《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 1](#_Toc515435289)

[（三）数据流分析 错误!未定义书签。](#_Toc515435290)

[（四）数据字典 2](#_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 8](#_Toc515435295)

[（四）概念模型分析（.PDM图） 8](#_Toc515435296)

[三、数据库逻辑结构设计 9](#_Toc515435297)

[（一）概念模型转化为逻辑模型 9](#_Toc515435298)

[1.一对一关系的转化](#_Toc515435299) **[错误!未定义书签。](#_Toc515435299)**

[2.一对多关系的转化](#_Toc515435300) **[错误!未定义书签。](#_Toc515435300)**

[3.多对多关系的转化](#_Toc515435301) **[错误!未定义书签。](#_Toc515435301)**

[（二）逻辑模型设计（.PDM图） 10](#_Toc515435302)

[四、数据库物理实现 10](#_Toc515435303)

[（一）表设计 10](#_Toc515435304)

[（二）创建表和完整性约束代码设计 11](#_Toc515435305)

[（三）创建视图、索引、存储过程和触发器 13](#_Toc515435306)

[五、数据库功能调试 15](#_Toc515435307)

[（一）职工管理模块 错误!未定义书签。](#_Toc515435308)

[（二）工程负责人管理模块 错误!未定义书签。](#_Toc515435309)

[（三）系统管理员管理模块 错误!未定义书签。](#_Toc515435310)

[六、设计系统前台软件 17](#_Toc515435311)

[（一）开发软件选择 17](#_Toc515435312)

[（二）软件功能要求与设计 17](#_Toc515435313)

[（三）软件功能实现 18](#_Toc515435314)

[（四）系统测试 错误!未定义书签。](#_Toc515435315)

[七、设计总结 20](#_Toc515435316)

# 一、系统需求分析

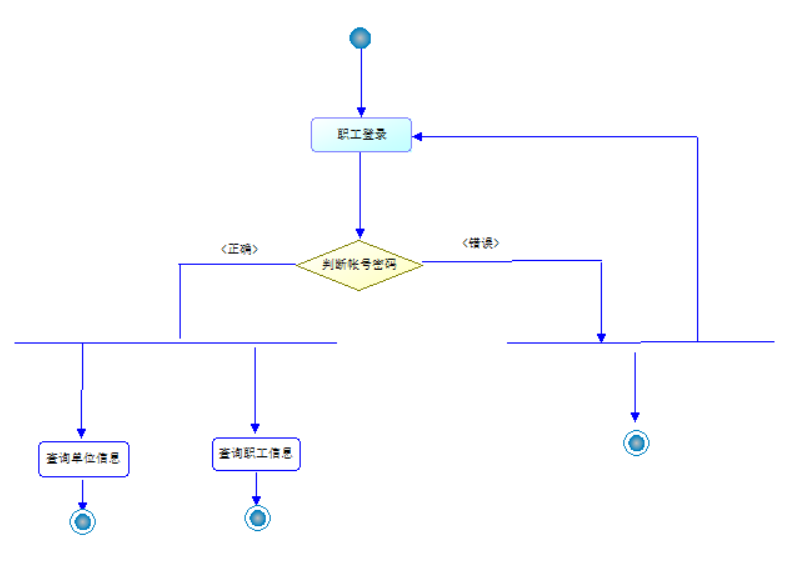
（一）需求概述

本次课程设计的题目是工厂管理系统，系统内容要求是：一个厂内有多个车间，每个车间有车间号、车间主任姓名、地址和电话;一个车间有多个工人，每个工人有职工号、姓名、年龄、性别和工种;一个车间生产多种产品，产品有产品号和价格。一个车间生产多种零件，一个零件也可能为多个车间制造。零件有零件号、重量和价格;一个产品由多种零件组成，一种零件也可装配出多种产品;产品与零件均存入仓库中;厂内有多个仓库，仓库有仓库号、仓库主任姓名和电话。

