

实验九 PowerDesigner数据库建模

姓名	学号	学院	日期
臧祝利	202011998088	人工智能学院	2022.11.1

实验目的

- (1) 熟悉 PowerDesigner 应用环境
- (2) 掌握 PowerDesigner 概念模型的定义和创建方法
- (3) 掌握利用 PowerDesigner 建立数据库后台的方法

实验内容

- (1) 分析题库管理系统中的实体关系模式，设计相应的 E-R 图
- (2) 根据实体关系模式，设计系统的逻辑结构，即将 E-R 图转换为关系结构
- (3) 利用 PowerDesigner 设计系统的物理结构，并根据数据库管理系统转换成相应的 SQL 文件

实验思路

- (1) 根据课上使用的题库管理系统，分析其内部的实体关系情况，并画出 E-R 图。
- (2) 某大学欲建个考试系统。包括一个题目数据库，里面有大量老师们出的试卷（可以多名老师合作出 1 份试卷）， tid , tname , tsex , tage , tdepa ;试卷有 eid , ename , etype , 学生可以根据需要报考某份试卷，答卷后获得一个分数。大学有若干个院系，存有院系名、办公地点和联系电话；每位老师同学只能隶属一个院系；老师学生分别有编号(教师编号/学生编号)、姓名、年龄、性别、所在院系；试卷有试卷编号、试卷名、试卷类型、考试时长等。

根据(2)可知，

包含四个实体，分别是老师、学生、院系和试卷；

包含四种联系，分别是出卷、报考、属于(老师)、属于(学生)

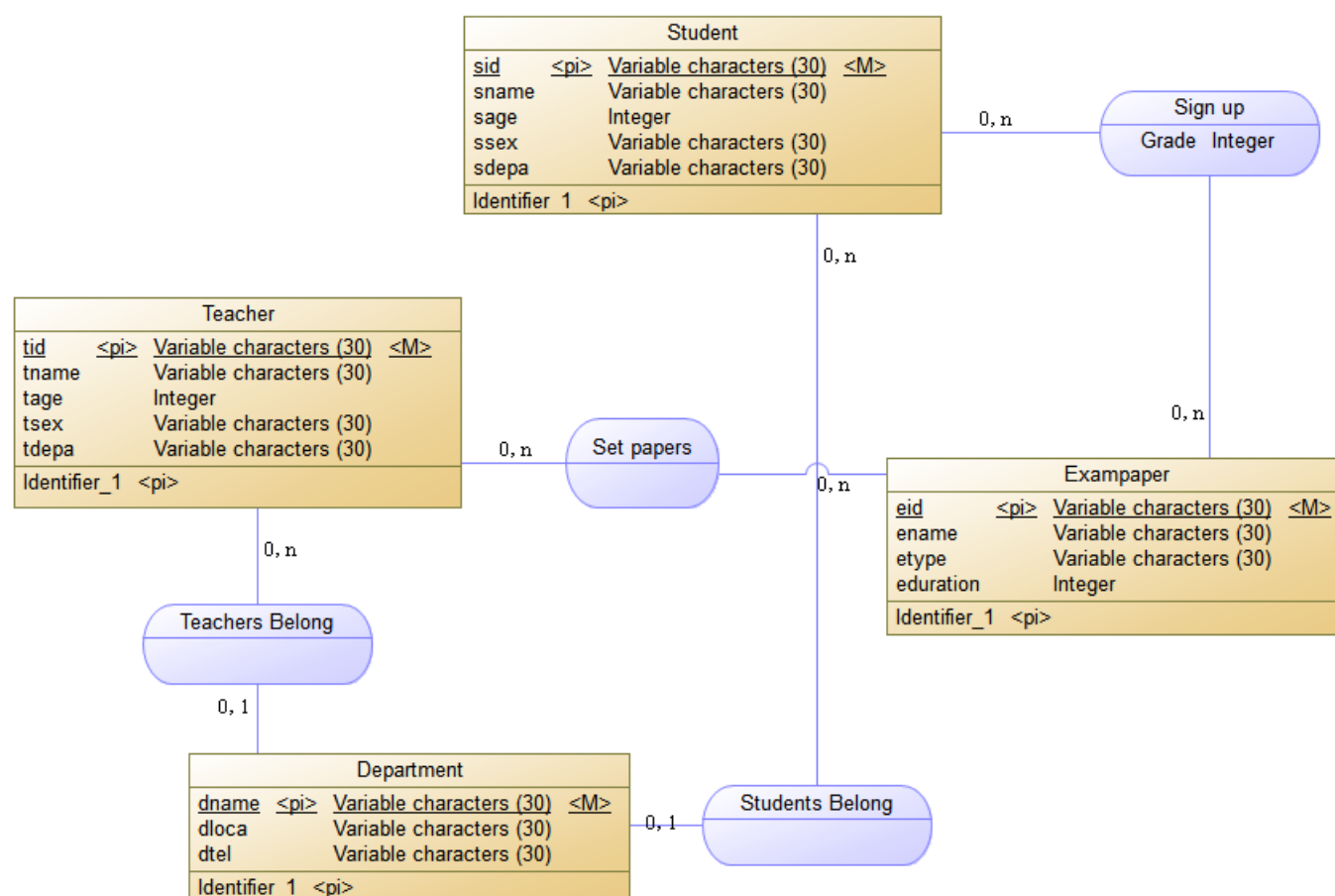
实体：【其中标下划线的为主键】

- Teacher 包含五个属性，分别为 tid , tname , tsex , tage , tdepa ；
- Student 包含五个属性，分别为 sid , sname , ssex , sage , sdepa ；
- Exampaper 包含四个属性，分别为 eid , ename , etype , eduration ；
- Department 包含三个属性，分别为 dname , dloca , dtel ；

