《数据库系统》课程设计

工厂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](#_Toc515435287)

[（一）需求概述 1](#_Toc515435288)

[（二）业务流分析](#_Toc515435289) **[错误！未定义书签。](#_Toc515435289)**

[（三）数据流分析 2](#_Toc515435290)

[（四）数据字典 3](#_Toc515435291)

[二、数据库概念结构设计 5](#_Toc515435292)

[（一）实体分析 5](#_Toc515435293)

[（二）属性分析 5](#_Toc515435294)

[（三）联系分析 7](#_Toc515435295)

[（四）概念模型分析（.PDM图） 7](#_Toc515435296)

[三、数据库逻辑结构设计 8](#_Toc515435297)

[（一）概念模型转化为逻辑模型 8](#_Toc515435298)

[1.一对一关系的转化 8](#_Toc515435299)

[2.一对多关系的转化 8](#_Toc515435300)

[3.多对多关系的转化 8](#_Toc515435301)

[（二）逻辑模型设计（.PDM图） 9](#_Toc515435302)

[四、数据库物理实现 9](#_Toc515435303)

[（一）表设计 9](#_Toc515435304)

[（二）创建表和完整性约束代码设计 10](#_Toc515435305)

[（三）创建视图、索引、存储过程和触发器 12](#_Toc515435306)

[五、数据库功能调试 17](#_Toc515435307)

[（一）职工管理模块 17](#_Toc515435308)

[（二）工程负责人管理模块 18](#_Toc515435309)

[（三）系统管理员管理模块 19](#_Toc515435310)

[六、设计系统前台软件 31](#_Toc515435311)

[（一）开发软件选择 31](#_Toc515435312)

[（二）软件功能要求与设计 31](#_Toc515435313)

[（三）软件功能实现 31](#_Toc515435314)

[（四）系统测试](#_Toc515435315) **[错误！未定义书签。](#_Toc515435315)**

[七、设计总结 35](#_Toc515435316)

# 一、系统需求分析

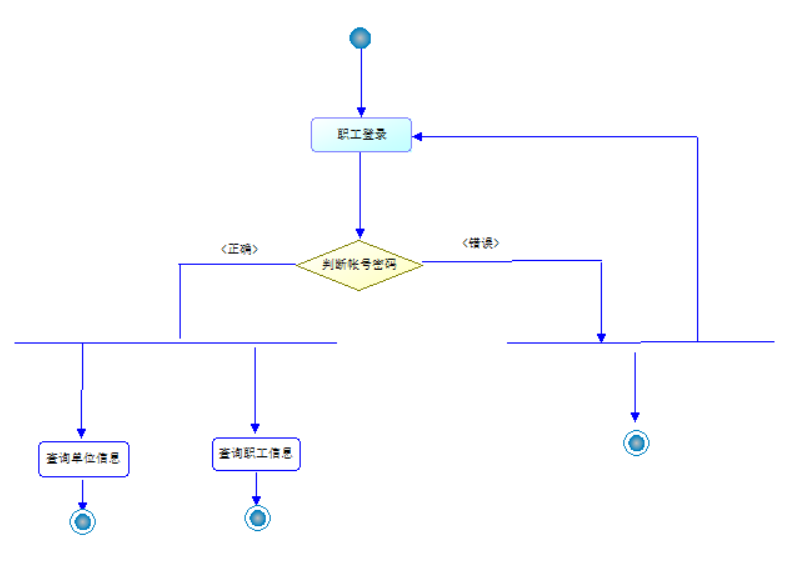
（一）需求概述

本次课程设计的题目是工厂管理系统，系统内容要求是：根据工厂管理系统建立一个数据库存储信息，数据库表名包含工厂，车间，工人，产品，零件，仓库。其中工厂包括多个车间和多个仓库，车间有多个工人 一个车间可以生产多个产品和多种零件，一个零件也可能被多个车间制造，一个产品由多个零件组成，一个零件也可以装配出多个产品，产品和零件均存入仓库中等信息。

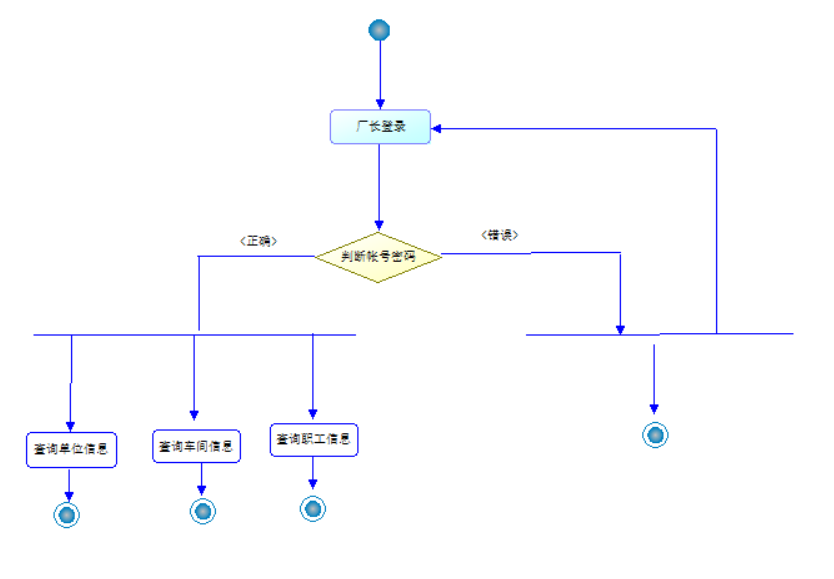
某工厂有多个下属车间，每一车间有多个职工，一个职工仅隶属于一个车间；且一个职工仅在一个车间中工作，但一个工程中有很多职工参加工作；有多个供应商为各个工程供应不同设备。

针对各工厂信息管理情况，设计出适合它们的计算机辅助工厂数据库管理系统。实现对工厂产品和职工基本信息进行计算机辅助管理，完成有关信息的记录、修改、查询、删除等一系列任务的计算机管理与实现。

