# 数据库系统开发实验

姓名：卜梦煜 学号：1120192419 班级：07111905

## 1.实验名称

选课管理

## 2.实验目的

通过本实验，熟悉并掌握使用ADO.NET开发Windows桌面数据库应用程序的原理、方法和流程。对于给定的应用环境与功能需求，能够开发出满足用户需求，操作简单，能有效存取数据，具有易用性、逻辑性、高效性、可拓展性特点的应用程序。

## 3.实验环境

SQL Server 2012

Power Designer 16.5

Visual Studio 2019

## 4.实验要求

设计数据库，基本信息包括学生、课程、教师，学生从已分配教师的课程列表中选择相应的课程（课程列表项信息包括：课程名称、学分、授课教师、授课时间等，可以手工录入已分配教师、上课时间的课程列表信息）。

选课规则：每个学生最少选择3门课程，学分高于8学分，最多可以选择5门课程，学分不超过12学分。

实现下列功能：

（1）录入学号，自动显示学生姓名。

（2）在可选择课程列表上方显示已选择的课程列表，可以从已选择的课程列表中删除已选择的课程。

（3）显示可选择的课程列表（不包含已选择的课程），可以按规则选择3-5门课程，选择的课程自动从可选课程列表中移至已选课程列表。

（4）显示当前学生选择的课程数量、总学分。

（5）可以临时保存学生选择的课程信息，如果符合选课规则（3-5门、学分：8-12学分），则可提交选课结果。

## 5.实验内容

### 5.1数据建模

#### 5.1.1建立实体

根据题目需求，共建立8个基本表，定义如下：

学生（学号，姓名，性别，电话，学院，年级，专业）

教师（职工号，姓名，性别，电话，学院，办公室，职务）

课程（课程号，课程名称，学分，先修课）

教室（教室号，教室类型，教室位置，教室容量）

