

高校成绩管理系统数据库设计与实现

班级 国际 1504 学号 201519630415 姓名 王成章

1、需求分析

由于时代的发展，数据更新换代，某校原来的 Excel 和手工的表结构已经不能满足当下的需求，特此实现一个针对高校的数据管理系统

1.1 数据需求描述

在使用数据库之前，多大部分是用纸质的资料和普通 office 文本储存的学生信息，因此设计一个具有能够导入现有信息并且加以优化的数据库信息。针对学生端和教师端做出不同优化。

1.2 系统功能需求

数据库的基本功能，以 C/S 的方式实现客户端界面，要求能使学生查看与统计自身的不同信息

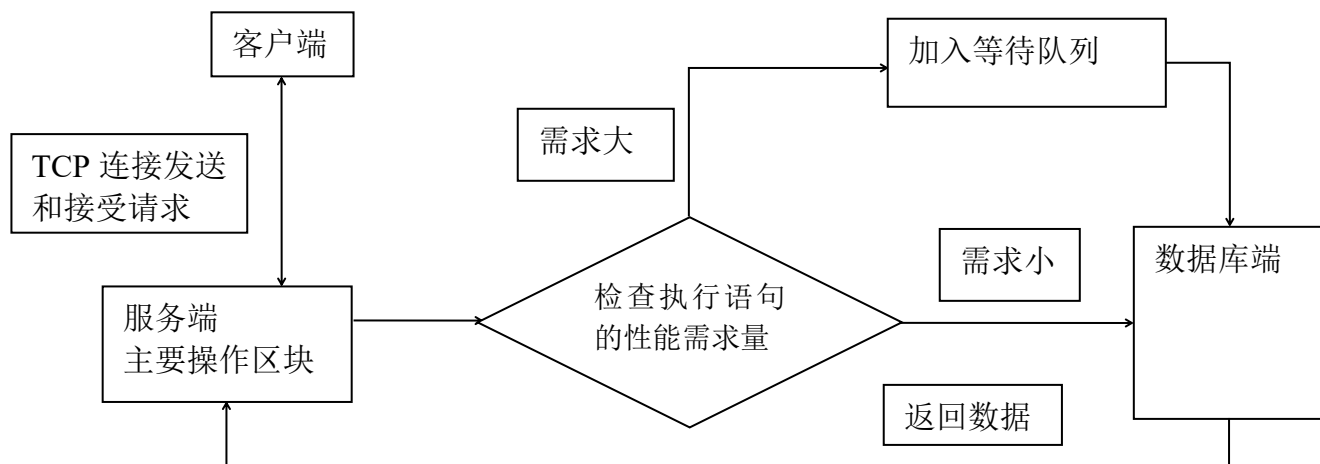
扩展功能，录入 Excel 文件，图表可视化显示详细信息，良好的系统消息应用提示。

1.3 其他性能需求

由于日前的数据库系统，尤其是高校选课的成绩管理系统存在低频的高密集的 I/O 响应，使得数据库没法一时间处理请求，针对此种情况，多线程乃至进程优化了系统结构。

2、概念结构设计

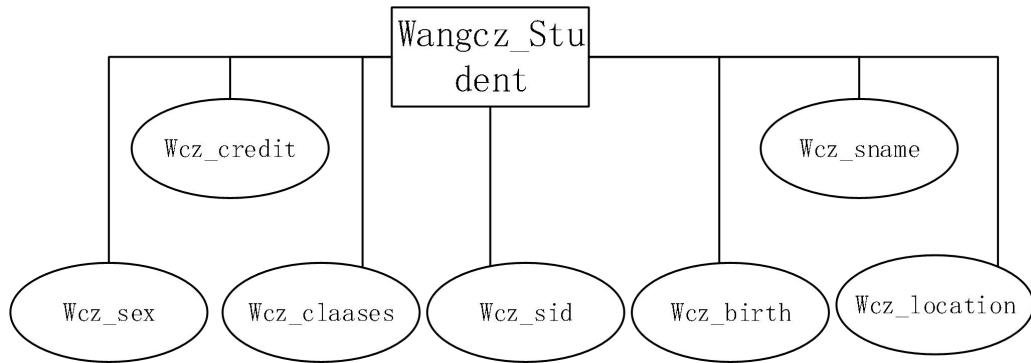
系统结构图



文字简述：由于在客户端上直接连接数据库会造成不安全的行为，因此在中间封装了一层高 io 的服务器层接收数据库和客户端的连接。同时开一个进程给任务队列去执行业务，考虑到查询数据必须不经过队列等待，在服务器应用层发送 sql 命令之前判断执行的业务类型来优化结构。