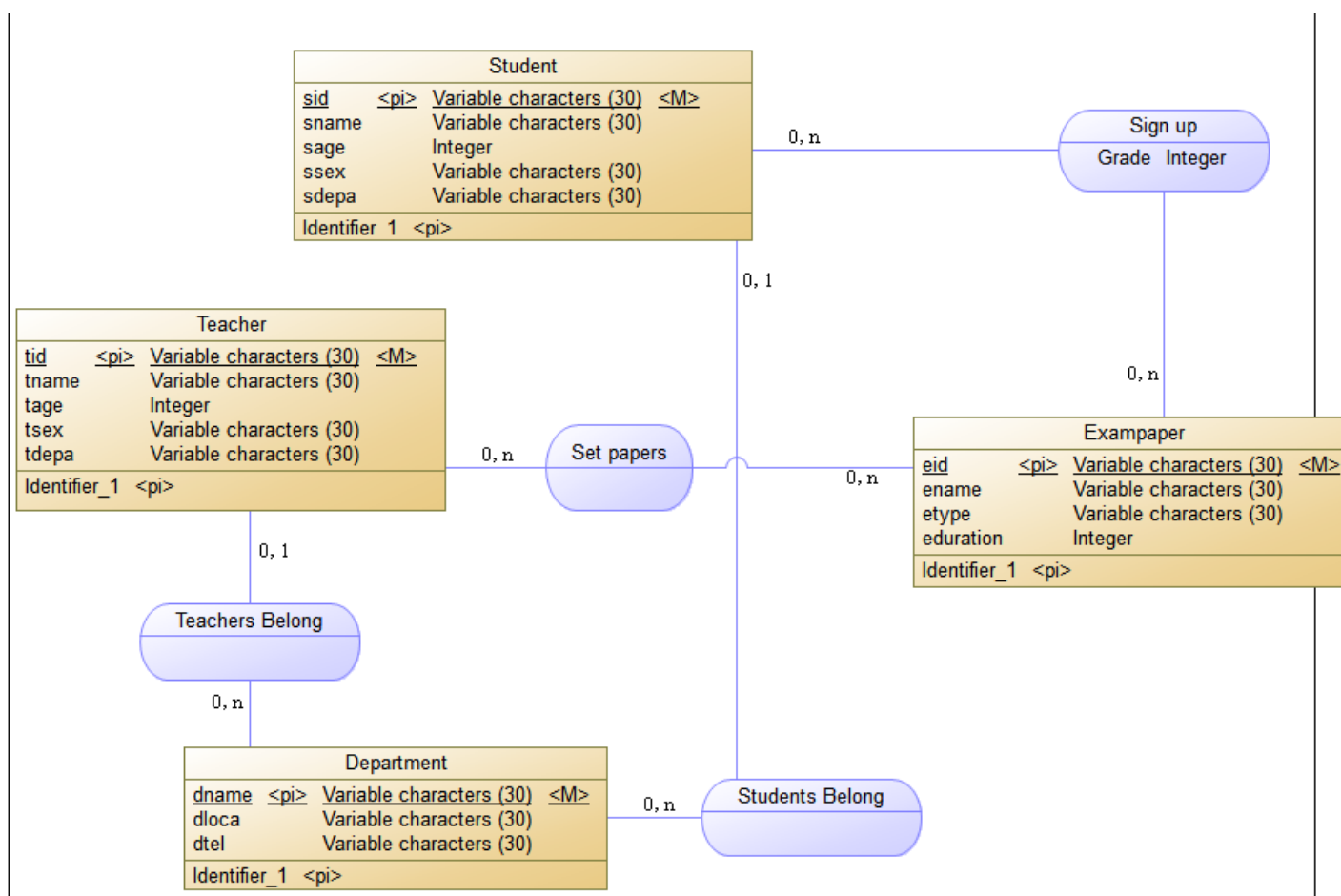
联系：

- Teacher 和 Exampaper 之间存在出卷关系 Set papers ；
- Student 和 Exampaper 之间存在报考关系 Sign up , 包含属性成绩 Grade ；
- Teacher 和 Department 之间是属于关系 Teachers Belong ；
- Student 和 Department 之间是属于关系 Srudents Belong ；

ER图如下：（ 主要问题在一对多关系，如果画图，结果如下 ）



如果要在powerdesigner中正确表示表结构，则需要将一对多关系的权值替换，图如下：



(3) 根据上述 E-R 图，分析表结构的属性特征（主键或外键关系）及实体之间的对应关系。

属性特征：

表结构分析过程如下：

- `Teacher (tid , tname , tsex , tage , tdepa);`
- `Student (sid , sname , ssex , sage , sdepa);`
- `Exampaper (eid , ename , etype , eduration);`
- `Department (dname , dloca , dtel);`
- `Set papers (tid,eid);`
 - 由于是多对多的关系，因此主键为 (tid,eid)；
 - tid 和 eid 分别为 Teacher 和 Exampaper 的外键
- `Sign up (sid,eid , Grade);`
 - 多对多的关系，因此主键为 (sid,eid)；
 - sid 和 eid 分别为 Student 和 Exampaper 的外键
- `Teachers Belong (tid , dname);`
 - 一对多的关系，选择多端作为主键;其中 dname 是 Department 表的外键；

