**数据库系统概论课程设计**

**笔记**

目录

[一、课程需求 5](#_Toc91853329)

[1、知识回顾 5](#_Toc91853330)

[2、数据库设计 5](#_Toc91853331)

[3、GUI开发--javaFX 5](#_Toc91853332)

[4、项目开发--成绩管理系统 5](#_Toc91853333)

[二、知识回顾 6](#_Toc91853334)

[1、数据库原理--范式 6](#_Toc91853335)

[2、E-R图 8](#_Toc91853336)

[3、SQL介绍 8](#_Toc91853337)

[4、事务 9](#_Toc91853338)

[5、DML 9](#_Toc91853339)

[三、数据库设计 10](#_Toc91853340)

[1、工具安装 10](#_Toc91853341)

[2、开发阶段 11](#_Toc91853342)

[3、概念数据模型 11](#_Toc91853343)

[4、物理数据模型 26](#_Toc91853344)

[6、生成数据库脚本 30](#_Toc91853345)

[7、生成物理数据库 33](#_Toc91853346)

[8、生成测试数据 33](#_Toc91853347)

[9、 33](#_Toc91853348)

[10、 33](#_Toc91853349)

[四、 34](#_Toc91853350)

[1、 34](#_Toc91853351)

[2、 34](#_Toc91853352)

[3、 34](#_Toc91853353)

[4、 34](#_Toc91853354)

[5、 35](#_Toc91853355)

[6、 35](#_Toc91853356)

[7、 35](#_Toc91853357)

[8、 35](#_Toc91853358)

[9、 35](#_Toc91853359)

[10、 36](#_Toc91853360)

[五、 36](#_Toc91853361)

[1、 36](#_Toc91853362)

[2、 36](#_Toc91853363)

[3、 37](#_Toc91853364)

[4、 37](#_Toc91853365)

[5、 37](#_Toc91853366)

[6、 37](#_Toc91853367)

[7、 37](#_Toc91853368)

[8、 38](#_Toc91853369)

[9、 38](#_Toc91853370)

[10、 38](#_Toc91853371)

[六、 38](#_Toc91853372)

[1、 38](#_Toc91853373)

[2、 39](#_Toc91853374)

[3、 39](#_Toc91853375)

[4、 39](#_Toc91853376)

[5、 39](#_Toc91853377)

[6、 40](#_Toc91853378)

[7、 40](#_Toc91853379)

[8、 40](#_Toc91853380)

[9、 40](#_Toc91853381)

[10、 40](#_Toc91853382)

[七、 41](#_Toc91853383)

[1、 41](#_Toc91853384)

[2、 41](#_Toc91853385)

[3、 41](#_Toc91853386)

[4、 42](#_Toc91853387)

[5、 42](#_Toc91853388)

[6、 42](#_Toc91853389)

[7、 42](#_Toc91853390)

[8、 42](#_Toc91853391)

[9、 43](#_Toc91853392)

[10、 43](#_Toc91853393)

[八、 43](#_Toc91853394)

[1、 43](#_Toc91853395)

[2、 43](#_Toc91853396)

[3、 44](#_Toc91853397)

[4、 44](#_Toc91853398)

[5、 44](#_Toc91853399)

[6、 44](#_Toc91853400)

[7、 45](#_Toc91853401)

[8、 45](#_Toc91853402)

[9、 45](#_Toc91853403)

[10、 45](#_Toc91853404)

[九、 46](#_Toc91853405)

[1、 46](#_Toc91853406)

[2、 46](#_Toc91853407)

[3、 46](#_Toc91853408)

[4、 46](#_Toc91853409)

[5、 47](#_Toc91853410)

[6、 47](#_Toc91853411)

[7、 47](#_Toc91853412)

[8、 47](#_Toc91853413)

[9、 47](#_Toc91853414)

[10、 48](#_Toc91853415)

[十、 48](#_Toc91853416)

[1、 48](#_Toc91853417)

[2、 48](#_Toc91853418)

[3、 48](#_Toc91853419)

[4、 49](#_Toc91853420)

[5、 49](#_Toc91853421)

[6、 49](#_Toc91853422)

[7、 49](#_Toc91853423)

[8、 50](#_Toc91853424)

[9、 50](#_Toc91853425)

[10、 50](#_Toc91853426)

[十一、 50](#_Toc91853427)

[1、 50](#_Toc91853428)

[2、 51](#_Toc91853429)

[3、 51](#_Toc91853430)

[4、 51](#_Toc91853431)

[5、 51](#_Toc91853432)

[6、 51](#_Toc91853433)

[7、 52](#_Toc91853434)

[8、 52](#_Toc91853435)

[9、 52](#_Toc91853436)

[10、 52](#_Toc91853437)

[十二、 53](#_Toc91853438)

[1、 53](#_Toc91853439)

[2、 53](#_Toc91853440)

[3、 53](#_Toc91853441)

[4、 53](#_Toc91853442)

[5、 54](#_Toc91853443)

[6、 54](#_Toc91853444)

[7、 54](#_Toc91853445)

[8、 54](#_Toc91853446)

[9、 54](#_Toc91853447)

[10、 55](#_Toc91853448)

[十三、 55](#_Toc91853449)

[1、 55](#_Toc91853450)

[2、 55](#_Toc91853451)

[3、 55](#_Toc91853452)

[4、 56](#_Toc91853453)

[5、 56](#_Toc91853454)

[6、 56](#_Toc91853455)

[7、 56](#_Toc91853456)

[8、 57](#_Toc91853457)

[9、 57](#_Toc91853458)

[10、 57](#_Toc91853459)

[十四、 57](#_Toc91853460)

[1、 57](#_Toc91853461)

[2、 58](#_Toc91853462)

[3、 58](#_Toc91853463)

[4、 58](#_Toc91853464)

[5、 58](#_Toc91853465)

[6、 58](#_Toc91853466)

[7、 59](#_Toc91853467)

[8、 59](#_Toc91853468)

[9、 59](#_Toc91853469)

[10、 59](#_Toc91853470)

[十五、 60](#_Toc91853471)

[1、 60](#_Toc91853472)

[2、 60](#_Toc91853473)

[3、 60](#_Toc91853474)

[4、 60](#_Toc91853475)

[5、 61](#_Toc91853476)

[6、 61](#_Toc91853477)

[7、 61](#_Toc91853478)

[8、 61](#_Toc91853479)

[9、 62](#_Toc91853480)

[10、 62](#_Toc91853481)

# 一、课程需求

## 1、知识回顾

复习

## 2、数据库设计

使用PD进行数据库工具

毕设中的数据库设计

ER图

物理数据库

毕设中的数据库相关文档

## 3、GUI开发--javaFX

javaFX项目开发

设计窗体

展示数据

添加数据

修改数据

删除数据

## 4、项目开发--成绩管理系统

需求

数据库设计

编码实现功能

报告

# 二、知识回顾

## 1、数据库原理--范式

范式

1NF

🡪第一范式

🡪列具有原子性

🡪列不可拆分，一事一地

规划表

|  |  |  |  |
| --- | --- | --- | --- |
| 学生信息表 | | | |
| 学号\* | 姓名 | 电话 | 性别 |
| 1 | 张三 | 110 | 男 |
| 1 | 张三 | 120 | 男 |

