

# 浙江大学

## OOP(object-oriented programming)



### PROJECT FINAL REPORT

1. Name : 杨乐辰 Student Number : 3180105955
2. Name : 王逸君 Student Number : 3180105884
3. Name : 沈栩捷 Student Number : 3180103479
4. Name : 石昊洋 Student Number : 3180102686

2019~2020 春夏学期 2020 年 7 月

## 1、 题目描述和要求

### 企业工资管理系统

- 1、企业最起码有管理人员、技术人员、工人三类人员，每一类人员有不同的工资计算方法（具体计算方法自行设计）；
- 2、可以存储、显示、修改和删除企业人员信息。
- 3、可以查询、统计工资信息。

## 2、 需求分析

在企业工资管理系统的实现过程中，我们认为主要有以下几点需求：

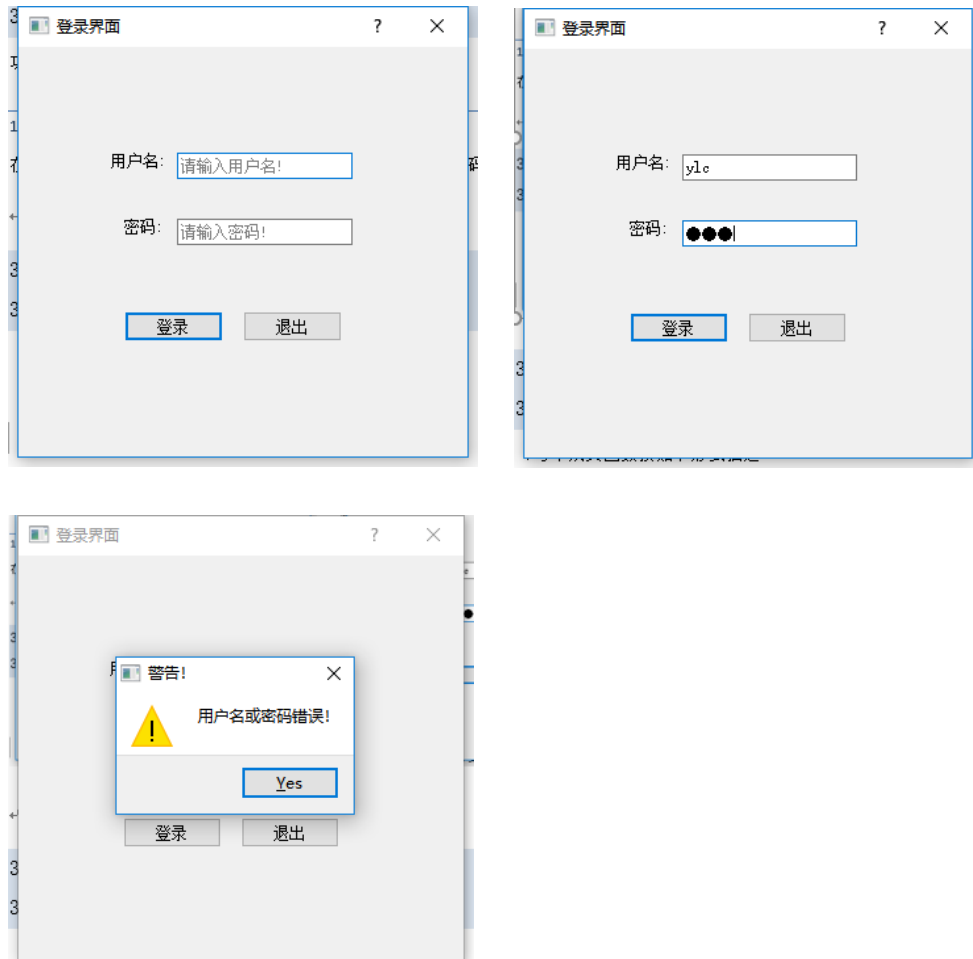
- 1、作为直接反映企业的工资情况的程序，我们需要设置登录界面对其采取一定的保密措施。
- 2、在程序中需要对员工信息进行增删改查的操作，所以设置专门的 Bar 进行点击事件。
- 3、对多种职务进行工资算法的细分，满足不同职位不同算法的要求。
- 4、对于每个部门以及整个公司发放的工资进行统计，让领导掌握公司工资情况。
- 5、对于每个员工的考勤、加班情况进行统计，发放全勤奖以及加班工资，对于旷工请假的员工扣除工资。

## 3、 总体设计

### 3.1 功能模块设计

#### 1. 登录界面

在打开程序后，输入用户名、密码登录系统，用户名或者密码错误则无法登入。（注：密码为 name : ylc , password : 123）



---

## 2. 添加员工信息

在主界面点击“修改员工信息”，再选择“新增员工信息”，则进入新建信息功能，在输入员工的基本信息后，点击 OK 则新建信息，Cancel 则取消，如果添加成功则显示“Insert Successfully! ”，失败则显示“Invalid input”，如已存在相同 Id 或者关键信息不全等问题。

1 添加员工信息 ? X

2

3

4

姓名: aaa

编号: 12334

性别: M

电话: 12233

部门: A

职务: C

薪资: 666

OK Cancel

工资管理系统

修改员工信息 工种 考勤 加班津贴 员工基本信息 总工资

	id	name	sex	phone number	department	job	salary
1	0000000001	Ylc	M	1111	Sci	Manager	11.22
2	0000000002	Sxj	M	1111	Atu	Manager	33.22
3	0000000003	Shy	M	1111	Wto	Manager	44.33
4	0000000004	Wvj	M	1111	Cvj	Manager	55.66