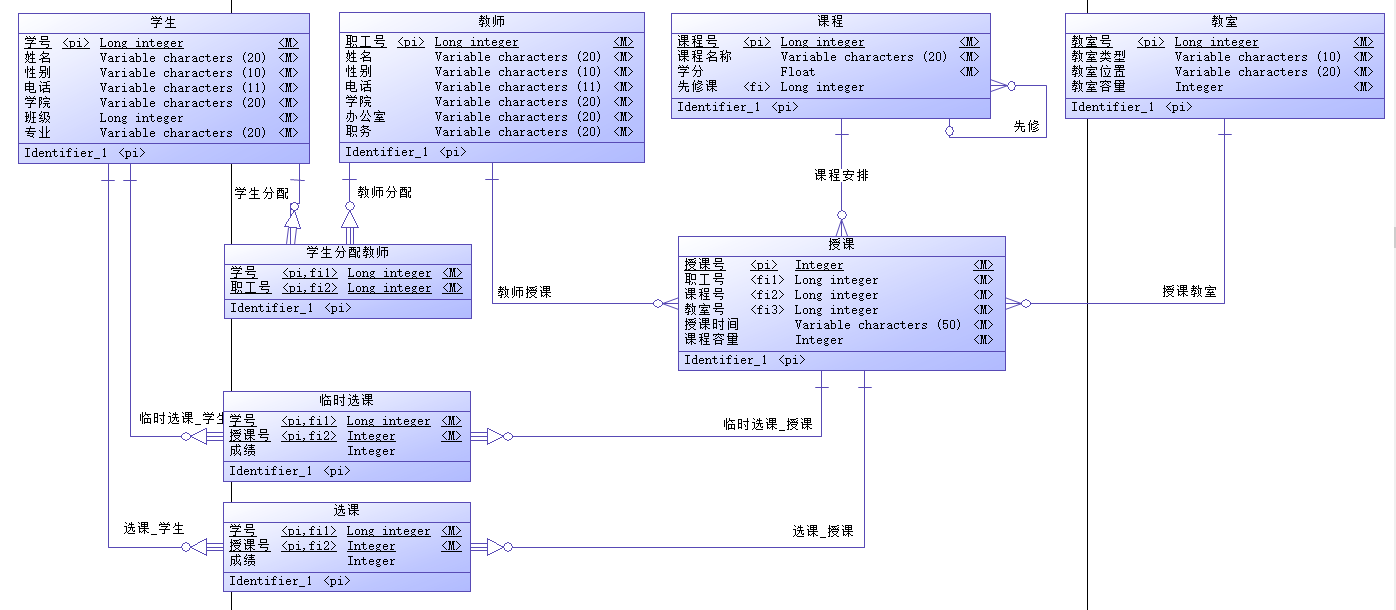
学生分配教师（学号，职工号）

授课（授课号，职工号，课程号，教室号，授课时间，课程容量）

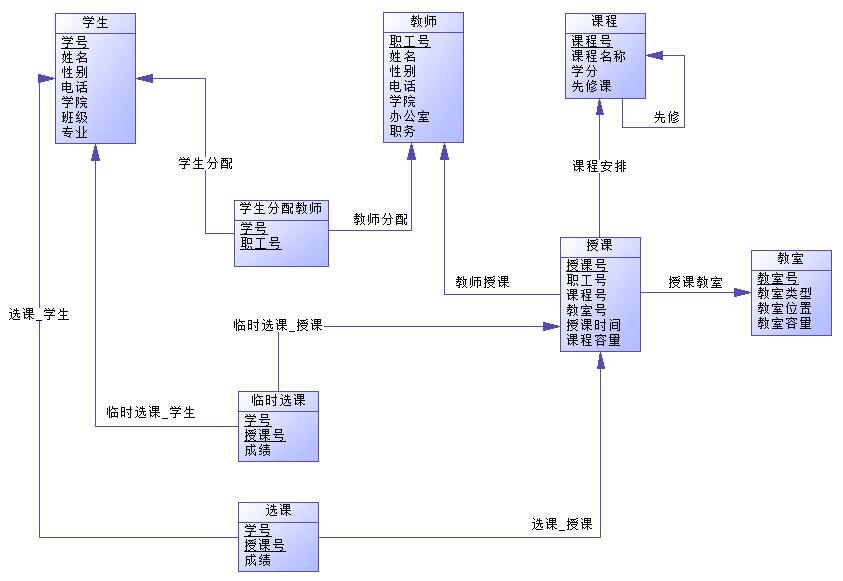
临时选课（学号，授课号，成绩）

选课（学号，授课号，成绩）

#### 5.1.2建立逻辑模型



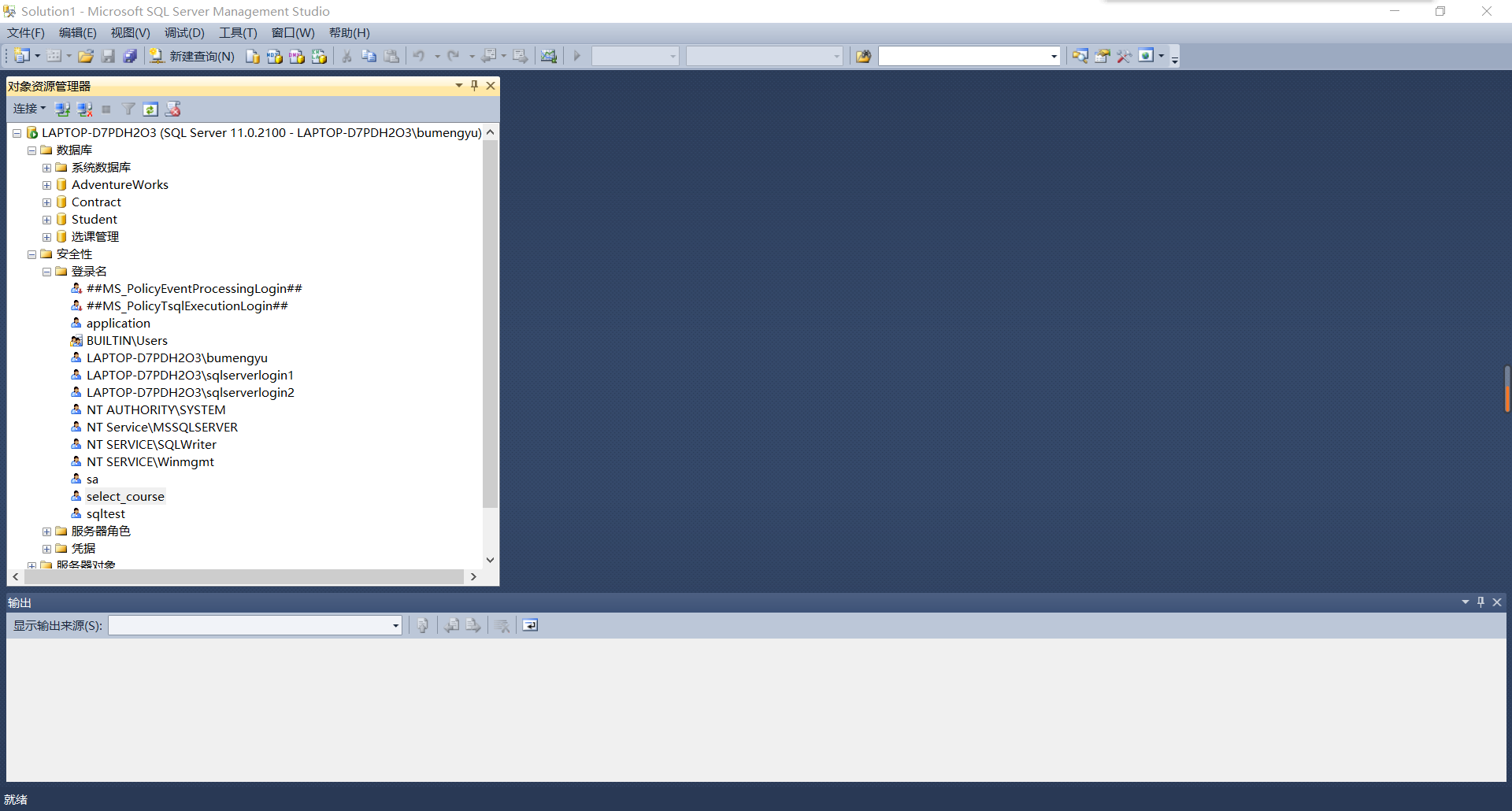
#### 5.1.3建立物理模型



### 5.2数据库配置

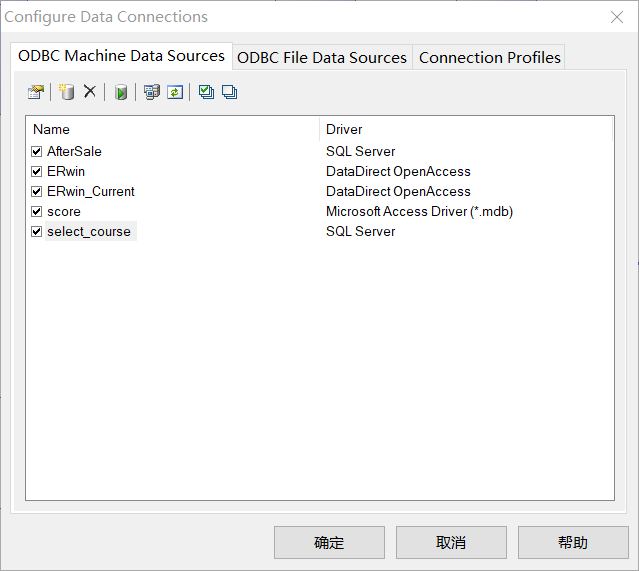
#### 5.2.1创建数据库

使用SQL Server创建数据库“选课管理”，并创建对应的登录名select\_course。

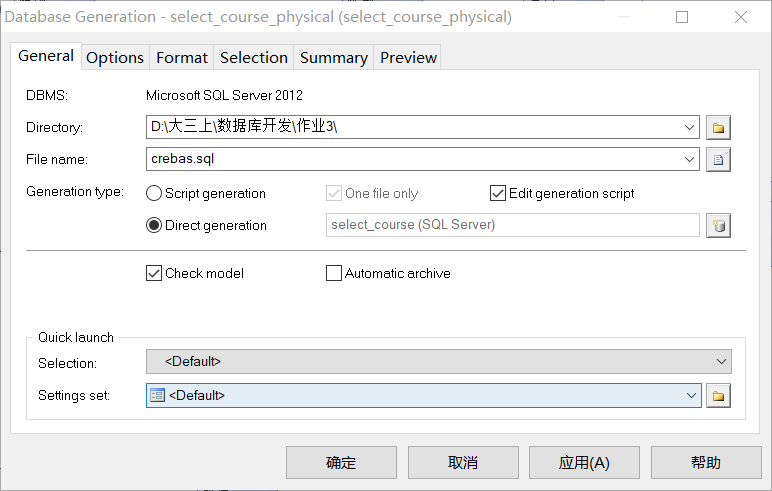


#### 5.2.2使用Power Designer生成数据库

创建数据源select\_course。



生成数据库。



#### 5.2.3 Visual Studio绑定数据库

创建数据连接。



### 5.3 设计窗体



### 5.4 功能实现

#### 5.4.1 录入学号，自动显示学生姓名

利用oleDbAdapter3数据适配器建立数据库查询，用查询结果生成dataSet3，再将dataset3中数据绑定到 “姓名textBox”中。当用户输入学生的学号时，会自动调用“学号TextBox\_TextChanged\_1”方法，取出textBox中数据，建立相应查询语句，执行相应查询操作，查询该学生姓名、查询该学生已选课程信息、查询该学生可选课程信息。当查询结果为空时姓名、已选学分、已选课程数、已选课程信息、可选课程信息均为空；当查到对应学生时会显示相应内容。

查询结果为空：