问题：增删改操作异常，查冗余数据

解决方案一：电话不具有原子性，可以继续拆分，并兼顾主键表达

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 学生信息表 | | | | |
| 学号\* | 姓名 | 电话1 | 电话2 | 性别 |
| 1 | 张三 | 110 | 120 | 男 |
| 2 | 李四 | 118 | 991 | 女 |

解决方案二：强制电话只有存储一个号码，并兼顾主键表达，数据完整性约束（电话是唯一）

|  |  |  |  |
| --- | --- | --- | --- |
| 学生信息表 | | | |
| 学号\* | 姓名 | 电话（\*） | 性别 |
| 1 | 张三 | 110 | 男 |
| 2 | 李四 | 991 | 女 |

2NF

🡪第二范式

🡪非主键列完全依赖于所有主键列

🡪符合第一范式时，主键列仅有一列，则自动满足第二范式

🡪非复合主键必须完全依赖于所有主键列

规划表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 学生成绩信息表 | | | | |
| 学号\* | 姓名 | 课程编号\* | 课程名称 | 成绩 |
| 1 | 张三 | 1 | java | 55 |
| 1 | 张三 | 2 | spark | 66 |
| 2 | 李四 | 3 | mysql | 44 |
| 3 | 王五 | 1 | java | 77 |

问题：查数据冗余，删改增操作异常

解决方案：一事一地，学号+课程编号🡪成绩，学号🡪姓名，课程编号🡪课程名称

|  |  |  |
| --- | --- | --- |
| 学生成绩信息表 | | |
| 学号\* | 课程编号\* | 成绩 |
| 1 | 1 | 55 |
| 1 | 2 | 66 |
| 2 | 3 | 44 |
| 3 | 1 | 77 |

|  |  |
| --- | --- |
| 学生信息表 | |
| 学号\* | 姓名 |
| 1 | 张三 |
| 2 | 李四 |
| 3 | 王五 |

|  |  |
| --- | --- |
| 课程信息表 | |
| 课程编号\* | 课程名称 |
| 1 | java |
| 2 | spark |
| 3 | mysql |

3NF

🡪第三范式

🡪不存在函数传递依赖关系

🡪因为fx(a)->fx(b),fx(b)->fx(c)所以得到结论fx(a)->fx(c)

🡪根据一事一地原则，拆分传递依赖关系

规划表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 学生信息表 | | | | | |
| 学号\* | 姓名 | 性别 | 系编号 | 系名称 | 系地址 |
| 1 | 张三 | 男 | 1 | 数学 | 1#812 |
| 2 | 李四 | 女 | 1 | 数学 | 1#812 |
| 3 | 王五 | 男 | 2 | 计算机 | 2#814 |

问题：查数据冗余，增删改异常

解决：一事一地原则，(学号->系编号)，(系编号->系名称，系地址)，(学号->系名称，系地址)

|  |  |  |  |
| --- | --- | --- | --- |
| 学生信息表 | | | |
| 学号\* | 姓名 | 性别 | 系编号 |
| 1 | 张三 | 男 | 1 |
| 2 | 李四 | 女 | 1 |
| 3 | 王五 | 男 | 2 |

|  |  |  |
| --- | --- | --- |
| 系信息表 | | |
| 系编号\* | 系名称 | 系地址 |
| 1 | 数学 | 1#812 |
| 2 | 计算机 | 2#814 |

……

范式级别

1NF < 2NF < 3NF < ……

实际开发

必须至少满足第3范式

坚持满足第3范式的设计

## 2、E-R图

Entity-Relationship图

实体-关系 图

标识实体

标识实体的属性

标识实体的关系

毕设中必须体现E-R图

毕设中必须体现数据字典

## 3、SQL介绍

SQL--结构化查询语言

DDL数据定义语言

对数据库中的各种对象进行创建、修改、删除等操作

create/alter/drop……

DML数据操纵语言

对表中的数据进行增、删、改、查等操作

select/update/insert/delete……

DCL数据控制语言

控制用户对数据库中的各种对象或表中数据进行限制访问

grant/revoke/deny……

TCL事务控制语言

管理数据库中的事务操作

begin transaction/commit transaction/rollback transaction/save transaction……

## 4、事务

事务就是指一个动作由N步逻辑操作所构成

转帐

B-

A+

ACID四个属性

原子性

一致性

隔离性

持久性

## 5、DML

增加

insert into 表名 values(值1,值2,……)

insert into 表名(列1,列2,……) values(值1,值2,……)

insert into 表名(列1,列2,……) select 列1,列2,…… from 表名

修改

update 表名 set 列1=值1, 列2=值2,…… where 条件查询

update 表名 set 列1=值1, 列2=值2,…… where 多表

删除

delete from 表名 where 条件查询

delete 表1,表2 from 表1,表2,表3 where 多表公共字段判断

查询

单表查询

select \* from 表名 where 条件表达式 group by 列 having 条件判断 order by 排序列

多表查询

select \* from 表名1,表名2 where 条件表达式

连接查询

select \* from 表名1 inner join /left join / right join 表名2 on 公共字段比较 where 条件表达式

子查询

select \* from 表名1 where 列1 in (查询语句)

相关子查询

select \* from 表名1 where exists (select \* from 表名2 where 表名2.列 = 表名2.列名)

