

## 《数据库系统》课程设计报告

题目	企业人事管理系统		
小组成员信息			
姓名	学号	班级	分工
王子雄	17343111	教务 2 班	全部内容

提交时间： 2019 年 6 月 24 日

### 一. 开发环境与开发工具

Windows 10

Visual Studio 2013、MySQL

(VS 需要对环境进行配置, 可参考

[https://blog.csdn.net/qq\\_21095573/article/details/82824693](https://blog.csdn.net/qq_21095573/article/details/82824693))

### 二. 系统需求分析(2 分)

员工信息表

列名	数据类型	长度	是否允许为空
员工编号	int		N
姓名	char	50	N
性别	char	20	N
政治面貌	char	100	Y

加班记录表

列名	数据类型	长度	是否允许为空
加班编号	int		N
时长	int		N

出差记录表

列名	数据类型	长度	是否允许为空
员工编号	int		N
天数	int		N

### 三. 功能需求分析 (3 分)

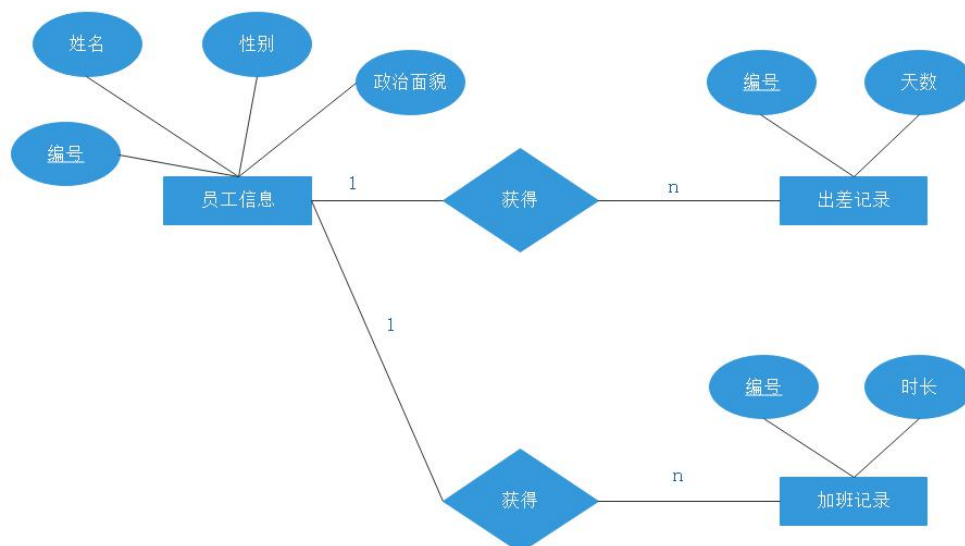
企业人事管理系统具有如下功能:

- (1) 人事变动管理: 包括新员工登记、员工离职登记等;
- (2) 人事档案管理: 包括员工基本信息的查询、修改等;
- (3) 加班出差管理: 包括员工加班出差信息的查询、添加、删除、修改等。



#### 四. 系统设计（10 分）

数据概念结构设计（系统 ER 图）



数据库关系模式设计

- (1) 员工信息（员工编号，姓名，性别，政治面貌）
- (2) 加班记录（加班编号，时长）
- (3) 出差记录（出差编号，天数）

(4) 员工获得加班 ( 员工编号, 加班编号 )

(5) 员工获得出差 ( 员工编号, 出差编号 )

数据库物理结构设计

```
create database employeeIMS;
```

```
use employeeIMS;
```

```
create table employee(
```

```
    eno integer,
```

```
    ena char(50) not null,
```

```
    sex char(20) not null,
```

```
    epo char(100),
```

```
    primary key(eno)
```

```
);
```

```
alter table employee change ena ena char(50) character set utf8 collate utf8_croatian_ci;
```

```
alter table employee change sex sex char(20) character set utf8 collate utf8_croatian_ci;
```

```
alter table employee change epo epo char(100) character set utf8 collate utf8_croatian_ci;
```

```
create table overtime(
```

```
    ono integer,
```

```
    otime integer,
```

```
    primary key(ono)
```

```
);
```

```
create table business(
```

```
    bno integer,
```

```
    bday integer,
```

```
    primary key(bno)
```

```
);
```

```
create table emp_over(
```

```
    eno integer,
```

```
    ono integer,
```

```
    foreign key(eno) references employee(en),
```

```
    foreign key(ono) references overtime(ono)
```

```
    on delete cascade
```

```
);
```

```
create table emp_bus(
```

```
    eno integer,
```

```
    bno integer,
```

```
    foreign key(en) references employee(en),
```

```
    foreign key(bno) references business(bno)
```

```
    on delete cascade
```