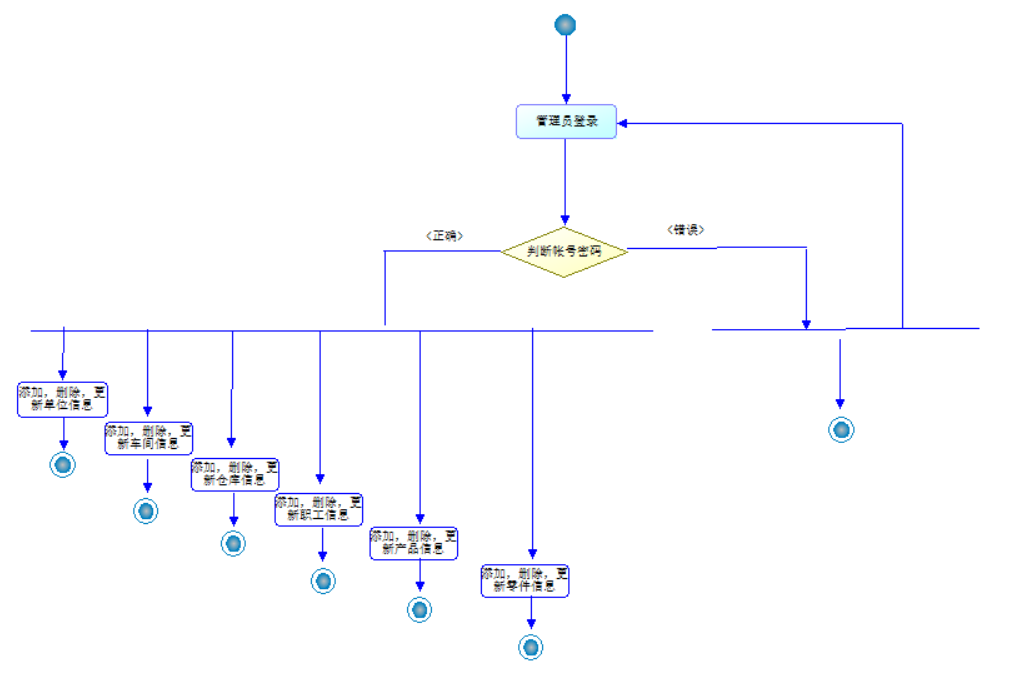
从职工角度：



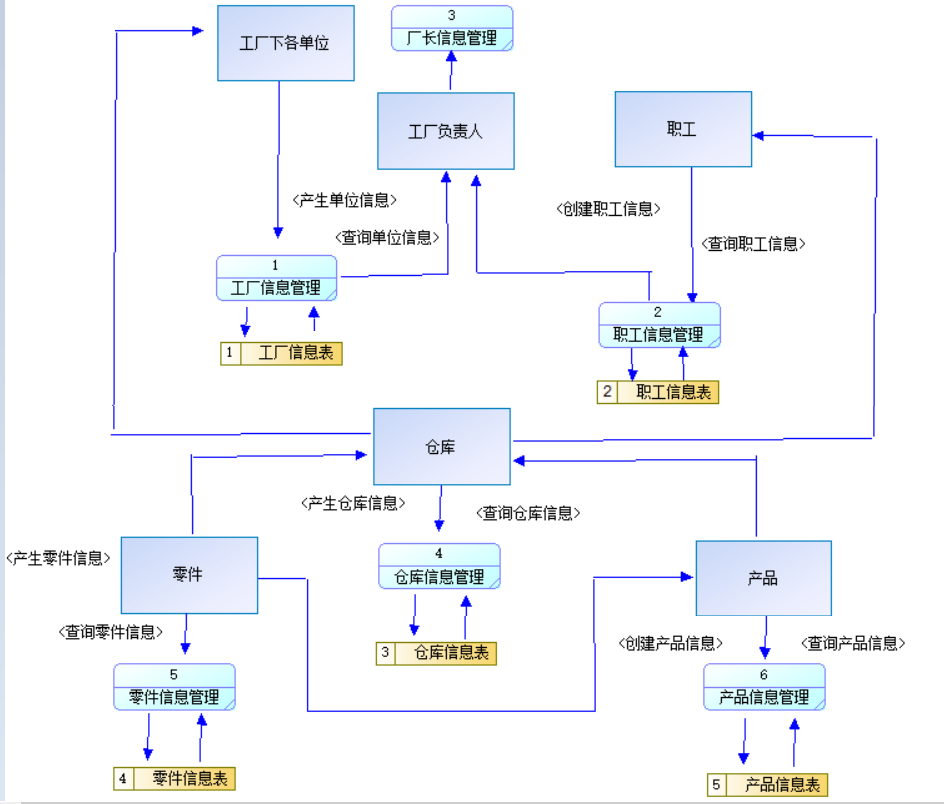
从工厂分责人角度：



从管理员角度：



（二）数据流分析



（三）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 |
| **工厂** | 区别单位 | char | 4 | 0001-9999 | 一个工厂对应一个厂名 |
| 厂名 | 标注单位 | vchar | 20 |  | 一个厂名对应一个厂长名 |
| 厂长姓名 | 标注单位 | Char | 11 |  | 一个厂长名对应一个工厂 |
| **车间** | 区别单位 | Char | 4 | 0001-9999 | 一个职工号对应一个职工名 |
| 车间号 | 标注单位 | vchar | 20 |  | 一个职工名对应一个职工号 |
| 车间主任 | 标注单位 | char | 2 | “男”或“女” |  |
| 地址 | 区别单位 | char | 2 |  |  |
| **工人** |  | Int | 6 | 1-999999 |  |
| 职工号 | 区别单位 | Char | 4 | 0001-9999 | 一个工人对应一个职工号 |
| 姓名 | 标注单位 | vchar | 20 |  | 一个职工号对应一个姓名 |
| 年龄 | 区别单位 | vchar | 20 |  | 一个姓名对应一个年龄 |
| 性别 | 区别单位 | char | 4 | 0001-9999 | 一个姓名对应一个性别 |
| 工种 | 区别单位 | vchar | 20 |  | 一个工人对应一个工种 |
| **仓库** | 区别单位 | Int | 4 | 1-9999 | 一个仓库对应一个仓库号 |
| 仓库号 | 标注单位 | char | 4 | 0001-9999 | 一个仓库号对应一个仓库 |
| **零件** | 区别单位 | vchar | 20 |  | 一个零件对应一个零件号 |
| 零件号 | 标注单位 | char | 11 |  | 一个零件号对应一个零件 |
| **产品** | 区别单位 | vchar | 20 |  | 一个产品对应一个产品号 |
| 产品号 | 标注单位 | int | 4 |  | 一个产品号对应一个产品 |

数据结构：

|  |  |  |
| --- | --- | --- |
| 数据结构名 | 含义名 | 组成 |
| 工厂 | 工厂信息 | 工厂编号、工厂名称、工厂电话、厂长 |
| 车间 | 车间信息 | 车间编号、车间主任 |
| 工人 | 工人信息 | 工人编号、工人名称、年龄、性别、工种 |
| 仓库 | 仓库信息 | 仓库编号、仓库主任、电话 |
| 零件 | 零件信息 | 零件编号、重量、价格 |
| 产品 | 产品信息 | 产品编号、价格 |

数据流：

|  |  |  |  |
| --- | --- | --- | --- |
| 数据流名 | 数据流来源 | 数据流去向 | 组成 |
| 工厂信息管理 | 工厂下属各单位 | 工厂负责人 | 工厂信息 |
| 车间信息管理 | 车间下属各单位 | 车间负责人 | 车间信息 |
| 工人信息管理 | 职工提供 | 登记表 | 职工信息 |
| 仓库信息管理 | 仓库单位 | 仓库负责人 | 仓库信息 |
| 零件信息管理 | 零件负责人 | 零件信息表 | 零件信息 |
| 产品信息管理 | 产品负责人 | 产品登记表 | 产品信息 |

