

数据库应用系统课程设计

项目名称： 人事管理系统

课程名称： 数据库原理与应用

班级： 计 162

姓名：

学号：

教师：

信息工程学院计算机系

目录

目录

| | |
|----------------------|----|
| 一、系统概述..... | 4 |
| 1.1、目的和要求..... | 4 |
| 1.2、实验内容..... | 4 |
| (1) 基本需求描述..... | 4 |
| (2) 数据库设计..... | 4 |
| 1.3、选题描述..... | 4 |
| 1.2、开发环境..... | 5 |
| 1.3 组内分工..... | 5 |
| 二、系统分析..... | 5 |
| 2.1、安全性，完整性分析..... | 5 |
| 2.2、需求分析..... | 6 |
| 2.3、系统模块设计..... | 6 |
| 三、数据库设计..... | 7 |
| 3.1、数据库字典..... | 7 |
| 3.2、系统 E-R 图..... | 9 |
| 3.3、数据库 E-R 图..... | 9 |
| 员工表: | 10 |
| 实习员工表: | 10 |
| 离职表: | 10 |
| 工龄表: | 11 |
| 培训表: | 11 |
| 职务调动表: | 12 |
| 合同表: | 12 |
| 部门表: | 12 |
| 出勤表: | 12 |
| 刷卡机表: | 13 |
| 请假表: | 13 |
| 加班表: | 13 |
| 出差表: | 14 |
| 工资表: | 14 |
| 奖惩表: | 14 |
| 管理员表: | 15 |
| 3.4、数据库逻辑设计..... | 15 |
| 四、系统的实现与调试..... | 16 |
| 4.1 表、数据库的建立..... | 16 |
| 4.2 建立基本表..... | 16 |
| 4.2.1 建立员工表..... | 16 |
| 4.2.2 建立实习员工表: | 17 |
| 4.2.3 建立离职表: | 18 |
| 4.2.4 建立职务调动表: | 18 |

| | |
|-------------------------------------|----|
| 4.2.5 建立合同表: | 19 |
| 4.2.6 建立奖惩表: | 19 |
| 4.2.7 建立加班表: | 20 |
| 4.2.8 建立出差表: | 20 |
| 4.2.9 建立出勤表: | 21 |
| 4.2.10 建立工资表: | 21 |
| 4.2.11 建立管理人员表: | 22 |
| 4.2.12 建立工龄表..... | 22 |
| 4.2.13 建立培训表..... | 23 |
| 4.2.14 建立请假记录表..... | 23 |
| 4.2.15 建立部门表..... | 24 |
| 4.2.16 索引的建立..... | 24 |
| 4.2.17 视图的建立..... | 24 |
| 4.2.18 Excel 与数据库信息的迁移..... | 25 |
| 4.3 程序功能与主要代码内容..... | 26 |
| 4.3.1 登录界面用户名和密码与数据库建立联系..... | 26 |
| 4.3.2 职务调动功能..... | 29 |
| 五、 问题与解决..... | 31 |
| 六、 思考与总结..... | 31 |
| 七、 参考文献..... | 32 |
| [1]何玉洁. 数据库原理与应用. 第三版. 机械工业出版社..... | 32 |
| [2]唐大. C#程序设计教程. 清华大学出版社..... | 32 |

一、系统概述

1.1、目的和要求

在学生对相关业务进行调研，在掌握其基本业务流程、规范后，解决基本的业务功能需求。每组所设计的信息管理系统必须有一定的工作量。在解决基本功能需求时应体现解决实际工程问题的过程，至少包含一个复杂工程问题的解决过程。以下几个题目可供选择，题目中列举了其中的基本要求。如果选择其他的题目，需要在第一周的前两天提交给相关教师，由教师确定该题目的难易程度是否适合课程设计。

1.2、实验内容

（1）基本需求描述

简述信息管理系统的基本功能，用户权限，用户角色，系统技术指标、开发周期以及相关开发过程中的工具选择（数据库、数据库设计工具、软件开发工具、文档版本管理、代码版本管理工具等）

（2）数据库设计

根据业务需求，画出系统的实体关系图。将概念模型转换成逻辑模型和物理模型，将数据填入数据库的表中，要求构造的总数据量不少于 2000 条。要求完成数据库的备份、导入、导出和备份等操作。要求数据库能够支持至少完成一个复杂工程问题。

1.3、选题描述

通常人事管理系统的用户是某企业或事业单位的人力资源管理人员，日常的业务主要围绕着人的管理。因此需求分析时，应当紧紧围绕人事管理业务以及工资系统展开。人事管理是复杂的，包括员工基本信息管理、单位部门管理、岗位管理、员工培训信息管理、薪酬管理、考勤管理以及劳动合同管理等功能。

要求设计模块不少于五个，且各模块功能相互独立。

需要解决的复杂工程问题举例如下：1、考勤管理模块中，需要读取员工进出厂区的刷卡信息，而每天的刷卡记录大约 8000 条，其中有一些是真正的上班刷

卡记录，厂区工人采用轮班制，每天三个班次，四组人员。需要根据有效的记录进行考勤核算；2、员工的薪酬管理系统，根据员工的工资套账，计算员工的五险一金、奖惩、请假、出差、考勤工资、大病统筹、效益工资以及税率等进行核算，对于特殊的计件、计时工资需要进行特殊处理。

1.2、开发环境

开发环境：Windows 10

数据库管理系统：MS SQL SERVER 2017 数据库

E-R 图绘制 visio 2010

数据导入 Excel

1.3 组内分工

二、系统分析

2.1、安全性，完整性分析

安全性控制的方法有用户的标识和鉴定，存取控制，定义视图，数据加密和审计等，在本系统中的安全性体现在用户的标识和鉴定，例如在登陆界面时，首先会输入账户名称，系统内部记录着所有合法用户的标识，每次用户要求进入系统时，由系统进行核实，通过鉴定后才提供机器上对数据库的使用权。当你不是该系统的合法用户时，则账号是错误的就无法进入该系统。就用户存取权限控制而言，在系统登陆界面时会选择是一般用户登陆还是管理员登陆，当为一般用户登陆时，只能查询员工信息和部门信息，而不能对其进行管理。当以管理员身份登陆时不仅仅可以查询部门和员工的基本信息，也可以对其进行添加、删除、修改等操作。

数据库的完整性是指保护数据库中数据的正确性、有效性和相容性，防止错误的数据进入数据库造成无效操作。在本系统对数据类型的约束，包括数据的

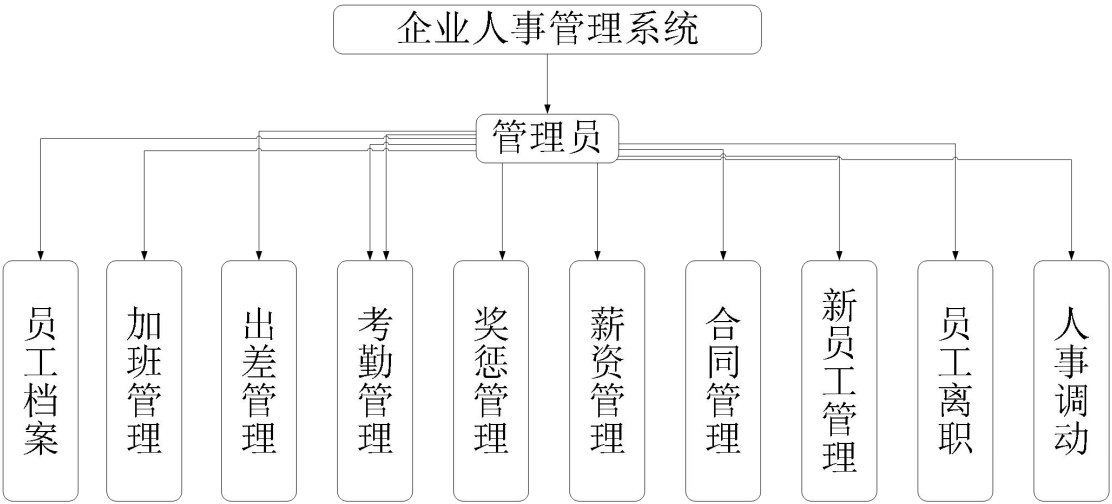
类型、长度、单位和精度等。例如规定员工性别的数据类型应为字符型，长度为2。对数据格式的约束例如规定出生日期的数据格式为 XXXX.XX.XX。对取值范围的约束，例如月份的取值范围为 1-12 月，日期为 1-31 号。对空格的约束，例如员工姓名和编号不能为空，但是员工出生日期即可为空。实体完整性约束说明了关系主键（主码）的属性列必须唯一，其值不能为全空或部分为空。