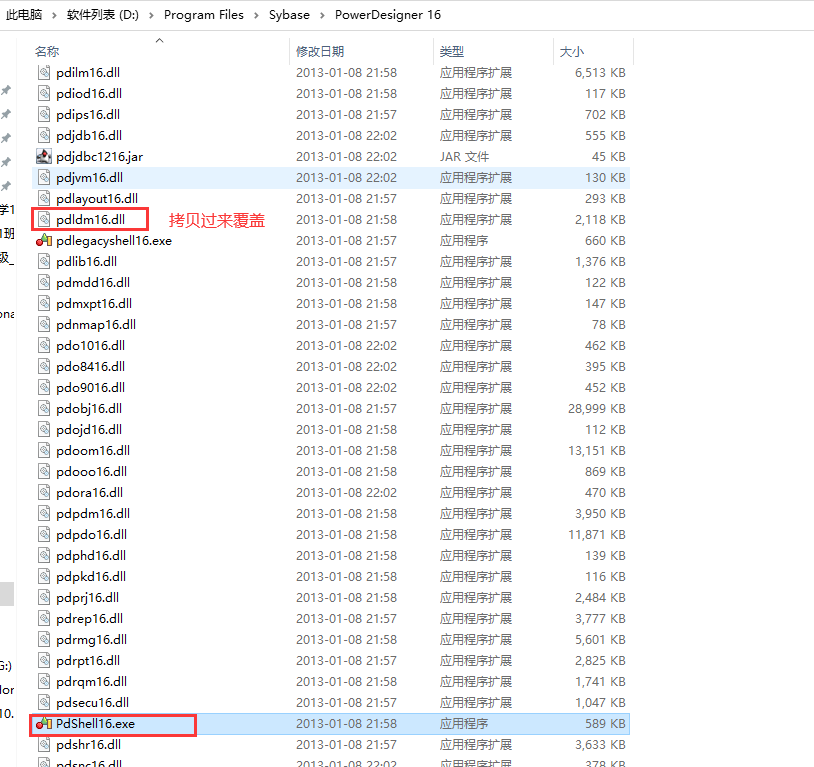
# 三、数据库设计

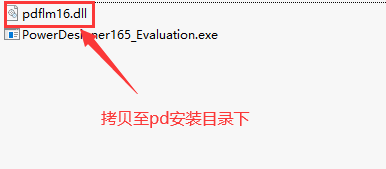
## 1、工具安装

下载

PowerDesigner165

安装





PowerDesigner最初由[Xiao-Yun Wang](https://zh.wikipedia.org/w/index.php?title=Xiao-Yun_Wang&action=edit&redlink=1)（[王曉昀](https://zh.wikipedia.org/w/index.php?title=%E7%8E%8B%E6%99%93%E6%98%80&action=edit&redlink=1)）在[SDP Technologies](https://zh.wikipedia.org/w/index.php?title=SDP_Technologies&action=edit&redlink=1)公司開發完成。在法國稱為AMC\*Designor，在國際市場上稱為S-Designor。在這兩個產品名字中都包含「or」，它實際上特指「Oracle」，因為在產品開發的最開始是為Oracle數據庫設計的，但是很快就發展並支持市場上所有主流的數據庫系統。[SDP Technologies](https://zh.wikipedia.org/w/index.php?title=SDP_Technologies&action=edit&redlink=1)是一個建於1983年的法國公司，1995年，Powersoft公司購買了該公司，而在1994年早期，Sybase已經買下了Powersoft公司。在這些併購之後，為了保持Powersoft的產品商標的一致，改名叫做「PowerDesigner」。 目前Sybase擁有PowerDesigner及其法文版PowerAMC的所有權利。