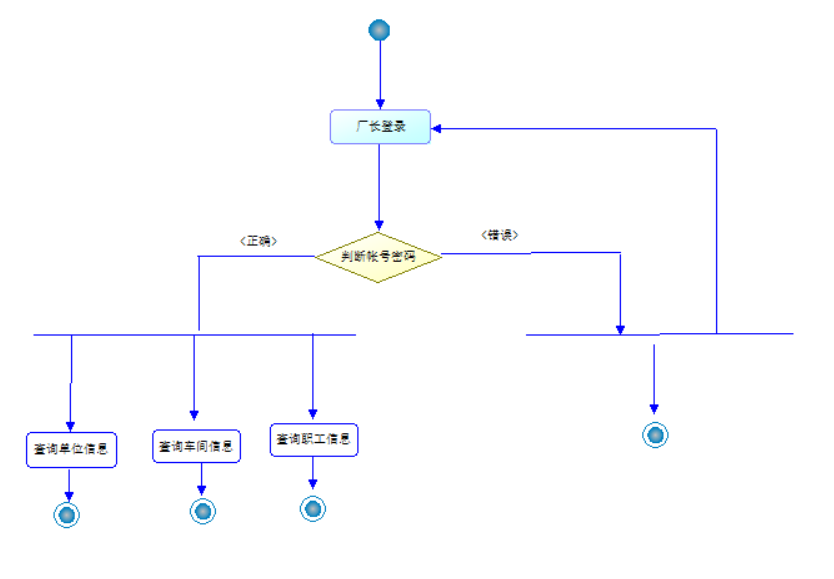
按照一定条件查询、统计工人和产品等基本信息，能模拟工厂生产过程中对原材料（零件）、产品检测、枯脆环节的管理。

（二）业务流分析

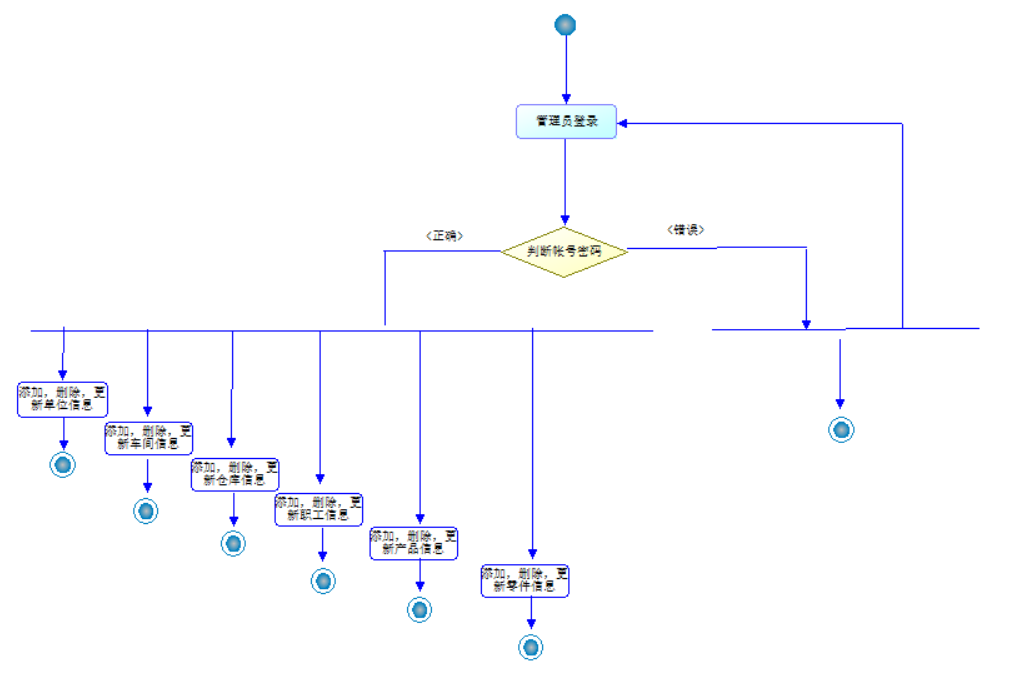
从职工角度：



从工厂分责人角度：



从管理员角度：



（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 |
| 厂名 | 工厂相关信息 | char | 20 |  | 一个工厂对应一个厂名 |
| 厂长名 | 工厂相关信息 | char | 6 |  | 一个工厂对应一个厂长名 |
| 车间号 | 车间相关信息 | Int | 10 | 0001-9999 | 一个车间对应一个车间号 |
| 车间主任 | 车间相关信息 | Char | 6 |  | 一个车间对应一个车间主任 |
| 地址 | 车间相关信息 | char | 20 |  |  |
| 电话 | 车间相关信息 | Int | 11 |  |  |
| 职位编号 | 员工相关信息 | char | 20 | 0001-9999 | 一个职工对应一个编号 |
| 姓名 | 员工相关信息 | char | 6 |  | 一个职工对应一个名字 |
| 年龄 | 员工相关信息 | Int | 4 |  |  |
| 性别 | 员工相关信息 | char | 2 | “男”或“女” |  |
| 工种 | 员工相关信息 | char | 20 |  |  |
| 地址 | 员工相关信息 | char | 20 |  | 一个员工对应一个地址 |
| 电话 | 员工相关信息 | Int | 11 |  | 一个员工对应一个电话 |
| 产品号 | 产品相关信息 | Int | 10 | 0001-9999 | 一个产品对应一个产品号 |
| 价格 | 产品相关信息 | Int | 10 |  | 一个产品对应一个价格 |
| 产品名称 | 产品相关信息 | char | 20 |  | 一个产品对应一个产品名称 |
| 车间号 | 产品相关信息 | Int | 10 |  |  |
| 零件号 | 零件相关信息 | Int | 10 |  | 一个零件对应一个零件号 |
| 重量 | 零件相关信息 | int | 10 |  |  |
| 价格 | 零件相关信息 | Int | 10 |  |  |
| 仓库保管员编号 | 仓库相关信息 | Int | 10 |  | 一个仓库对应一个保管员编号 |
| 姓名 | 仓库相关信息 | char | 6 |  |  |
| 电话 | 仓库相关信息 | int | 11 |  |  |