查到该学生：



#### 5.4.2 在可选择课程列表上方显示已选择的课程列表，可以从已选择的课程列表中删除已选择的课程

设置数据适配器oleDbAdapter1查询选课管理数据库中学生的可选课程，将查询结果生成到dataSet1，将dataSet1绑定到dataGridView1，从而实现对用户的数据可视化。dataGridView1中显示的绑定到数据源的属性有：授课号、课程名称、学分、授课教师、教室位置、授课时间，未绑定的属性有：是否退课。

对于已选课程中的记录，可点击“是否退课”列的“退课”按钮，将该条选课记录从数据库的选课表或临时选课表中删除，数据库中删除成功后在dataGridView1中删除该条选课记录。



#### 5.4.3 显示可选择的课程列表（不包含已选择的课程），可以按规则选择3-5门课程，选择的课程自动从可选课程列表中移至已选课程列表

对于可选课程，处理逻辑与已选课程类似。设置数据适配器oleDbAdapter2查询选课管理数据库中学生的选课记录，将查询结果生成到dataSet2，将dataSet2绑定到dataGridView2，从而实现对用户的数据可视化。dataGridView2中显示的绑定到数据源的属性有：授课号、课程名称、学分、授课教师、教室位置、授课时间，未绑定的属性有：是否选课。

对于已选课程中的记录，可点击“是否选课”列的“选课”按钮，将该条选课记录添加到数据库的临时选课表汇总，数据库中插入成功后在dataGridView1中添加该条选课记录，并提示选课成功。



选课已达到5门时，若继续选课，会提示“已选5门课，不能继续选课！”，并拒绝选课请求。



对于手工录入课程信息，考虑到手工录入出错的可能性更大，因此录入时错误检测更加严格，检测的错误包括：已选修过这门课、当前选修课程数超过5门、录入的课程在授课表没有相应记录。

错误1：不存在这门课：