2.2、需求分析

调查本地的企业，根据企业的具体情况分析、设计和实现企业人事管理系统。其主要功能如下：

- 1、 人事档案管理：户口状况、政治面貌、身份证号、合同管理等。
- 2、 考勤、打卡、请假。
- 3、 人事变动：新进员工登记、员工离职登记、人事变更记录。
- 4、 薪资、考核奖惩。
- 5、 员工培训。
- 6、 系统维护：操作员管理、权限设置等。

2.3、系统模块设计



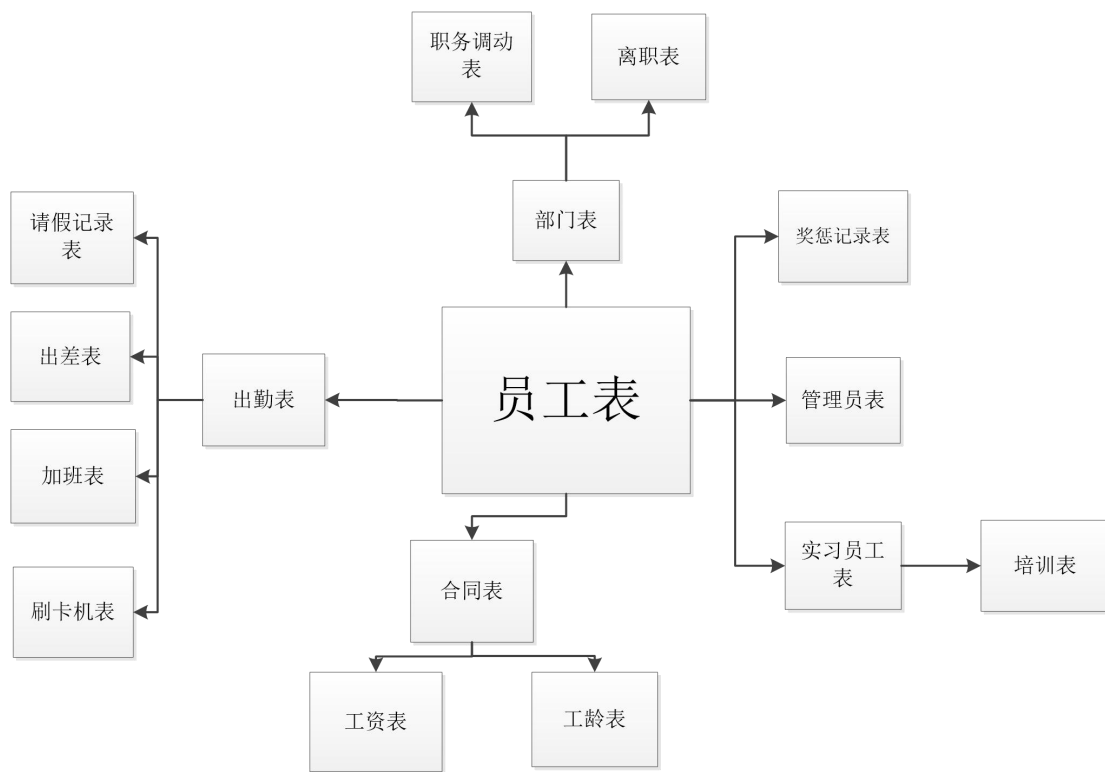
三、数据库设计

3.1、数据库字典

| 数据项 | 来源 | 数据类型 | 数据大小 | 约束条件 |
|--------|-------|----------|------|-------------|
| 部门号 | 部门表 | char | 10 | primary key |
| 部门名 | 部门表 | char | 10 | not null |
| 领导人号 | 员工表 | char | 10 | not null |
| 出差编号 | 出差表 | char | 10 | primary key |
| 出差起始 | 出差表 | char | 10 | |
| 出差结束 | 出差表 | char | 10 | not null |
| 出勤号 | 出勤表 | char | 6 | primary key |
| 上班日期 | 出勤表 | datetime | | not null |
| 上班时间 | 出勤表 | datetime | | not null |
| 下班时间 | 出勤表 | datetime | | not null |
| 刷卡机号 | 刷卡机表 | char | 4 | primary key |
| 打卡次数 | 出勤表 | nchar | 10 | |
| 工龄记录号 | 工龄表 | char | 10 | primary key |
| 入职日期 | 工龄表 | datetime | | |
| 工资编号 | 工资表 | char | 6 | primary key |
| 基本工资 | 工资表 | money | | not null |
| 罚款 | 工资表 | money | | |
| 起始时间 | 工资表 | datetime | | |
| 截止日期 | 工资表 | datetime | | |
| 发薪日期 | 工资表 | datetime | | |
| 管理员账号 | 管理员表 | char | 10 | primary key |
| 密码 | 管理员表 | char | 10 | not null |
| 级别 | 管理员表 | char | 10 | |
| 合同编号 | 合同表 | char | 10 | primary key |
| 合同起始日期 | 合同表 | datetime | | |
| 合同终止日期 | 合同表 | datetime | | |
| 加班编号 | 加班表 | char | 10 | primary key |
| 加班起始 | 加班表 | datetime | | |
| 加班结束 | 加班表 | datetime | | |
| 奖惩编号 | 奖惩记录表 | char | 6 | primary key |
| 奖惩类型 | 奖惩记录表 | char | 4 | |
| 奖惩金额 | 奖惩记录表 | money | | |
| 奖惩日期 | 奖惩记录表 | datetime | | |
| 批复部门号 | 部门表 | char | 10 | |

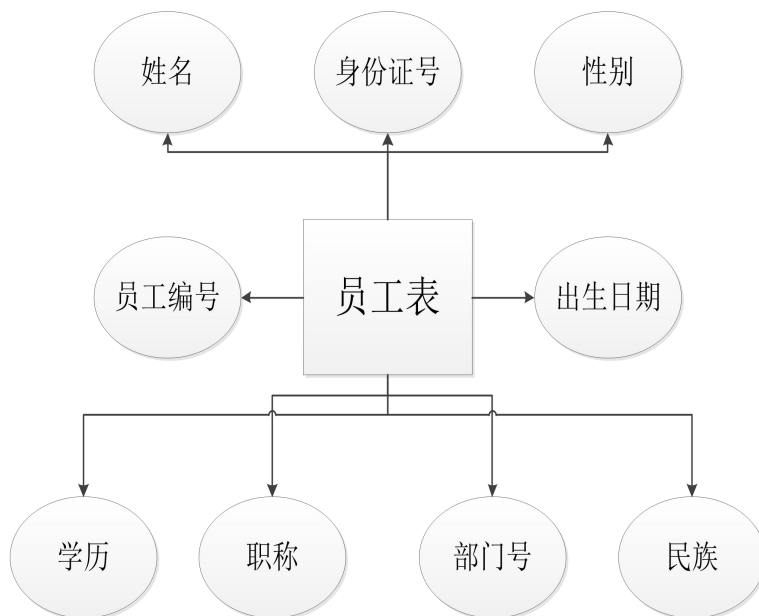
| | | | | |
|-------|-------|----------|----|-------------|
| 批复日期 | 奖惩记录表 | datetime | | |
| 奖惩原因 | 奖惩记录表 | varchar | 50 | |
| 离职编号 | 离职表 | char | 10 | primary key |
| 离职日期 | 离职表 | char | 10 | |
| 培训编号 | 培训表 | char | 10 | primary key |
| 培训项目 | 培训表 | char | 10 | |
| 培训成绩 | 培训表 | char | 10 | |
| 开始日期 | 培训表 | datetime | | |
| 结束日期 | 培训表 | datetime | | |
| 请假编号 | 请假记录表 | char | 6 | primary key |
| 请假时间 | 请假记录表 | datetime | | |
| 销假时间 | 请假记录表 | datetime | | |
| 请假天数 | 请假记录表 | int | | |
| 审批人编号 | 请假记录表 | char | 4 | |
| 登记编号 | 新员工表 | char | 10 | primary key |
| 职位 | 新员工表 | char | 10 | |
| 入职日期 | 新员工表 | datetime | | |
| 姓名 | 员工表 | char | 10 | not null |
| 性别 | 员工表 | char | 2 | |
| 出生日期 | 员工表 | dateime | | |
| 学历 | 员工表 | varchar | 32 | not null |
| 身份证号 | 员工表 | varchar | 32 | not null |
| 民族 | 员工表 | varchar | 16 | not null |
| 职称 | 员工表 | char | 10 | |
| 员工编号 | 员工表 | char | 10 | primary key |
| 调动前职务 | 员工表 | char | 10 | |
| 调动后职务 | 员工表 | char | 10 | |
| 调动部门号 | 部门表 | char | 10 | not null |
| 调动日期 | 职务调动表 | datetime | | |
| 批复日期 | 职务调动表 | datetime | | |
| 调动原因 | 职务调动表 | varchar | 50 | |
| 调动编号 | 职务调动表 | char | 10 | primary key |