- 由于此联系和 Teacher 同主键，因此进行 **合并**：
 - Teacher (tid , tname , tsex , tage , tdepa , dname);
 - dname 是 **外键**；
- Students Belong (sid , dname);
 - 一对多的关系，选择多端作为 **主键**；其中 dname 是 Department 表的 **外键**；
 - 由于此联系和 Student 同主键，因此进行 **合并**：
 - Student (sid , sname , ssex , sage , sdepa , dname);
 - dname 是 **外键**；

表结构如下:

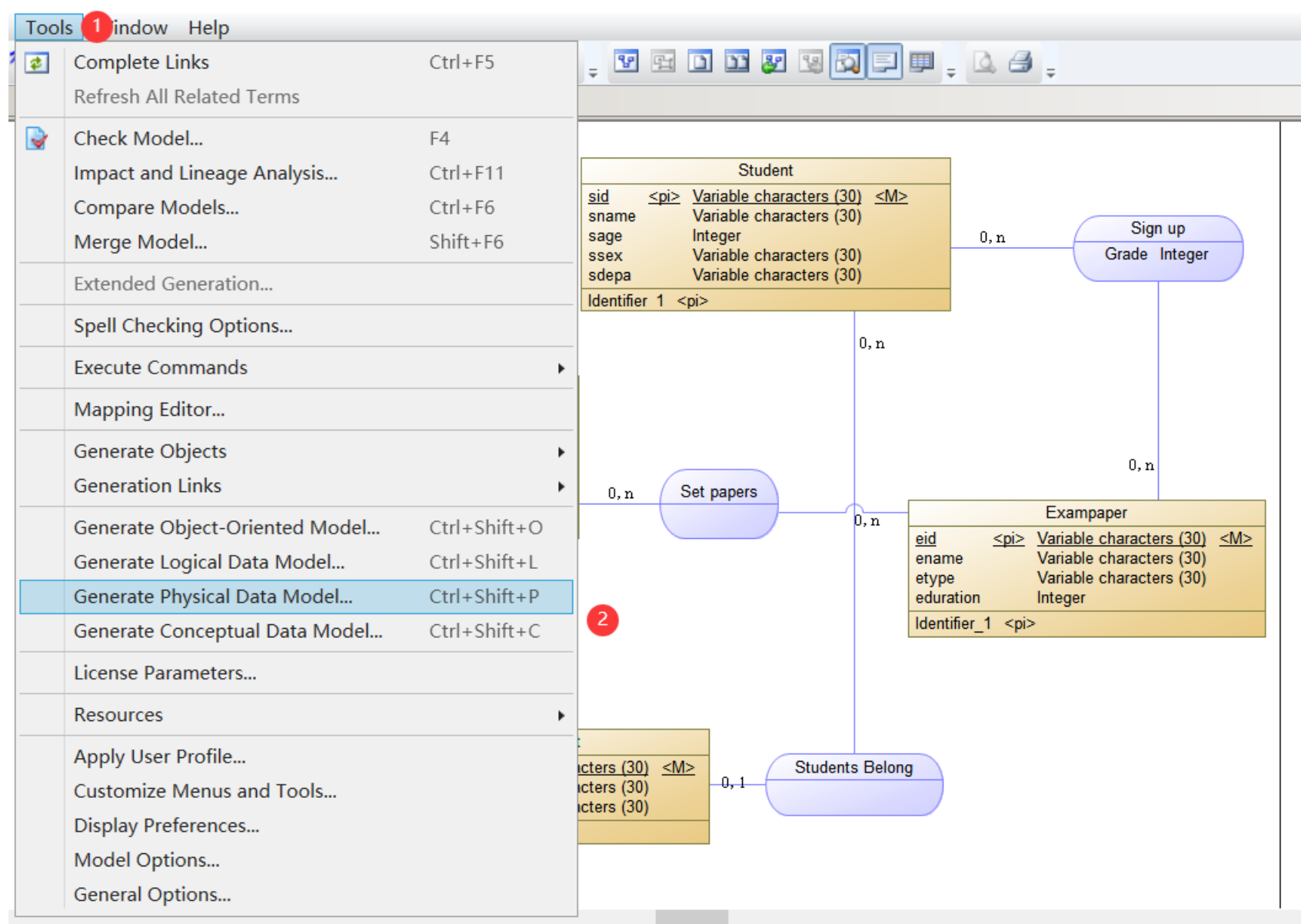
- Teacher (tid , tname , tsex , tage , tdepa , dname);
- Student (sid , sname , ssex , sage , sdepa , dname);
- Exampaper (eid , ename , etype , eduration);
- Department (dname , dloca , dtel);
- Set papers (tid,eid);
- Sign up (sid,eid , Grade);

实体之间的对应关系：

- ①根据“可以多名老师合作出 1 份试卷”，我们也可以知道“1个老师可以出多份试卷”，因此老师 Teacher 和试卷 Exampaper 之间为“多对多”的关系；
- ②根据“每位老师、同学只能隶属一个院系”，说明一个院系可以有多个老师、学生，但是一个老师和学生都只能属于一个院系，可知老师 Teacher 和院系 Department、学生 Student 和院系 Department 之间都是“一对多”的关系；
- ③每个学生可以选择报考多张试卷，每张试卷可以由多个学生报考，因此学生 Student 和试卷 Exampaper 之间为“多对多”的关系；

(4) 通过 PowerDesigner 建立数据库后台，根据概念设计模型生成物理属性模型，最后转化成 SQL 语句。

Step1.点击菜单中的"Tools"-->"Generate Physical Data Model";



Step2.选择 PostgreSQL 9.x 点击确定，生成如下图的物理模型；

PDM Generation Options

General Detail Target Models Selection

☒ Generate new Physical Data Model

DBMS: PostgreSQL 9.x

Name: lab10

Code: lab10

Configure Model Options...