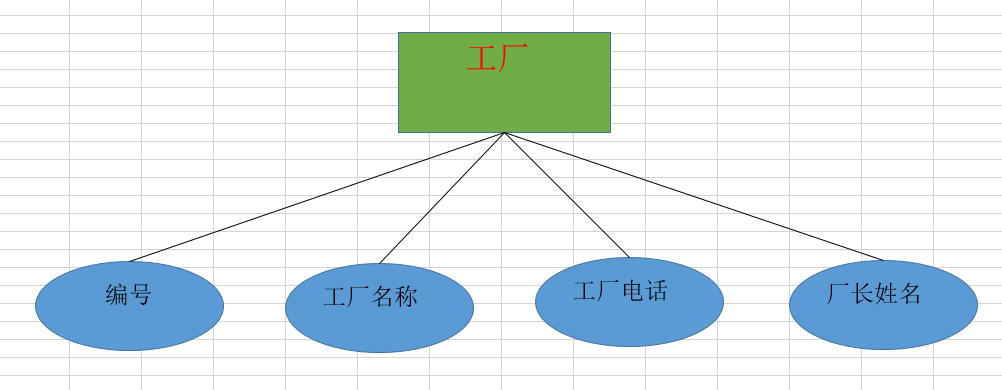
# 二、数据库概念结构设计

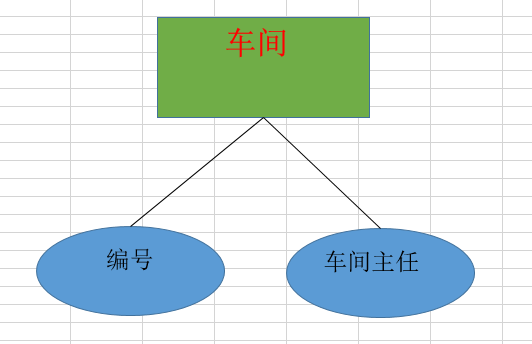
（一）实体分析

经需求分析，本次课程设计中包含六个实体，他们分别是：工厂实体、车间实体、职工实体、仓库实体、零件实体、产品实体。

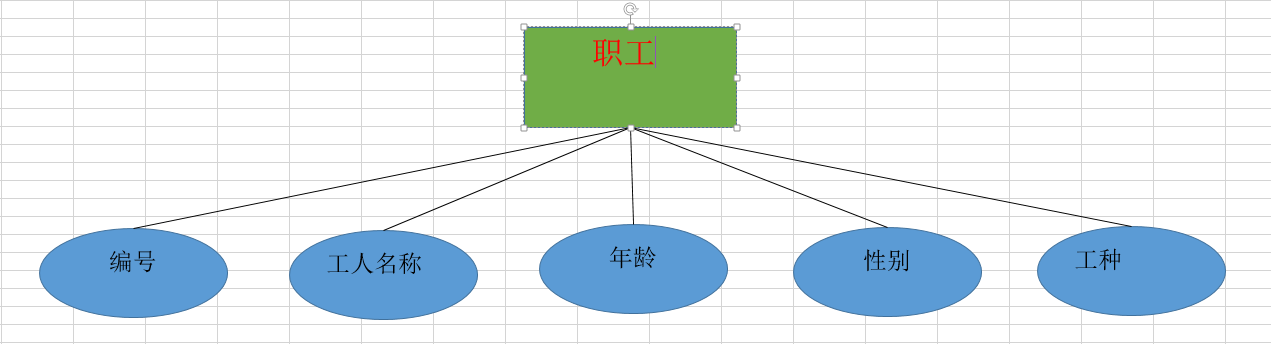
（二）属性分析

工厂实体属性：工厂编号、工厂名称、电话、厂长。如下：

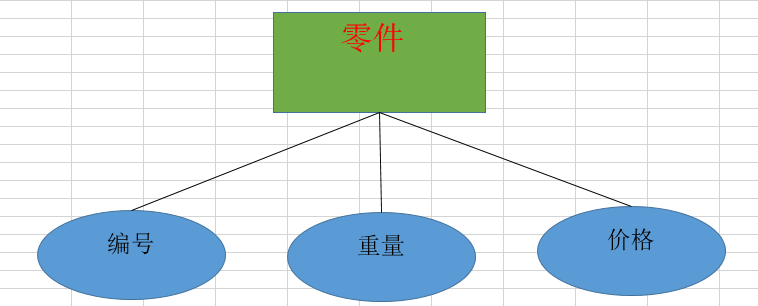
车间实体属性：车间编号，车间主任。如下：



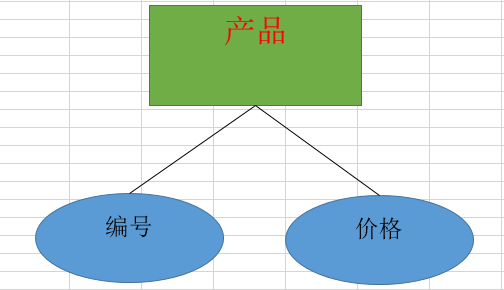
职工实体属性：职工编号、工人姓名、年龄、性别、工种。如下：



零件实体属性：零件编号、重量、价格。如下：



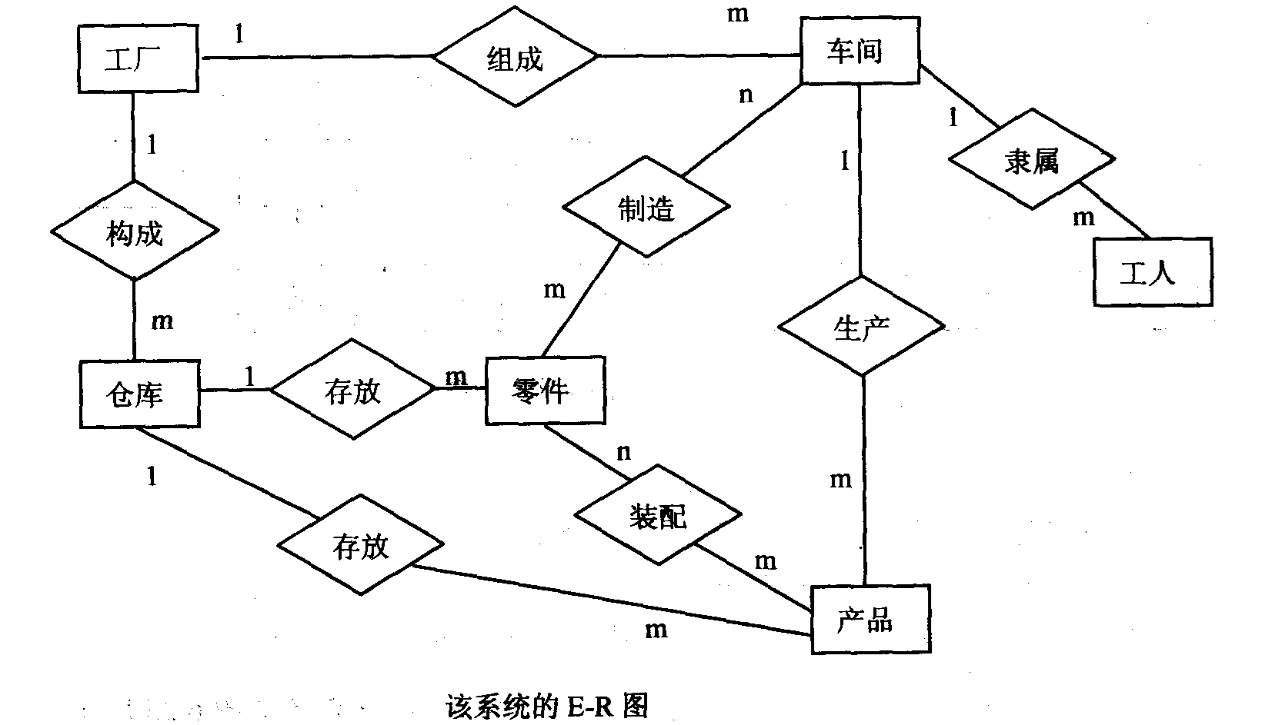
产品实体属性：产品编号，价格。如下：



（三）联系分析

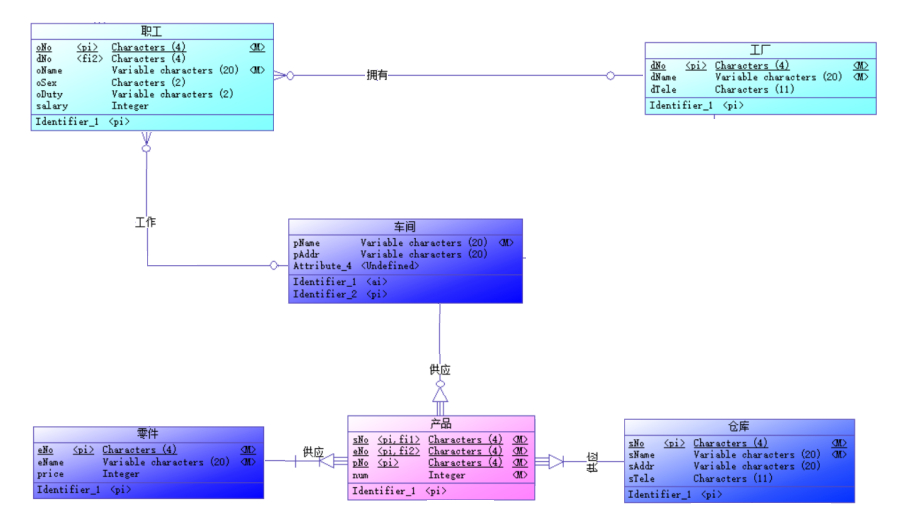
一个普通职工只属于一个车间，但一个车间有多个职工，职工与车间之间的联系是多对一的联系；一个车间只能属于一个工厂，一个工厂拥有多个车间参与，车间与工厂的关系是多对一的关系；一个仓库只能属于一个工厂，一个工厂拥有多个仓库参与，仓库与工厂的关系是多对一的关系；多个车间生产多个零件和产品，产品与车间的关系是多对多，零件与车间的关系是多对多；多个仓库存储多个零件和产品，产品与仓库的关系是多对多，零件与仓库的关系是多对多；

E-R图分析如下：



（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

在工厂管理系统中没有一对一关系的转化

2.一对多关系的转化

一个职工只属于一个车间，但一个车间有多个职工。

一个车间只属于一个工厂，一个工厂拥有多个车间。

职工（职工编号（主键）、车间编号（外键）、仓库编名（外键）、姓名、性别、年龄、工种）

3.多对多关系的转化

多个仓库存储多个零件和多个产品，零件和产品与仓库关系是多对多。

多个车间生产多个零件和多个产品，零件和产品与车间关系是多对多。

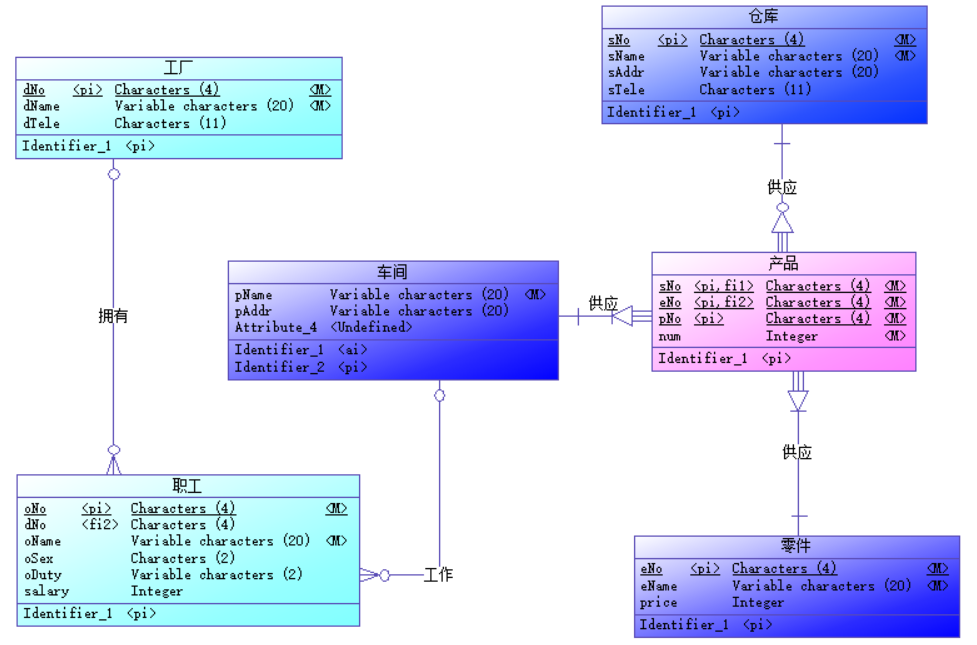
仓库（仓库编名（主键）、仓库主任、电话）

车间（车间编号（主键）、车间主任、地址、电话）

零件（零件编号（主键）、重量、价格）

产品（产品编号（主键）、价格）

（二）逻辑模型设计（.PDM图）



# 四、数据库物理实现

（一）表设计

**工厂信息表设计**

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名称** | **字段解释** | **数据类型** | **备注** |
| **fno** | **工厂编号** | **char(4)** | **主键** |
| **fname** | **厂长名称** | **vchar(20)** | **非空** |

**车间信息表设计：**

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名称** | **字段解释** | **数据类型** | **备注** |
| **workshop\_no** | **车间编号** | **char(4)** | **主键** |
| **wname** | **车间主任** | **vchar(20)** | **非空** |
| **waddress** | **车间地址** | **Char(2)** |  |
| **wtel** | **车间电话** | **char(2)** |  |
| **fno** | **工厂编号** | **char(4)** | **外键** |

**职工信息表设计：**

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名称** | **字段解释** | **数据类型** | **备注** |
| **wno** | **职工编号** | **char(4)** | **主键** |
| **wname** | **姓名** | **vchar(20)** | **非空** |
| **wsex** | **性别** | **Char(2)** |  |
| **wage** | **年龄** | **int** |  |
| **wtype** | **工种编号** | **char(4)** | **外键** |
| **workshop\_no** | **车间编号** | **char(4)** | **外键** |

**仓库信息表设计：**

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名称** | **字段解释** | **数据类型** | **备注** |
| **eno** | **仓库编号** | **char(4)** | **主键** |
| **ename** | **仓库主任** | **vchar(20)** | **非空** |
| **etel** | **电话** | **int** |  |