3.2、系统 E-R 图

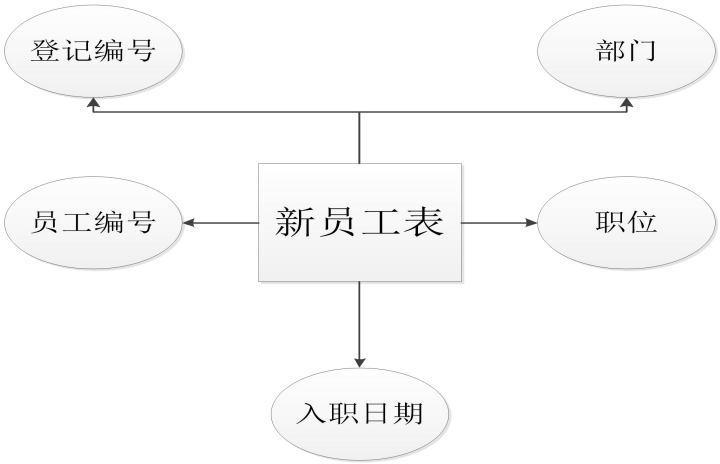


3.3、数据库 E-R 图

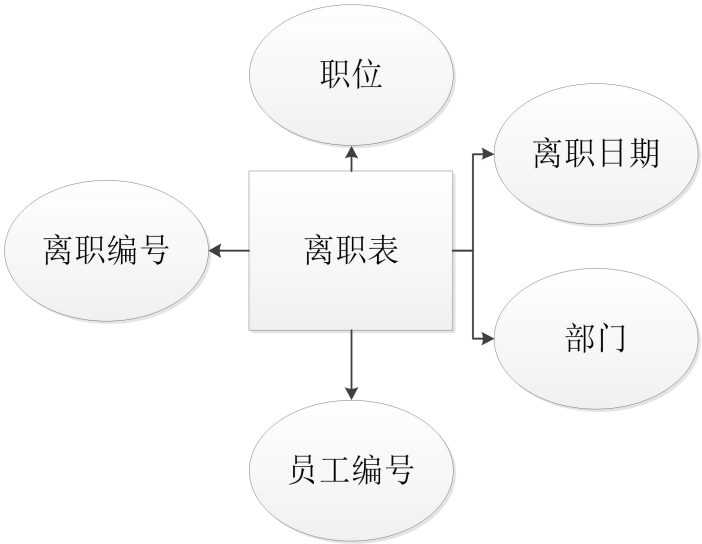
员工表：



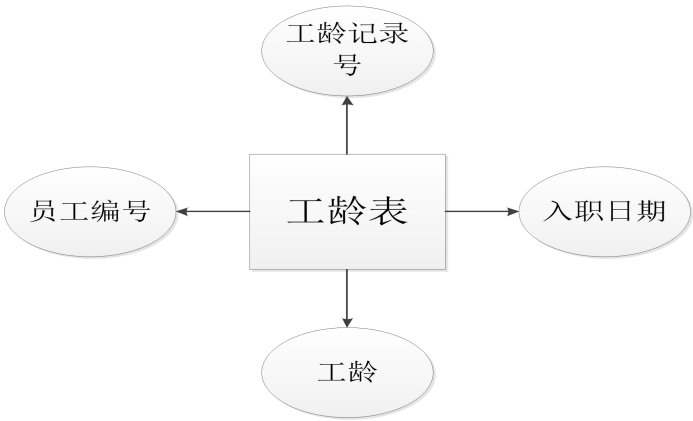
新员工表:



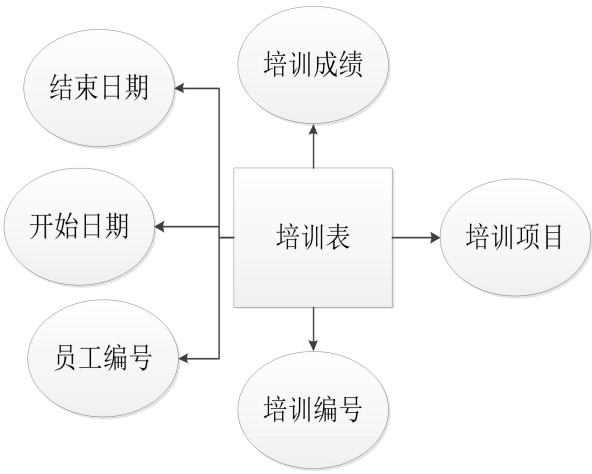
离职表:



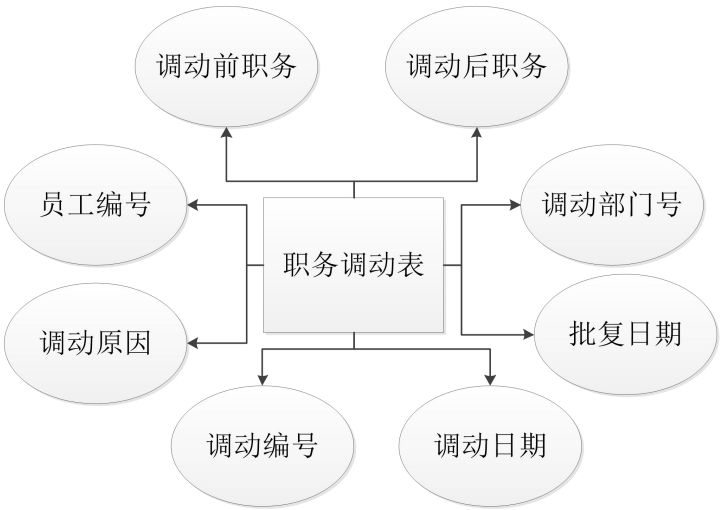
工龄表:



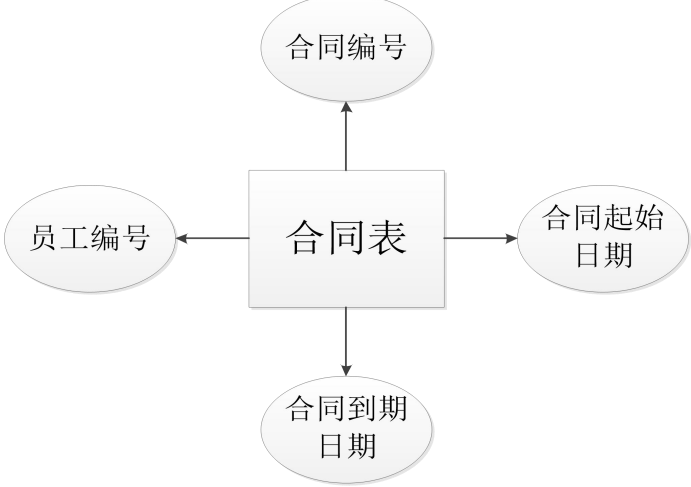
培训表：



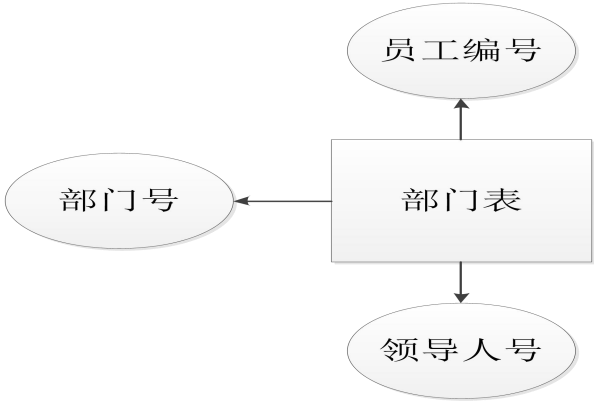
职务调动表：



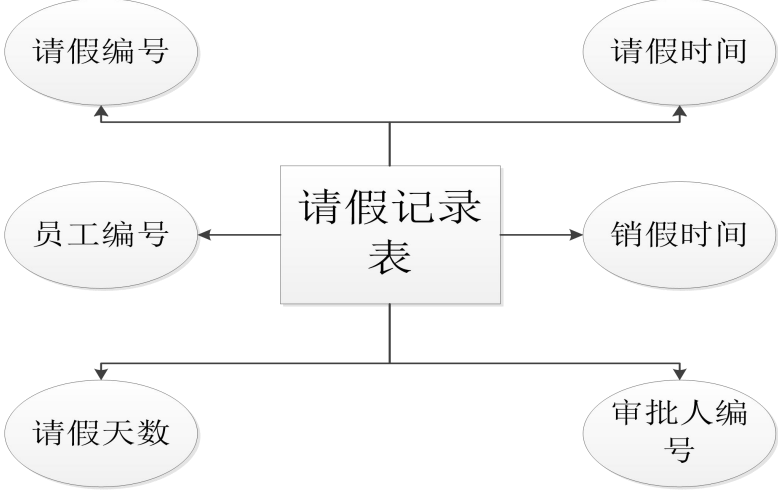
合同表：



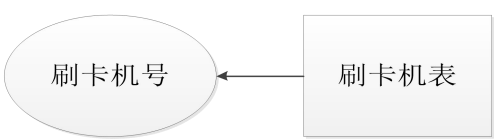
部门表:



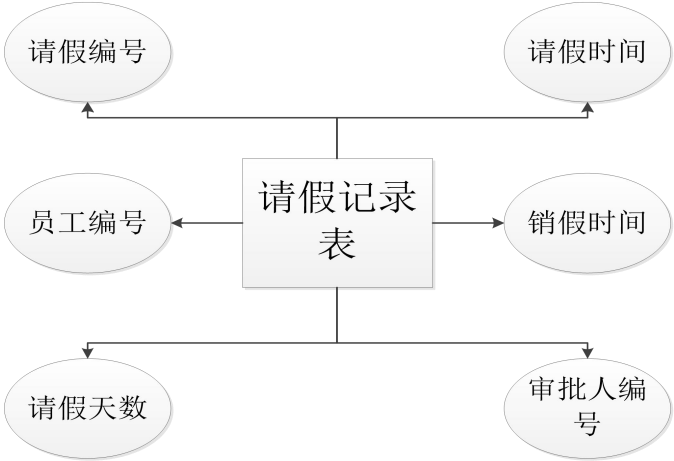
出勤表:



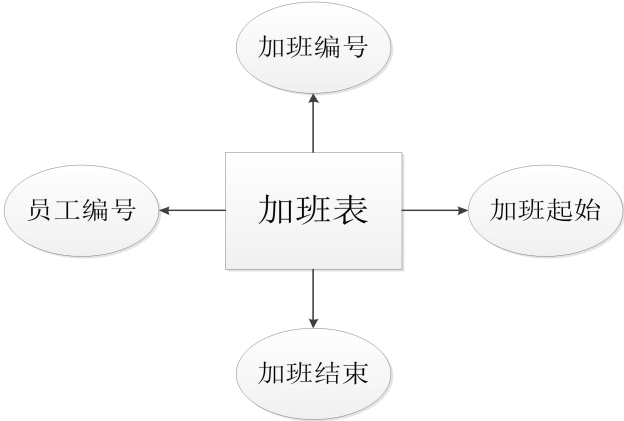
刷卡机表:



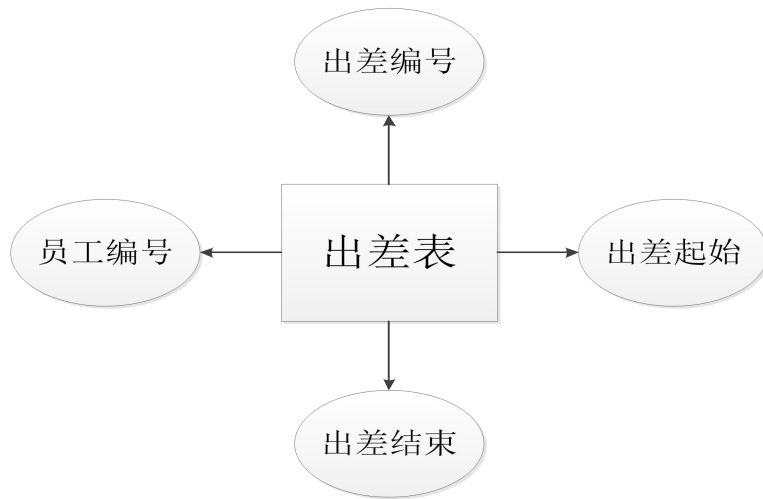
请假表：



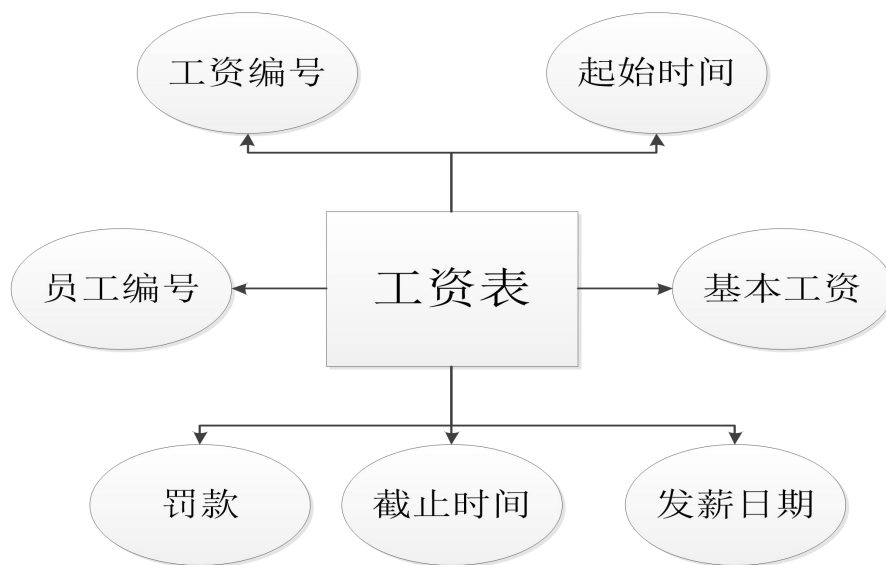
加班表：



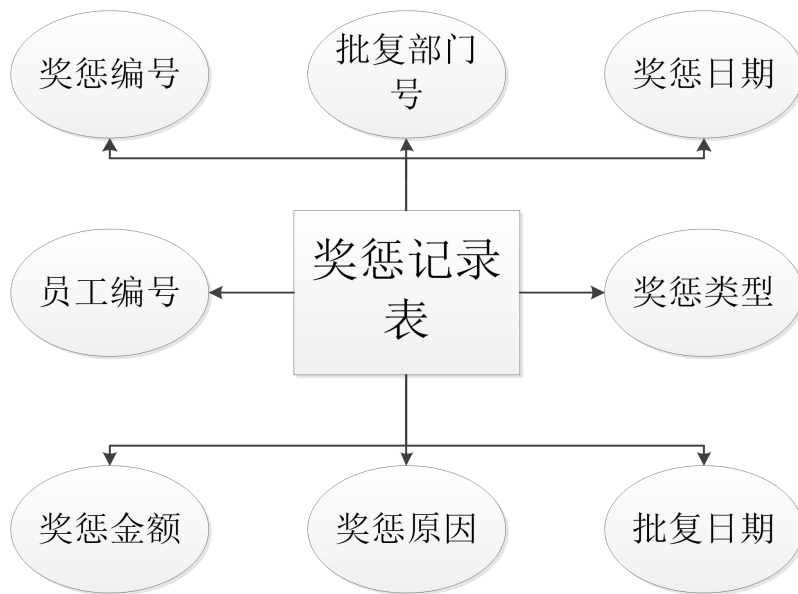
出差表：



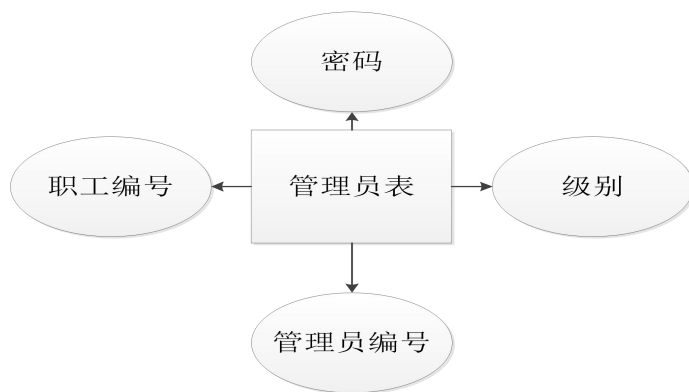
工资表:



奖惩表:



管理员表:



3.4、数据库逻辑设计

由 E-R 图，可得以下关系

部门表（部门号 部门名 领导人号）

出差表（出差编号 员工编号 出差起始 出差结束）

出勤表（出勤号 员工编号 上班日期 上班时 下班时间 刷卡机号 打卡次数）

工龄表（工龄记录号 员工编号 工龄 入职日期）

工资表（工资编号 员工编号 基本工资 罚款 起始时间 截止时间 发薪日期）

管理员表（管理员账号 职工编号 密码 级别）

合同表（合同编号 员工编号 合同起始日期 合同到期日期）

加班表（加班编号 员工编号 加班起始 加班结束）

奖惩记录表（奖惩编号 员工编号 奖惩类型 奖惩金额 奖惩日期 批复部门号

批复日期 奖惩原因)

离职表(离职编号 员工编号 部门 职位 离职日期)

培训表(培训编号 员工编号 培训项目 培训成绩 开始日期 结束日期)

请假记录表(请假编号 员工编号 请假时间 销假时间 请假天数 审批人编号)

刷卡机表(刷卡机号)

实习员工表(登记编号 员工编号 部门 职位 入职日期)

员工表(员工编号 姓名 性别 出生日期 学历 身份证号 民族 职称 部门号)

职务调动表(调动编号 员工编号 调动前职务 调动后职务 调动部门号 调动日期 批复日期 调动原因)

四、系统的实现与调试

4.1 表、数据库的建立

```
create database 人事管理系统
on
(name=人事管理系统 dat,
filename='E:\Study\SQL\mydata\人事管理系统 dat.mdf',
size=100,
maxsize=300,
filegrowth=10)
log on
(name=人事管理系统 log,
filename='E:\Study\SQL\mydata\人事管理系统 dat.ldf',
size=50,
maxsize=150,
filegrowth=10%)
```

4.2 建立基本表

4.2.1 建立员工表

```
create table 员工表(
员工编号 char(10) primary key,
姓名 char(10) not null,
性别 char(2) check(性别 in('男','女')) not null,
出生日期 datetime not null,
```


学历 varchar (32)
 身份证号 varchar(32) not null,
 民族 varchar(16) default '汉',
 职位 char(10),
 部门号 char(10),
);

| | 列名 | 数据类型 | 允许 Null 值 |
|---|------|-------------|-------------------------------------|
| ▶ | 员工编号 | char(10) | <input type="checkbox"/> |
| | 姓名 | char(10) | <input type="checkbox"/> |
| | 性别 | char(2) | <input checked="" type="checkbox"/> |
| | 出生日期 | datetime | <input checked="" type="checkbox"/> |
| | 学历 | varchar(32) | <input type="checkbox"/> |
| | 身份证号 | varchar(32) | <input type="checkbox"/> |
| | 民族 | varchar(16) | <input type="checkbox"/> |
| | 职称 | char(10) | <input checked="" type="checkbox"/> |
| | 部门号 | char(10) | <input checked="" type="checkbox"/> |

4.2.2 建立实习员工表:

create table 新员工表(
 登记编号 char(10) primary key,
 员工编号 char(10),
 部门号 char(10),
 职位 char(10),
 入职日期 datetime,
 foreign key(员工编号) references 员工表(员工编号)
);


| | 列名 | 数据类型 | 允许 Null 值 |
|---|------|----------|-------------------------------------|
| ▶ | 登记编号 | char(10) | <input type="checkbox"/> |
| | 员工编号 | char(10) | <input checked="" type="checkbox"/> |
| | 部门号 | char(10) | <input checked="" type="checkbox"/> |
| | 职位 | char(10) | <input checked="" type="checkbox"/> |
| | 入职日期 | datetime | <input checked="" type="checkbox"/> |
| | | | <input type="checkbox"/> |

4.2.3 建立离职表:

```

create table 离职表(
    离职编号 char(10) primary key,
    员工编号 char(10),
    部门 char(10),
    职位 char(10),
    离职日期 char(10),
    离职原因 nvarchar(50),
    foreign key(员工编号) references 员工表(员工编号)
);

```


| | 列名 | 数据类型 | 允许 Null 值 |
|---|------|--------------|-------------------------------------|
|  | 离职编号 | char(10) | <input type="checkbox"/> |
| | 员工编号 | char(10) | <input checked="" type="checkbox"/> |
| | 部门 | char(10) | <input checked="" type="checkbox"/> |
| | 职位 | char(10) | <input checked="" type="checkbox"/> |
| | 离职日期 | char(10) | <input checked="" type="checkbox"/> |
| | 离职原因 | nvarchar(50) | <input checked="" type="checkbox"/> |
| | | | <input type="checkbox"/> |

4.2.4 建立职务调动表:

```


create table 职务调动表(
    调动编号 char(10) primary key,
    员工编号 char(10),
    调动前职称 char(10),
    调动后职称 char(10),
    调后部门号 char(10),
    批复日期 datetime,
    调动日期 datetime,
    调动原因 char(50),
    foreign key(员工编号) references 员工表(员工编号)
);

```

| | 列名 | 数据类型 | 允许 Null 值 |
|---|--------|-------------|-------------------------------------|
|  | 调动编号 | char(10) | <input type="checkbox"/> |
| | 员工编号 | char(10) | <input type="checkbox"/> |
| | 调动前职称 | char(10) | <input checked="" type="checkbox"/> |
| | 调动后职称 | char(10) | <input checked="" type="checkbox"/> |
| | 调动后部门号 | char(10) | <input type="checkbox"/> |
| | 调动日期 | datetime | <input checked="" type="checkbox"/> |
| | 批复日期 | datetime | <input checked="" type="checkbox"/> |
| | 调动原因 | varchar(50) | <input checked="" type="checkbox"/> |