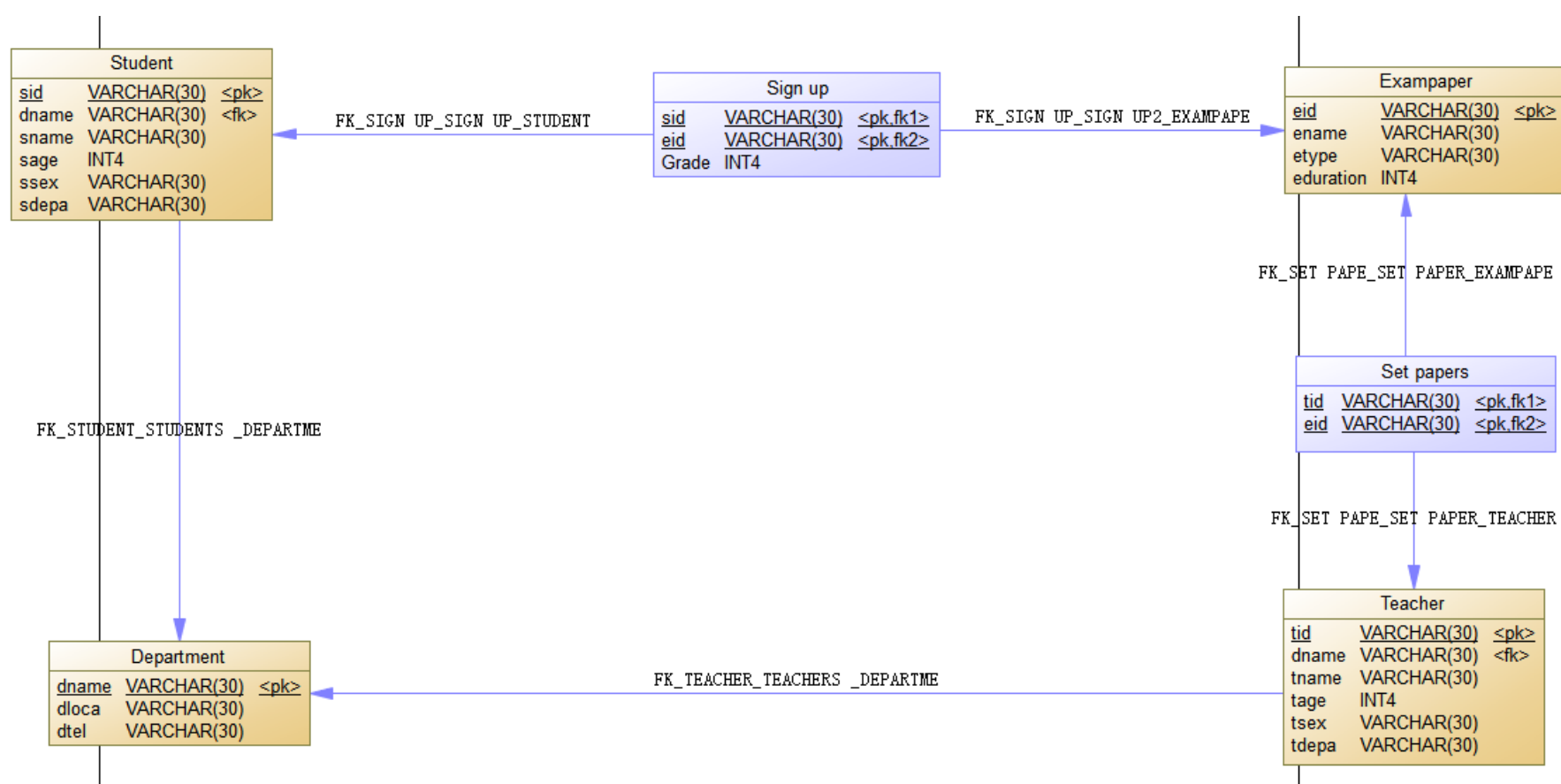
☐ Update existing Physical Data Model

Select model: lab9

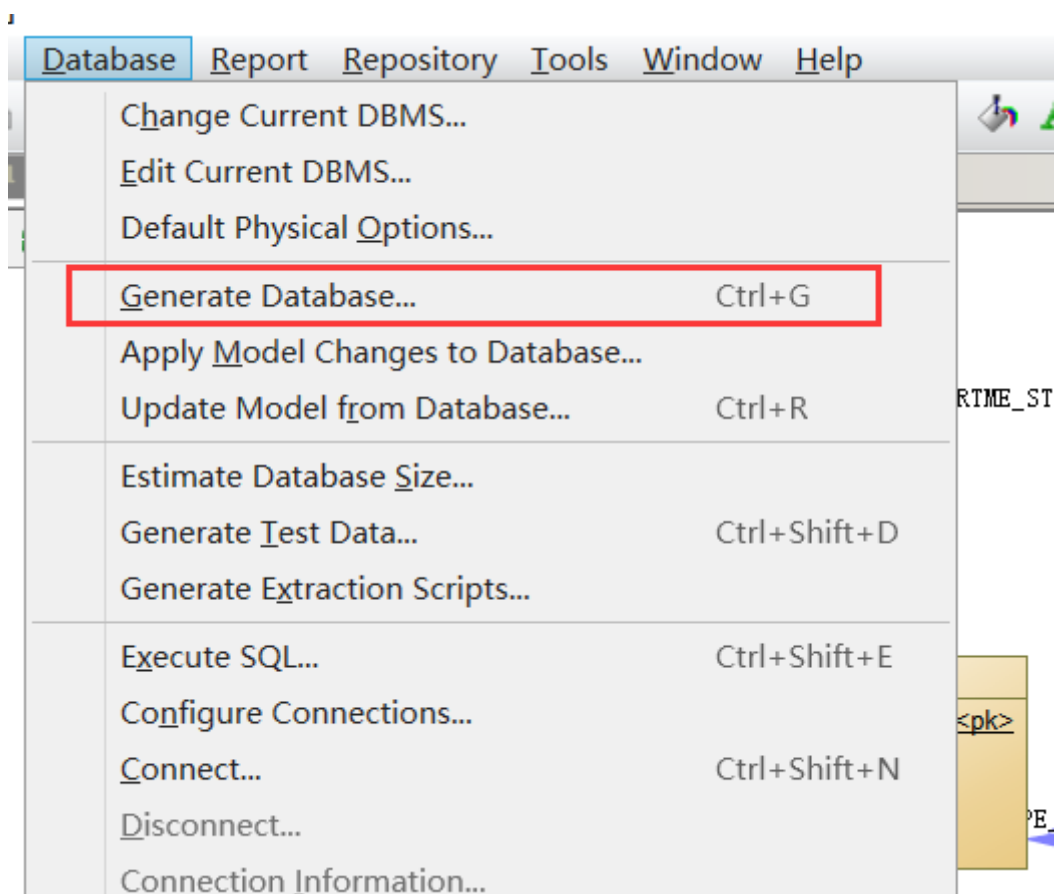