**零件信息表设计：**

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名称** | **字段解释** | **数据类型** | **备注** |
| **cno** | **零件编号** | **char(10)** | **主键** |
| **cweight** | **零件重量** | **vzhar(10)** | **非空** |
| **cprice** | **产品价格** | **vchar(10)** |  |

**产品信息表设计：**

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名称** | **字段解释** | **数据类型** | **备注** |
| **pno** | **产品编号** | **char(10)** | **主键** |
| **pprice** | **产品价格** | **vchar(20)** | **非空** |

（二）创建表和完整性约束代码设计

1. 创建工厂信息表及完整性约束：

create table factory

(

fno varchar(10) NOT NULL primary key,

fdname varchar(10)

)

1. 创建车间信息表及完整性约束：

create table workshop

(

workshop\_no char(10) NOT NULL primary key,

wdname varchar(10),

waddress varchar(10),

wtel varchar(20),

fno varchar(10) NOT NULL,

foreign key (fno) references factory(fno)

)

1. 创建职工信息表及完整性约束：

create table workman

(

wno char(10) NOT NULL primary key,

wname varchar(10),

wage int,

wtype varchar(10),

workshop\_no char(10) NOT NULL,

foreign key (workshop\_no) references workshop(workshop\_no)

)

1. 创建产品信息表及完整性约束:

create table product

(

pno varchar(10) NOT NULL primary key,

pname varchar(10),

pprice float,

workshop\_no char(10) NOT NULL,

sno char(10) NOT NULL,

foreign key (workshop\_no) references workshop(workshop\_no),

foreign key (sno) references storage(sno)

)

1. 创建零件信息表及完整性约束：

create table element

(

eno char(10) NOT NULL primary key,

eweight float,

eprice float,

sno char(10) NOT NULL,

foreign key (sno) references storage(sno)

)

1. 创建车间与零件信息表表及完整性约束:

create table workshop\_element

(

workshop\_no char(10) NOT NULL ,

eno char(10) NOT NULL ,

primary key(workshop\_no,eno),

foreign key (workshop\_no) references workshop(workshop\_no),

foreign key (eno) references element(eno)

)

1. 创建产品与零件信息表及完整性约束:

create table product\_element

(

pno varchar(10) NOT NULL ,

eno char(10) NOT NULL ,

primary key(pno,eno),

foreign key (pno) references product(pno),

foreign key (eno) references element(eno)

)

1. 创建仓库信息表及完整性约束:

create table storage

(

sno char(10) NOT NULL primary key,

sdname varchar(10),

stel varchar(20),

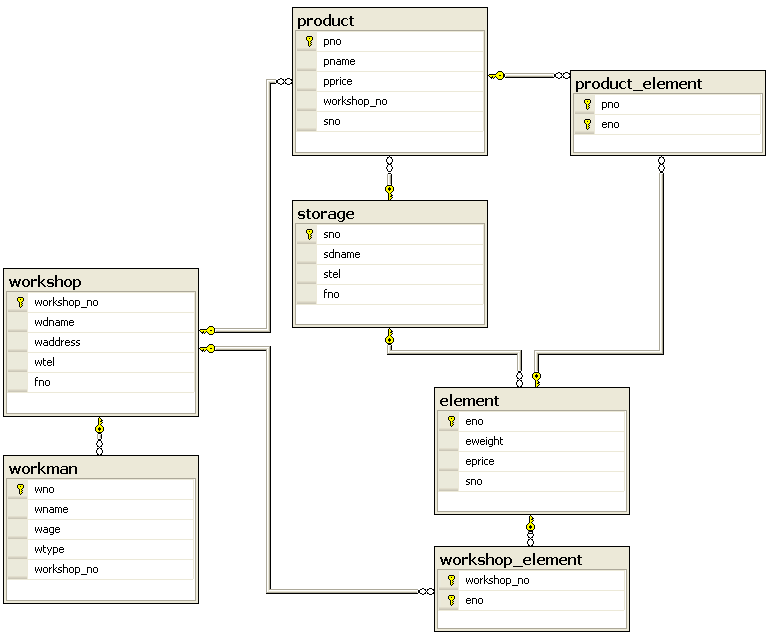
fno varchar(10) NOT NULL,

foreign key (fno) references factory(fno)

)

1. （三）创建视图、索引、存储过程和触发器
2. **创建视图**

**各表关系如下图：**



(1)给产品表创建一个拥有paname(产品名称)，price(产品价格)，以及外键的sno(仓库号)的视图

create view IS\_product

as

select sno,pname,pprice

from product

效果图如下：



(2)给工人信息表创建一个包含有workman(生产表)的所有信息的视图

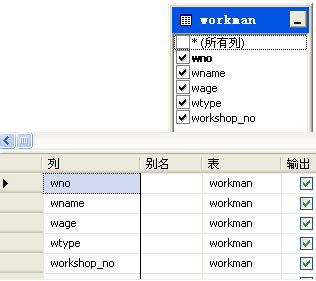
create view IS\_workman

as

select \*

from workman

效果图如下:



1. **索引的建立**

建立索引使查询的时候按顺序排列

create unique index propno on product(pno) --为product表的pno(产品号)建立索引

create unique index eeno on element(eno) --为element表的eno(零件号)建立索引

1. **创建存储过程**

create procedure manage --创建一个名字为manage的存储过程

(@eno char(10), --定义一些参数

@wname varchar(10),

@sdname char(10),

@sno char(10)

)

as

declare @eweight1 float=80.4 --申明一个变量@eweight1

if exists(select eweight from element where eweight< @eweight1)

update element

set eweight=80.4 --将element(零件)表中的eweight都定义为80.4

select \*

from element

select eprice

from element

where eno=@eno

update workman

set wage=10 where wname = @wname

select \*

from workman

update storage

set sdname=@sdname where sno=@sno

select \*

from storage

print'修改后的零件表'

exec manage

@eno='e03', --当eno中为e03时，再查询此时此刻的pprice(价格)

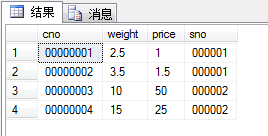
@wname='张大',

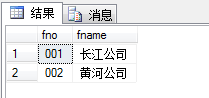
@sdname='王二',

@sno='B2'

首先创建了一个名字为manage的存储过程，定义一些可以用到得相关参数，如@eno，@wname，@sdname，@sno，通过这些参数传递确定的数据，利用SQL语句结合在存储过程从而实现想要更改，删除，插入等相关的操作.

**存储过程执行后的效果图为：**

****

****

**4. 触发器的建立与实现操作**

为product表创建一个基于update操作和delete操作的复合型触发器，当修改了该表的pprice(价格)或者删除了pno(产品号)时，触发器被激活生效，显示相关的操作信息

（1）创建触发器

create trigger tri\_update\_delete\_product --创建触发器

on product

for update,delete

as

if update(pprice) --更新价格

begin

select inserted.pno,deleted.pprice as 原价格,inserted.pprice as 新价格

from deleted,inserted

where deleted.pno=inserted.pno

end

else if columns\_updated()=0

begin

select deleted.pno as 被删除的产品号,deleted.pname,deleted.pprice as 原价格

from deleted

end

（2）触发触发器

1．在查询命令窗口中输入以下update语句，修改产品号为’01’的价格，激发触发器

update product

set pprice=pprice+1

where pno='01'

视图如下：



2．在查询命令窗口中输入以下delete语句，删除产品为号为’03’的记录，激发触发器

delete product

where pno='03'