4.2.5 建立合同表:

```
create table 合同表(
    合同编号 char(10) primary key,
    员工编号 char(10),
    合同起始 datetime,
    合同到期 datetime,
    foreign key(员工编号) references 员工表(员工编号)
);
```

| | 列名 | 数据类型 | 允许 Null 值 |
|---|------|----------|-------------------------------------|
|  | 合同编号 | char(10) | <input type="checkbox"/> |
| | 员工编号 | char(10) | <input checked="" type="checkbox"/> |
| | 合同起始 | datetime | <input checked="" type="checkbox"/> |
| | 合同到期 | datetime | <input checked="" type="checkbox"/> |
| | | | <input type="checkbox"/> |

4.2.6 建立奖惩表:

```
create table 奖惩记录表(
    奖惩编号 char(6) primary key,
    员工编号 char(10),
    奖惩类型 char(4),
    奖惩金额 money,
    奖惩日期 datetime,
    批复部门号 char(10),
    批复日期 datetime,
    奖惩原因 varchar(50),
```

```
foreign key(员工编号) references 员工表(员工编号)
);
```

| | 列名 | 数据类型 | 允许 Null 值 |
|---|-------|-------------|-------------------------------------|
| 🔑 | 奖惩编号 | char(6) | <input type="checkbox"/> |
| | 员工编号 | char(10) | <input type="checkbox"/> |
| | 奖惩类型 | char(4) | <input checked="" type="checkbox"/> |
| | 奖惩金额 | money | <input checked="" type="checkbox"/> |
| | 奖惩日期 | datetime | <input checked="" type="checkbox"/> |
| | 批复部门号 | char(10) | <input checked="" type="checkbox"/> |
| | 批复日期 | datetime | <input checked="" type="checkbox"/> |
| | 奖惩原因 | varchar(50) | <input checked="" type="checkbox"/> |

4.2.7 建立加班表:

```
create table 加班表(
加班编号 char(10) primary key,
员工编号 char(10),
加班起始 datetime,
加班结束 datetime,
foreign key(员工编号) references 员工表(员工编号)
);
```

| | 列名 | 数据类型 | 允许 Null 值 |
|---|------|----------|-------------------------------------|
| 🔑 | 加班编号 | char(10) | <input type="checkbox"/> |
| | 员工编号 | char(10) | <input checked="" type="checkbox"/> |
| | 加班起始 | datetime | <input checked="" type="checkbox"/> |
| | 加班结束 | datetime | <input checked="" type="checkbox"/> |
| | | | <input type="checkbox"/> |

4.2.8 建立出差表:

```
create table 出差表(
出差编号 char(10) primary key,
员工编号 char(10),
出差起始 datetime,
出差结束 datetime,
foreign key(员工编号) references 员工表(员工编号)
);
```

| | 列名 | 数据类型 | 允许 Null 值 |
|---|------|----------|-------------------------------------|
| ▶ | 出差编号 | char(10) | <input type="checkbox"/> |
| | 员工编号 | char(10) | <input checked="" type="checkbox"/> |
| | 出差起始 | char(10) | <input checked="" type="checkbox"/> |
| | 出差结束 | char(10) | <input checked="" type="checkbox"/> |
| | | | <input type="checkbox"/> |

4.2.9 建立出勤表:

```
create table 出勤表(
  出勤编号 char(10) primary key,
  员工编号 char(10),
  上班日期 datetime,
  上班时间 datetime,
  下班时间 datetime,
  刷卡机号 char(4),
  打卡次数 nchar(10),
  foreign key(员工编号) references 员工表(员工编号)
);
```

| | 列名 | 数据类型 | 允许 Null 值 |
|---|------|-----------|-------------------------------------|
| ▶ | 出勤号 | char(6) | <input type="checkbox"/> |
| | 员工编号 | char(10) | <input type="checkbox"/> |
| | 上班日期 | datetime | <input type="checkbox"/> |
| | 上班时间 | datetime | <input type="checkbox"/> |
| | 下班时间 | datetime | <input type="checkbox"/> |
| | 刷卡机号 | char(4) | <input checked="" type="checkbox"/> |
| | 打卡次数 | nchar(10) | <input checked="" type="checkbox"/> |
| | | | <input type="checkbox"/> |

4.2.10 建立工资表:

```
create table 工资表(
  工资编号 char(6) primary key,
  员工编号 char(10),
  基本工资 money,
  实发工资 money,
  起始时间 datetime,
  截止时间 datetime,
```

发薪日期 datetime,
foreign key(员工编号) references 员工表(员工编号)
);

| | 列名 | 数据类型 | 允许 Null 值 |
|------|----|----------|-------------------------------------|
| 工资编号 | | char(6) | <input type="checkbox"/> |
| 员工编号 | | char(10) | <input type="checkbox"/> |
| 基本工资 | | money | <input type="checkbox"/> |
| 实发工资 | | money | <input checked="" type="checkbox"/> |
| 起始时间 | | datetime | <input checked="" type="checkbox"/> |
| 截止时间 | | datetime | <input checked="" type="checkbox"/> |
| 发薪日期 | | datetime | <input checked="" type="checkbox"/> |
| | | | <input type="checkbox"/> |

4.2.11 建立管理人员表:

create table 管理人员(
管理员编号 char(10) primary key,
员工编号 char(10),
密码 char(10),
级别 char(10),
);

| | 列名 | 数据类型 | 允许 Null 值 |
|-------|----|----------|-------------------------------------|
| 管理员账号 | | char(10) | <input type="checkbox"/> |
| 职工编号 | | char(10) | <input type="checkbox"/> |
| 密码 | | char(10) | <input type="checkbox"/> |
| 级别 | | char(10) | <input checked="" type="checkbox"/> |
| | | | <input type="checkbox"/> |

4.2.12 建立工龄表

create table 工龄表(
工龄记录号 char(10) primary key,
员工编号 char(10),
工龄 char(10),
入职日期 datetime,
foreign key(员工编号) references 员工表(员工编号)
);

| | 列名 | 数据类型 | 允许 Null 值 |
|---|-------|----------|-------------------------------------|
| ▶ | 工龄记录号 | char(10) | <input type="checkbox"/> |
| | 员工编号 | char(10) | <input checked="" type="checkbox"/> |
| | 工龄 | char(10) | <input checked="" type="checkbox"/> |
| | 入职日期 | datetime | <input checked="" type="checkbox"/> |
| | | | <input type="checkbox"/> |

4.2.13 建立培训表

```
create table 培训表(
    培训编号 char(10) primary key,
    员工编号 char(10),
    培训项目 char(10),
    培训成绩 char(10),
    开始时间 datetime,
    结束时间 datetime,
    foreign key(员工编号) references 员工表(员工编号)
);
```

| | 列名 | 数据类型 | 允许 Null 值 |
|---|------|----------|-------------------------------------|
| ▶ | 培训编号 | char(10) | <input type="checkbox"/> |
| | 员工编号 | char(10) | <input checked="" type="checkbox"/> |
| | 培训项目 | char(10) | <input checked="" type="checkbox"/> |
| | 培训成绩 | char(10) | <input checked="" type="checkbox"/> |
| | 开始时间 | datetime | <input checked="" type="checkbox"/> |
| | 结束时间 | datetime | <input checked="" type="checkbox"/> |
| | | | <input type="checkbox"/> |

4.2.14 建立请假记录表

```
create table 请假记录表(
    请假编号 char(6) primary key,
    员工编号 char(10),
    请假时间 datetime,
    销假时间 datetime,
    请假天数 int,
    审批人编号 char(4),
```

```
foreign key(员工编号) references 员工表(员工编号)
);
```

| | 列名 | 数据类型 | 允许 Null 值 |
|---|-------|----------|-------------------------------------|
| 🔑 | 请假编号 | char(6) | <input type="checkbox"/> |
| | 员工编号 | char(10) | <input type="checkbox"/> |
| | 请假时间 | datetime | <input checked="" type="checkbox"/> |
| | 销假时间 | datetime | <input checked="" type="checkbox"/> |
| | 请假天数 | int | <input checked="" type="checkbox"/> |
| | 审批人编号 | char(4) | <input checked="" type="checkbox"/> |
| | | | <input type="checkbox"/> |