数据结构：

|  |  |  |
| --- | --- | --- |
| 数据结构名 | 含义名 | 组成 |
| 工厂表 | 定义了工厂的相关信息 | 厂名、厂长 |
| 车间表 | 定义了车间的相关信息 | 车间编号、车间名称、车间主任编号、备注 |
| 员工表 | 定义了员工表的相关信息 | 员工号、姓名、工种、职位编号、年龄、性别、电话、地址 |
| 产品表 | 定义了产品的相关信息 | 编号、产品名称、价格、车间编号、备注 |
| 零件表 | 定义了零件的相关信息 | 零件号、重量、价格 |
| 车间-零件表 | 定义了车间与零件的相关信息 | 车间编号、零件号 |
| 产品-零件表 | 定义了产品与零件的相关信息 | 产品编号、零件号 |
| 仓库表 | 定义了仓库相关的信息 | 编号、管理员姓名、电话 |
| 零件-仓库表 | 定义了零件与仓库的相关信息 | 仓库编号（主键）、零件编号 |
| 产品-仓库表 | 定义了产品与仓库的相关信息 | 仓库编号（主键）、零件编号 |

数据流：

|  |  |  |  |
| --- | --- | --- | --- |
| 数据流名 | 数据流来源 | 数据流去向 | 组成 |
| 产品信息 | 车间 | 仓库 | 单位信息 |
| 零件信息 | 车间 | 仓库 | 职工信息 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

数据存储：

|  |  |  |  |
| --- | --- | --- | --- |
| 数据存储名 | 输入的数据流 |  | 组成 |
| 车间信息 | 存储了车间相关信息 |  | 车间编号,车间名称,车间主任编号,备注 |
| 员工信息 | 存储了员工相关信息 |  | 员工号,姓名,工种,职位编号,年龄,性别,电话,地址 |
| 产品信息 | 存储了产品相关信息 |  | 编号,产品名称,价格,车间编号,备注 |
| 零件信息 | 定义了零件相关信息 |  | 零件号,重量,价格 |
| 仓库信息 | 定义了仓库相关信息 |  | 编号,管理员姓名,电话 |

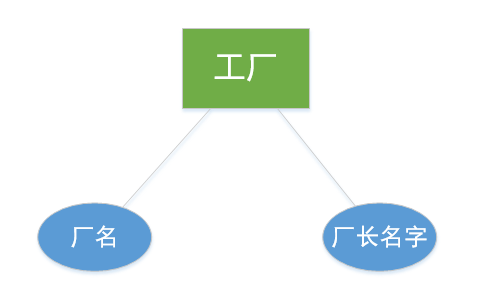
# 二、数据库概念结构设计

（一）实体分析

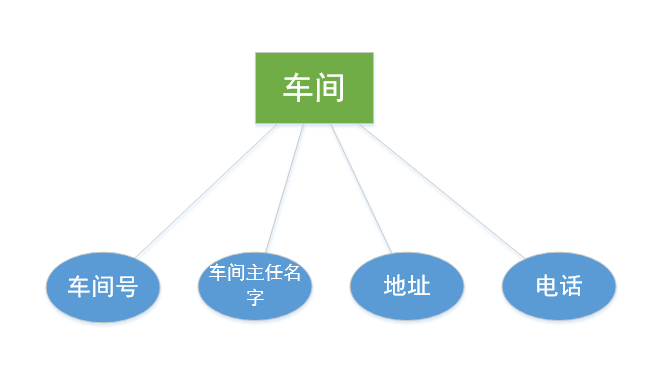
经需求分析，本次课程设计中包含六个实体，他们分别是：工厂实体、车间实体、工人实体、仓库实体、零件实体、产品实体。