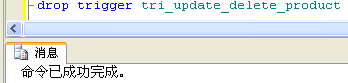
视图如下：



（3）删除新创建的触发器

drop trigger tri\_update\_delete\_product

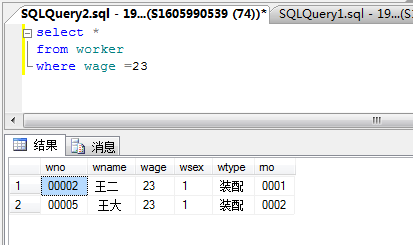
视图如下：



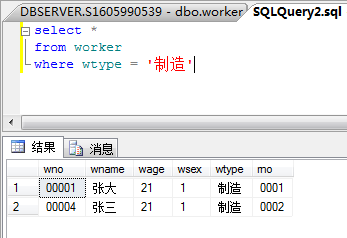
# 五、数据库功能调试

（一）职工管理模块

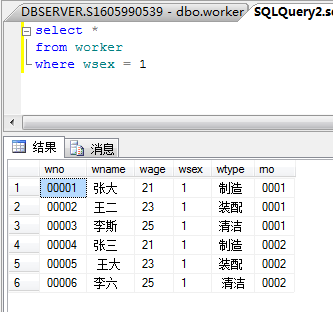
1. 查询职工信息



1. 查询职工所在单位信息

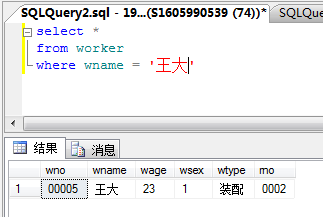


1. 查询职工所属工程信息

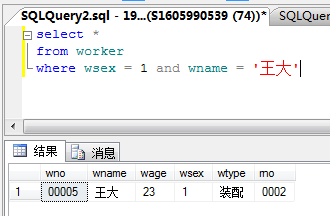


（二）工厂负责人管理模块

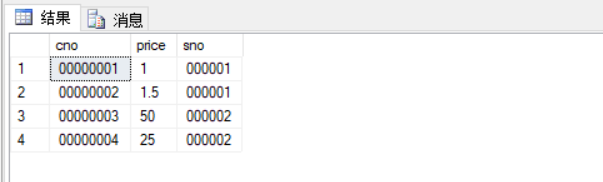
1. 查询工厂信息



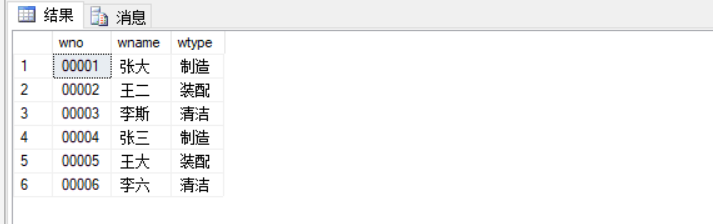
1. 查询工厂所需设备信息



1. 查询工厂供应商信息



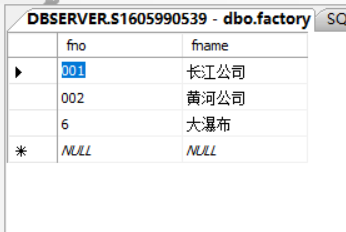
1. 查询参与工厂的职工信息



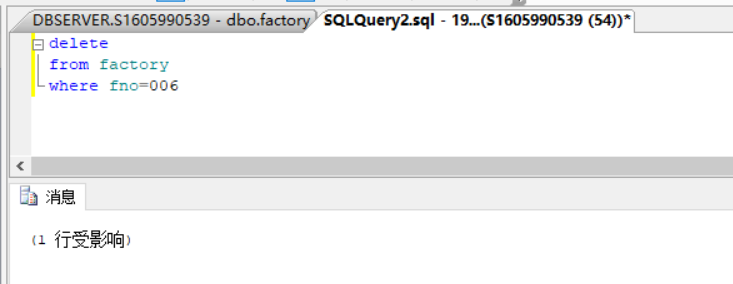
（三）系统管理员管理模块

1. 对工厂的信息管理

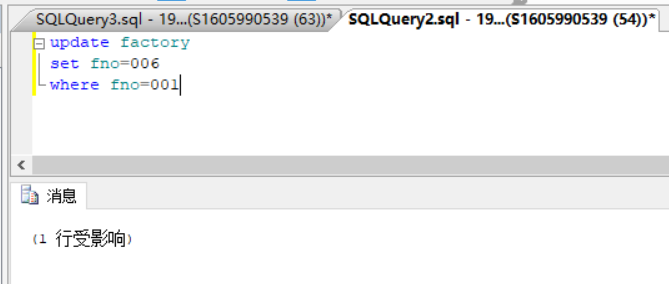
1.1插入工厂信息

1.2删除工厂信息

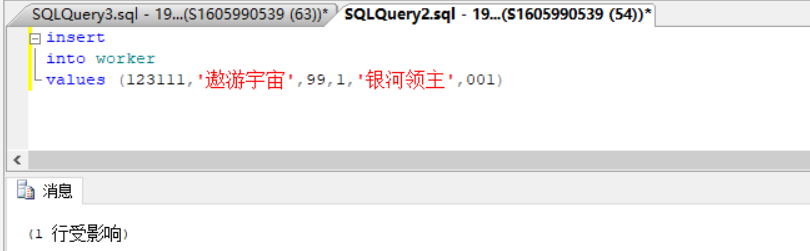
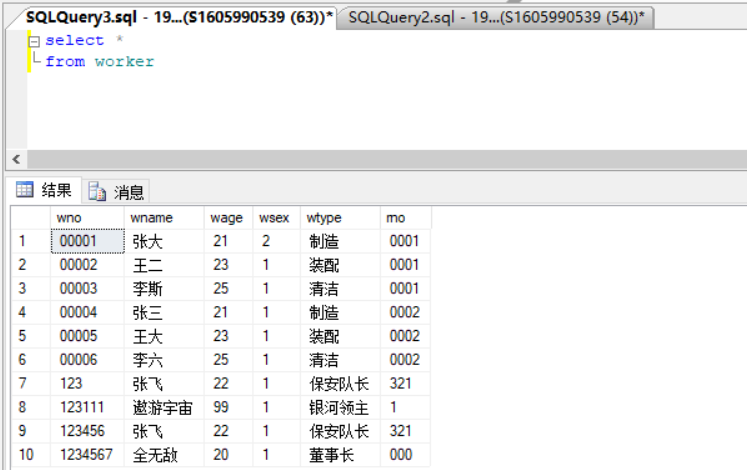
 