错误2：已选满5门课：



错误3：已选修过这门课：



成功手动录入：



#### 5.4.4 显示当前学生选择的课程数量、总学分

dataGridView1中为已选课程的信息，因此可通过对dataGridView1的行数查询操作完成对已选课程数的统计，通过遍历每一行，取出学分相加可完成对已选学分数的统计。



在查询到相应学生、退课、选课、手动录入课程成功后，学分数和已选课程数会相应地更新，并显示到右上角的两个textBox中。

#### 5.4.5 可以临时保存学生选择的课程信息，如果符合选课规则（3-5门、学分：8-12学分），则可提交选课结果

每次退课、选课、手工录入课程信息均是立刻存到临时选课表中，相当于“自动保存”功能，每次显示已选课程显示的都是在临时选课表、选课表查询到的选课记录的交集。



点击提交按钮，会自动检测已选课程数是否是3-5门，已选学分数是否是8-12学分，当不满足条件时会给出详细的错误提示，如“当前选修课程数小于3且选修学分数小于5”等；当满足条件时，会将该同学在临时选课表中的所有临时选课记录存入选课表，并删除临时选课表中的选课信息，并显示提交成功。



## 6.回答问题

### 6.1 使用哪种数据提供程序？

使用OLEDB中的.NET Framework数据提供程序。

### 6.2 使用的数据连接对象是哪一个？连接对象是如何建立的？最后生成的连接对象中的连接字符串是什么？代表什么含义？

1）使用的数据连接对象是OleDbConnection。

2）连接对象是在完成数据适配器OldDbAdapter配置向导之后自动创建的。

3）连接字符串：“Provider=SQLNCLI11;Data Source=(local);Integrated Security=SSPI;Initial Catalog=选课管理”。