## 2、开发阶段

需求分析阶段

概要设计阶段

详细设计阶段

编码

测试

部署

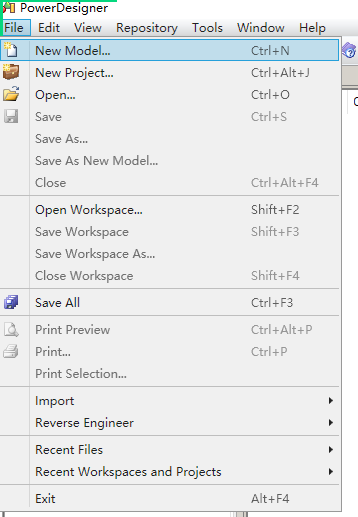
## 3、概念数据模型

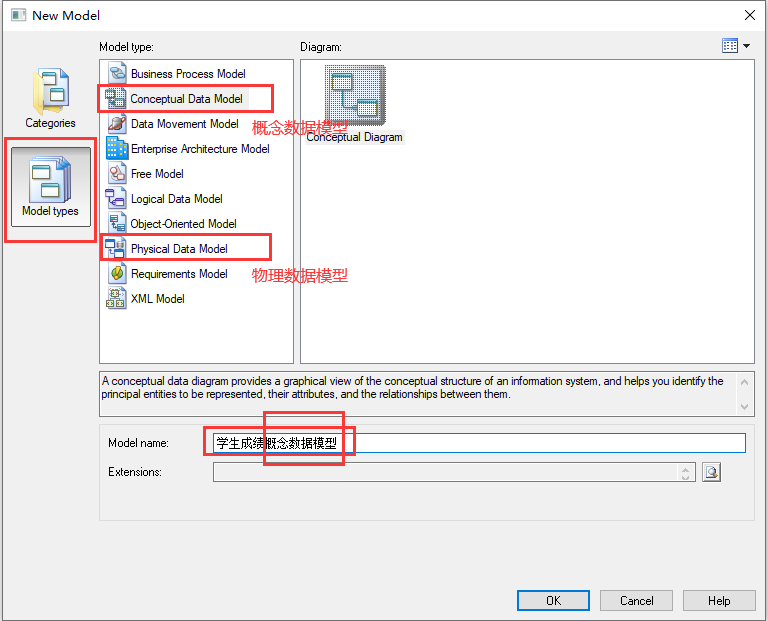
概念数据模型

与具体的数据库产品无关

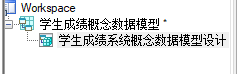
逻辑设计🡪ER图

概念数据模型，成功后，可以随意生成各种特定的数据库脚本

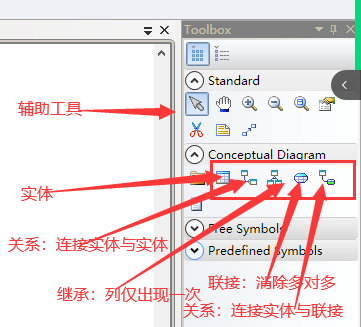




修改图表名称，方便文档输出



工具栏



确定需求

学生成绩管理系统

学生信息

学号

姓名

电话

地址

课程信息

课程编号

课程名称

学分

课时数

授课老师

成绩信息

学号

课程编号

成绩

考试日期

老师信息

工号

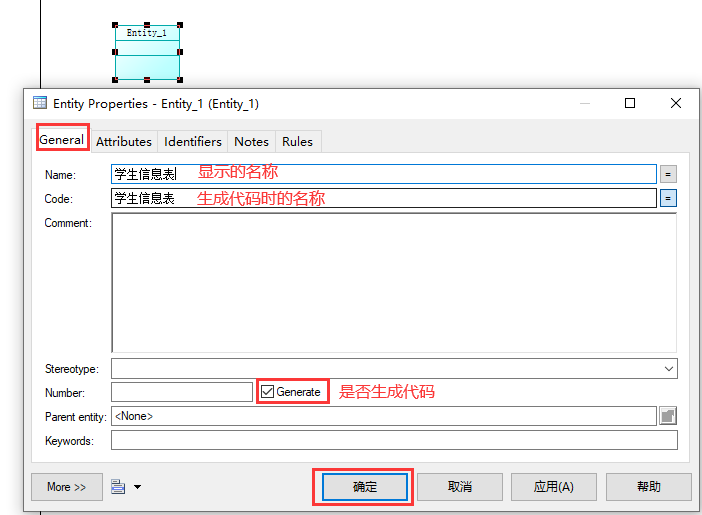
姓名

电话

地址

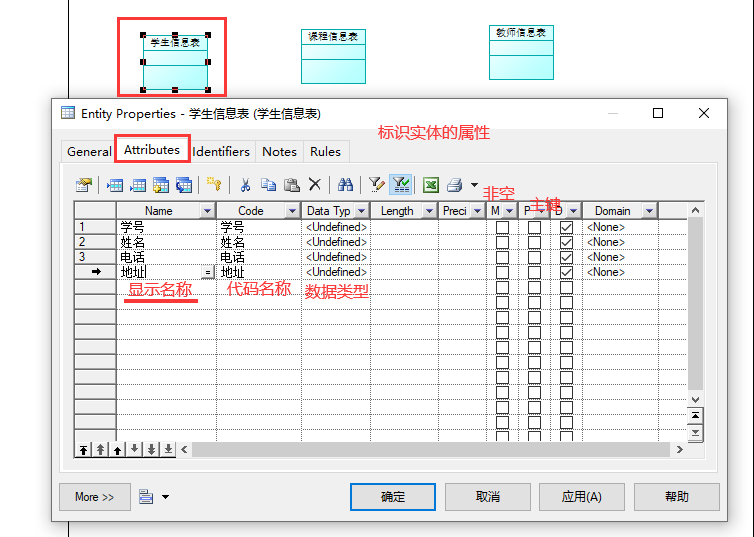