1.3修改工厂信息

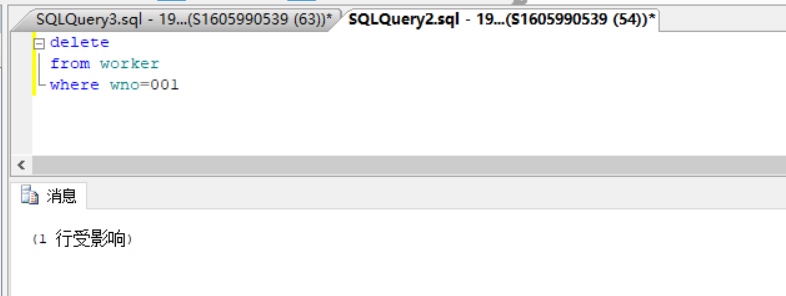
 

1. 对职工的信息管理

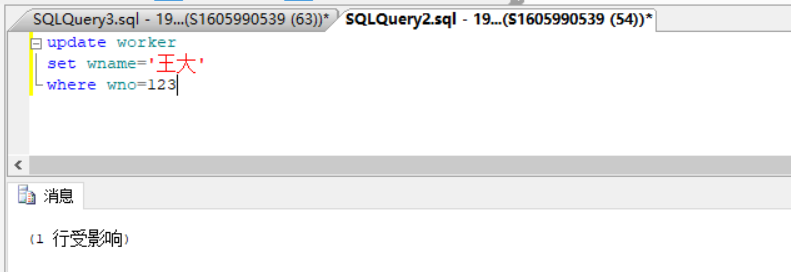
2.1插入职工信息

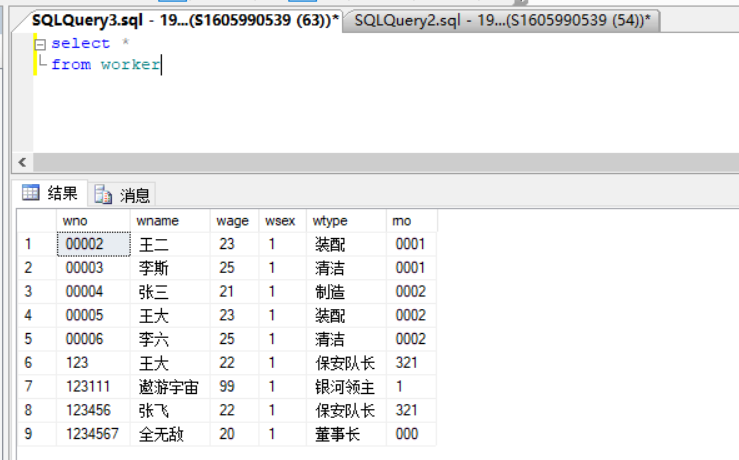
 

2.2删除职工信息

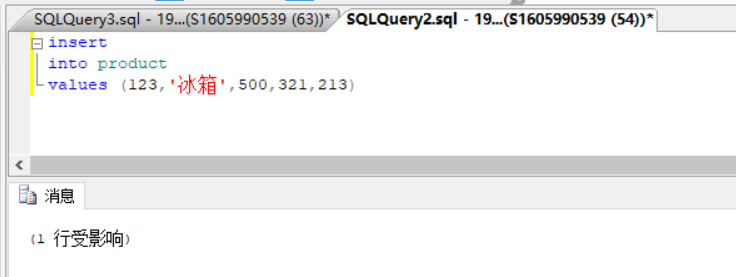
2.3修改职工信息

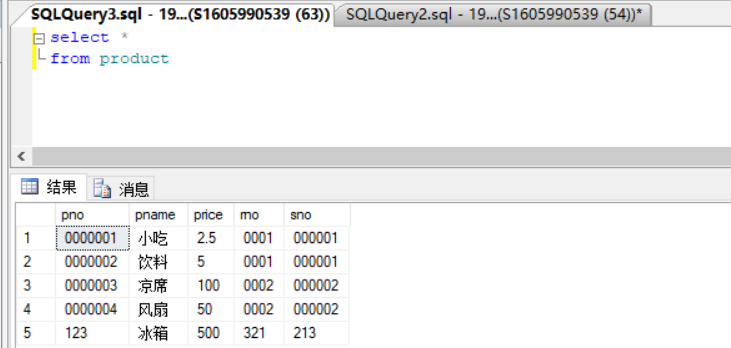




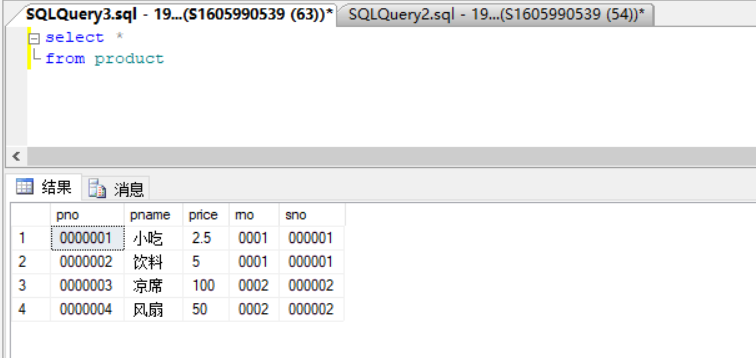
1. 对产品的信息管理

3.1插入产品信息

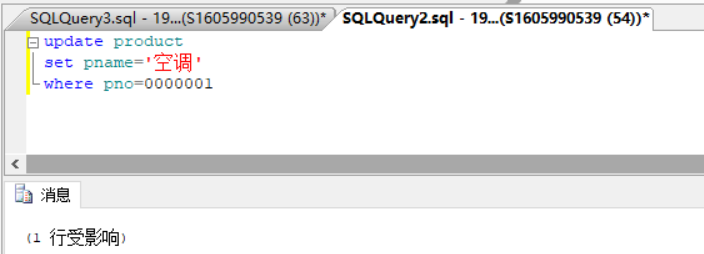
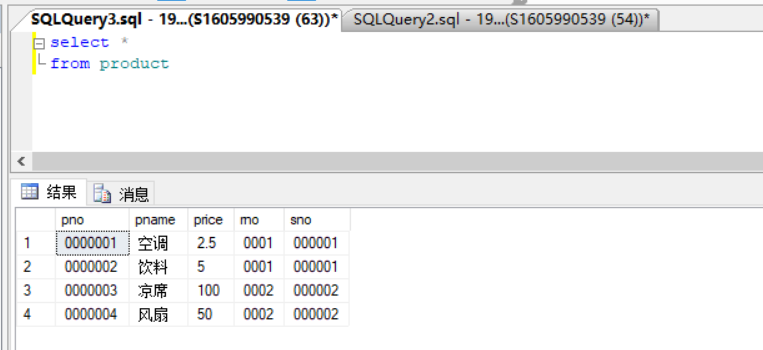




3.2删除产品信息

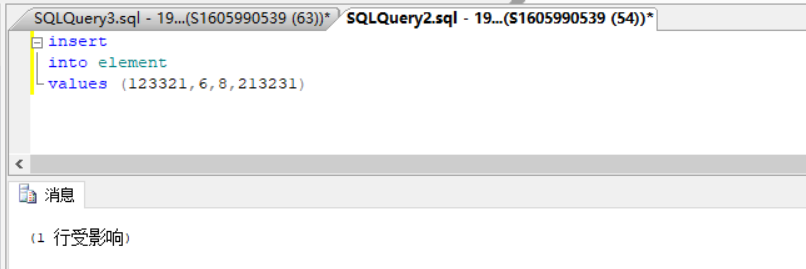
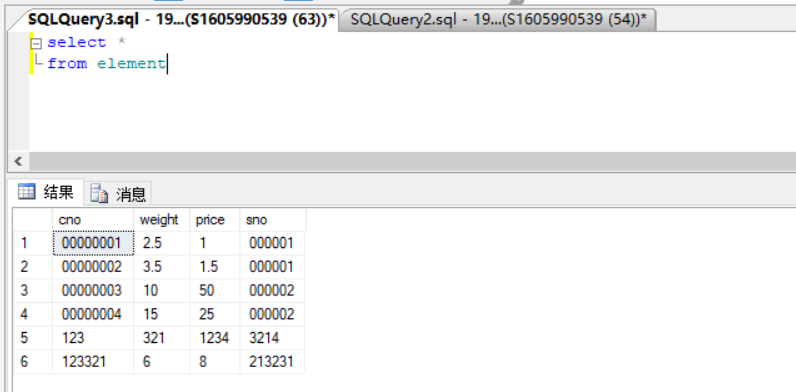
 

3.3修改产品信息

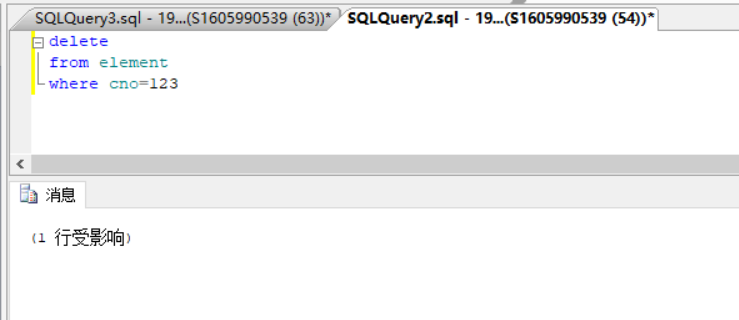
 

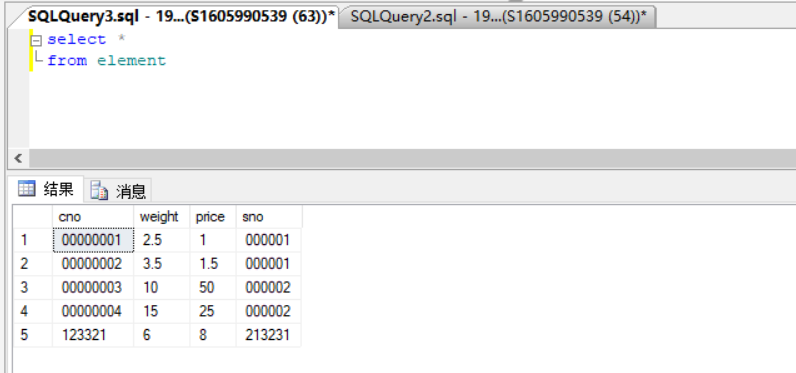
1. 对零件的信息管理

4.1插入零件信息

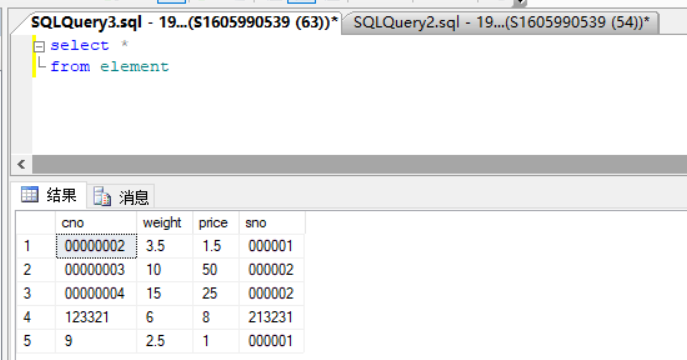
 

4.2删除零件信息



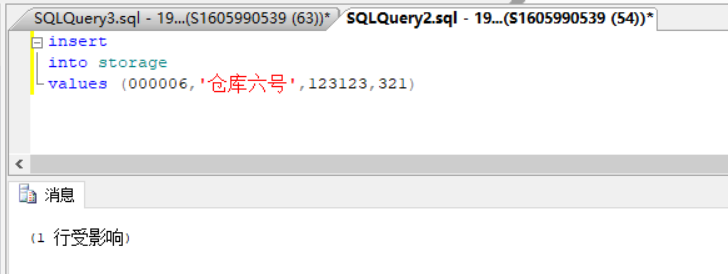
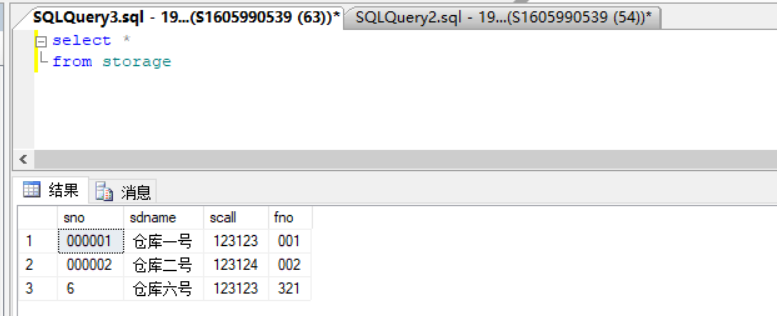


4.3修改零件信息

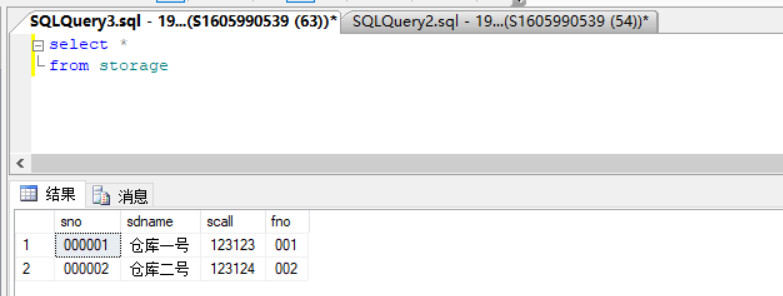
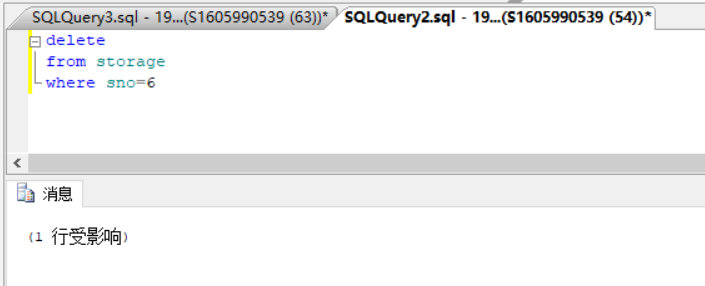
 