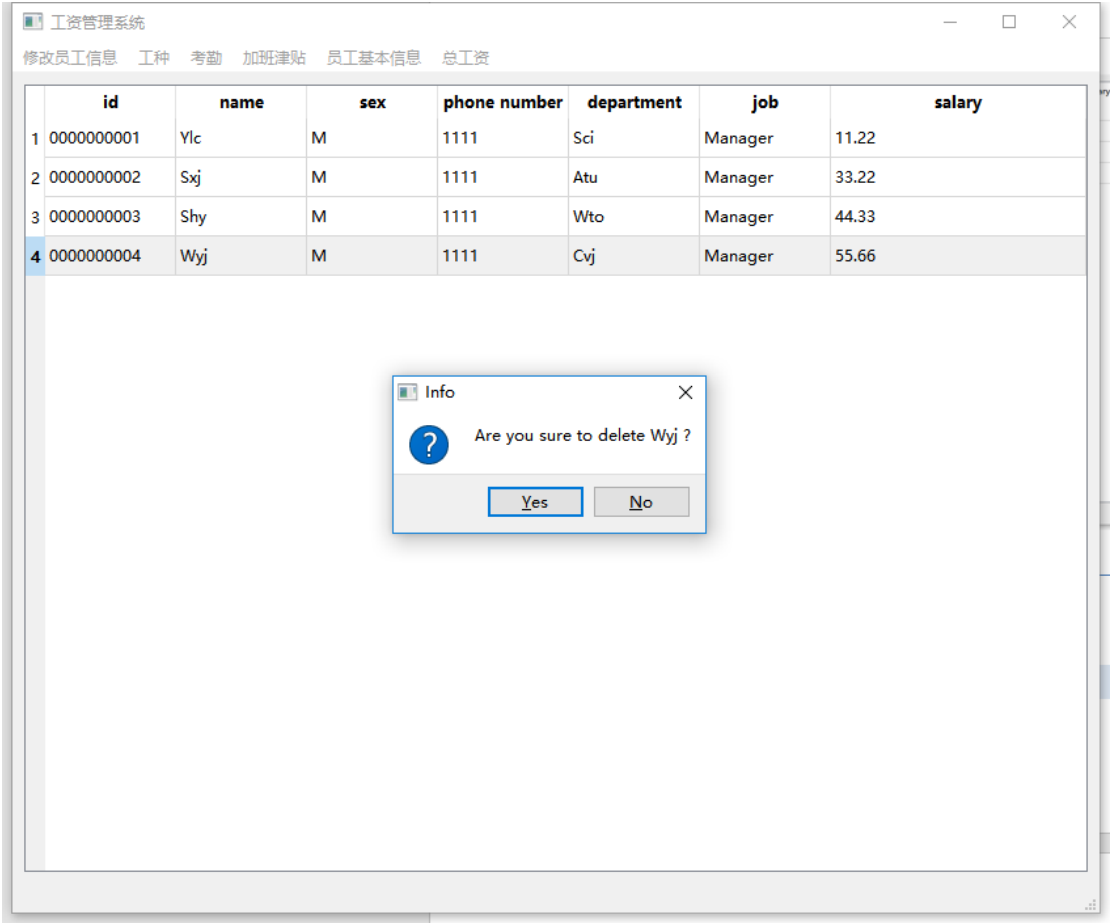
Info X

Insert Success

Yes

### 3. 删除员工信息

先主界面选中要删除的条目，然后点击删除员工信息，则显示：



点击 Yes 则删除信息，No 则取消；删除成功则显示 “Delete Success”，没有权限显示 "You don't have permission for deletion"，其他情况显示 "Error occured"。

#### 4. 修改员工信息

点击修改员工信息则出现修改栏，输入修改员工的 Id 以及其他信息，如果可以修改，点击 Ok 则显示 Modify Successfully，如果不能修改则显示 “invalid input”