4）含义：使用SQLNCLI11作为访问接口，连接的数据库服务器名称为(local)，安全接口为Microsoft的安全支持提供器接口SSPI，初始目录为“选课管理”数据库。

### 6.3 使用的数据适配器对象是什么？其中的查询或更新语句是什么？如果有参数则参数是如何处理的？

1）使用的数据适配器是OleDbAdapter。

2）查询或更新语句是动态sql语句，每次通过OleDbAdapter.SelectCommand.CommandText重新设置，更新语句的修改与查询语句类似。

3）参数通过动态sql注入：利用OleDbAdapter.SelectCommand.Parameters传递参数。

### 6.4 使用的数据集对象是什么？数据集中有哪些数据表？数据表是由哪些适配器对象生成的？（或采用其它方法）。

1）采用的数据集对象是dataSet1、dataSet2、dataSet3，由相应的oleDbAdapter生成。

2）数据集中为查询结果，分别是已选课程、可选课程、学生姓名。

3）数据表有两种方式生成，一种是Adapter查询生成器生成，一种是通过修改OleDbAdapter.SelectCommand.CommandText生成。

## 7.实验总结

本次实验整体顺利，按照实验指导书中内容可以完成基本的适配器连接、创建数据集、数据绑定等的操作。在查阅相关资料后，我知道了如何修改adapter中的select语句、使用非图形化界面的形式手动建立command、adapter并完成相应任务。

实验中对一些可能出现的问题做了特判处理，以增强程序的健壮性，如学号错误、手工添加课程时的一系列可能的错误等。

本次实验中也遇到了很多问题，大多是缺乏工程设计经验、不熟悉C#、不熟悉.NET Framwork框架导致的，比如最初无法对错误学号做判断导致程序异常退出，后来在查询、插入、删除时也出现很多问题，但都一一克服了。

实验改进：进一步增加程序的健壮性，使得程序在非法输入情况下能够做出相应的错误提醒，而不是直接运行异常导致退出。

## 8.附录（生成数据库的sql脚本）

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

/\* DBMS name: Microsoft SQL Server 2012 \*/

/\* Created on: 2021/9/18 19:43:58 \*/

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

if exists (select 1

from sys.sysreferences r join sys.sysobjects o on (o.id = r.constid and o.type = 'F')

where r.fkeyid = object\_id('临时选课') and o.name = 'FK\_临时选课\_临时选课\_学生\_学生')

alter table 临时选课

drop constraint FK\_临时选课\_临时选课\_学生\_学生

go

if exists (select 1

from sys.sysreferences r join sys.sysobjects o on (o.id = r.constid and o.type = 'F')

where r.fkeyid = object\_id('临时选课') and o.name = 'FK\_临时选课\_临时选课\_授课\_授课')

alter table 临时选课

drop constraint FK\_临时选课\_临时选课\_授课\_授课

go

if exists (select 1

from sys.sysreferences r join sys.sysobjects o on (o.id = r.constid and o.type = 'F')

where r.fkeyid = object\_id('学生分配教师') and o.name = 'FK\_学生分配教师\_学生分配\_学生')

alter table 学生分配教师

drop constraint FK\_学生分配教师\_学生分配\_学生

go

if exists (select 1

from sys.sysreferences r join sys.sysobjects o on (o.id = r.constid and o.type = 'F')

where r.fkeyid = object\_id('学生分配教师') and o.name = 'FK\_学生分配教师\_教师分配\_教师')

alter table 学生分配教师

drop constraint FK\_学生分配教师\_教师分配\_教师

go

if exists (select 1

from sys.sysreferences r join sys.sysobjects o on (o.id = r.constid and o.type = 'F')

where r.fkeyid = object\_id('授课') and o.name = 'FK\_授课\_授课教室\_教室')

alter table 授课

drop constraint FK\_授课\_授课教室\_教室

go

if exists (select 1

from sys.sysreferences r join sys.sysobjects o on (o.id = r.constid and o.type = 'F')

where r.fkeyid = object\_id('授课') and o.name = 'FK\_授课\_教师授课\_教师')

alter table 授课

drop constraint FK\_授课\_教师授课\_教师

go

if exists (select 1

from sys.sysreferences r join sys.sysobjects o on (o.id = r.constid and o.type = 'F')

where r.fkeyid = object\_id('授课') and o.name = 'FK\_授课\_课程安排\_课程')