DBMS: SAP SQL Anywhere 17

☒ Preserve modifications

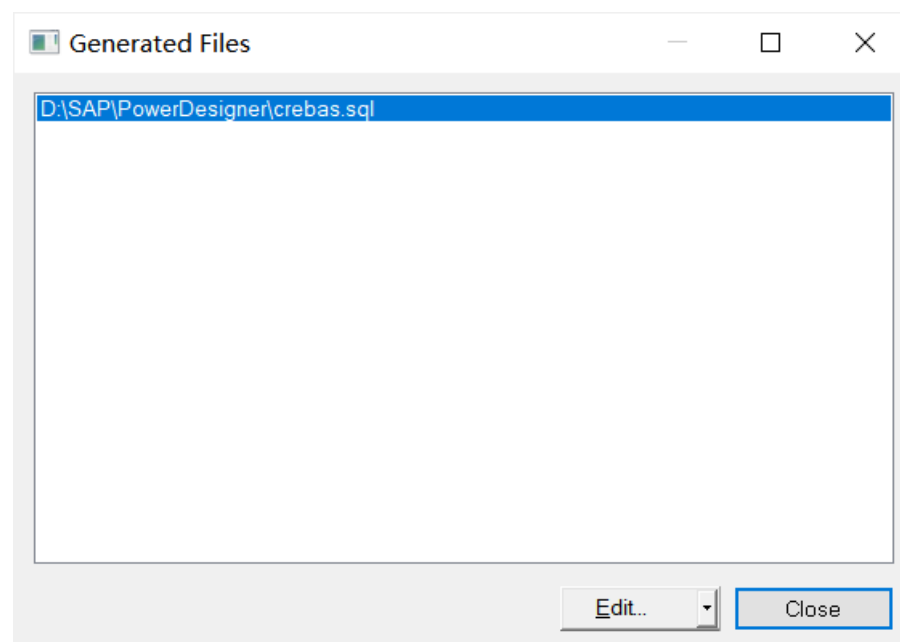
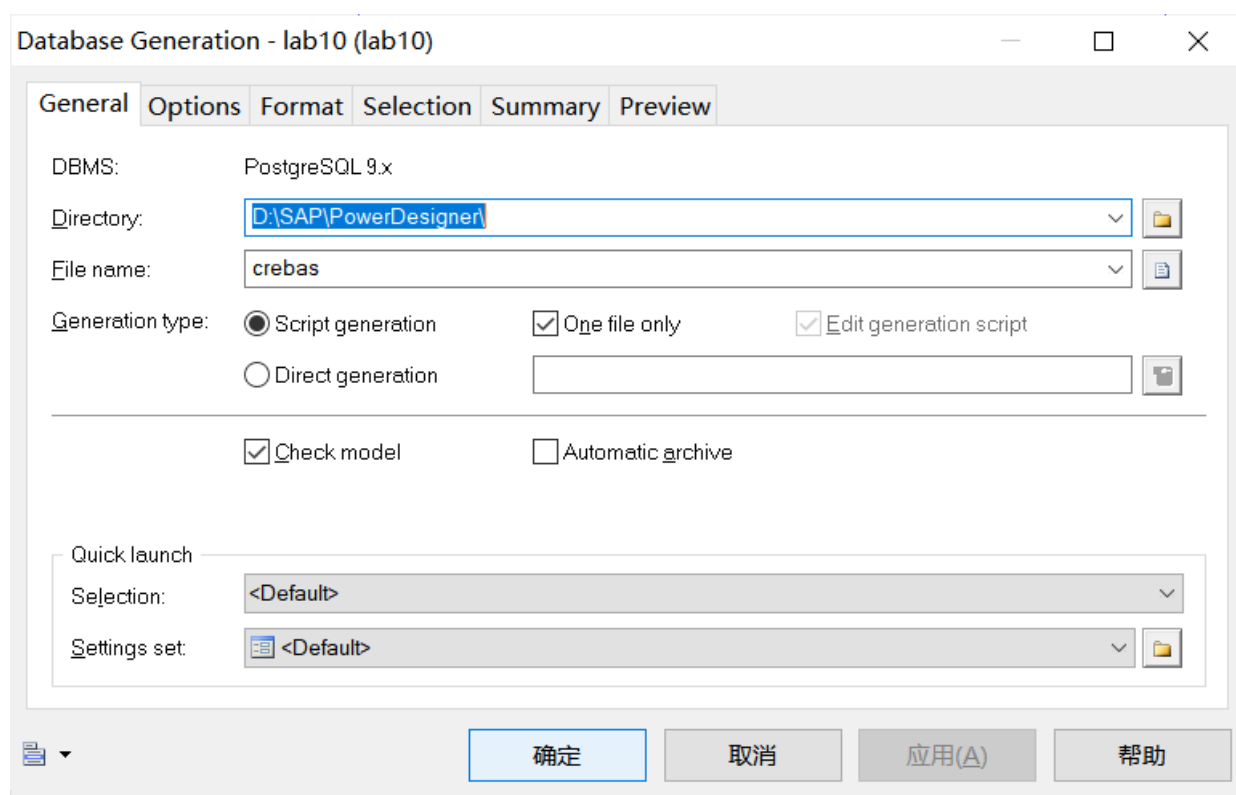
确定 取消 应用(A) 帮助