修改员工信息

输入要修改员工的id: 23

修改信息

输入要修改员工的姓名: 31

输入要修改员工的性别: 323

输入要修改员工的电话: 213

输入要修改员工的部门: 133213

输入要修改员工的职务: 321

OK Cancel

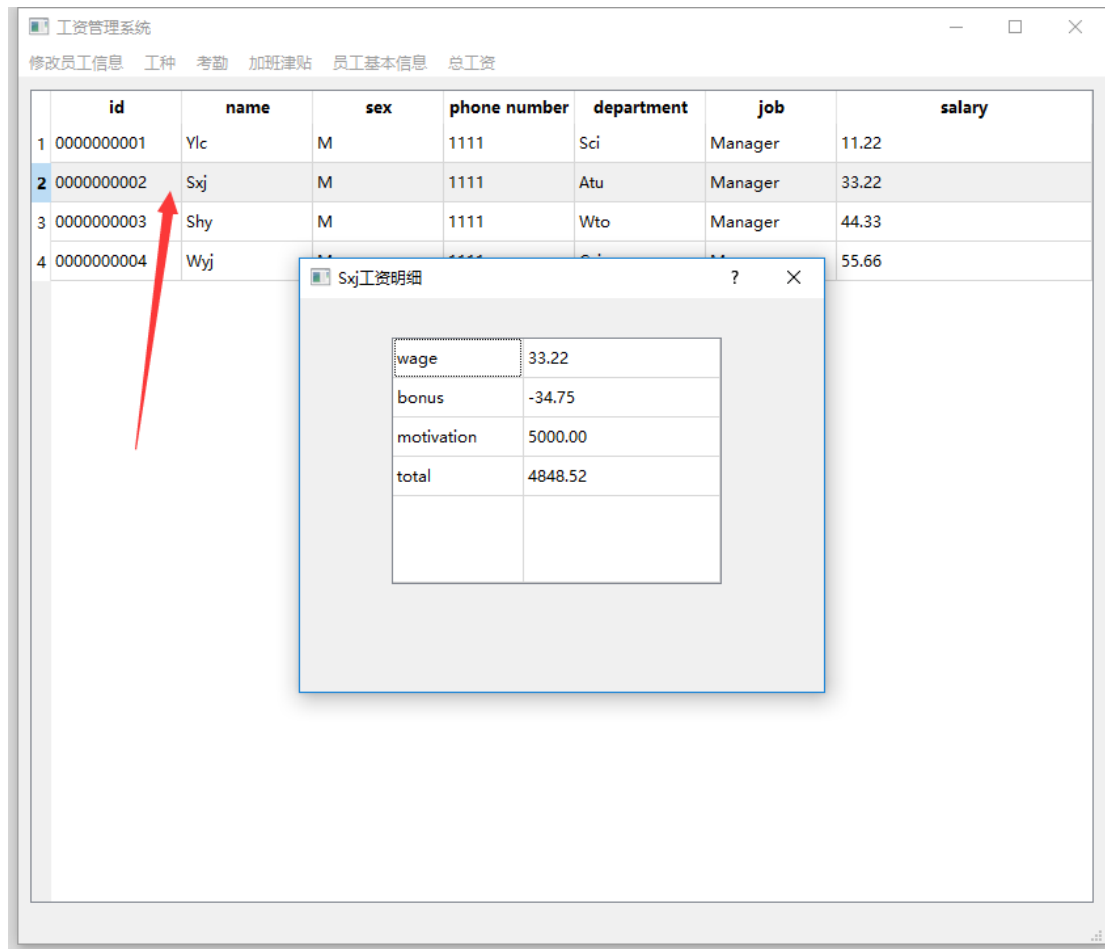
工资管理系统

修改员工信息 工种 考勤 加班津贴 员工基本信息 总工资

	id	name	sex	phone number	department	job	salary
1	0000000001	Ylc	M	1111	Sci	Manager	11.22
2	0000000002	Sxj	M	1111	Atu	Manager	33.22
3	0000000003	Shy	M	1111	Wto	Manager	44.33
4	0000000004	Wyj	M	1111	Cvj	Manager	55.66

Info Invalid input Yes

## 5. 双击员工信息显示详细工资



工资管理系统

修改员工信息 工种 考勤 加班津贴 员工基本信息 总工资

	id	name	sex	phone number	department	job	salary
1	0000000001	Ylc	M	1111	Sci	Manager	11.22
2	0000000002	Sxj	M	1111	Atu	Manager	33.22
3	0000000003	Shy	M	1111	Wto	Manager	44.33
4	0000000004	Wyj	M	1111	Sci	Manager	55.66