alter table 授课

drop constraint FK\_授课\_课程安排\_课程

go

if exists (select 1

from sys.sysreferences r join sys.sysobjects o on (o.id = r.constid and o.type = 'F')

where r.fkeyid = object\_id('课程') and o.name = 'FK\_课程\_先修\_课程')

alter table 课程

drop constraint FK\_课程\_先修\_课程

go

if exists (select 1

from sys.sysreferences r join sys.sysobjects o on (o.id = r.constid and o.type = 'F')

where r.fkeyid = object\_id('选课') and o.name = 'FK\_选课\_选课\_学生\_学生')

alter table 选课

drop constraint FK\_选课\_选课\_学生\_学生

go

if exists (select 1

from sys.sysreferences r join sys.sysobjects o on (o.id = r.constid and o.type = 'F')

where r.fkeyid = object\_id('选课') and o.name = 'FK\_选课\_选课\_授课\_授课')

alter table 选课

drop constraint FK\_选课\_选课\_授课\_授课

go

if exists (select 1

from sysindexes

where id = object\_id('临时选课')

and name = '临时选课\_授课\_FK'

and indid > 0

and indid < 255)

drop index 临时选课.临时选课\_授课\_FK

go

if exists (select 1

from sysindexes

where id = object\_id('临时选课')

and name = '临时选课\_学生\_FK'

and indid > 0

and indid < 255)

drop index 临时选课.临时选课\_学生\_FK

go

if exists (select 1

from sysobjects

where id = object\_id('临时选课')

and type = 'U')

drop table 临时选课

go

if exists (select 1

from sysobjects

where id = object\_id('学生')

and type = 'U')

drop table 学生

go

if exists (select 1

from sysindexes

where id = object\_id('学生分配教师')

and name = '教师分配\_FK'

and indid > 0

and indid < 255)

drop index 学生分配教师.教师分配\_FK

go

if exists (select 1

from sysindexes

where id = object\_id('学生分配教师')

and name = '学生分配\_FK'

and indid > 0

and indid < 255)

drop index 学生分配教师.学生分配\_FK

go

if exists (select 1

from sysobjects

where id = object\_id('学生分配教师')

and type = 'U')

drop table 学生分配教师

go

if exists (select 1

from sysindexes

where id = object\_id('授课')

and name = '授课教室\_FK'

and indid > 0

and indid < 255)

drop index 授课.授课教室\_FK

go

if exists (select 1

from sysindexes

where id = object\_id('授课')

and name = '课程安排\_FK'

and indid > 0

and indid < 255)

drop index 授课.课程安排\_FK

go

if exists (select 1

from sysindexes

where id = object\_id('授课')

and name = '教师授课\_FK'

and indid > 0

and indid < 255)

drop index 授课.教师授课\_FK

go

if exists (select 1

from sysobjects

where id = object\_id('授课')

and type = 'U')

drop table 授课

go

if exists (select 1

from sysobjects

where id = object\_id('教室')

and type = 'U')

drop table 教室

go

if exists (select 1

from sysobjects

where id = object\_id('教师')

and type = 'U')

drop table 教师

go

if exists (select 1

from sysindexes

where id = object\_id('课程')

and name = '先修\_FK'

and indid > 0

and indid < 255)

drop index 课程.先修\_FK

go

if exists (select 1

from sysobjects

where id = object\_id('课程')

and type = 'U')

drop table 课程

go

if exists (select 1

from sysindexes

where id = object\_id('选课')

and name = '选课\_授课\_FK'

and indid > 0

and indid < 255)

drop index 选课.选课\_授课\_FK

go

if exists (select 1

from sysindexes

where id = object\_id('选课')

and name = '选课\_学生\_FK'

and indid > 0

and indid < 255)

drop index 选课.选课\_学生\_FK

go

if exists (select 1

from sysobjects

where id = object\_id('选课')

and type = 'U')

drop table 选课

go

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

/\* Table: 临时选课 \*/

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

create table 临时选课 (

学号 bigint not null,

授课号 int not null,

成绩 int null,

constraint PK\_临时选课 primary key nonclustered (学号, 授课号)

)

go

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

/\* Index: 临时选课\_学生\_FK \*/

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

create index 临时选课\_学生\_FK on 临时选课 (

学号 ASC

)

go

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

/\* Index: 临时选课\_授课\_FK \*/

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

create index 临时选课\_授课\_FK on 临时选课 (

授课号 ASC

)

go

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

/\* Table: 学生 \*/

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

create table 学生 (

学号 bigint not null,

姓名 varchar(20) not null,

性别 varchar(10) not null,

电话 varchar(11) not null,

学院 varchar(20) not null,

班级 bigint not null,

专业 varchar(20) not null,

constraint PK\_学生 primary key nonclustered (学号)