);

##### 五. 系统功能的实现 (5 分)

界面采用命令行格式

主界面

```
连接数据库成功

0--退出
1--人事变动管理
2--人事档案管理
3--加班出差管理
请输入您的选择:
```

人事变动管理界面

```
0--返回
1--新员工登记
2--离职员工登记
请输入您的选择: _
```

新员工登记界面及操作

```
<连接数据库成功
<
<0--退出
<1--人事变动管理
<2--人事档案管理
<3--加班出差管理
请输入您的选择: 1
n
0--返回
s1--新员工登记
2--离职员工登记
n请输入您的选择: 1
n
n请输入新员工的编号: 002
'e请输入新员工的姓名: 陈
i请输入新员工的性别: 男
请输入新员工的政治面貌: 党员
f新员工登记成功
r
```

新员工登记函数实现（使用 insert 在数据库表 employee 中插入新员工）

```

92 void newemployee()
93 {
94     char eno[] = "";
95     char ena[] = "";
96     char sex[] = "";
97     char epo[] = "";
98     char strquery[100] = "insert into employee (eno,ena,sex,epo) values('";
99     cout << "\n请输入新员工的编号: ";
100    cin >> eno;
101    strcat(strquery, eno);
102    strcat(strquery, ",");
103    cout << "请输入新员工的姓名: ";
104    cin >> ena;
105    strcat(strquery, ena);
106    strcat(strquery, ",");
107    cout << "请输入新员工的性别: ";
108    cin >> sex;
109    strcat(strquery, sex);
110    strcat(strquery, "');");
111    cout << "请输入新员工的政治面貌: ";
112    cin >> epo;
113    strcat(strquery, epo);
114    strcat(strquery, "');");
115    if (mysql_query(&mysql, strquery) == 0) {
116        cout << "新员工登记成功" << endl;
117    }
118    else {
119        cout << "新员工登记失败" << endl;
120    }
121    changmanu();
122 }
123

```

## 离职员工登记界面及操作

```

连接数据库成功
0--退出
1--人事变动管理
2--人事档案管理
3--加班出差管理
v 请输入您的选择: 1
{
0--返回
1--新员工登记
2--离职员工登记
请输入您的选择: 2

请输入离职员工的编号: 001
离职员工登记成功

```

离职员工函数实现（使用 delete 删去数据库表 employee 中对应的员工）

```

124 void resignemployee()
125 {
126     char eno[] = "";
127     char strquery[100] = "delete from employee where eno = ";
128     cout << "\n请输入离职员工的编号: ";
129     cin >> eno;
130     strcat(strquery, eno);
131     if (mysql_query(&mysql, strquery) == 0) {
132         cout << "离职员工登记成功" << endl;
133     }
134     else {
135         cout << "离职员工登记失败" << endl;
136     }
137     changmanu();
138 }
139

```

## 人事档案管理界面

```
0--返回
1--查询员工信息
2--修改员工信息
请输入您的选择:
```

查询员工信息界面及操作（可查询到 002 号员工，已查询不到在上面演示中离职的 001 号员工）

```
连接数据库成功
0--退出
1--人事变动管理
2--人事档案管理
3--加班出差管理
请输入您的选择: 2

0--返回
1--查询员工信息
2--修改员工信息
请输入您的选择: 1

请输入需要查询员工的编号: 002
员工编号: 2
员工姓名: 陈
员工性别: 男
政治面貌: 党员

0--返回
1--查询员工信息
2--修改员工信息
请输入您的选择:
```

```

连接数据库成功

0--退出
1--人事变动管理
2--人事档案管理
3--加班出差管理
请输入您的选择: 2

0--返回
1--查询员工信息
2--修改员工信息
请输入您的选择: 1

请输入需要查询员工的编号: 001
该员工不存在

0--返回
1--查询员工信息
2--修改员工信息
请输入您的选择: _

```

查询员工信息函数实现（使用 select 在 employee 中查询对应的员工信息）

```

void queryinf()
{
    MYSQL_RES *result;
    MYSQL_ROW row;
    char eno[] = "";
    char strquery[100] = "select eno, ena, sex, epo from employee where eno = ";
    cout << "\n请输入需要查询员工的编号: ";
    cin >> eno;
    strcat(strquery, eno);
    if (mysql_query(&mysql, strquery) == 0) {
        result = mysql_store_result(&mysql);
        if (result->row_count != 0) {
            while (row = mysql_fetch_row(result)) {
                cout << "员工编号: " << row[0] << endl;
                cout << "员工姓名: " << row[1] << endl;
                cout << "员工性别: " << row[2] << endl;
                cout << "政治面貌: " << row[3] << endl;
            }
        }
        else {
            cout << "该员工不存在" << endl;
        }
    }
    profilemanu();
}