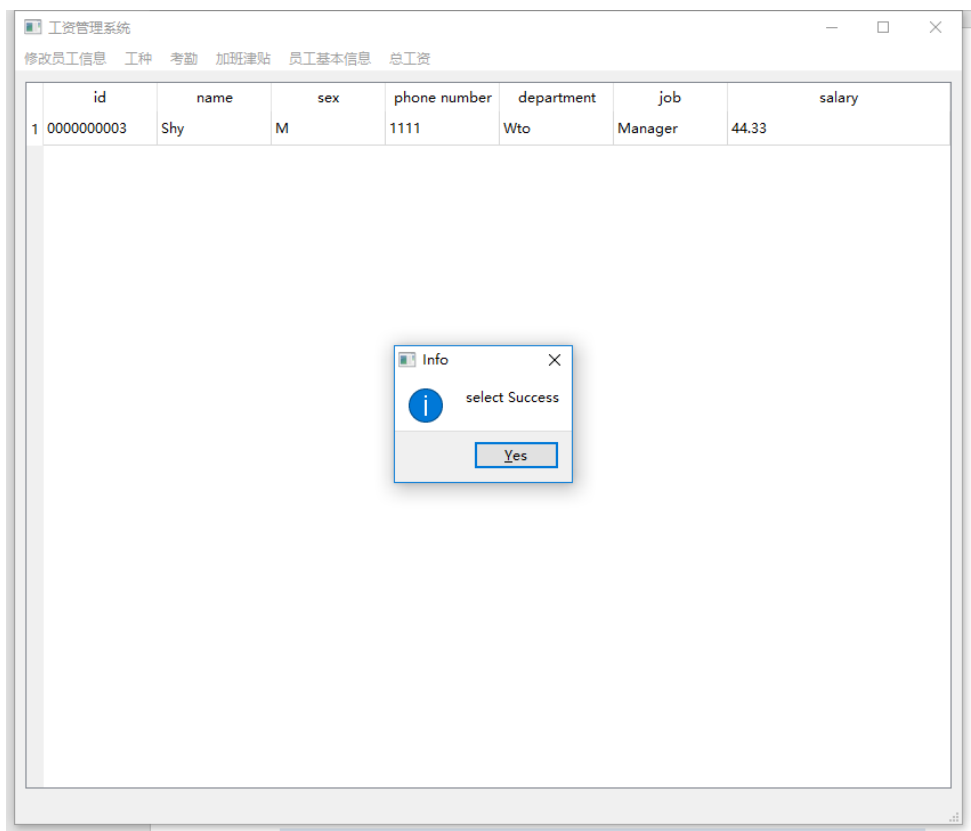
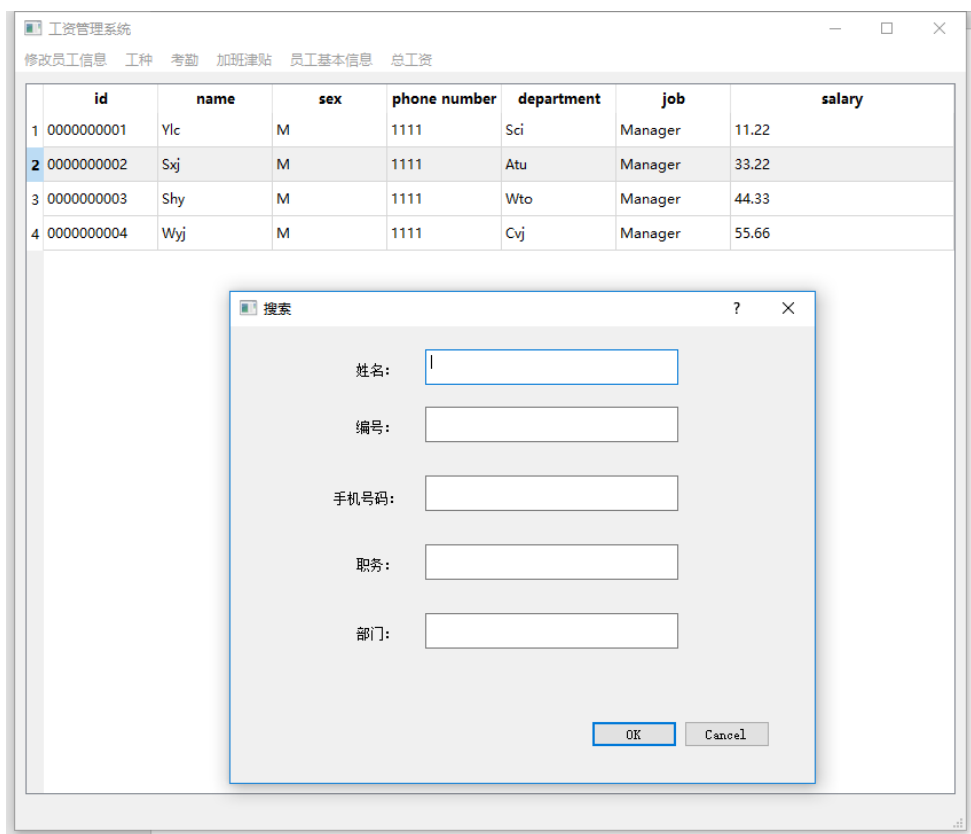
  

Sxj工资明细

wage	33.22
bonus	-34.75
motivation	5000.00
total	4848.52

## 6. 显示员工信息

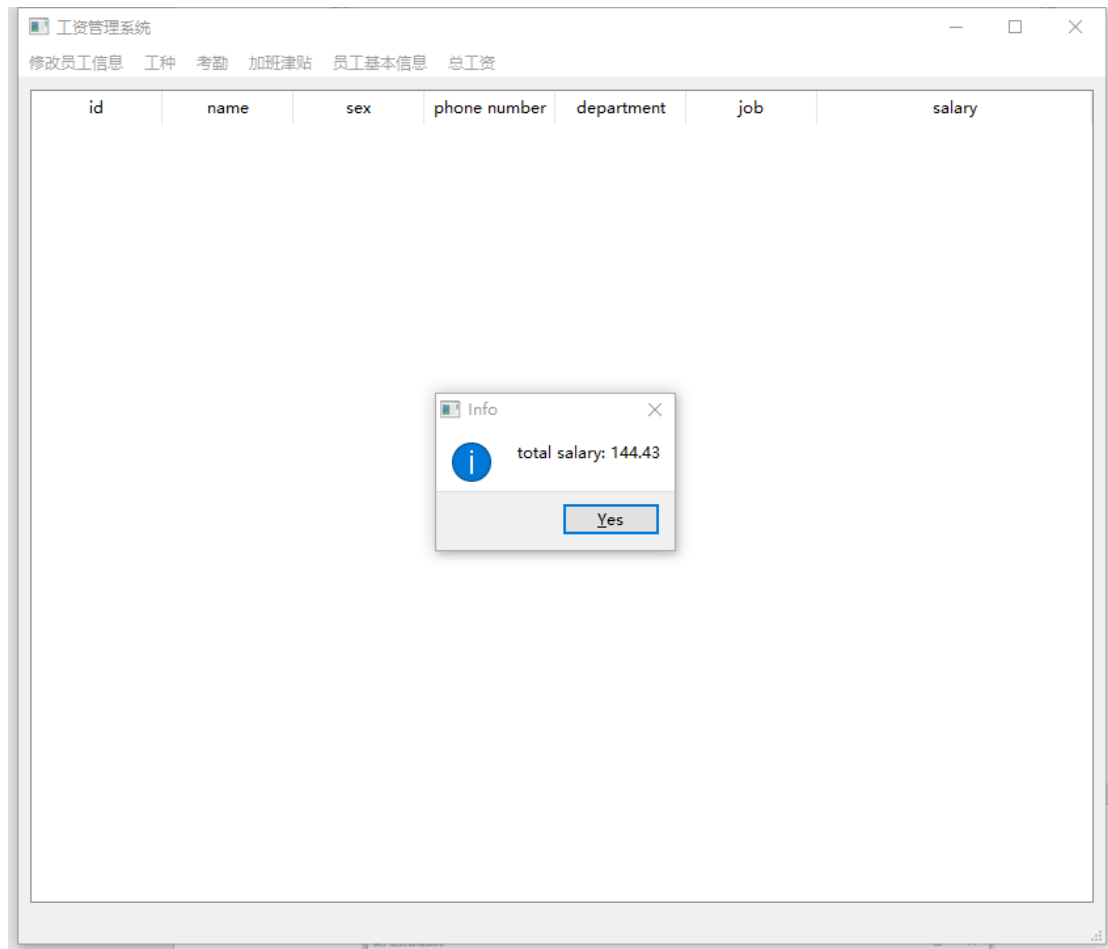
由于员工可能会很多，所以我们希望通过模糊搜索进行查找，在员工基本信息中选择显示，则可以通过输入已知信息进行搜索，如果不输入任何信息则显示总表。