Step3.生成SQL语句；在生成的物理模型页面点击菜单中的"Database"-->"Generate Database..."；



点击确定；

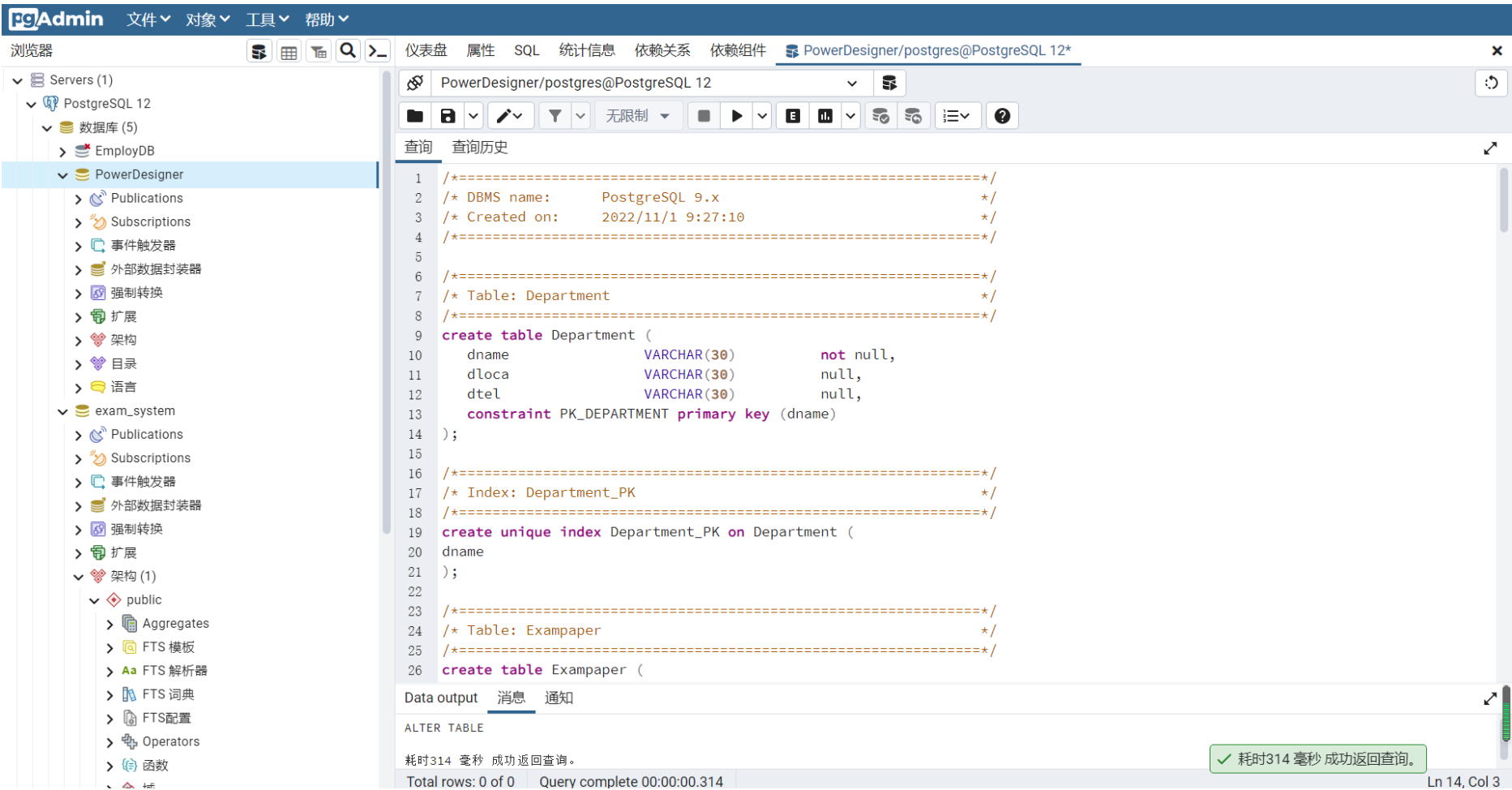


生成的代码如下：

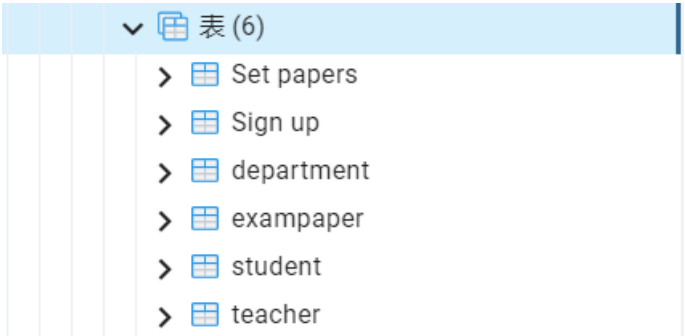
```
crebas.sql  X
D: > SAP > PowerDesigner > crebas.sql
1  /*=====*/
2  /* DBMS name:      PostgreSQL 9.x          */
3  /* Created on:     2022/11/1 9:27:10        */
4  /*=====*/
5
6
7  drop index Department_PK;
8
9  drop table Department;
10
11 drop index Exampaper_PK;
12
13 drop table Exampaper;
14
15 drop index "Set papers2_FK";
16
17 drop index "Set papers_FK";
18
19 drop index "Set papers_PK";
20
21 drop table "Set papers";
22
23 drop index "Sign up2_FK";
24
25 drop index "Sign up_FK";
26
27 drop index "Sign up_PK";
28
29 drop table "Sign up";
30
31 drop index "Students Belong_FK";
32
33 drop index Student_PK;
34
35
```

(5) 将生成的 SQL 语句导入到 Postgresql 中，查看相应表结构。

先创建一个数据库，名称为 PowerDesigner，将代码复制，去掉开始的 drop 语句，运行代码；



生成了六个表；



各个表的结构如下：

- teacher 表
 - 属性



- 约束
 - 主键

teacher

General列高级约束参数安全SQL

主键外键检查唯一排除

名称

列

pk_teacher

tid

■ 外键

teacher

General列高级约束参数安全SQL

主键外键检查唯一排除

名称

列

引用表

FK_TEACHER_TEACHERS_DEPARTM

(dname) -> (dname)

public.department

• student 表

◦ 属性

student

General列高级约束参数安全SQL

继承自表

选择要从其继承...

列

名称

数据类型

长度/精度

规模

不为 NUL...

主键?

默认值

sid

character varying

30

dname

character varying

30

sname

character varying

30

sage

integer

ssex

character varying

30

sdepa

character varying

30

◦ 约束

■ 主键

student

General列高级约束参数安全SQL

主键外键检查唯一排除

名称

列

pk_student