教龄

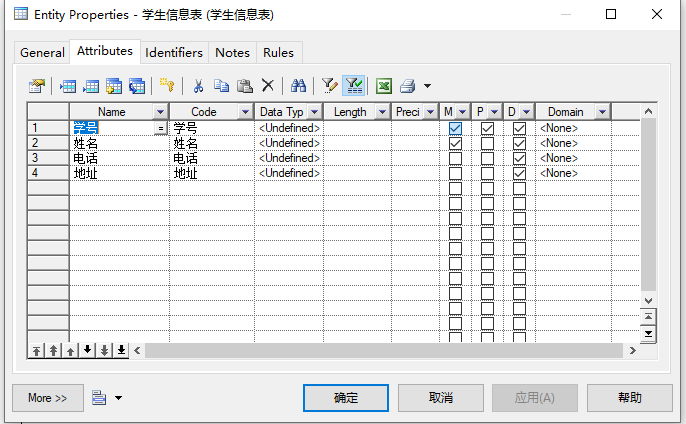
标识实体

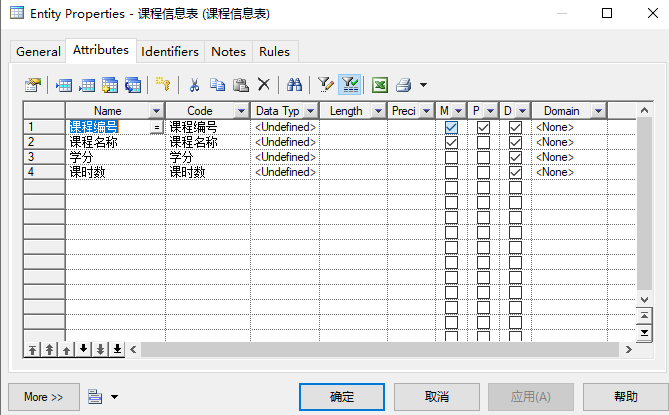




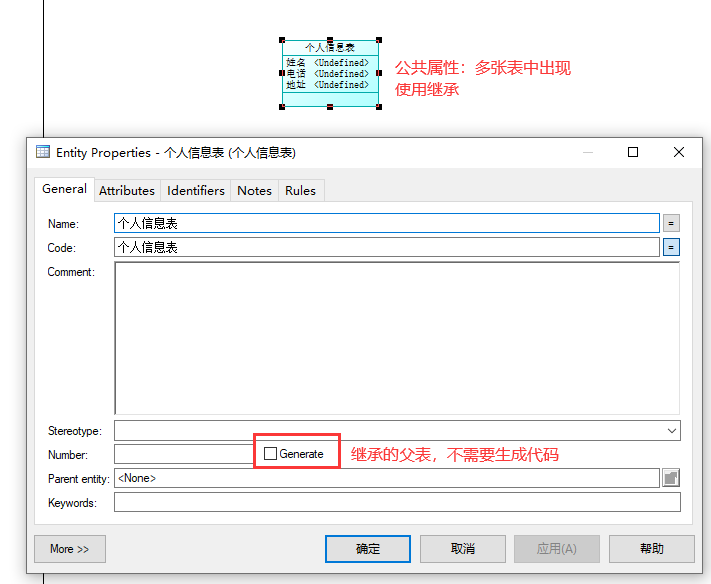
标识实体的属性



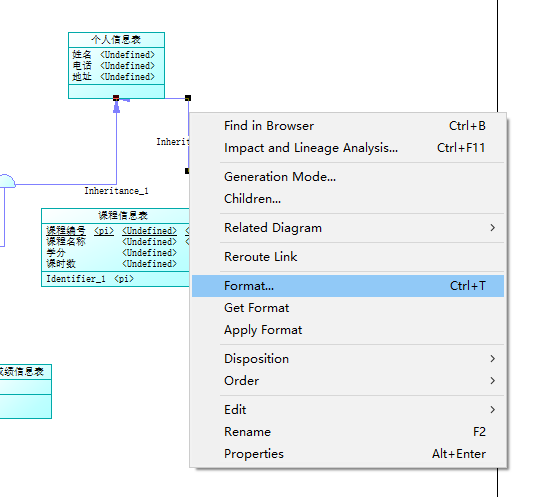


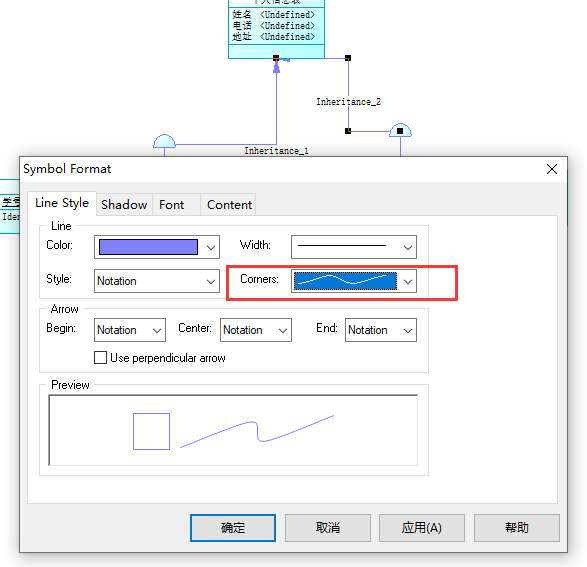


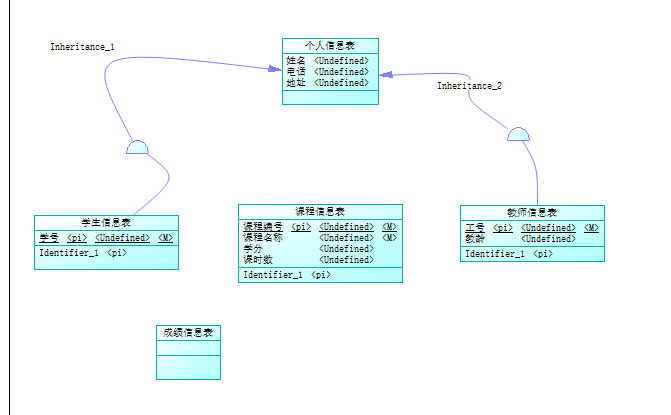
使用继承



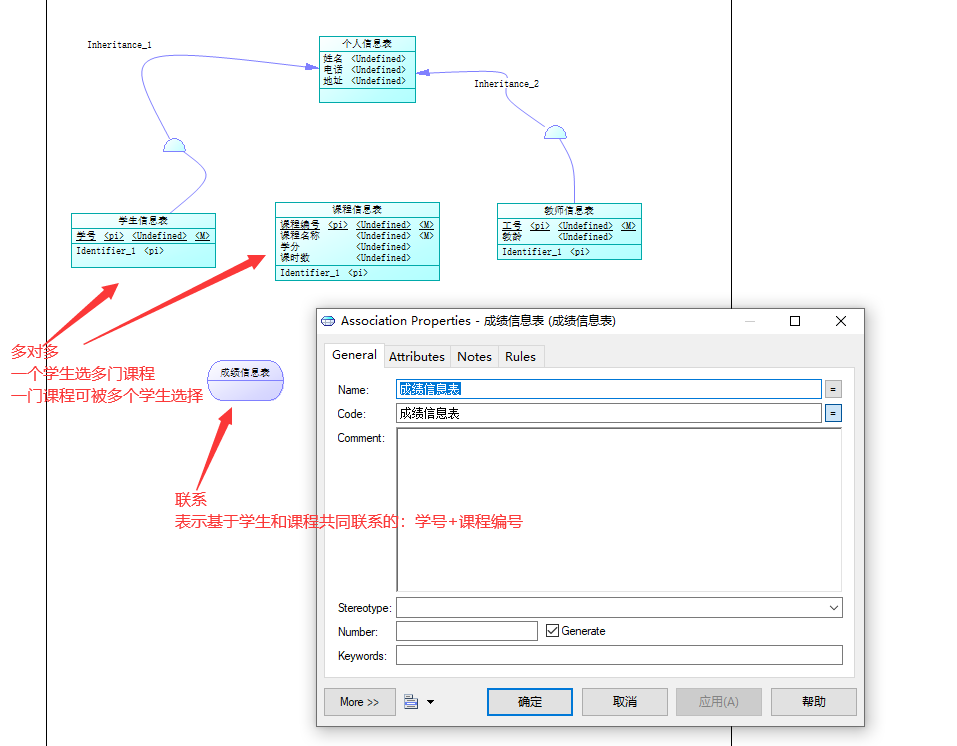
不使用直角线条



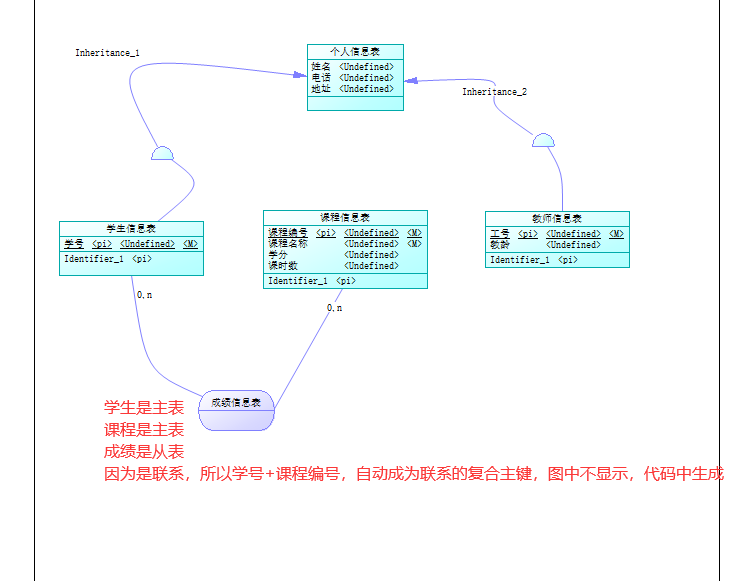




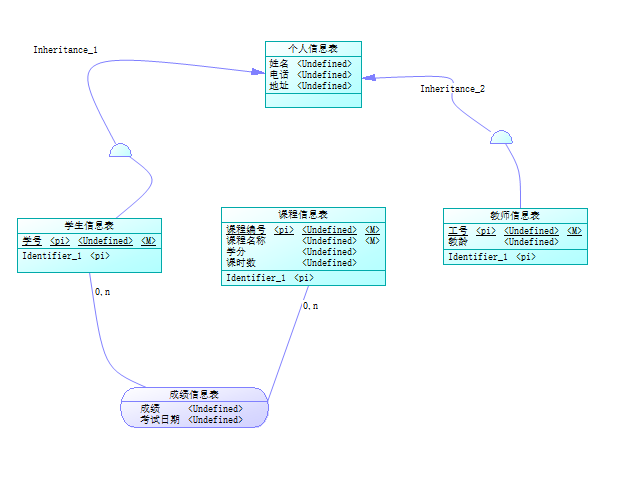
使用联系



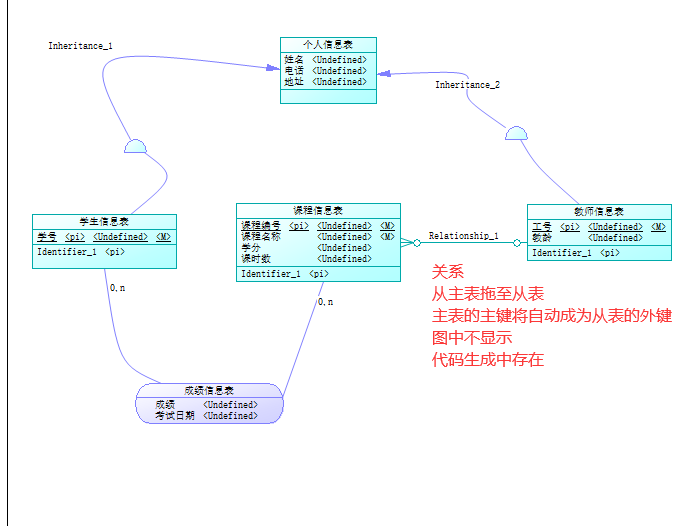
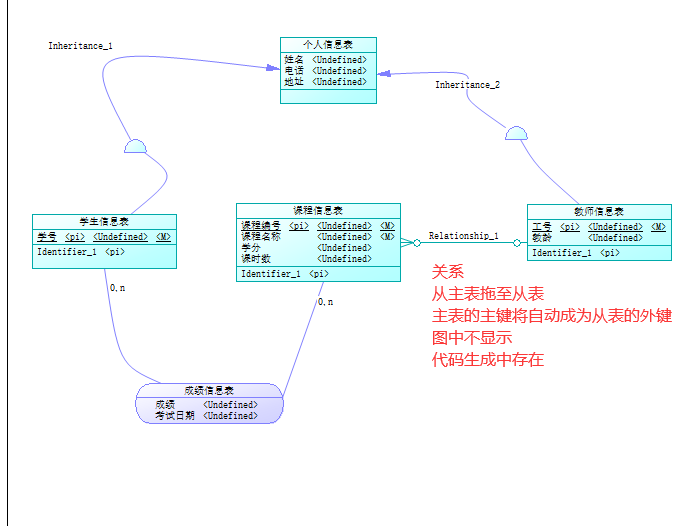
标识关系