搜索失败则显示空表。



## 7. 显示总工资



### 3.2 数据结构设计

数据库是主要的存储手段；将数据从数据库中取出后采用 **employee** 类进行数据封装，方便对数据的访问。

类关系如下：

**employee**：所有员工的抽象基类，其他类公有继承其中的成员。

**worker:** 工资由加班工资、绩效工资构成；

**management:** 有绩效工资、激励工资（开发中简化处理为一个常数）

**sales:** 有绩效工资、销售提成（开发中简化处理为常数）

**tech:** 有绩效、项目、技术工资，其中除了绩效之外的工资均简化处理为常数

**employee** 类中的关键方法是纯虚函数 `print_salary_detail()`，用于将不同工种员工的工资明细打印。

### 3.3 类主要功能描述

#### 1.DatabaseServer

- a. 主要功能：用于数据库的连接，获得数据库的部分数据
- b. 成员变量：`QSqlDatabase database` ,表示数据库连接句柄的实例化
- c. 成员函数：

##### (1) ConnectDB

函数原型：`bool connectDB()`

功能描述：连接数据库

参数描述： 无

返回值描述：连接数据库成功

重要局部变量定义：无

重要局部变量用途描述：无

函数算法描述：调用 `QSqlDatabase` 中的相关函数连接数据库

## (2) `createDB`

函数原型：`void createDB()`

功能描述：创建数据库

参数描述：无

返回值描述：无

重要局部变量定义：无

重要局部变量用途描述：无

函数算法描述：创建两张数据库的表 `information` 和 `salary`

## (3) `initDB`

函数原型：`void initDB()`

功能描述：初始化数据库

参数描述：无

返回值描述：无

重要局部变量定义：无

重要局部变量用途描述：无

函数算法描述：用 `QSqlQuery` 插入一些测试数据

## (4) `getExtraWork`

函数原型: `static QString getExtraWork(const QString id)`

功能描述: 获得表 salary 中相应 id 对应的 extra\_work 的数据

参数描述: 记录的 id

返回值描述: 返回加班的天数

重要局部变量定义: `QSqlQuery query, QString sql`

重要局部变量用途描述: 执行 sql 语句

函数算法描述: 用 `QSqlQuery` 操作数据库, 得到相应的记录

## (5) getAttendance

函数原型: `static int getAttendance(const QString id)`

功能描述: 获得表 salary 中相应 id 对应的 attendance 的数据

参数描述: 记录的 id

返回值描述: 返回考勤到岗的天数

重要局部变量定义: `QSqlQuery query, QString sql`

重要局部变量用途描述: 执行 sql 语句

函数算法描述: 用 `QSqlQuery` 操作数据库, 得到相应的记录

## (6) setSalaryTaxed

函数原型: `static void setSalaryTaxed(const QString id, const QString salaryTaxed);`

功能描述: 设置 salary 中的 salary\_taxed 字段

参数描述: 记录的 id, 需要修改的 salaryTaxed

返回值描述: 无

重要局部变量定义: `QSqlQuery query, QString sql`

重要局部变量用途描述: 执行 `sql` 语句

函数算法描述: 用 `QSqlQuery` 操作数据库, 改变相应的记录

## 2.MainWindow

a) 主要功能: 用于主界面的显示

b) 成员函数:

### i. `connectDatabase()`

函数原型: `bool MainWindow::connectDatabase()`

功能描述: 用于连接数据库

参数描述: 无

返回值描述: 无

### ii. `on_action_triggered()`

函数原型: `void on_action_triggered();`

功能描述: `layout` 中按钮的触发器, 点击 `layout` 中的按钮, 会触发韩洙中的内容

参数描述: 无

返回值描述: 无

### iii. `changePeople()`

函数原型: `bool changePeople(const QString name,const QString id, const QString sex, const QString phonenum, const QString department, const QString job);`