1. 对仓库的信息管理

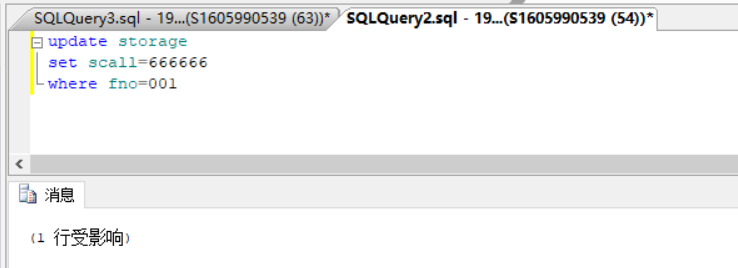
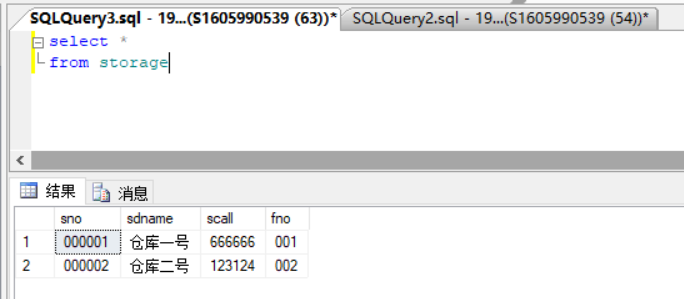
5.1插入仓库信息

5.2删除仓库信息

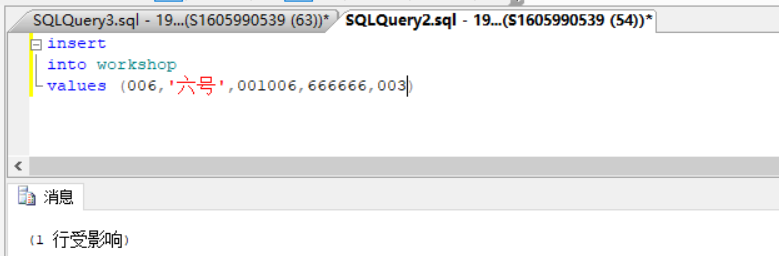
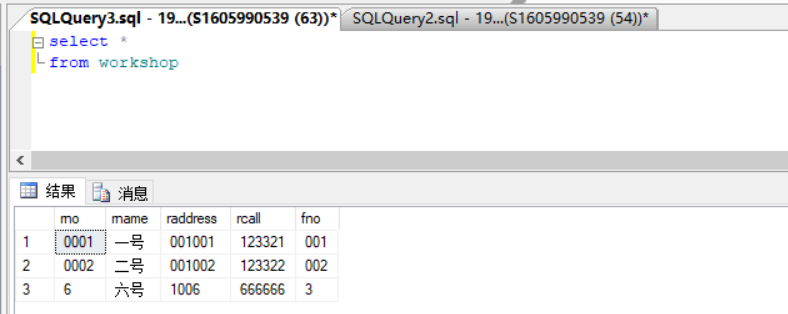


5.3修改仓库信息

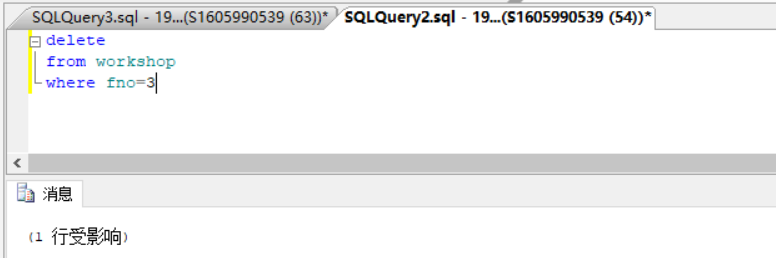
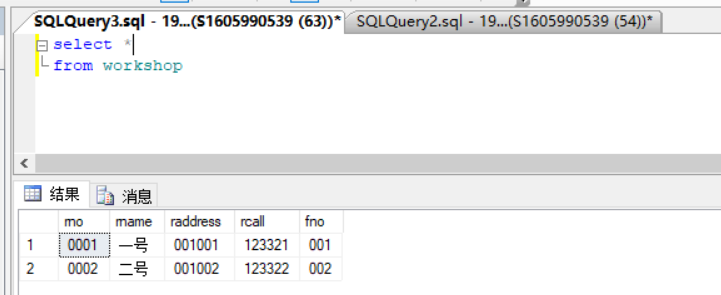
 

1. 对车间的信息管理

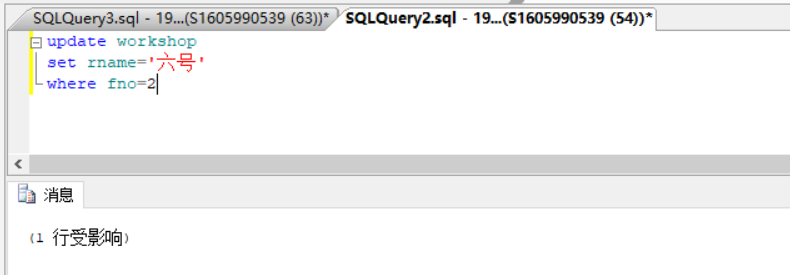
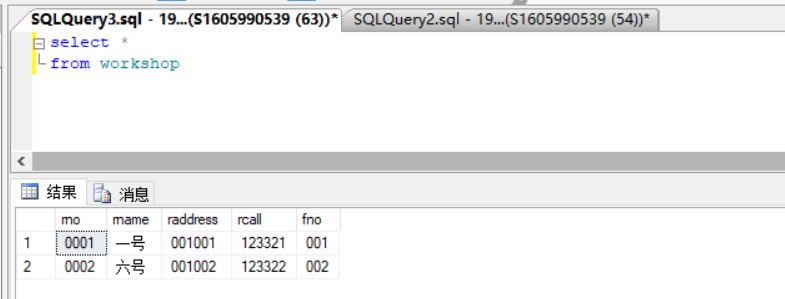
6.1插入车间信息

6.2删除车间信息

6.3修改车间信息

# 六、设计系统前台软件

（一）开发软件

**Adobe Dreamweaver CS6**

**Sql server 2008;**

**Power designner;**

（二）软件功能要求与设计

Adobe Dreamweaver，简称“[DW](https://baike.baidu.com/item/DW/15975506" \t "https://baike.baidu.com/item/Adobe%20Dreamweaver/_blank)”，中文名称 "梦想编织者"，最初为美国[MACROMEDIA](https://baike.baidu.com/item/MACROMEDIA" \t "https://baike.baidu.com/item/Adobe%20Dreamweaver/_blank)公司开发，2005年被[Adobe公司](https://baike.baidu.com/item/Adobe%E5%85%AC%E5%8F%B8" \t "https://baike.baidu.com/item/Adobe%20Dreamweaver/_blank)收购。DW是集[网页](https://baike.baidu.com/item/%E7%BD%91%E9%A1%B5" \t "https://baike.baidu.com/item/Adobe%20Dreamweaver/_blank)制作和管理网站于一身的所见即所得[网页代码](https://baike.baidu.com/item/%E7%BD%91%E9%A1%B5%E4%BB%A3%E7%A0%81" \t "https://baike.baidu.com/item/Adobe%20Dreamweaver/_blank)编辑器。利用对 [HTML](https://baike.baidu.com/item/HTML" \t "https://baike.baidu.com/item/Adobe%20Dreamweaver/_blank)、[CSS](https://baike.baidu.com/item/CSS/5457" \t "https://baike.baidu.com/item/Adobe%20Dreamweaver/_blank)、[JavaScript](https://baike.baidu.com/item/JavaScript" \t "https://baike.baidu.com/item/Adobe%20Dreamweaver/_blank)等内容的支持，[设计师](https://baike.baidu.com/item/%E8%AE%BE%E8%AE%A1%E5%B8%88/5112897" \t "https://baike.baidu.com/item/Adobe%20Dreamweaver/_blank)和[程序员](https://baike.baidu.com/item/%E7%A8%8B%E5%BA%8F%E5%91%98" \t "https://baike.baidu.com/item/Adobe%20Dreamweaver/_blank)可以在几乎任何地方快速制作和进行[网站建设](https://baike.baidu.com/item/%E7%BD%91%E7%AB%99%E5%BB%BA%E8%AE%BE/185526" \t "https://baike.baidu.com/item/Adobe%20Dreamweaver/_blank)。

Adobe Dreamweaver使用所见即所得的接口，亦有[HTML](https://baike.baidu.com/item/HTML" \t "https://baike.baidu.com/item/Adobe%20Dreamweaver/_blank)（[标准通用标记语言](https://baike.baidu.com/item/%E6%A0%87%E5%87%86%E9%80%9A%E7%94%A8%E6%A0%87%E8%AE%B0%E8%AF%AD%E8%A8%80" \t "https://baike.baidu.com/item/Adobe%20Dreamweaver/_blank)下的一个应用）编辑的功能，借助经过简化的智能编码引擎，轻松地创建、编码和管理[动态网站](https://baike.baidu.com/item/%E5%8A%A8%E6%80%81%E7%BD%91%E7%AB%99" \t "https://baike.baidu.com/item/Adobe%20Dreamweaver/_blank)。访问代码提示，即可快速了解 HTML、CSS 和其他[Web](https://baike.baidu.com/item/Web/150564" \t "https://baike.baidu.com/item/Adobe%20Dreamweaver/_blank) 标准。使用[视觉](https://baike.baidu.com/item/%E8%A7%86%E8%A7%89/5228" \t "https://baike.baidu.com/item/Adobe%20Dreamweaver/_blank)辅助功能减少错误并提高[网站开发](https://baike.baidu.com/item/%E7%BD%91%E7%AB%99%E5%BC%80%E5%8F%91/1294506" \t "https://baike.baidu.com/item/Adobe%20Dreamweaver/_blank)速度。