```

修改员工信息界面及操作

```

连接数据库成功

0--退出
1--人事变动管理
2--人事档案管理
3--加班出差管理
请输入您的选择：2

0--返回
1--查询员工信息
2--修改员工信息
请输入您的选择：2

请输入需要修改员工的编号：002
请修改员工的姓名：王
请修改员工的性别：女
请修改员工的政治面貌：团员
员工修改信息成功

0--返回
1--查询员工信息
2--修改员工信息
请输入您的选择：

```

修改员工信息函数实现（使用 update，修改 employee 中对应编号员工的信息）



```

void updateinf()
{
    char eno[] = "";
    char ena[] = "";
    char sex[] = "";
    char epo[] = "";
    char strquery[100] = "update employee ";
    cout << "\n请输入需要修改员工的编号: ";
    cin >> eno;
    strcat(strquery, "set ena = '");
    cout << "请修改员工的姓名: ";
    cin >> ena;
    strcat(strquery, ena);
    strcat(strquery, "', sex = '");
    cout << "请修改员工的性别: ";
    cin >> sex;
    strcat(strquery, sex);
    strcat(strquery, "', epo = '");
    cout << "请修改员工的政治面貌: ";
    cin >> epo;
    strcat(strquery, epo);
    strcat(strquery, "'");
    strcat(strquery, "where eno = ");
    strcat(strquery, eno);
    if (mysql_query(&mysql, strquery) == 0) {
        cout << "员工修改信息成功" << endl;
    }
    else {
        cout << "员工修改信息失败" << endl;
    }
    profilemanu();
}

```

加班出差管理界面

```

0--返回
1--加班信息管理
2--出差信息管理
请输入您的选择:

```

加班信息管理界面

```

0--返回
1--加班信息查询
2--加班信息添加
3--加班信息删除
4--加班信息修改
请输入您的选择:

```

加班信息查询操作

```

连接数据库成功
0--退出
1--人事变动管理
2--人事档案管理
3--加班出差管理
请输入您的选择: 3

0--返回
1--加班信息管理
2--出差信息管理
请输入您的选择: 1

0--返回
1--加班信息查询
2--加班信息添加
3--加班信息删除
4--加班信息修改
请输入您的选择: 1

请输入需要查询员工加班信息的编号: 002
加班编号: 1
加班时长: 3

加班编号: 2
加班时长: 6

```

加班信息查询函数实现（使用 select 在 emp\_over 和 overtimework 中查找对应信息）

```

void over_query()
{
    MYSQL_RES *result;
    MYSQL_ROW row;
    char eno[] = "";
    char strquery[100] = "select eno, emp_over.ono, otime from emp_over, overtimework where eno = ";
    cout << "\n请输入需要查询员工加班信息的编号: ";
    cin >> eno;
    strcat(strquery, eno);
    strcat(strquery, " and emp_over.ono = overtimework.ono;");
    if (mysql_query(&mysql, strquery) == 0) {
        result = mysql_store_result(&mysql);
        if (result->row_count != 0) {
            while (row = mysql_fetch_row(result)) {
                cout << "加班编号: " << row[1] << endl;
                cout << "加班时长: " << row[2] << "\n" << endl;
            }
        }
        else {
            cout << "该加班不存在" << endl;
        }
    }
    overtimemanu();
}

```

加班信息添加界面及实现

```

连接数据库成功
0--退出
1--人事变动管理
2--人事档案管理
3--加班出差管理
请输入您的选择: 3

0--返回
1--加班信息管理
2--出差信息管理
请输入您的选择: 1

0--返回
1--加班信息查询
2--加班信息添加
3--加班信息删除
4--加班信息修改
请输入您的选择: 2

请输入加班员工的编号: 002
请输入本次加班的编号: 003
请输入本次加班的时长: 12
加班信息添加成功