功能描述: 改变个人的内容

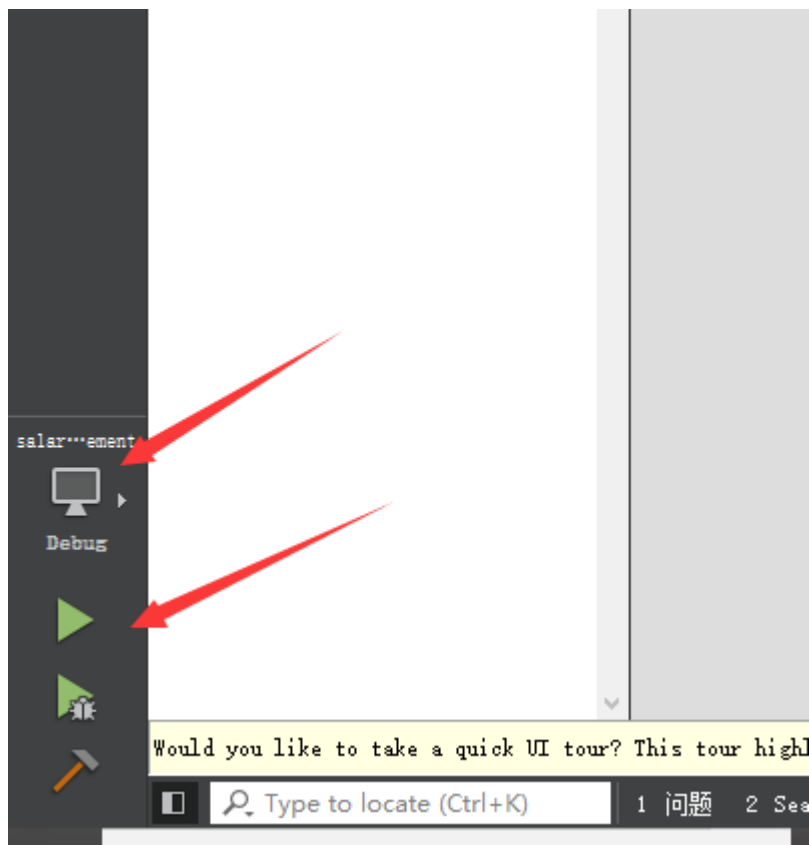
参数描述，可以由名字、id 或者性别来进行搜索

返回值描述：如果修改成功则返回 **true**，如果修改不成功则为 **false**。

## 4. 部署与运行

### 4.1 编译安装运行说明

- 1、 下载压缩包后首先解压，然后在已经安装 QT 以及 MinGW 的计算机中就可以直接点击.pro 文件进入 QT creator。
- 2、 在程序中首先选择 MinGW 编译器，然后待程序加载完成后点击左下角“Play”箭头即可。



- 3、 进入程序后首先输入“ylc”、“123”作为用户名以及密码，然后就可以按照程序中的菜单进行操作，注意请勿使用多余的回车键。



#### 4.2 典型测试情况

- 1 在测试修改信息时，其余修改部分都可以正常使用，但是对 Job 进行修改时则出现无法修改的问题，后解决

修改员工信息

输入要修改员工的id: 1

修改信息

输入要修改员工的姓名:

输入要修改员工的性别:

输入要修改员工的电话:

输入要修改员工的部门: Tech

输入要修改员工的职务:

OK Cancel

工资管理系统

修改员工信息 工种 考勤 加班津贴 员工基本信息 总工资

	id	name	sex	phone number	department	job	salary
1	0000000001	Ylc	M	1111	Sci	Manager	11.22
2	0000000002	Sxj	M	1111	Atu	Manager	33.22
3	0000000003	Shy	M	1111	Wto	Manager	44.33
4	0000000004	Wyj	M	1111	Cvj	Manager	55.66
5	0000000005	qqq	F	12321	saxsa	Tech	44
6	0000000006	4156	M	4245	44	Sales	99
7	0000000222	123213	F	122	123	Worker	666

Info Invalid input Yes

- 2 在测试修改员工考勤时，发现无法插入，最后则是输入框槽函数参数反了，之后做出修改。



- 3 设置加班天数和考勤天数时忘记考虑有一部分职务无需考勤以及加班，导致出现问题。

## 5. 组内分工

### 5.1 组内分工情况

**杨乐辰：**登录前后端以及程序界面设计

**石昊洋：**负责数据模型（即员工类以及各种不同工种类）的封装、显示及与数据库的交互；轻度涉及工资明细的界面设计。

**沈栩捷：**负责数据库相关的操作和前后端中需要数据库操作的信号与槽的部分。

**王逸君：**登陆后后端界面以及 layout 和主界面弹窗设计。文稿的修改和最终审核。文件结构最终修改

### 5.2 个人实践过程中遇到的难点及解决方案

**杨乐辰：**

- （1）设置登录界面涉及逻辑较多，对许多的程序登录界面研究进行设计。
- （2）界面 UI 设计需要与组内同学沟通，要完成每个同学的要求。
- （3）QT 的学习和数据库使用也话花费了不少时间。

**石昊洋：**

- （1）设计的功能中对于不同的工种有不同的工资统计方式；采用虚函数的机制很自然地解决了这个问题。
- （2）设计的逻辑中，双击主页面员工信息即会显示工资明细界面，开发过程中曾出现弹出多个相同的工资明细界面的 bug；后来发现问题出在前端程序员在刷新 tableview 的时候调用了现成的 tableview 初始化函数，实际上

创建了多个重叠的 tableview，双击时全都进行了响应并弹出了多个窗口。将刷新时调用的代码更改为仅更新表的内容之后问题解决。

（3）内存管理：从主界面生成显示工资明细的弹出窗口容易造成内存泄露；本次开发中利用了 Qt 的内存管理机制，通过指定弹出窗口的 parent 窗口，在 parent 窗口析构时会自动析构其下的所有弹出窗口，解决了这一问题；窗口中的子控件的析构较为简单，在主窗口析构函数中一齐析构，不在话下。

