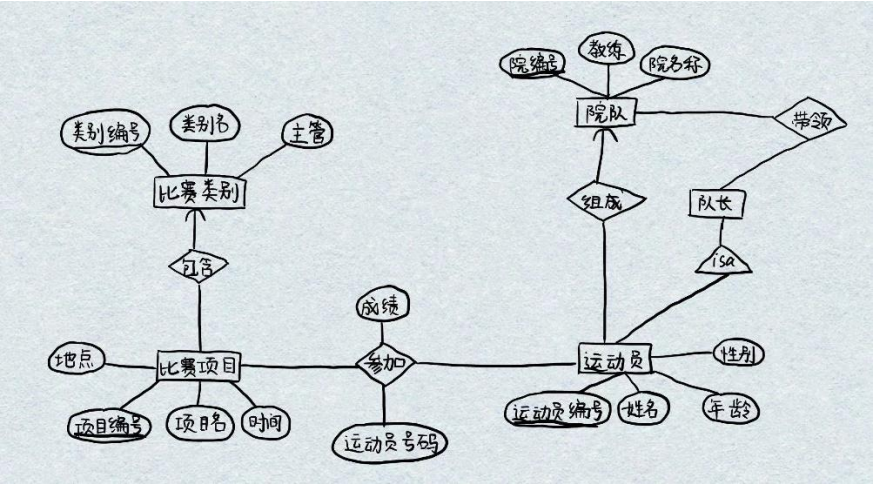
一、 需求描述

（1）每一比赛类别包括类别编号、类别名称和主管的属性，并包含多个比赛项目，依靠类别编号识别。
（2）每个比赛项目包含项目编号、项目名称、比赛时间和地点属性，并且只包含在一个比赛类别里，依靠项目编号识别
（3）每个院队有院编号、院名称、教练属性，并有一位队长带领，由多个运动员组成，依靠院编号识别
（4）每位运动员有运动员编号，姓名、年龄、性别属性，依靠运动员编号识别
（5）每位运动员参加某个项目时都会有对应的成绩和号码。
（6）每个项目可以有多个运动员参与，每个运动员可以参与多个项目。

二、 采用教材中介绍的方法实现设计

a) ER 图的绘制



b) 将 ER 图转换为关系模式

以下是关系模式，包括主键（PK）和外键（FK）：

比赛类别（类别编号（PK），类别名称，主管）
比赛项目（项目编号（PK），项目名称，比赛时间，地点，类别编号（FK））
院队（院编号（PK），院名称，教练， 队长编号(FK)）
运动员（运动员编号（PK），姓名，年龄，性别，院编号（FK））
参赛记录（运动员编号（PK,FK），项目编号（PK,FK），成绩，运动员号码）

c) 该数据库模式上 5 个查询语句样例

（1）单表查询

查询所有男运动员的名字：

```
Select 姓名
From 运动员
Where 性别=“男”
```

（2）多表连接查询

查询张三所参加的所有项目的项目编号：

```
Select 项目编号
From 运动员 natural join 参加
Where 姓名=“张三”
```

（3）多表嵌套查询

查询张三所参加的所有项目的项目名字

```
Select 项目名
From 比赛项目
Where 项目编号 in
（Select 项目编号
From 运动员 natural join 参加
Where 姓名=“张三”）
```

（4）EXSISTS 查询

查找只参加了一个项目的运动员的编号：

```
Select 运动员编号
Form 参加 a1
Where not exists
（Select *
From 参加
Where a1.运动员编号=运动员编号 and a1.项目编号<>项目编号）
```

