

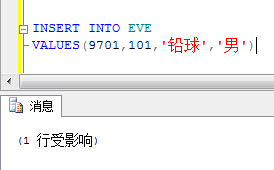
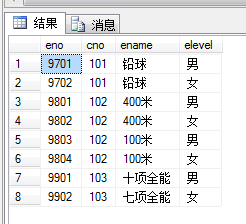
 

3.3修改比赛类别信息

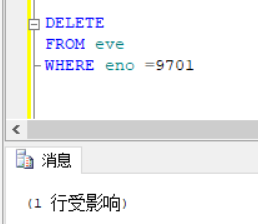
 

1. 对比赛项目的信息管理

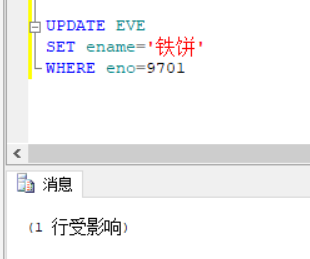
4.1插入比赛项目信息

4.2删除比赛项目信息

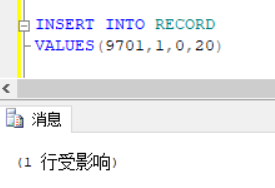
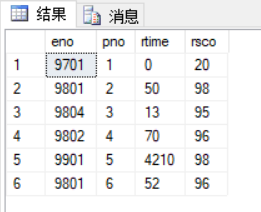


4.3修改比赛项目信息

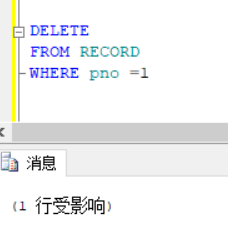
 

1. 对参加比赛的记录信息管理

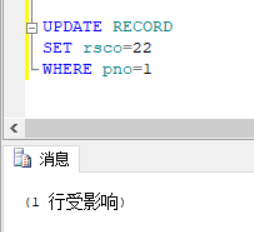
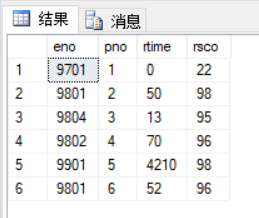
5.1插入参加比赛的记录信息

5.2删除参加比赛的记录信息

5.3修改参加比赛的记录信息

# 六、设计系统前台软件

（一）开发软件选择

Dreamweaver 8

Sql server 2008

（二）软件功能要求与设计

Dreamweaver通过数据源与数据库连接，通过添加记录集的方式导入数据库的各种表，以此来实现设计体育比赛项目管理系统的页面，共4种页面。

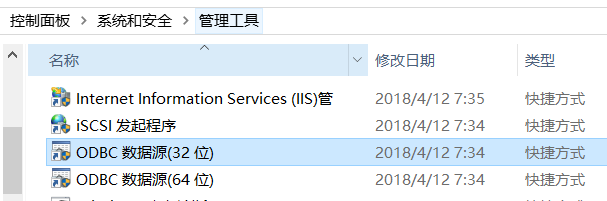
首页页面、插入页面、更新页面、登录页面

（三）软件功能实现

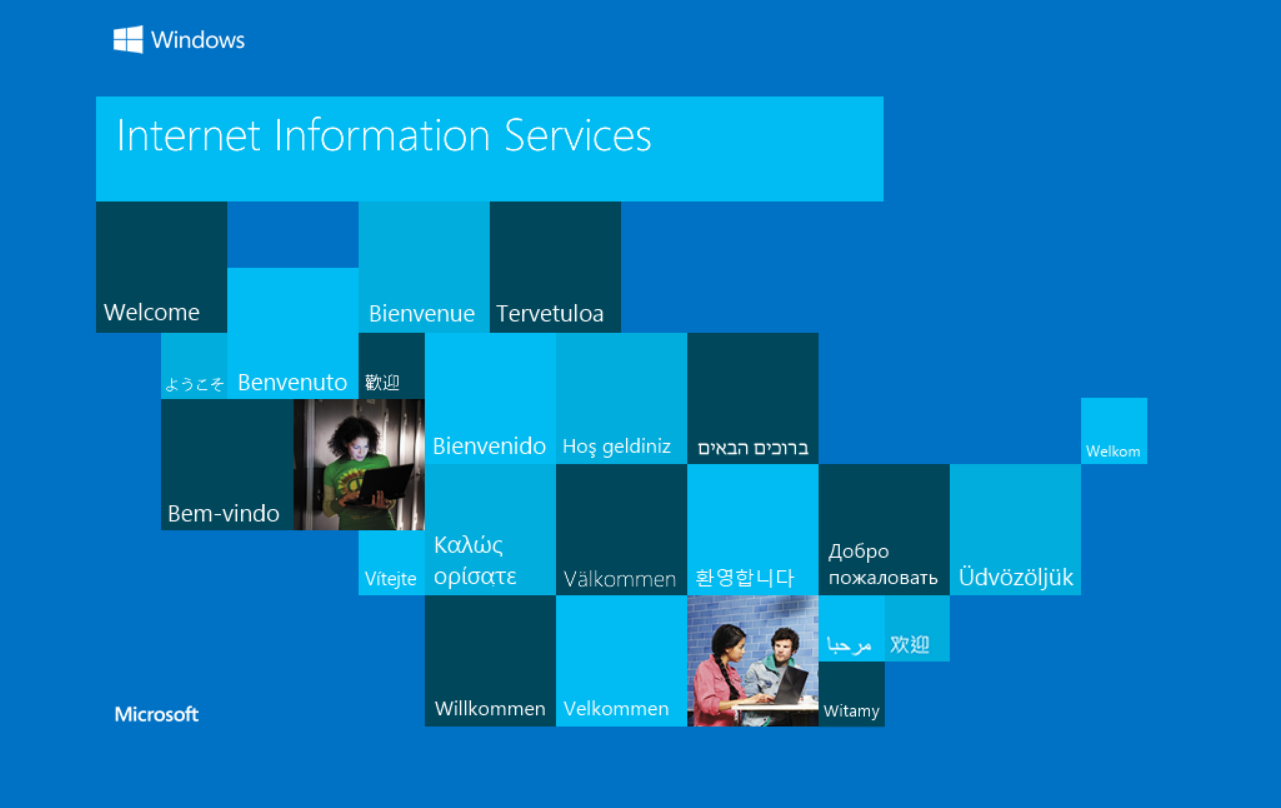
Dreamweaver与数据库连接，建立数据源，添加记录集，插入表格，插入记录集，设置数据显示条数等，实现首页设计建立，更新，查询，插入等页面。完成后通过预览，检查功能操作是否正常，再进一步进行优化提升，使系统更完善。