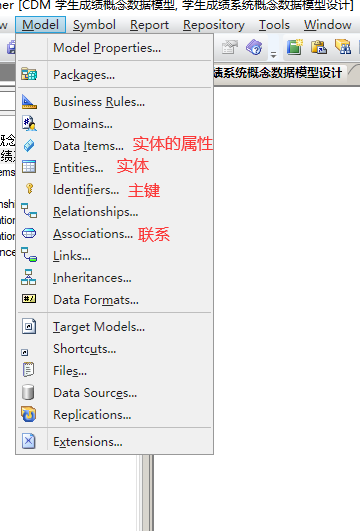


标识联系的属性

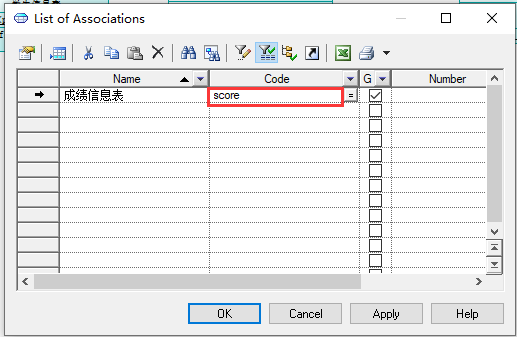


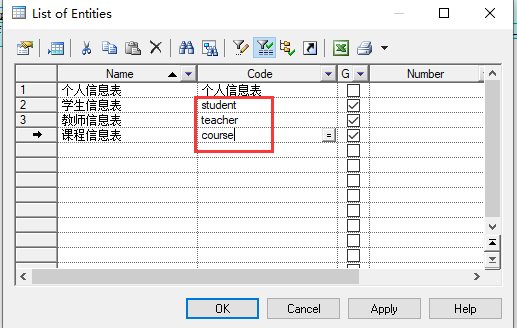
标识关系

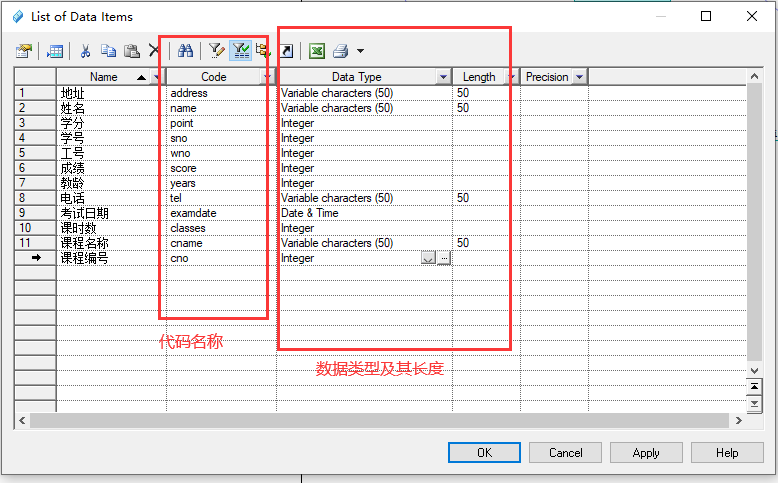


设置实体和联系

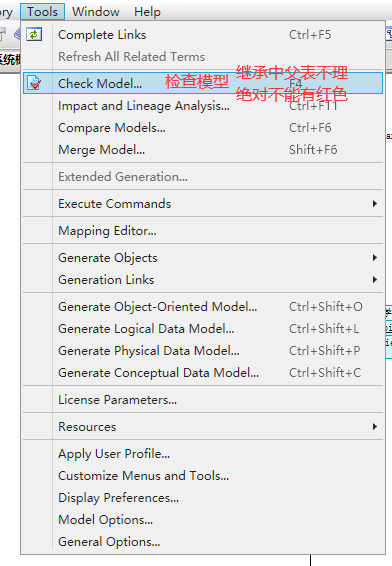


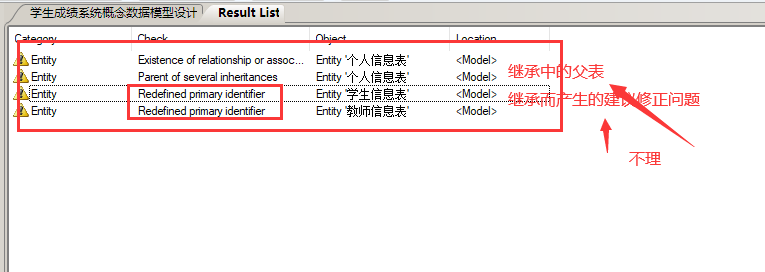


设置属性

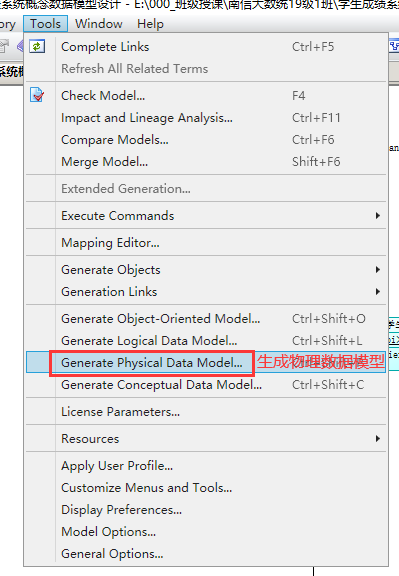


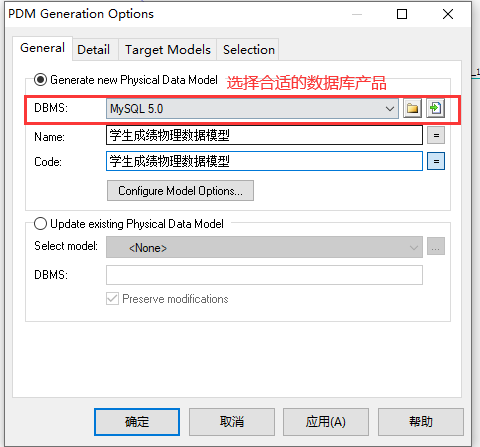
检查模型

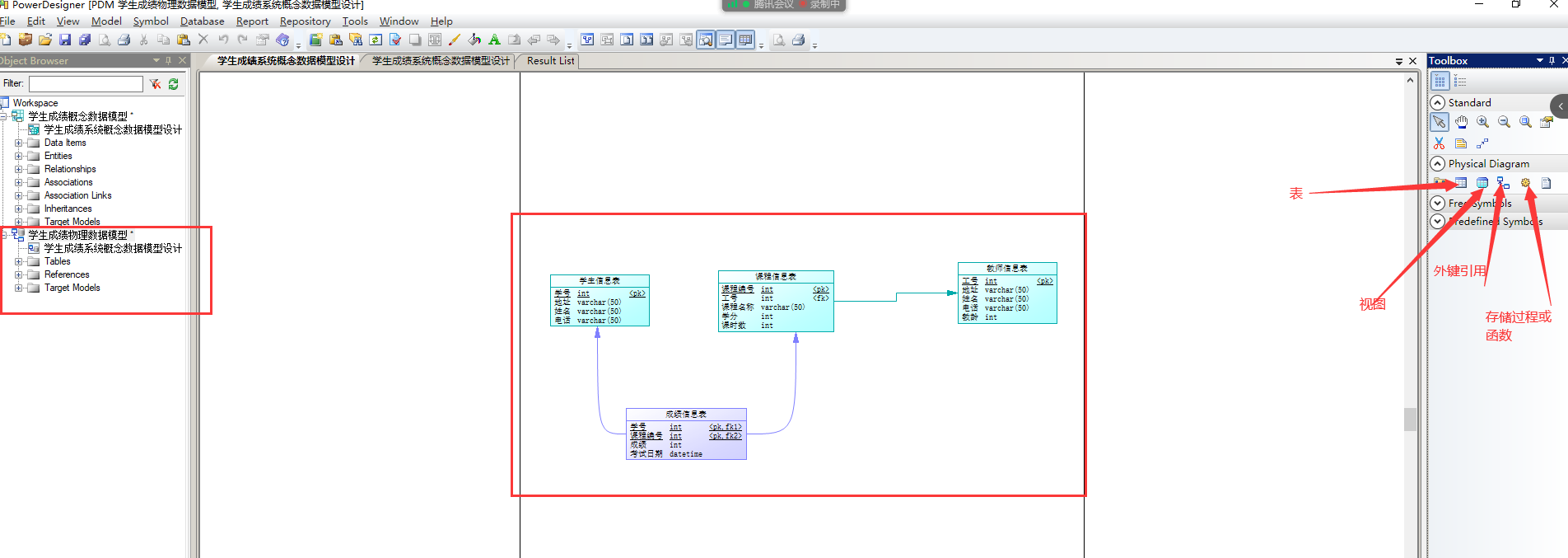




生成物理数据模型







## 4、物理数据模型

物理数据模型

与特定的数据库产品相关🡪如MySQL5.0

针对具体的物理数据库操作

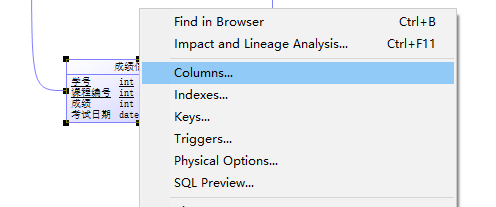
表、主键、约束、存储过程、函数、索引、触发器等

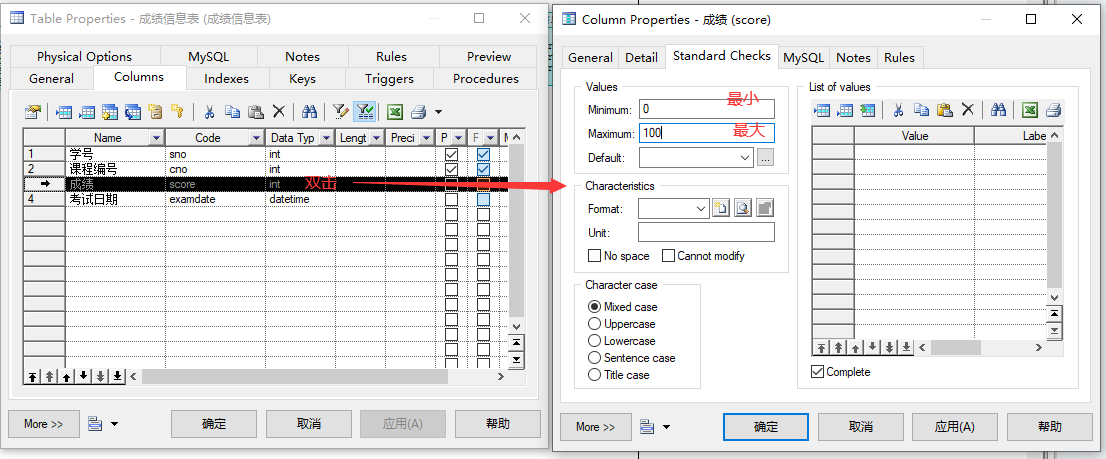
代码需要自己编写

自动增长列

需要通过代码或生成物理数据库以后，再进行调整

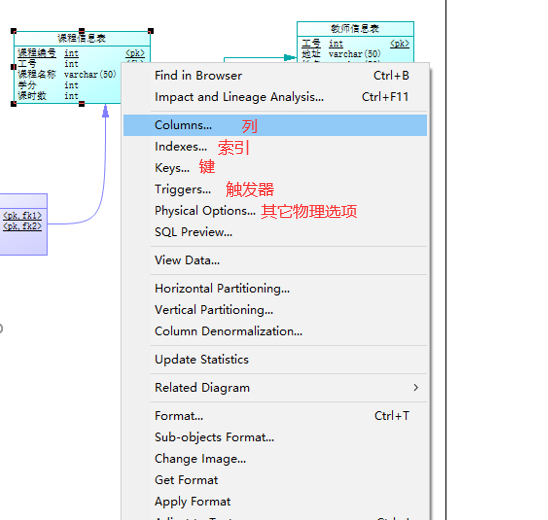
调整约束



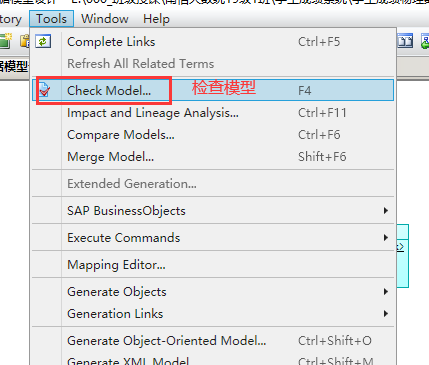


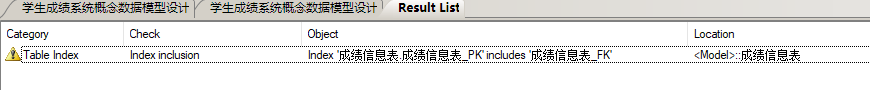
其它调整

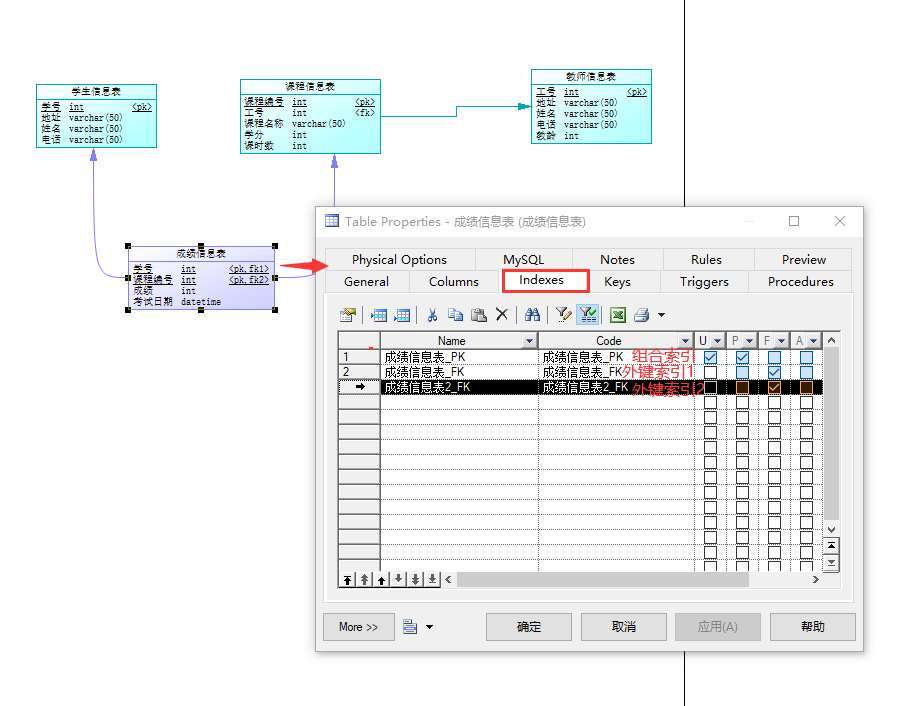
列、索引、键、触发器、其它物理选项……



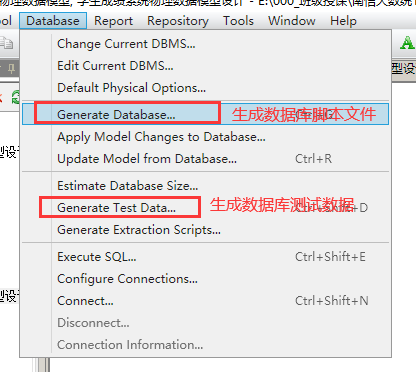
检查模型

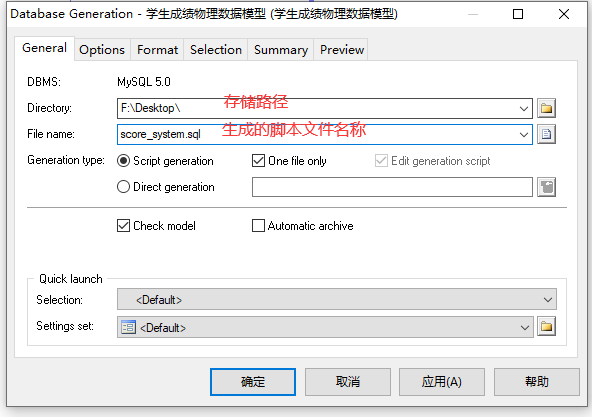




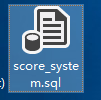


## 6、生成数据库脚本





生成了脚本文件



查看代码

/\*==============================================================\*/

/\* DBMS name: MySQL 5.0 \*/

/\* Created on: 2021-12-31 14:26:42 \*/

/\*==============================================================\*/

drop table if exists course;

drop table if exists score;