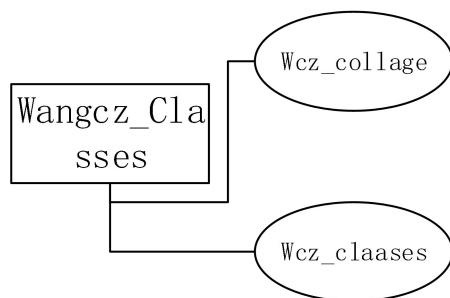
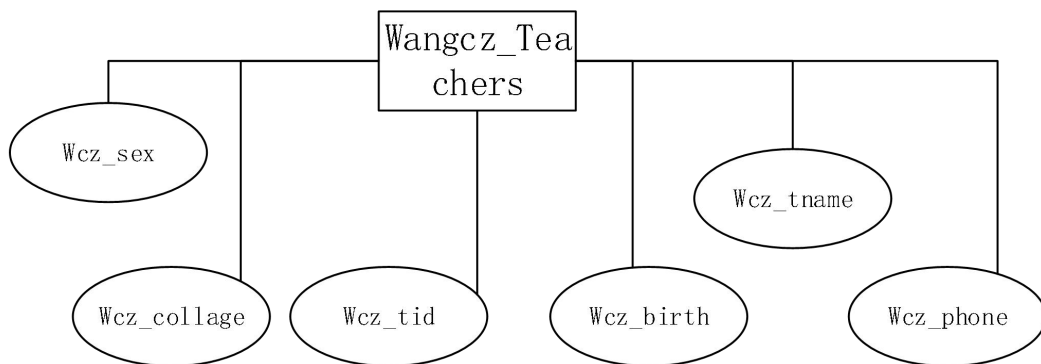
E-R 图

学生



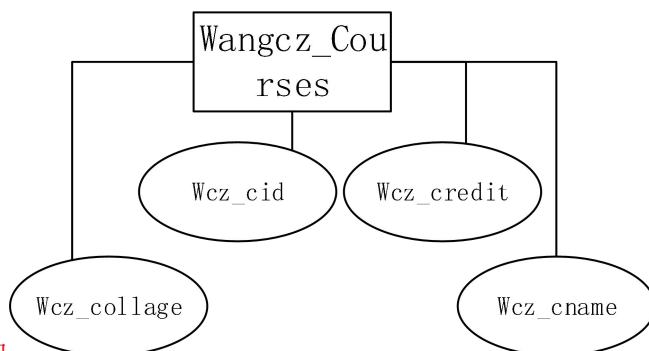
教

师



班级

课程



3、逻辑结构设计

3.1 关系模式设计

将 E-R 图转换为关系模式，定义实体型、属性及其联系。

学生 Wangcz_Students(学号 wcz_sid, 班级名称 wcz_classes, 姓名 wcz_sname, 性别 wcz_sex, 出生年月 wcz_birth, 生源所在地 wcz_location, 已修总学分 wcz_credit)

班级 Wangcz_Classes (学院 wcz_collage, 班级名称 wcz_classes)

教师 Wangcz_Teachers(工号 wcz_tid, 姓名 wcz_tname, 性别 wcz_sex, 出生年月 wcz_birth, 联系电话 wcz_phone, 所属学院 wcz_collage)

课程 Wangcz_Courses(课程编号 wcz_cid, 课程名字 wcz_cname, 所属班级 wcz_classes, 学分 wcz_credit, 限制人数 wcz_maxnum)

成绩 Wangcz_Reports(学号 wcz_sid, 课程编号 wcz_cid, 教师名称 wcz_tid, 成绩 wcz_score)

学院 Wangcz_Collage(学院 wcz_collage)

消息通知 Wangcz_Msgbox(消息编号 wcz_mid, 编号 wcz_id, 内容 wcz_msg)

登录验证 Wangcz_Passwords(账号 wcz_id, 密码, wcz_password)

3.2 数据类型定义

对关系模式中的属性定义类型、长度和约束

(1) 学生

属性	类型	长度	约束
<u>学号 wcz_sid</u>	int	/	primary key
班级名称 wcz_classes	char	20	foreign key
姓名 wcz_sname	char	20	not null
性别 wcz_sex	char	2	“男” or “女”
出生年月 wcz_birth	date	/	
生源所在地 wcz_location	char	20	
已修总学分 wcz_credit	float	/	