（二）属性分析

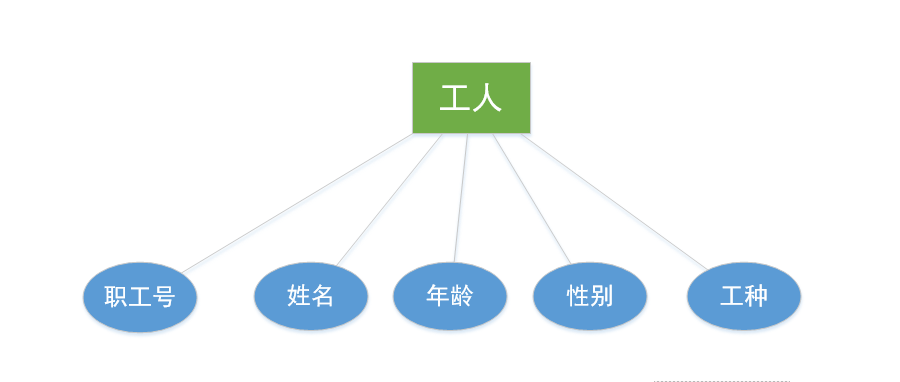
工厂实体属性：厂名、厂长名。如下：



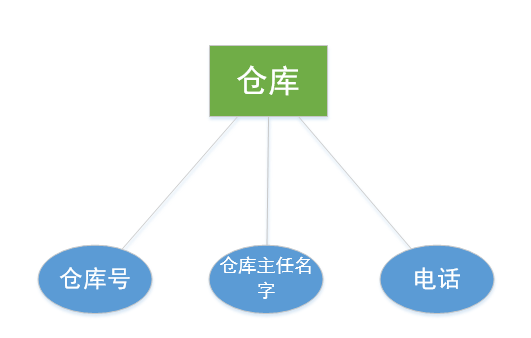
车间实体属性：车间号、车间主任名字、地址、电话。如下：



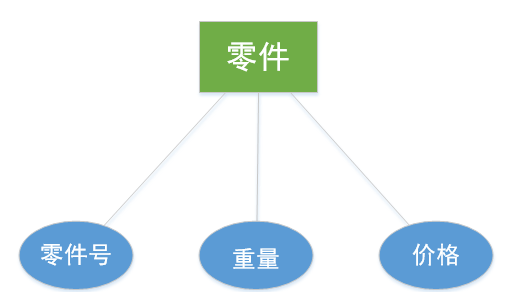
工人实体属性：职工号、姓名、年龄、性别、工种、车间号。如下：



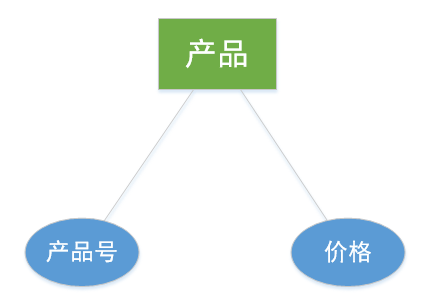