drop table if exists student;

drop table if exists teacher;

/\*==============================================================\*/

/\* Table: course \*/

/\*==============================================================\*/

create table course

(

cno int not null,

wno int,

cname varchar(50) not null,

point int,

classes int,

primary key (cno)

);

/\*==============================================================\*/

/\* Table: score \*/

/\*==============================================================\*/

create table score

(

sno int not null,

cno int not null,

score int,

examdate datetime,

primary key (sno, cno)

);

/\*==============================================================\*/

/\* Table: student \*/

/\*==============================================================\*/

create table student

(

sno int not null,

address varchar(50),

name varchar(50),

tel varchar(50),

primary key (sno)

);

/\*==============================================================\*/

/\* Table: teacher \*/

/\*==============================================================\*/

create table teacher

(

wno int not null,

address varchar(50),

name varchar(50),

tel varchar(50),

years int,

primary key (wno)

);

alter table course add constraint FK\_Relationship\_1 foreign key (wno)

references teacher (wno) on delete restrict on update restrict;

alter table score add constraint FK\_score foreign key (sno)

references student (sno) on delete restrict on update restrict;

alter table score add constraint FK\_score2 foreign key (cno)

references course (cno) on delete restrict on update restrict;

## 7、生成物理数据库

## 8、生成测试数据

## 9、

## 10、

# 四、

## 1、

## 2、

## 3、

## 4、

## 5、

## 6、

## 7、

## 8、

## 9、

## 10、

# 五、

## 1、

## 2、

## 3、

## 4、

## 5、

## 6、

## 7、

## 8、

## 9、

## 10、

# 六、

## 1、

## 2、

## 3、

## 4、

## 5、

## 6、

## 7、

## 8、

## 9、

## 10、

# 七、

## 1、

## 2、

## 3、

## 4、

## 5、

## 6、

## 7、

## 8、

## 9、

## 10、

# 八、

## 1、

## 2、

## 3、

## 4、

## 5、

## 6、

## 7、

## 8、

## 9、

## 10、

# 九、

## 1、

## 2、

## 3、

## 4、

## 5、

## 6、

## 7、

## 8、

## 9、

## 10、

# 十、

## 1、

## 2、

## 3、

## 4、

## 5、

## 6、

## 7、

## 8、

## 9、

## 10、

# 十一、

## 1、

## 2、

## 3、

## 4、

## 5、

## 6、

## 7、

## 8、

## 9、

## 10、

# 十二、

## 1、

## 2、

## 3、

## 4、

## 5、

## 6、

## 7、

## 8、

## 9、

## 10、

# 十三、

## 1、

## 2、

## 3、

## 4、

## 5、

## 6、

## 7、

## 8、

## 9、

## 10、

# 十四、

## 1、

## 2、

## 3、

## 4、

## 5、

## 6、

## 7、

## 8、

## 9、

## 10、

# 十五、

## 1、

## 2、

## 3、

## 4、

## 5、

## 6、

## 7、

## 8、

## 9、

## 10、