教师

属性	类型	长度	约束
工号 wcz_tid	int	/	primary key
姓名 wcz_tname	char	20	not null
性别 wcz_sex	char	2	“男” or “女”
出生年月	date	/	

wcz_birth			
联系电话 wcz_phone	char	20	
所属学院 wcz_collage	char	20	Foreign key

课程

属性	类型	长度	约束
课程编号 wcz_cid	int	/	primary key
课程名字 wcz_cname	char	20	Not null
所属班级 wcz_classes	char	20	foreign key
学分 wcz_credit	int	/	
限制人数 wcz_maxnum	Int	/	

班级

属性	类型	长度	约束
班级名字 wcz_classes	char	20	primary key
所属学院 wcz_collage	char	20	foreign key

成绩

属性	类型	长度	约束
学号 wcz_sid	int	/	foreign key
课程编号 wcz_cid	int	/	foreign key
教师名称 wcz_tid	int	/	foreign key
成绩 wcz_score	float	/	
学年 wcz_year	int	/	

学院

属性	类型	长度	约束
学院 wcz_collage	Char	20	primary key

消息

属性	类型	长度	约束
消息编号 wcz_mid	int	/	primary key
编号 wcz_id	int	/	foreign key
内容 wcz_msg	Char	100	

登录

属性	类型	长度	约束
账号 wcz_id	int	/	foreign key

密码 wcz_password	int	/	Default 123456
--------------------	-----	---	----------------

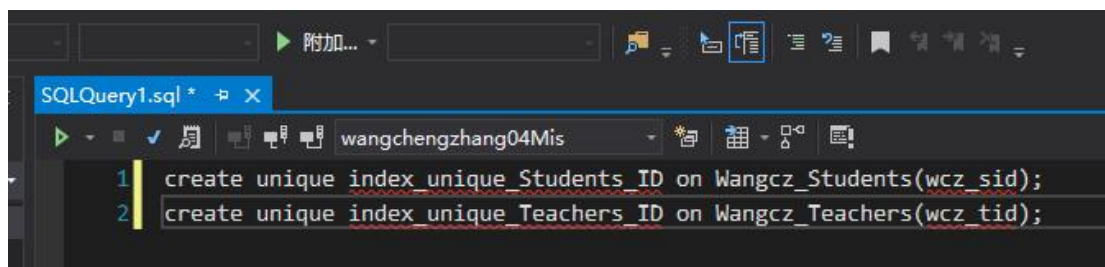
4、物理结构设计

4.1 聚簇设计

确定每个关系需要或不需要聚簇索引，为什么需要聚簇索引，在哪些列上建立聚簇索引。

4.2 索引设计

```
create unique index_unique_Students_ID on Wangcz_Students(wcz_sid);
create unique index_unique_Teachers_ID on Wangcz_Teachers(wcz_tid);\
```



4.3 分区设计

数据库文件和日志文件的分区问题。

5、数据库实施

（全部操作都要在 SQL Server 2005 的 SSMS 环境中用命令实现，并要求截图）
比如：

5.1 基本表建立

以下以学生表为例说明报告撰写过程：

（1）表学生 Wangcz_Students 的建立
SQL 语句为：

```
create table Wangcz_Students (
wcz_id int Primary key,
wcz_name char(20) not null,
wcz_sex char(2) check(wcz_sex='男' or wcz_sex='女'),
wcz_birth date,
wcz_classes char(20),
wcz_location char(20),
wcz_credit int,
```

```
constraint Wangcz_Classes_Wangcz_Students foreign key (wcz_classes) references
Wangcz_Classes
)
```

```
SQLQuery1.sql *
1 create table Wangcz_Students (
2   wcz_id int Primary key,
3   wcz_name char(20) not null,
4   wcz_sex char(2) check(wcz_sex='男' or wcz_sex='女'),
5   wcz_birth date,
6   wcz_classes char(20),
7   wcz_location char(20),
8   wcz_credit int,
9   constraint Wangcz_Classes_Wangcz_Students foreign key (wcz_classes) references Wangcz_Classes
10 )
```