厂库实体属性：仓库号、仓库主任姓名、电话、厂名。如下：



零件实体属性：零件号、重量、价格、仓库号。如下：



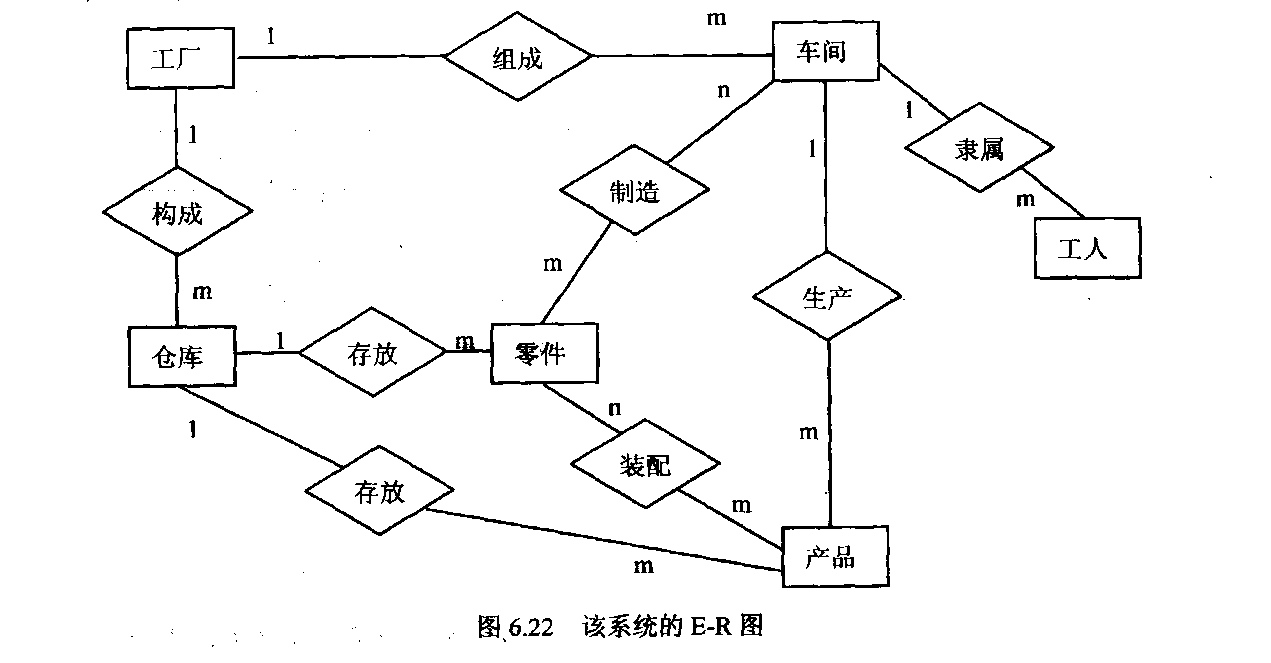
产品实体属性：车间号、零件号。如下：



（三）联系分析

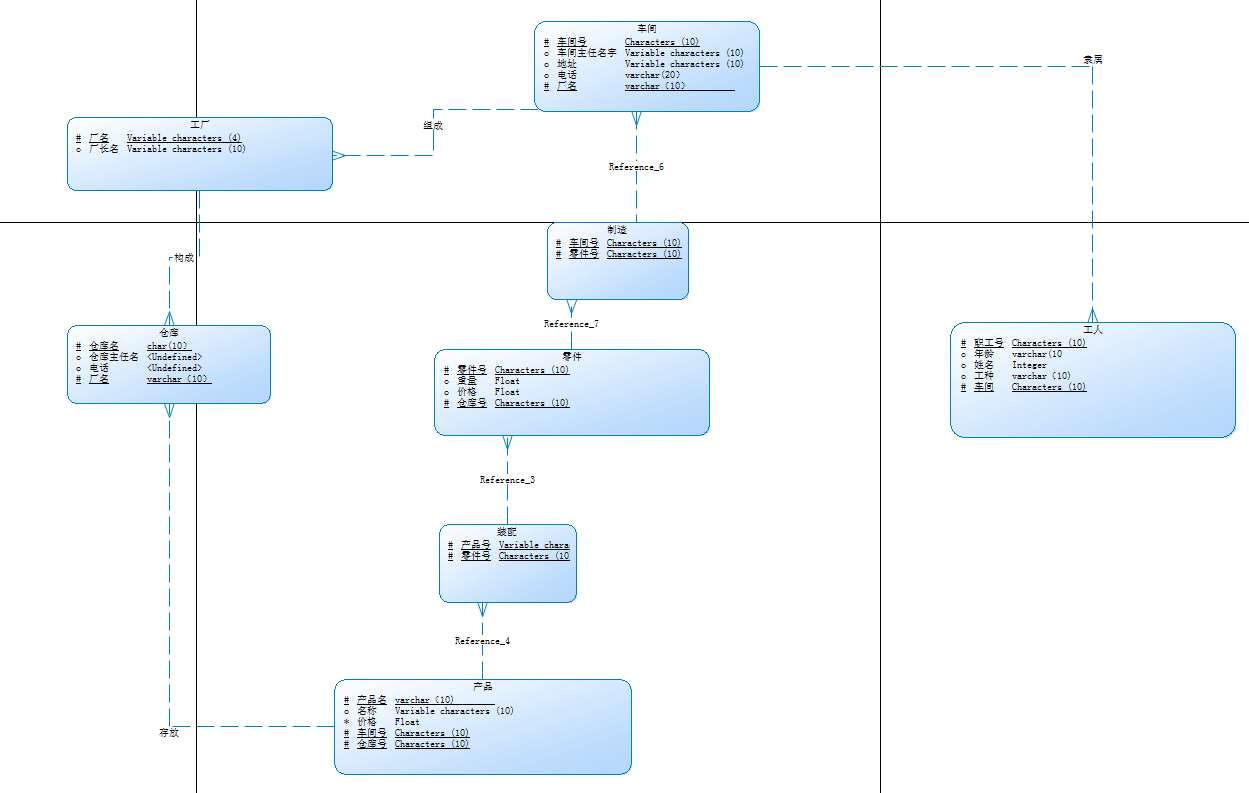
工人在车间制造零件和生产产品，零件在车间由装配成产品，零件和产品存放在仓库，车间和仓库构成车间；