Sql 2008的功能是通过信息学院的服务器进入自己的专属数据库，建自己实验中所需要的各种表，插入数据，建立视图，触发器，存储等等；

Powerdesigner的功能是对自己实验的ER图，以及关系模式的分析，画出物理模型，同时转化成概念模型，以及数据流图；

Dreamweaver是将数据库与其连接，将数据库的表导入Dreamweaver的站点中，对其进行界面的设计，比如更新表，插入表，登录界面的设计

（三）软件功能实现

插入记录集



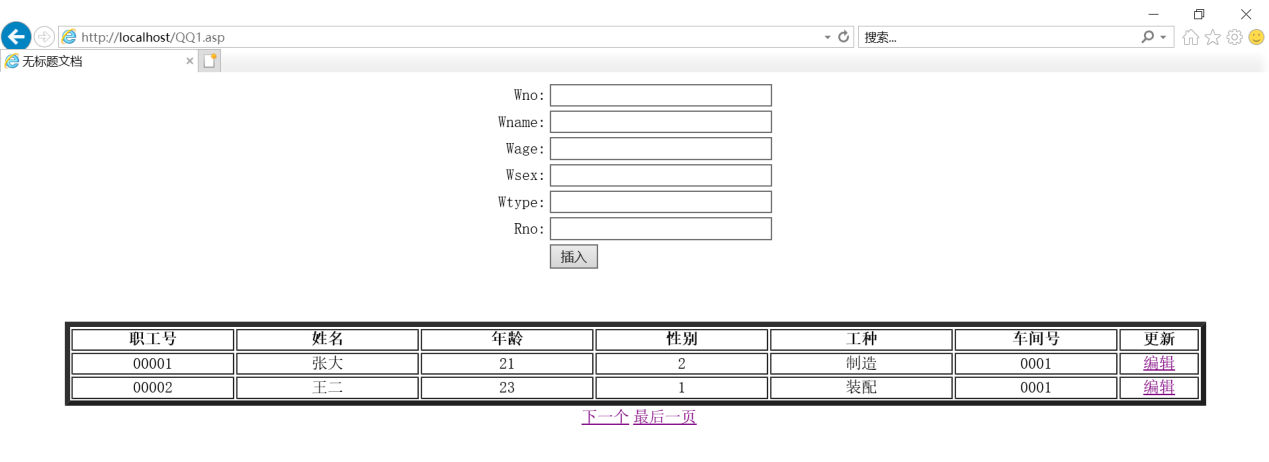


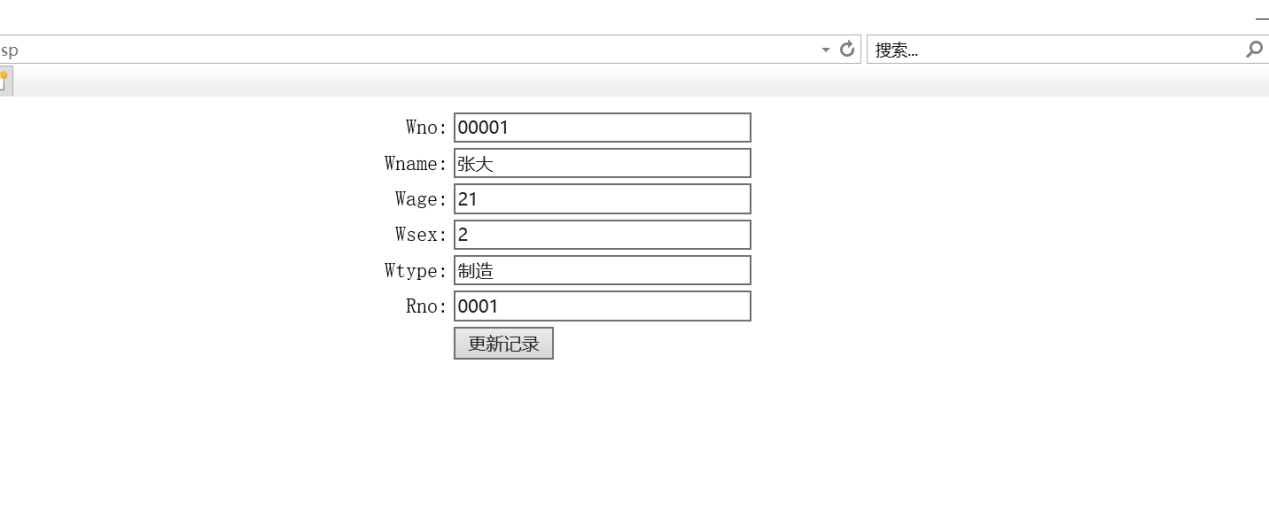
插入数据



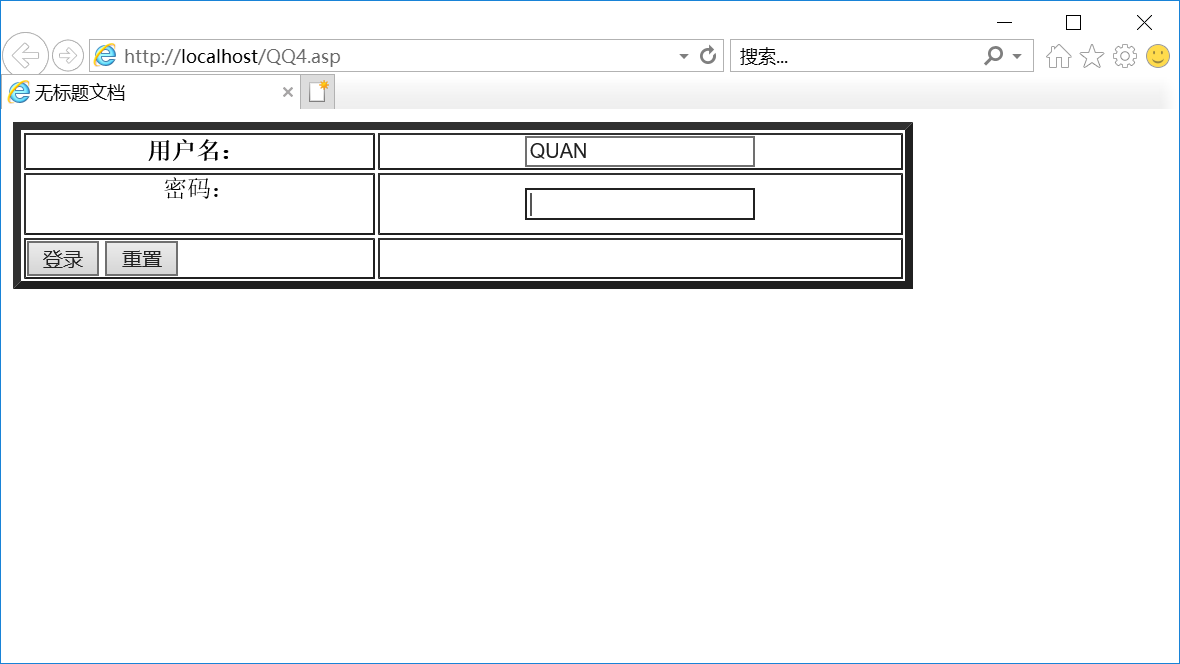


更新数据





账号登录



# 七、设计总结

这次课程设计加深了我对数据库系统概论相关知识和SQL SERVER相关功能的理解。例如对建立基本表、视图、索引等，都比以前更加的熟练。并在解决遇到的各种问题的过程中，学到了很多新的知识,在本次课程设计之前，我一直都没搞懂很多知识点，而且运用起来很生疏，尽管刚刚做得时候困难重重，思路非常不明确。但是经过我请教老师和同学之后，我从学习中逐渐找到了清晰的思路，并把知识点连接了起来，从自己亲自实践之后才能发现，只有自己动手过才能深刻理解书本知识，而不是单独的死记硬背。

另外我也从本次课程设计中深深的体会到了，学习过程中不能有一点马虎，每个完美的课程设计都是严谨的作风和清晰的思路创造出来的，所以以后无论学什么，态度是最重要的，态度端正，学习认真严谨，这样才能学好很多东西。有时候，我自己觉得我写的程序非常的正确，但是就是编译通不过，在查找错误的过程中，面临着否认自己的过程，非常的痛苦，而且由于自己的经验及各方面的能力的不足，所以进展的速度非常的缓慢，往往几天的时间还没有一点进展。这时候，我一般是先自己通过书本，手册和资料找解决办法，实在没辙了才向专家请教。尽管向专家请教解决问题比较快，自己钻研花的时间较多，但我强迫自己独立的思考对我的学习提高帮助非常大。

由于我的知识浅薄，经验不足及阅历颇浅，在设计的过程中也存在一些问题。比如需求分析阶段的工作完成的不是很充分，在后面的设计过程中，前后时常会出现矛盾，不一致现象，因此要反复的修改前面的结果，导致整个工作进行的不是很顺利。另外比如功能过少等问题，我会在工作的使用过程中，根据工作的具体要求不断的修改，完善，争取使该系统慢慢趋向完美，以上是我本次课程设计的总的收获。