（四）系统测试

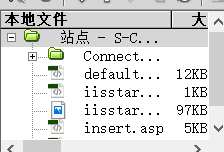
1.创建ODBC数据源



1. IIS的安装及本地服务器的发布



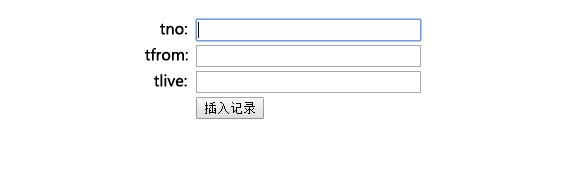
1. Dreamweaver中站点的创建

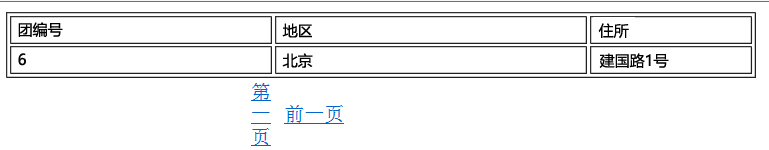
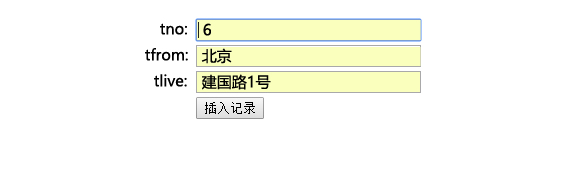


4.ASP首页的制作及分页的实现

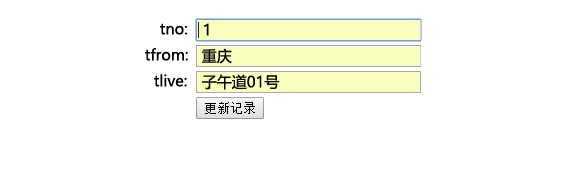


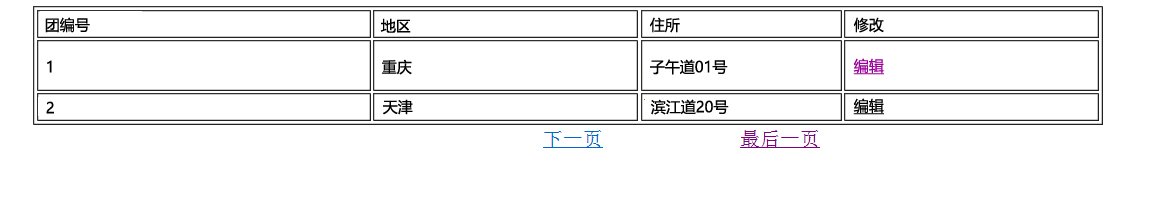
5.插入页面的实现



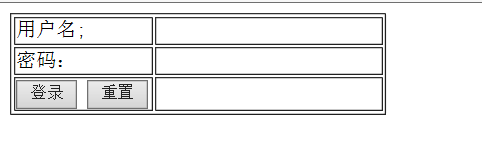


6.更新页面的实现，点击编辑即可修改信息





7.登录页面的实现



# 七、设计总结

本次课程设计我选择的体育比赛项目管理系统，开始实验之前读了大量的相关论文和资料，体育比赛项目管理系统越来越普遍，生活中很多大大小小的体育活动都会需要到，一开始，对此系统的设计很不清晰，很多功能不知道怎么划分在一起，尤其是属性的设计，外键之间的联系等，再参考了众多资料之后，才有了头绪。在这次课程设计中，我们懂得了在实现一个系统的时候，首先要做好必须作好需求分析，建立好数据库，如果需求分析不成功，到了后来只有推翻重新设计。

我确定了系统的主要模块。然后先分析需求，搞清楚系统的主要业务是哪些，需要哪些表，哪些属性，相对应画出cdm和pdm。到了创建数据库的时候，我发现有很多小问题，而这些小问题都是因为前期的业务分析没有到位，表的属性没有考虑完全，所以也进行了多次的修改。通过本次课程设计，使我对本学期所学的数据库有了一个更深层次的了解，系统的将所学的知识联系到了一起，同时我也认识到自己对一些知识学习的疏忽。虽然我建立的系统功能比较少，不是很完善，但是在做这个系统的过程中，还是经历了很多波折，也从中认识到了自己很多不足，在以后的学习中我会将本门课程的知识与生活中的实践联系到一起，不断提高自己的文化知识和实践能力。