E-R图分析如下：



（四）概念模型分析（.CDM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

逻辑结构设计的任务是将概念结构设计的E-R图，转化为与选用的DBMS所支持的数据模型相符的逻辑结构，形成逻辑类型

关系模型（第一数据项为主键）

工厂(厂名，厂长名)

车间（车间号，车间主任，地址，电话，备注）

员工（职工号，姓名，工种，职位编号，年龄，性别，电话，地址）

产品(产品号，产品名称，价格，车间编号，备注)

零件（零件号，重量，价格）

仓库（仓库号，仓库保管员，姓名，电话）

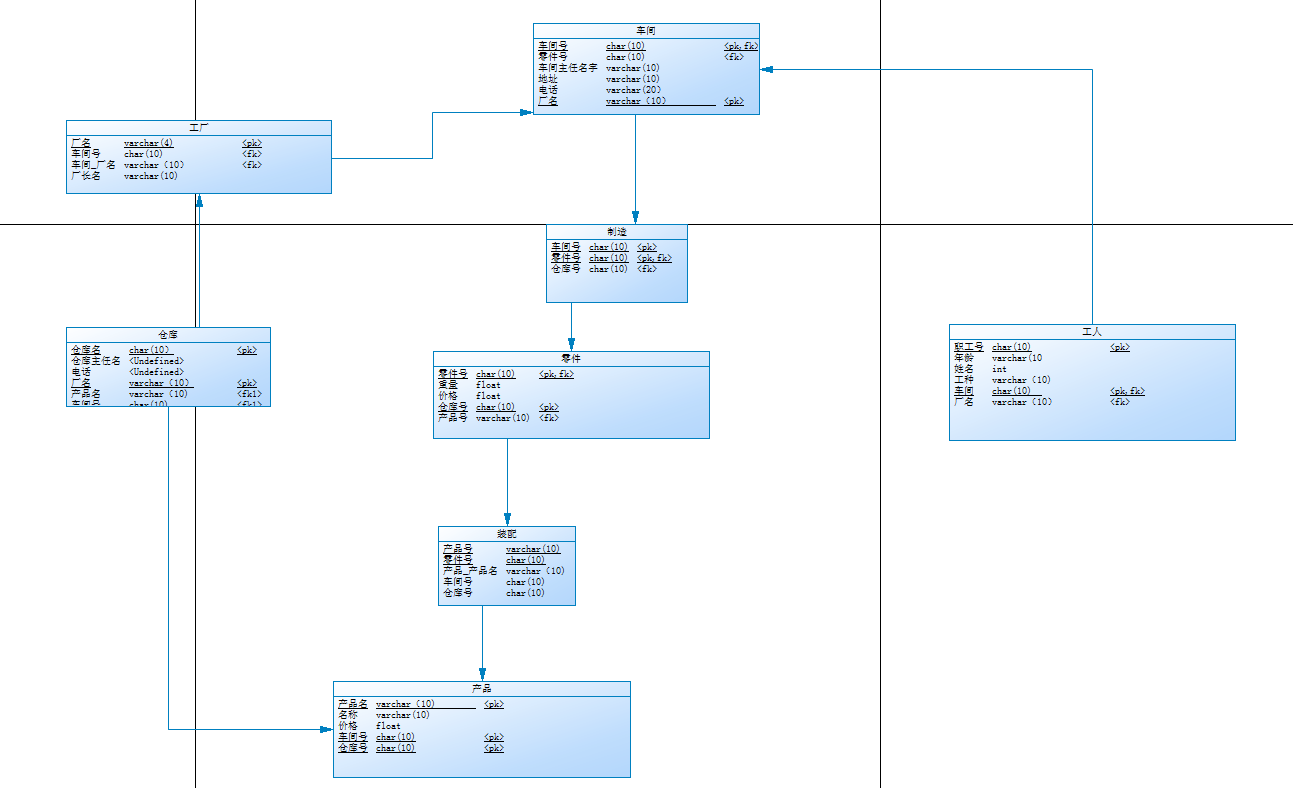
车间-零件（车间号，零件号）

产品-零件（产品号，零件号）

零件-仓库（仓库号，零件号）

产品-仓库（仓库号，产品号）

（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：**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **字段解释** | **数据类型** | **备注** |  |
| **dno** | **厂名** | **varchar(4)** | **NOT NULL** | **主键** |
| **fdname** | **厂长名** | **varchar(10)** |  |  |