表 Wangcz_Teachers 的建立

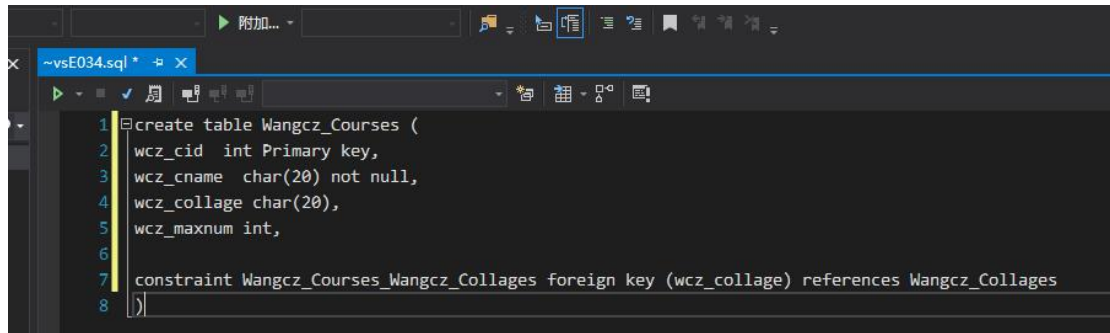
```
create table Wangcz_Teachers (
wcz_tid int Primary key,
wcz_tname char(20) not null,
wcz_sex char(2) check(wcz_sex='男' or wcz_sex='女'),
wcz_phone char(20),
wcz_collage char(20)
constraint Wangcz_Teachers_Wangcz_Collages foreign key (wcz_collage) references
Wangcz_Collages
)
```

```
~vsE034.sql *
1 create table Wangcz_Teachers (
2   wcz_tid int Primary key,
3   wcz_tname char(20) not null,
4   wcz_sex char(2) check(wcz_sex='男' or wcz_sex='女'),
5   wcz_phone char(20),
6   wcz_collage char(20)
7   constraint Wangcz_Teachers_Wangcz_Collages foreign key (wcz_collage) references Wangcz_Collages
8 )
```

表 Wangcz_Courses 的建立

```
create table Wangcz_Courses (
wcz_cid int Primary key,
wcz_cname char(20) not null,
wcz_collage char(20),
wcz_maxnum int,

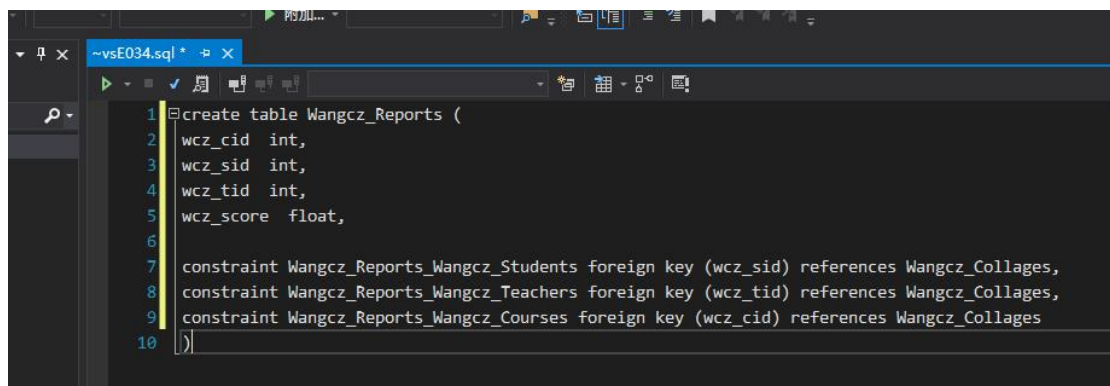
constraint Wangcz_Courses_Wangcz_Collages foreign key (wcz_collage) references
Wangcz_Collages
)
```



```
1 create table Wangcz_Courses (  
2   wcz_cid int Primary key,  
3   wcz_cname char(20) not null,  
4   wcz_collage char(20),  
5   wcz_maxnum int,  
6  
7   constraint Wangcz_Courses_Wangcz_Collages foreign key (wcz_collage) references Wangcz_Collages  
8 )
```