```

加班信息添加函数实现（使用 insert 在表 overtime 和 emp\_over 中添加加班信息）

```

void over_add()
{
    char eno[] = "";
    char ono[] = "";
    char otime[] = "";
    char strquery1[100] = "insert into overtime (ono,otime) values('";
    char strquery2[100] = "insert into emp_over (eno,ono) values('";
    cout << "\n请输入加班员工的编号: ";
    cin >> eno;
    strcat(strquery2, eno);
    strcat(strquery2, "','");
    cout << "请输入本次加班的编号: ";
    cin >> ono;
    strcat(strquery1, ono);
    strcat(strquery2, ono);
    strcat(strquery1, "','");
    cout << "请输入本次加班的时长: ";
    cin >> otime;
    strcat(strquery1, otime);
    strcat(strquery1, "','");
    strcat(strquery2, otime);
    if (mysql_query(&mysql, strquery1) == 0 && mysql_query(&mysql, strquery2) == 0){
        cout << "加班信息添加成功" << endl;
    }
    else{
        cout << "加班信息添加失败" << endl;
    }
    overtimeMenu();
}

```

## 加班信息删除界面及操作

```

连接数据库成功

0--退出
1--人事变动管理
2--人事档案管理
3--加班出差管理
1请输入您的选择: 3

[0--返回
[1--加班信息管理
[2--出差信息管理
[请输入您的选择: 1

n0--返回
t1--加班信息查询
"2--加班信息添加
n3--加班信息删除
t4--加班信息修改
t请输入您的选择: 3

1请输入加班员工的编号: 002
请输入需要删除加班的编号: 001
删除加班记录成功
  
```

加班信息删除界面实现（使用 `delete` 在表 `overtimework` 和 `emp_over` 中删除对应编号员工的加班信息）

```

354 void over_delete()
355 {
356     char eno[] = "";
357     char ono[] = "";
358     char strquery1[100] = "delete from emp_over where eno = ";
359     char strquery2[100] = "delete from overtimework where ono = ";
360     cout << "\n请输入加班员工的编号: ";
361     cin >> eno;
362     strcat(strquery1, eno);
363     cout << "请输入需要删除加班的编号: ";
364     cin >> ono;
365     strcat(strquery1, " and ono = ");
366     strcat(strquery1, ono);
367     strcat(strquery2, ono);
368     if (mysql_query(&mysql, strquery1) == 0 && mysql_query(&mysql, strquery2) == 0) {
369         cout << "删除加班记录成功" << endl;
370     }
371     else{
372         cout << "删除加班记录失败" << endl;
373     }
374     overtimemanu();
375 }
  
```

## 加班信息修改界面及操作

```

连接数据库成功
0--退出
1--人事变动管理
2--人事档案管理
3--加班出差管理
请输入您的选择：3

0--返回
1--加班信息管理
2--出差信息管理
请输入您的选择：1

0--返回
1--加班信息查询
2--加班信息添加
3--加班信息删除
4--加班信息修改
请输入您的选择：4

请输入需要修改加班的编号：002
请修改本次加班的时长：1
加班信息修改成功

```

加班信息修改实现（使用 update 修改 overtime 中对应编号的加班信息）

```

void over_update()
{
    char eno[] = "";
    char ono[] = "";
    char otime[] = "";
    char strquery1[100] = "update overtime set otime = ' ";
    cout << "\n请输入需要修改加班的编号：";
    cin >> ono;
    cout << "请修改本次加班的时长：";
    cin >> otime;
    strcat(strquery1, otime);
    strcat(strquery1, " ");
    strcat(strquery1, "where ono = ");
    strcat(strquery1, ono);
    if (mysql_query(&mysql, strquery1) == 0){
        cout << "加班信息修改成功" << endl;
    }
    else{
        cout << "加班信息修改失败" << endl;
    }
    overtimeMenu();
}

```

查询加班信息认证上述结果（已删除 002 员工的 001 加班，添加 003 加班，修改 001 加班时长为 1）

```
连接数据库成功
0--退出
1--人事变动管理
2--人事档案管理
3--加班出差管理
请输入您的选择：3

0--返回
1--加班信息管理
2--出差信息管理
请输入您的选择：1

0--返回
1--加班信息查询
2--加班信息添加
3--加班信息删除
4--加班信息修改
请输入您的选择：1

请输入需要查询员工加班信息的编号：002
加班编号：2
加班时长：1

加班编号：3
加班时长：12
```

出差功能的界面和函数实现与加班功能十分接近，报告中不单独列出。

## 六. 总结

本次课程设计运用了数据库课程中诸多知识：

- (1) 用 ER 模式进行概念数据库设计
- (2) MySQL 语言的应用
- (3) 关系的完整性约束
- (4) 对关系数据的查询
- (5) C++语言嵌入式 MySQL 的设计