**车间信息表：**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **字段解释** | **数据类型** | **备注** |  |
| **workshop\_no** | **车间号** | **char(10)** | **NOT NULL** | **主键** |
| **wdname** | **车间主任名字** | **varchar(10)** |  |  |
| **waddress** | **地址** | **varchar(10)** |  |  |
| **wtel** | **电话** | **varchar(20)** |  |  |
| **fno** | **厂名** | **Varchar(10)** | **NOT NULL** | **外键** |

**工人信息表：**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **字段解释** | **数据类型** | **备注** |  |
| **wno** | **职工号** | **char(10)** | **NOT NULL** | **主键** |
| **wname** | **姓名** | **varchar(10)** |  |  |
| **wage** | **年龄** | **int** |  |  |
| **wtype** | **工种** | **varchar(10)** |  |  |
| **Workshop\_no** | **车间** | **char(10)** | **NOT NULL** | **外键** |

**产品信息表：**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **字段解释** | **数据类型** | **备注** |  |
| **pno** | **产品名** | **varchar(10)** | **NOT NULL** | **主键** |
| **pname** | **名称** | **varchar(10)** |  |  |
| **pprice** | **价格** | **float** |  |  |
| **workshop\_no** | **车间号** | **char(10)** | **NOT NULL** | **外键** |
| **sno** | **仓库号** | **Char(10)** | **NOT NULL** | **外键** |

**零件信息表：**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **字段解释** | **数据类型** | **备注** |  |
| **eno** | **零件号** | **char(10)** | **NOT NULL** | **主键** |
| **eweight** | **重量** | **float** |  |  |
| **eprice** | **价格** | **float** |  |  |
| **sno** | **仓库号** | **Char(10)** | **NOT NULL** | **外键** |

**车间与零件信息表：**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **字段解释** | **数据类型** | **备注** |  |
| **Workshop\_no** | **车间号** | **char(10)** | **NOT NULL** | **主键** |
| **eno** | **零件号** | **char(10)** | **NOTNULL** | **主键** |

**产品与零件信息表：**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **字段解释** | **数据类型** | **备注** |  |
| **pno** | **产品号** | **varchar(10)** | **NOT NULL** | **主键** |
| **eno** | **零件号** | **char(10)** | **NOT NULL** | **主键** |

**仓库信息表：**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **字段解释** | **数据类型** | **备注** |  |
| **sno** | **仓库名** | **char(10)** | **NOT NULL** | **主键** |
| **sdname** | **仓库主任姓名** |  |  |  |
| **stel** | **电话** |  |  |  |
| **fno** | **厂名** | **Varchar（10）** | **NOT NULL** | **外键** |

（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. 创建工人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及完整性约束：

create table workshop\_elemnet

(

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计完整约束性

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. 创建仓库信息表计完整约束性

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)

)

（三）创建视图、索引、存储过程和触发器

1. 创建视图

（1）给产品表创建一个拥有pname(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

（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

1. 创建索引

建立索引使查询的时候按顺序排列

create unique index propno on product(pno)

create unique index eeno on element(eno)

1. 创建存储过程

create procedure manage

(@eno char(10),

@wname varchar(10),

@sdname char(10),

@sno char(10)

)

as

declare @eweight1 float=80.4

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

update element

set eweight=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',

@wname='贺江平',

@sdname='张攀攀',

@sno='B2'

创建触发器

create trigger tri\_update\_delete\_product

on product

for update,delete

as

if UPDATE(pprice)

begin

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

from deleted,inesrtd

where deleted.pno=inserted.pno

end

else if columns\_updated()=0

begin

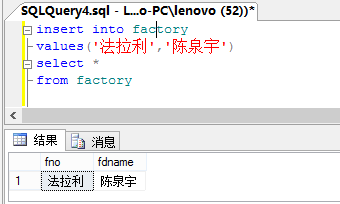
select deleted.pno as 被删除的产品号,deleted.pname,deleted.pprice as 原价格

from deleted

end

# 五、数据库功能调试

1. 查询工厂信息



1. 查询车间信息



1. 查询工人信息

insert into workman

values('A01','贺江平','10','软件工程师','A')

insert into workman

values('A02','彭成辉','20','劳力活','A')

insert into workman

values('A03','张卉卉','19','顾问','A')

insert into workman

values('A04','尹京伟','20','总经理','A')

insert into workman

values('A05','刘清波','20','总监','A')

insert into workman

select \*

from workman



1. 查询产品信息



1. 查询零件信息

insert into element

values('e01','80.4','100','B')

insert into element

values('e02','80.4','200','B')

insert into element

values('e03','80.4','300','B')

insert into element

values('e04','80.4','400','B')

insert into element

values('e05','80.4','500','B')

insert into element

select \*

from element



6．查询车间零件表

insert into workshop\_element

values('A','e10')