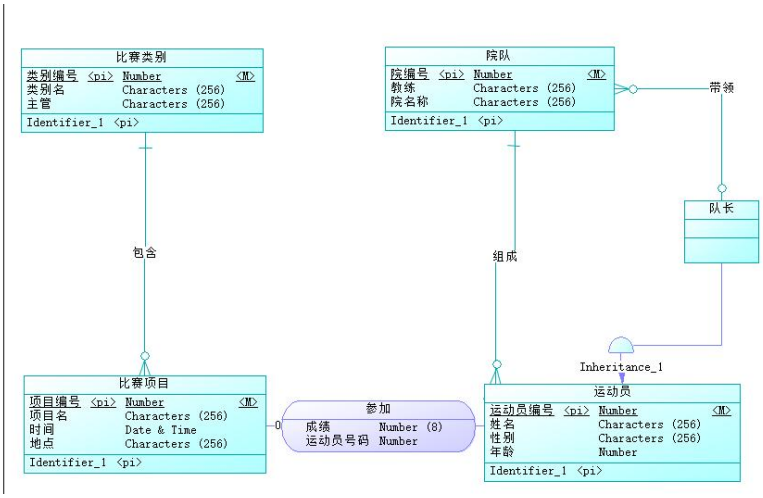
（5）聚合操作查询

各个编号的运动员的所有项目的平均成绩是多少

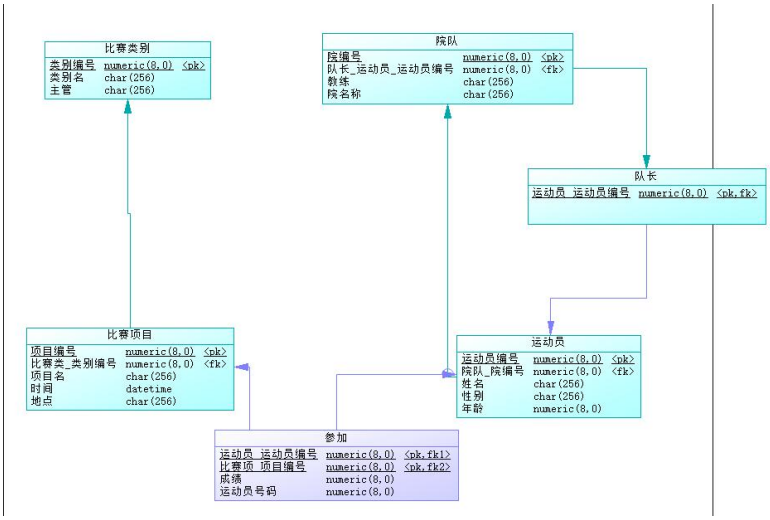
```
Select 运动员编号，AVG（成绩）
From 参加
Group by 运动员编号
```

三、 Power design 实现设计

a) 概念模型 ER 图



b) 关系模型图



c)　　创建数据库的 SQL 语句

```
drop table if exists 参加;

drop table if exists 比赛类别;

drop table if exists 比赛项目;

drop table if exists 运动员;

drop table if exists 队长;

drop table if exists 院队;

/*==============================================================*/
/* Table: 参加                                                  */
/*==============================================================*/
create table 参加
(
   运动员编号                  numeric(8,0) not null,
   项目编号                   numeric(8,0) not null,
   成绩                      numeric(8,0),
   运动员号码                   numeric(8,0),
   primary key (运动员编号, 项目编号)
);

/*==============================================================*/
/* Table: 比赛类别                                              */
/*==============================================================*/
create table 比赛类别
(
```

```
   类别编号                  numeric(8,0) not null,
   类别名                    char(256),
   主管                      char(256),
   primary key (类别编号)
);

/*==============================================================*/
/* Table: 比赛项目                                              */
/*==============================================================*/
create table 比赛项目
(
   项目编号                  numeric(8,0) not null,
   类别编号                  numeric(8,0) not null,
   项目名                    char(256),
   时间                      datetime,
   地点                      char(256),
   primary key (项目编号)
);

/*==============================================================*/
/* Table: 运动员                                                */
/*==============================================================*/
create table 运动员
(
   运动员编号                  numeric(8,0) not null,
   院编号                    numeric(8,0) not null,
   姓名                      char(256),
   性别                      char(256),
   年龄                      numeric(8,0),
   primary key (运动员编号)
);

/*==============================================================*/
/* Table: 队长                                                  */
/*==============================================================*/
create table 队长
(
   运动员编号                  numeric(8,0) not null,
   primary key (运动员编号)
);

/*==============================================================*/
/* Table: 院队                                                  */
/*==============================================================*/
```

```
create table 院队
(
   院编号                  numeric(8,0) not null,
   运动员编号                numeric(8,0),
   教练                   char(256),
   院名称                  char(256),
   primary key (院编号)
);

alter table 参加 add constraint FK_参加 foreign key ()
      references 运动员 (运动员编号);

alter table 参加 add constraint FK_参加2 foreign key ()
      references 比赛项目 (项目编号);

alter table 比赛项目 add constraint FK_包含 foreign key ()
      references 比赛类别 (类别编号);

alter table 运动员 add constraint FK_组成 foreign key ()
      references 院队 (院编号);

alter table 队长 add constraint FK_Inheritance_1 foreign key ()
      references 运动员 (运动员编号);

alter table 院队 add constraint FK_带领 foreign key ()
      references 队长 (运动员编号);
```

四、 分析比较上述两种方法

a) PowerDesigner 在设计中引入了一个额外的表“队长”来表示院队的队长。这是一个单独的表，与“运动员”表具有相同的主键，表示队长是运动员的一个子类。这样的设计有助于在需要时扩展队长的相关信息，但增加了查询和维护的复杂性

b) Powerdesign 生成的 SQL 语句在每一个 create table 中，会先声明用户定义的属性，最后使用 primary key 关键字来声明主键，而外键不在 create table 内部声明。附加的语句 alter table 则是为了声明外键，即一对多关系中一的那一方的主键放在另一方的属性里当成外键，有效表示了那些没有独立属性的关系，无需再为没有独立属性的关系再创建一个表(即多对一关系无需单独创建表)。