**沈栩捷：**

（1）在连接数据库的时候需要学习 SQLite3 与 QT 的连接，因为 SQLite 比较轻量，所以与之前比较熟悉的 MySQL 有点区别，需要重新学习一下

（2）数据库在 Table View 中的显示需要了解一下 QT 中的 model，把数据库中的表以表格的形式展示出来

（3）QT 中的 SQL 语句的实现以及语法需要学习 Git 的学习与使用

**王逸君：**

（1） 界面设计需要学习，槽函数等概念来连接界面和 controller。

（2） 协调不同的工作进度以及报告最后的审核修改。

（3） 需要了解 MVC 结构的开发

## 6. 合作纪要

2020 年 7 月 21 日：



本次会议主要讨论确定最终开发环境，代码风格等内容。

Github 上开发记录

下面是开发人员

Contributors 4



nujiy nujiy



Luke-Skycrawler Luke-Skycrawler



FreezingBP FreezingBP

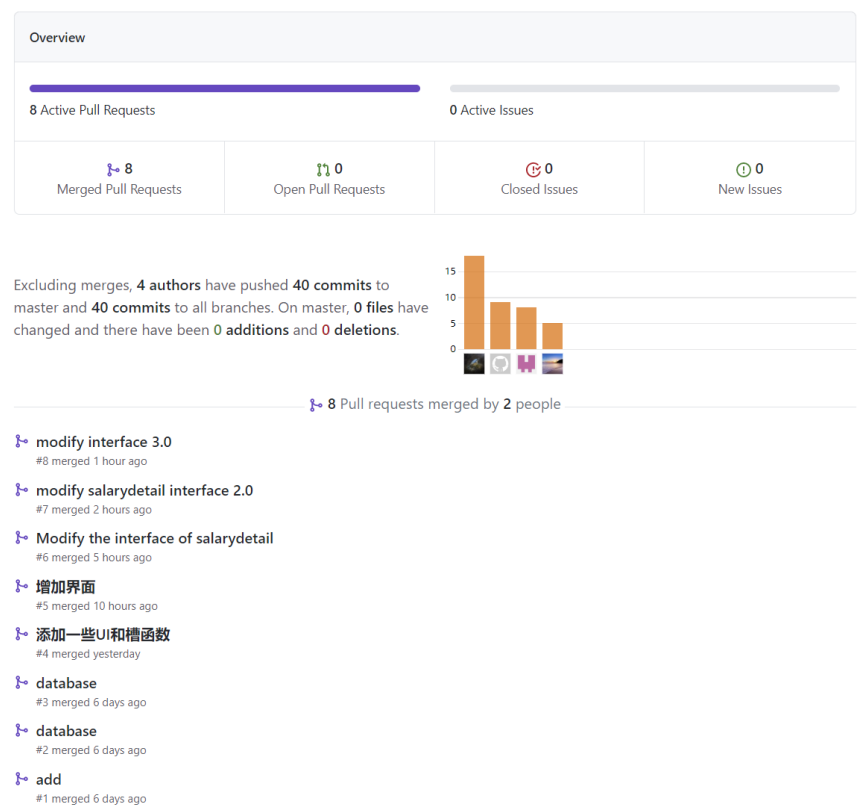


fishcrap fishcrap

合并记录

June 30, 2020 – July 30, 2020

Period: 1 month

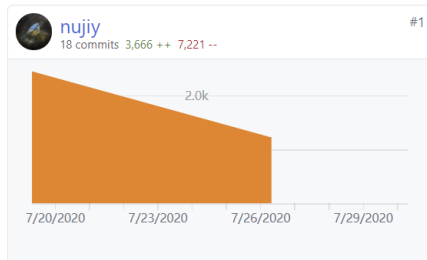
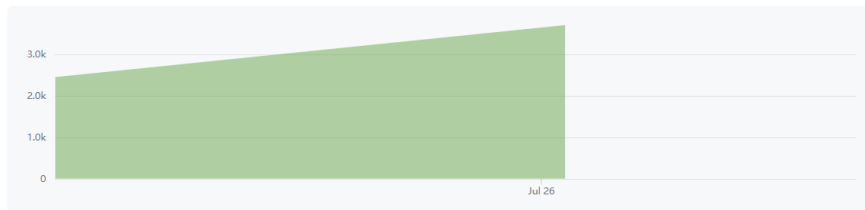


添加代码量

Jul 19, 2020 – Jul 30, 2020

Contributions: Additions ▾

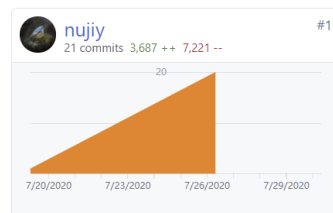
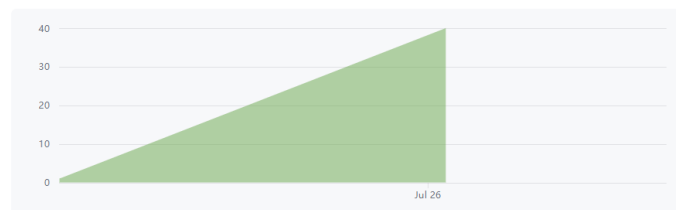
Contributions to master, excluding merge commits