select \*

from wrokshop\_element

1. 查询产品零件表

insert into product\_element

values('01','e10')

select \*

from product\_element

1. 查询仓库表

insert into storage

values('B','张攀','120','')

insert into storage

values('B2','张攀','130','')

select \*

from storage

# 六、设计系统前台软件

（一）开发软件选择

MacromediaDreamweaver 8、IIS（互联网信息服务）

（二）软件功能要求与设计

MacromediaDreamweaver 8功能：建立Web站点和应用程序的专业工具。它将可视布局工具、**[应用程序开发](https://baike.baidu.com/item/%E5%BA%94%E7%94%A8%E7%A8%8B%E5%BA%8F%E5%BC%80%E5%8F%91" \t "https://baike.baidu.com/item/dreamweaver8/_blank)**功能和代码编辑支持组合在一起，其功能强大，使得各个层次的开发人员和设计人员都能够快速创建基于标准的网站和应用程序的优美界面。开发人员可以使用Dreamweaver及所选择的服务器技术来创建功能强大的Internet应用程序，从而使用户能连接到数据库、Web服务和旧式系统。

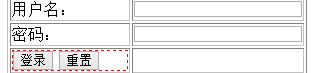
**（三）软件功能实现**

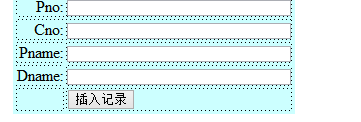
Sql 2008的功能是在数据库中建表，并且对表插入数据，以及视图，触发器，存储的建立；为Dreamweaver的连接建立基础；

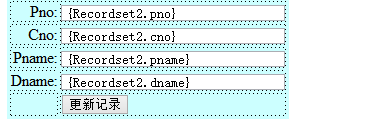
Dreamweaver能与数据库相连接，并且实现表页面的设计，例如更新，插入等等，通过在浏览器中预览，可以实现表页面的显示，以及输入数据后可以跳转到别的页面；

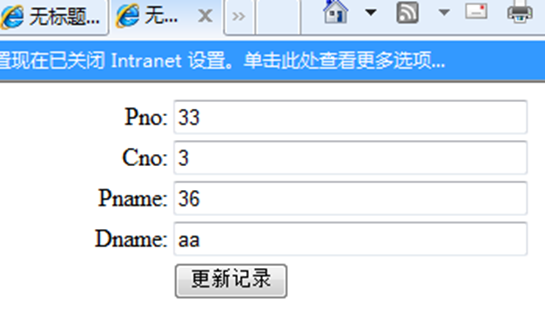
Powerdesigner的主要功能就是通过画物理模型，与此同时将物理模型转化为代码，例如学生表转化为代码等等；而且能将物理模型转化为概念模型；

（四）系统测试

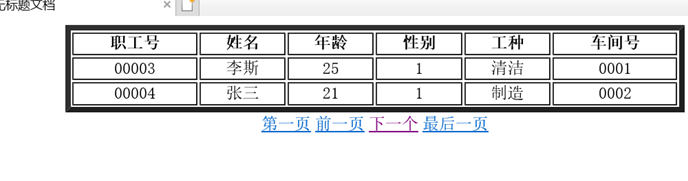












# 七、设计总结

通过这次课程设计，让我发现了很多问题。虽说上了数据库上过管理信息系统等相关的课程，但是没有亲身经历过相关的设计工作细节。还是有一定挑战的，其中存在许多困难，但是这次课程设计在考前对我来说也是一次很好的复习知识点的机会，在实践中我对于很多知识有了更进一步的了解，收获颇多。

通过这次课程设计发现这其中需要的很多知识我们没有接触过，去图书馆查资料的时候发现我们前边所学到的仅仅是皮毛，还有很多需要我们掌握的东西我们根本不知道。同时也发现有很多已经学过的东西我们没有理解到位，不能灵活运用于实际，不能很好的用来解决问题，这就需要我们不断的大量的实践，通过不断的自学，不断地发现问题，思考问题，进而解决问题。在这个过程中我们将深刻理解所学知识，同时也可以学到不少很实用的东西。

从各种文档的阅读到开始的需求分析、概念结构设计、逻辑结构设计、物理结构设计。亲身体验了一回系统的设计开发过程。很多东西书上写的很清楚，貌似看着也很简单，思路非常清晰。但真正需要自己想办法去设计一个系统的时候才发现其中的难度。经常做到后面突然就发现自己一开始的设计有问题，然后又回去翻工，在各种反复中不断完善自己的想法。

但是总的来说，完成的还可以，虽不能说很好，但我觉得我在这次课程设计中学到了很多，这就很不错了。