4. 2. 15 建立部门表

```
create table 部门表(
    部门名 char(10) primary key,
    部门号 char(10),
    领导人号 char(10),
    foreign key(员工编号) references 员工表(员工编号)
);
```

| | 列名 | 数据类型 | 允许 Null 值 |
|---|------|----------|--------------------------|
| 🔑 | 部门号 | char(10) | <input type="checkbox"/> |
| | 部门名 | char(10) | <input type="checkbox"/> |
| | 领导人号 | char(10) | <input type="checkbox"/> |
| | | | <input type="checkbox"/> |

4. 2. 16 索引的建立

```
Go
/*索引的建立*/
create index 部门_领导人号_index on 部门(领导人号)
create index 出勤_学历_index on 出勤(学历 )
create index 出勤_出生日期_index on 员工(出生日期 )
create index 出勤_上班日期_index on 出勤(上班日期 )
create index 工资_基本工资_index on 工资(基本工资 )
create index 职务调动_调动前职务_index on 职务调动(调动前职务 )
create index 奖惩记录_奖惩类型_index on 奖惩记录(奖惩类型 )
```



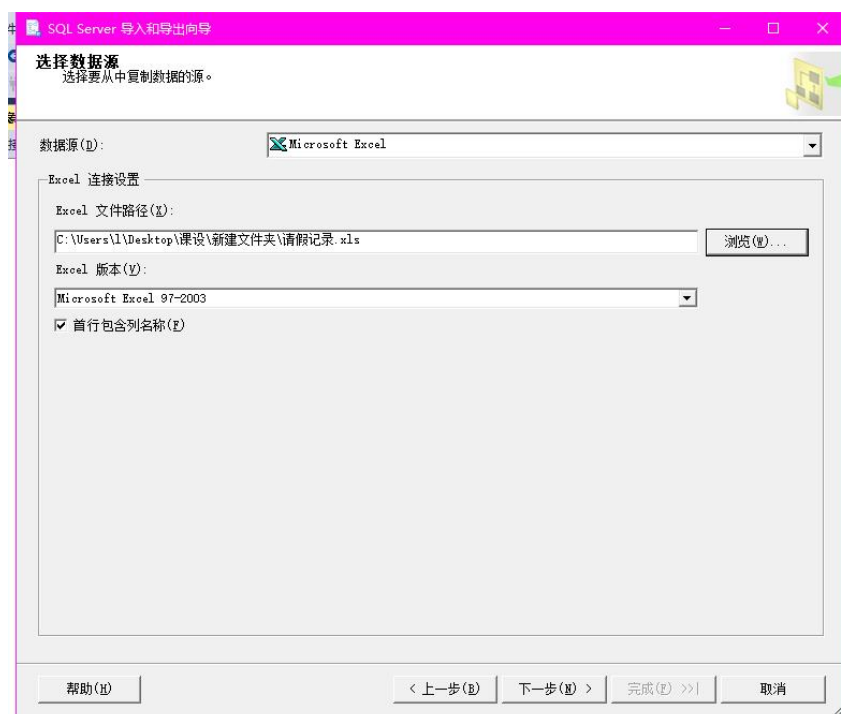
```
create index 请假记录_请假天数_index on 请假记录(请假天数)
Go
```

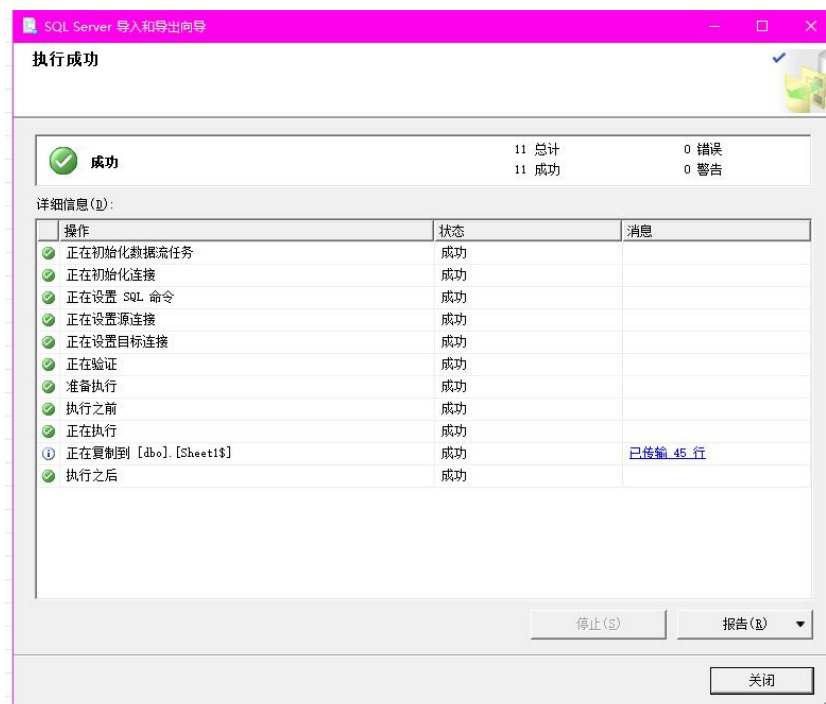
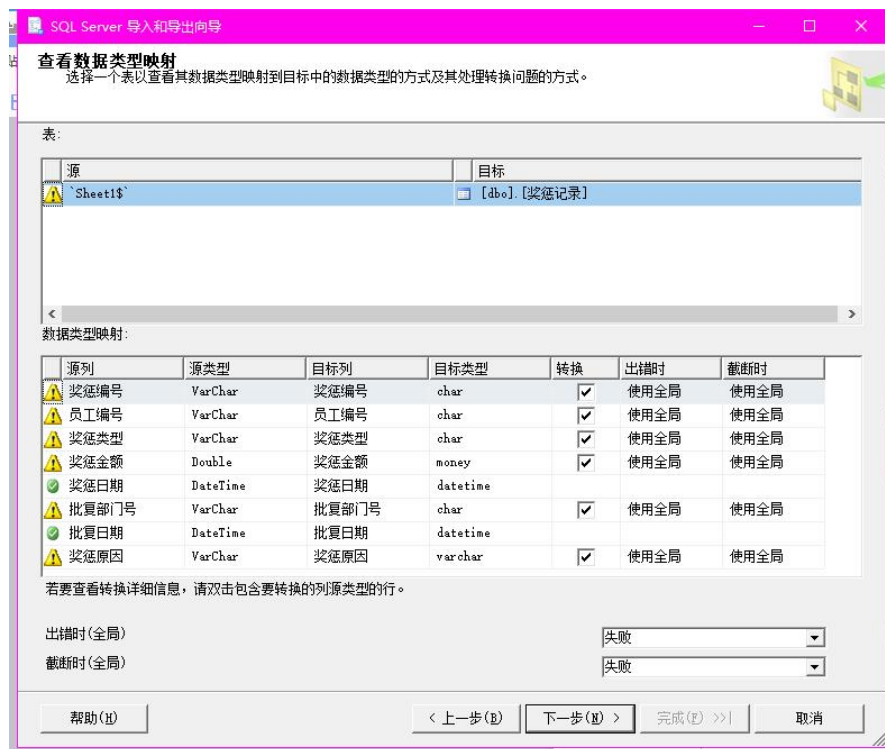
4.2.17 视图的建立

(1) 建立一个视图，反映员工姓名及工资情况。

```
create view 员工_工资
as
select 员工.姓名, 工资.*
from 员工,工资
where 员工.员工编号=工资表.员工编号
```

4.2.18 Excel 与数据库信息的迁移





4.3 程序功能与主要代码内容

4.3.1 登录界面用户名和密码与数据库建立联系

```
string username = textBox1.Text.Trim();
string userpwd = textBox2.Text.Trim();
```

```

string sqlStr = "server=MK96;database=人事管理系
统;Trusted_Connection=yes";//连接数据库字符串
uid=mk96;password=mengkuncurry
SqlConnection con = new SqlConnection(sqlStr);//连接数据库
con.Open(); //打开连接
string cmdStr = "select * from 员工 where 员工编号=' " +
username + "' and 员工编号=' " + userpwd + "'";//定义查询语
句
SqlCommand sqlCom = new SqlCommand(cmdStr, con);//定义查询
命令

SqlDataReader sRead = sqlCom.ExecuteReader();//执行查