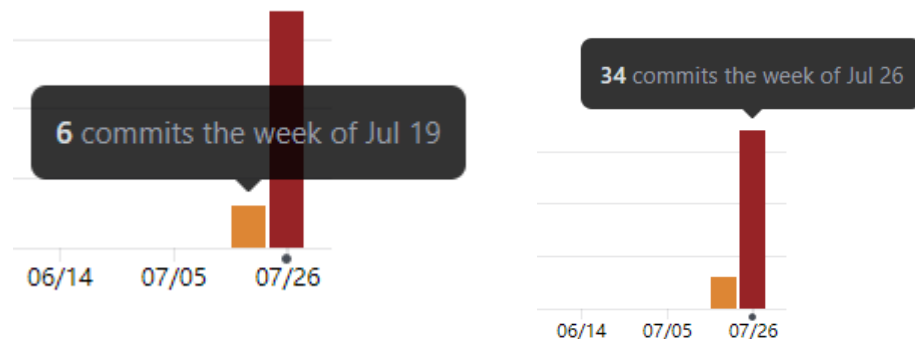
Jul 19, 2020 – Jul 30, 2020

Contributions: Commits ▾

Contributions to master, excluding merge commits



提交代码主要日期



## 7. 总结

### 7.1 程序亮点或创新之处

该程序完成了这个题目下的所有要求，操作合理人性化，并且增加了许多其他功能，比如双击查看工资明细、职务系统的工资修改以及对于不同输入的判断等。相比于其他的工资系统我们采用 QT 自带的 sqlite 进行操作，所以使用范围很广，无需安装 SQL 等数据库，可直接运行，系统兼容性好。

程序大体上采用了 MVC 架构来组织，代码架构较清晰，开发的耦合性低，组员分头开发不易冲突。

### 7.2 应用知识点

外部第三方库的使用：SQL 数据库，QT Windows 界面开发；

课程内容：

1. 对于不同的工种，采用了虚函数和 `dynamic_cast` 的机制来实现不同的工资统计与显示。
2. 对于常用的统计税后收入的函数采用了内联来提高效率。
3. 采用了友元类把数据从员工的类提供给显示的窗口类，而其他类中均不可访问，从而提高了员工类的封装性。

### 7.3 心得体会

**总体描述：**本项目使用 Github 进行协同开发。项目一共分为两个部分，一个为 View 层为主的前端开发，一个为 Model 层为主的后端开发。其中前端使用 QT Creator 作为前端开发环境，后端以 SQL lite 数据库为主。本组 View 层主要由王逸君、杨乐辰编写；Model 层主要由石昊洋，沈栩捷编写。

**杨乐辰：**在项目的设计中，我首先学习了 QT 的 C++语法，熟悉 QT 中不同类的使用，对于课内学到的虚函数以及类的复用有了一定认识，其次也在程序的设计过程中学习了一部分 SQLite 的知识，对于数据库的使用有个初步的了解。而且在程序的编写过程中与各个组内同学进行了许多的交流，大家也对我给予了很多的帮助。对于一个应用程序的界面设计也有了初步认识，认识到要根据用户需求设计程序让其更加人性化。这次的 oop 大程锻炼了我的编程能力，让我对今后学习的方向有了一定的认识。

**王逸君：**在项目的实现中，进一步深入了解了 Git 和 Github 之间的联系以及不同仓库之间的拉取请求、合并等操作细节，真正使 Git 成为自己代码管理的工具。在本次开发中，学习了使用 QT Creator 进行窗口界面的开发，了解槽函数等概念，能够进行使用 GUI 的程序开发。此外，本次项目开发过程中，我的同伴也给予了我很多帮助，并容忍我的拖延等毛病，我向他们深表感谢。

**石昊洋：**本次多人协作开发的经历中是调试部分和开发的部分都给了我颇多收获。

在开始编写前，组员之间就已经大致理解了各自的分工，分工的明确减少了不少交流的困难；版本控制用了 git，可能是因为大家都能比较熟练地掌握，合作起来很舒服，几次合并也很顺利。

调试的过程就不太能用“舒服”来形容，但大家也是尽力施展各种了手腕，比如熟悉不同模块的组员开视频会议共享屏幕 debug 等。总之，处理问题的时候不会有单打独斗的感觉，可能也是因为组员之间交流的信息量够大吧（电话交流的确比打字有效多了）。

至于使用到了多少 oop 课程教授的知识点，虽然我认为不是最重要的收获，但也犯了些许编程错误，得了教训。其中印象较深的是在一个非虚函数里调用了虚函数导致了 undefined vtable 报错。网上同类问题的解决方法还挺搞笑的。



最后，开发完成后，看着 github 上的提交数和满满的一页的代码文件，居然还有点成就感，很像个“大型项目”的架子。这就够了。

**沈栩捷：**在完成项目的过程中，温故了很多 C++ 的知识和数据库的知识，对于 QT 中操作数据库的类和函数有了更加深入的了解和认识。对于 QT 中槽和信号的机制也有了更深一步的认知。在本次项目中，还新学到了使用 Git 项目管理工具进行管理开发的方法，第一次四个人在 GitHub 上共同管理开发。在一起合作开发项目的同时，我们也尽可能地互相沟通，语音沟通，充分体会到了共同合作需要及时沟通才能更好地完成任务。

## 8、参考文献和资料

《数据库系统概念》

QT 官方文档

<https://www.qt.io/>

<https://blog.csdn.net/lk142500/article/details/81109083>

<https://blog.csdn.net/xiaomucgwlmx/article/details/82914756>

[https://blog.csdn.net/HK\\_5788/article/details/80959076](https://blog.csdn.net/HK_5788/article/details/80959076)

还用很多就不一一列举了