sid

■ 外键

student

General列高级约束参数安全SQL

主键外键检查唯一排除

	名称	列	引用表
<div></div>	FK_STUDENT_STUDENTS_DEPARTM	(dname) -> (dname)	public.department

- exampaper 表

 - 属性

exampaper

General列高级约束参数安全SQL

继承自表

选择要从其继承...

列

	名称	数据类型	长度/精度	规模	不为 NUL...	主键?	默认值
<div></div>	eid	character varying <div></div>	30		<div></div>	<div></div>	
<div></div>	ename	character varying <div></div>	30		<div></div>	<div></div>	
<div></div>	etype	character varying <div></div>	30		<div></div>	<div></div>	
<div></div>	eduration	integer <div></div>			<div></div>	<div></div>	

- 约束
 - 主键

exampaper

General列高级约束参数安全SQL

主键外键检查唯一排除

名称	列
pk_exampaper	eid

- 无外键

- department 表

 - 属性

department

General列高级约束参数安全SQL

继承自表

选择要从其继承...

列

	名称	数据类型	长度/精度	规模	不为 NUL...	主键?	默认值
<div></div>	dname	character varying <div></div>	30		<div></div>	<div></div>	
<div></div>	dloca	character varying <div></div>	30		<div></div>	<div></div>	
<div></div>	dtel	character varying <div></div>	30		<div></div>	<div></div>	

- 约束
 - 主键

- 约束
 - 主键

Set papers

General列高级约束参数安全SQL

主键外键检查唯一排除

+

名称

PK_SET PAPERS

列

tid,eid

- 外键

Set papers

General

列

高级

约束

参数

安全

SQL





主键

外键

检查

唯一

排除

		名称	列	引用表
		FK_SET PAPE_SET PAPER_EXAMPAP	(eid) -> (eid)	public.exampaper
		FK_SET PAPE_SET PAPER_TEACHER	(tid) -> (tid)	public.teacher

实验结果

j见实验思路部分

源程序

导出的 `sql` 完整代码如下:

```

/*=====*/
/* DBMS name:      PostgreSQL 9.x                               */
/* Created on:      2022/11/1 9:27:10                           */
/*=====*/