)

go

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

/\* Table: 学生分配教师 \*/

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

create table 学生分配教师 (

学号 bigint not null,

职工号 bigint not null,

constraint PK\_学生分配教师 primary key nonclustered (学号, 职工号)

)

go

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

/\* Index: 学生分配\_FK \*/

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

create index 学生分配\_FK on 学生分配教师 (

学号 ASC

)

go

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

/\* Index: 教师分配\_FK \*/

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

create index 教师分配\_FK on 学生分配教师 (

职工号 ASC

)

go

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

/\* Table: 授课 \*/

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

create table 授课 (

授课号 int not null,

职工号 bigint not null,

课程号 bigint not null,

教室号 bigint not null,

授课时间 varchar(50) not null,

课程容量 int not null,

constraint PK\_授课 primary key nonclustered (授课号)

)

go

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

/\* Index: 教师授课\_FK \*/

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

create index 教师授课\_FK on 授课 (

职工号 ASC

)

go

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

/\* Index: 课程安排\_FK \*/

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

create index 课程安排\_FK on 授课 (

课程号 ASC

)

go

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

/\* Index: 授课教室\_FK \*/

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

create index 授课教室\_FK on 授课 (

教室号 ASC

)

go

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

/\* Table: 教室 \*/

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

create table 教室 (

教室号 bigint not null,

教室类型 varchar(10) not null,

教室位置 varchar(20) not null,

教室容量 int not null,

constraint PK\_教室 primary key nonclustered (教室号)

)

go

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

/\* Table: 教师 \*/

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

create table 教师 (

职工号 bigint not null,

姓名 varchar(20) not null,

性别 varchar(10) not null,

电话 varchar(11) not null,

学院 varchar(20) not null,

办公室 varchar(20) not null,

职务 varchar(20) not null,

constraint PK\_教师 primary key nonclustered (职工号)

)

go

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

/\* Table: 课程 \*/

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

create table 课程 (

课程号 bigint not null,

课程名称 varchar(20) not null,

学分 float not null,

先修课 bigint null,

constraint PK\_课程 primary key nonclustered (课程号)

)

go

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

/\* Index: 先修\_FK \*/

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

create index 先修\_FK on 课程 (

先修课 ASC

)

go

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

/\* Table: 选课 \*/

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

create table 选课 (

学号 bigint not null,

授课号 int not null,

成绩 int null,

constraint PK\_选课 primary key nonclustered (学号, 授课号)

)

go

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

/\* Index: 选课\_学生\_FK \*/

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

create index 选课\_学生\_FK on 选课 (

学号 ASC

)

go

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

/\* Index: 选课\_授课\_FK \*/

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

create index 选课\_授课\_FK on 选课 (

授课号 ASC

)

go

alter table 临时选课

add constraint FK\_临时选课\_临时选课\_学生\_学生 foreign key (学号)

references 学生 (学号)

go

alter table 临时选课

add constraint FK\_临时选课\_临时选课\_授课\_授课 foreign key (授课号)

references 授课 (授课号)

go

alter table 学生分配教师

add constraint FK\_学生分配教师\_学生分配\_学生 foreign key (学号)

references 学生 (学号)

go

alter table 学生分配教师

add constraint FK\_学生分配教师\_教师分配\_教师 foreign key (职工号)

references 教师 (职工号)

go

alter table 授课

add constraint FK\_授课\_授课教室\_教室 foreign key (教室号)

references 教室 (教室号)

go

alter table 授课

add constraint FK\_授课\_教师授课\_教师 foreign key (职工号)

references 教师 (职工号)

go

alter table 授课

add constraint FK\_授课\_课程安排\_课程 foreign key (课程号)

references 课程 (课程号)

go

alter table 课程

add constraint FK\_课程\_先修\_课程 foreign key (先修课)

references 课程 (课程号)

go

alter table 选课

add constraint FK\_选课\_选课\_学生\_学生 foreign key (学号)

references 学生 (学号)

go

alter table 选课

add constraint FK\_选课\_选课\_授课\_授课 foreign key (授课号)

references 授课 (授课号)

go