```

//*****进行权限的判断，如果在数据库管理员中能够读取得到信息就说明该用户具有管理权限，窗体会自动进入到管理员界面，如果查询不到则会进去到普通用户的界面*****//

```

if (sRead.Read())//if(sRead.Read())的意思是，如果能查找到
{
    sRead.Close();
    string cmdStr1 = "select * from 管理人员 where 职工编
号=' " + username + "' and 职工编号=' " + userpwd + "'";//定义查询语句
    SqlCommand sqlCom1 = new SqlCommand(cmdStr1, con);//
定义查询命令
    SqlDataReader sRead1 = sqlCom1.ExecuteReader();//执行
查询
    if (sRead1.Read())//if(sRead1.Read())的意思是，如果能查
找到
    {
        MessageBox.Show("欢迎来到管理员界面");
        主页面 f = new 主页面(textBox1.Text,
textBox2.Text);
        f.Show();
        this.Hide();
    }
}

```



管理员界面

```

else
{
    MessageBox.Show("欢迎来到用户界面");
    用户界面 m=new 用户界面();
    m.Show();
    this.Hide();
}

```



用户界面

```
}
```

```
else
```

```
{
```

```
    MessageBox.Show("用户名与密码不匹配，请重新输入!");
```

```
}
```

```
}
```

4.3.2 职务调动功能

//职务调动功能,将数据更新到职务调动的表里然后员工表的信息也会随之更新
//

```
SqlCommand cmd = new SqlCommand(sql, con);  
string sql1 = "UPDATE 员工 SET 员工.部门号 = 职务调动.调动后部门号 , 员工.职称 =  
职务调动.调动后职称 FROM 员工 , 职务调动 WHERE 员工.员工编号 = 职务调动.员工编  
号";  
  
SqlCommand cmd1 = new SqlCommand(sql1, con);  
//cmd.ExecuteNonQuery();  
cmd.ExecuteNonQuery();  
int count = cmd1.ExecuteNonQuery();  
  
if (count > 0)  
{  
    MessageBox.Show("亲, 数据更新成功!");  
}  
else  
{  
    MessageBox.Show("亲, 数据更新失败!");  
}  
}
```

The screenshot shows a Windows application window titled '人事调动' (Personnel Transfer). The form contains several input fields and dropdown menus for recording a personnel transfer. A modal dialog box is displayed in the center, showing a success message: '亲, 数据更新成功!' (Hey, data update successful!).

| Field | Value |
|----------------------------|------------------------|
| 调动编号 (Transfer ID) | dd03 |
| 员工编号 (Employee ID) | 0001 |
| 调动前职务 (Previous Position) | 普通员工 (Common Employee) |
| 调动后职务 (New Position) | 总经理 (General Manager) |
| 调动部门号 (New Department ID) | B004 |
| 调动日期 (Transfer Date) | 2019-03-02 |
| 批复时间 (Approval Time) | 2019-03-02 |
| 调动原因 (Reason for Transfer) | 公司 (Company) |

Buttons at the bottom: 提交 (Submit) and 返回 (Return). The dialog box has a 确定 (Confirm) button.

职务调动

```
//*****信息的查询*****//
```

```
string cmdStr = "select * from 员工 where 员工编号='" + a + "' and 姓名='" + b + "' and 身份证号='" + c + "'"; //定义查询语句
SqlCommand sqlCom = new SqlCommand(cmdStr, con); //定义查询命令
//SqlDataReader sRead = sqlCom.ExecuteReader(); //执行查询
System.Data.SqlClient.SqlDataAdapter adapter = new
SqlDataAdapter(cmdStr, con);
DataSet ds = new DataSet();
adapter.Fill(ds);
DataTable dt = ds.Tables[0];
dataGridView1.DataSource = dt;
```

```
//*****信息的删除*****//
```

```
con.Open(); //打开连接
string select_id =
dataGridView1.SelectedRows[0].Cells[0].Value.ToString(); //选择的当前行第一列的值，也就是ID
string delete_by_id = "delete from 员工 where 员工编号='" +
select_id.Trim() + "'"; //sql 删除语句
SqlCommand cmd = new SqlCommand(delete_by_id, con);
cmd.ExecuteNonQuery();
MessageBox.Show("删除成功!");
信息浏览 n = new 信息浏览();
n.Show();
this.Hide();
```

```
//*****信息的添加*****//
```

```
SqlDataAdapter dataAdapter = new SqlDataAdapter("SELECT * FROM 员工 ", con);
SqlCommandBuilder cb = new SqlCommandBuilder(dataAdapter);
DataSet ds = new DataSet();
dataAdapter.Fill(ds, "员工");
DataTable tb = ds.Tables["员工"];

object[] o = new object[9] { this.textBox1.Text.TrimEnd(),
this.textBox2.Text.TrimEnd(), this.comboBox1.Text.TrimEnd(),
this.textBox4.Text.TrimEnd(), this.comboBox2.Text.TrimEnd(),
this.textBox6.Text.TrimEnd(), this.textBox7.Text.TrimEnd(),
this.comboBox3.Text.TrimEnd(), this.comboBox4.Text.TrimEnd() };
tb.Rows.Add(o);
dataAdapter.Update(ds, "员工");
ds.AcceptChanges();
```

五、问题与解决

在实验中期，我们过于着重代码和模板的设计，导致在数据库方面的关系、建表和完善程度不够。

对于人事管理系统，数据库中表与表之间的关系错综复杂，每一部分都应该环环相扣、相互关联。但我们忽略了完善数据库表的重要性，仅在数据库中实现了一张表，且关系不够密切。

沙芸老师指点我们要完善表与表之间的关系，数据库的内容是庞大且复杂的，要将重点放在数据库表的完善上。

经过老师的指点和简单举例，我们找到了错误，将课程设计的重点从代码界面的设计完成，转变为对于数据库表的建立和表与表之间关系的完善。

期初我们将员工信息、部门信息、工资等等详细信息汇总在了一张表中。经过改进，我们将各种信息细分，将数据库扩展到 16 张表，并且每张表之间的关系都有所体现。不仅完善了数据库，还在代码上完成了数据库的增删改查，有相同属性的几张表，只更改一张表就可以完成整个数据库的更改。

六、思考与总结

由于在数据库的设计初期，没有对人事管理系统进行准确地需求分析，导致后期设计过程中的数据库的概念设计阶段无法进行。此次数据库课程设计使我们认识到了需求分析对于软件系统设计的重要性。也认识到规范化的设计流程对于后期的设计来说，是十分重要的。前面的设计是后面的铺垫，只有对前面的设计过程进行准确地分析，才能节省整个系统的开发时间。

通过此次人事资源管理系统的数据库的课程设计，真正达到了学与用的结合，增强了我们对数据库方面应用的理解，对自己今后参与开发数据库系统积累了不少经验，这次的课程设计，让我们明白学习是一个长期积累的过程，在以后的工作、生活中都应该不断的学习，努力提升自己知识和综合素质。2 周的设计使我们的关系更进一步了，有什么不懂的大家在一起商量，听听不同的看法，从中更好的理解知识，我觉得这样的课程设计特别有价值和实践意义。

在实验过程中，从建立数据开始，对于数据库设计理念及思想上有更高的认识，从需求分析，到概念设计和逻辑设计，E-R 图的表示，数据流图的创建，懂得了

不少有关数据库开发过程中的知识，在实验中建表，及其关系模式，关系代数的建立及理解，增强了自己数据库中的应用 SQL 语言的灵活性，在学习过程中，我也能过上网查了不少资料，也看了一些别人设计的人事资源管理信息系统的设计报告，学以致用，完成了这份自己的报告，从中在学到用，从用又到学，不断修改，系统更新。虽然不能达到完善系统，但也做到了尽善尽美，加强理论学习对完善系统会有很多帮助。

七、参考文献

- [1]何玉洁. 数据库原理与应用. 第三版. 机械工业出版社
- [2]唐大. C#程序设计教程. 清华大学出版社