```

```
drop index Department_PK;
```

```
drop table Department;
```

```
drop index Exampaper_PK;
```

```
drop table Exampaper;
```

```
drop index "Set papers2_FK";
```

```
drop index "Set papers_FK";
```

```
drop index "Set_papers_PK";
```

```
drop table "Set papers";
```

```
drop index "Sign up2_FK";
```

```
drop index "Sign up_FK";
```

```
drop index "Sign up_PK";
```

```
drop table "Sign up";

drop index "Students Belong_FK";

drop index Student_PK;

drop table Student;

drop index "Teachers Belong_FK";

drop index Teacher_PK;

drop table Teacher;

/*=====*/
/* Table: Department */
/*=====*/
create table Department (
    dname          VARCHAR(30)          not null,
    dloca          VARCHAR(30)          null,
    dtel           VARCHAR(30)          null,
    constraint PK_DEPARTMENT primary key (dname)
);

/*=====*/
/* Index: Department_PK */
/*=====*/
create unique index Department_PK on Department (
dname
);

/*=====*/
/* Table: Exampaper */
/*=====*/
create table Exampaper (
    eid            VARCHAR(30)          not null,
    ename          VARCHAR(30)          null,
    etype          VARCHAR(30)          null,
    eduration      INT4                 null,
    constraint PK_EXAMPAPER primary key (eid)
);

/*=====*/
/* Index: Exampaper_PK */
/*=====*/
create unique index Exampaper_PK on Exampaper (
eid
);

/*=====*/
/* Table: "Set papers" */
/*=====*/
create table "Set papers" (
    tid            VARCHAR(30)          not null,
    eid            VARCHAR(30)          not null,
    constraint "PK_SET PAPERS" primary key (tid, eid)
);

/*=====*/
```

```

/* Index: "Set papers_PK" */
/*=====*/
create unique index "Set papers_PK" on "Set papers" (
tid,
eid
);

/*=====*/
/* Index: "Set papers_FK" */
/*=====*/
create index "Set papers_FK" on "Set papers" (
tid
);

/*=====*/
/* Index: "Set papers2_FK" */
/*=====*/
create index "Set papers2_FK" on "Set papers" (
eid
);

/*=====*/
/* Table: "Sign up" */
/*=====*/
create table "Sign up" (
    sid          VARCHAR(30)          not null,
    eid          VARCHAR(30)          not null,
    Grade        INT4                 null,
    constraint "PK_SIGN UP" primary key (sid, eid)
);

/*=====*/
/* Index: "Sign up_PK" */
/*=====*/
create unique index "Sign up_PK" on "Sign up" (
sid,
eid
);

/*=====*/
/* Index: "Sign up_FK" */
/*=====*/
create index "Sign up_FK" on "Sign up" (
sid
);

/*=====*/
/* Index: "Sign up2_FK" */
/*=====*/
create index "Sign up2_FK" on "Sign up" (
eid
);

/*=====*/
/* Table: Student */
/*=====*/
create table Student (
    sid          VARCHAR(30)          not null,
    dname        VARCHAR(30)          null,
    sname        VARCHAR(30)          null,

```

```

sage                INT4                null,
ssex                VARCHAR(30)          null,
sdepa               VARCHAR(30)          null,
constraint PK_STUDENT primary key (sid)
);

/*=====*/
/* Index: Student_PK                                */
/*=====*/
create unique index Student_PK on Student (
sid
);

/*=====*/
/* Index: "Students Belong_FK"                      */
/*=====*/
create index "Students Belong_FK" on Student (
dname
);

/*=====*/
/* Table: Teacher                                */
/*=====*/
create table Teacher (
tid                VARCHAR(30)          not null,
dname              VARCHAR(30)          null,
tname              VARCHAR(30)          null,
tage               INT4                 null,
tsex               VARCHAR(30)          null,
tdepa              VARCHAR(30)          null,
constraint PK_TEACHER primary key (tid)
);

/*=====*/
/* Index: Teacher_PK                                */
/*=====*/
create unique index Teacher_PK on Teacher (
tid
);

/*=====*/
/* Index: "Teachers Belong_FK"                      */
/*=====*/
create index "Teachers Belong_FK" on Teacher (
dname
);

alter table "Set papers"
add constraint "FK_SET PAPE_SET PAPER_TEACHER" foreign key (tid)
references Teacher (tid)
on delete restrict on update restrict;

alter table "Set papers"
add constraint "FK_SET PAPE_SET PAPER_EXAMPAPER" foreign key (eid)
references Exampaper (eid)
on delete restrict on update restrict;

alter table "Sign up"
add constraint "FK_SIGN UP_SIGN UP_STUDENT" foreign key (sid)
references Student (sid)

```

```
on delete restrict on update restrict;

alter table "Sign up"
add constraint "FK_SIGN UP_SIGN UP2_EXAMPAPE" foreign key (eid)
references Exampaper (eid)
on delete restrict on update restrict;

alter table Student
add constraint "FK_STUDENT_STUDENTS _DEPARTME" foreign key (dname)
references Department (dname)
on delete restrict on update restrict;

alter table Teacher
add constraint "FK_TEACHER_TEACHERS _DEPARTME" foreign key (dname)
references Department (dname)
on delete restrict on update restrict;
```

实验体会

PowerDesigner 的“一对多”关系的画图需要注意，一开始的时候按正常 ER 图画的，导致出现了 Department 合并 sid 和 tid 的情况，以后在使用过程中需要注意；