表 Wangcz_Reports 的建立

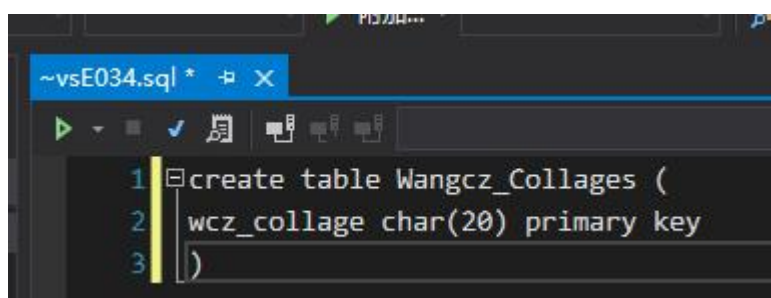
```
create table Wangcz_Reports (  
wcz_cid int,  
wcz_sid int,  
wcz_tid int,  
wcz_score float,  
  
constraint Wangcz_Reports_Wangcz_Students foreign key (wcz_sid) references  
Wangcz_Collages,  
constraint Wangcz_Reports_Wangcz_Teachers foreign key (wcz_tid) references  
Wangcz_Collages,  
constraint Wangcz_Reports_Wangcz_Courses foreign key (wcz_cid) references Wangcz_Collages  
)
```



```
1 create table Wangcz_Reports (  
2   wcz_cid int,  
3   wcz_sid int,  
4   wcz_tid int,  
5   wcz_score float,  
6  
7   constraint Wangcz_Reports_Wangcz_Students foreign key (wcz_sid) references Wangcz_Collages,  
8   constraint Wangcz_Reports_Wangcz_Teachers foreign key (wcz_tid) references Wangcz_Collages,  
9   constraint Wangcz_Reports_Wangcz_Courses foreign key (wcz_cid) references Wangcz_Collages  
10  )
```

表 Wangcz_Collages 的建立

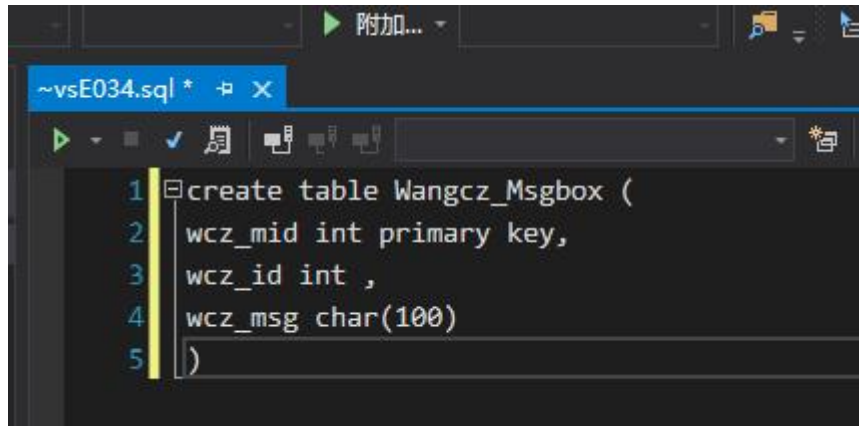
```
create table Wangcz_Collages (  
wcz_collage char(20) primary key  
)
```



```
1 create table Wangcz_Collages (  
2   wcz_collage char(20) primary key  
3 )
```


表 Wangcz_Msgbox 的建立

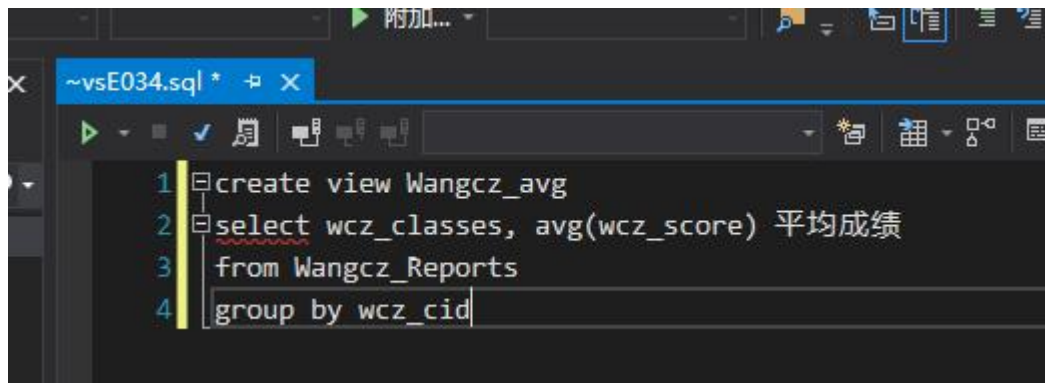
```
create table Wangcz_Msgbox (  
    wcz_mid int primary key,  
    wcz_id int ,  
    wcz_msg char(100)  
)
```



5.2 视图的建立

(1) 视图

```
create view Wangcz_avg  
select wcz_classes, avg(wcz_score) 平均成绩  
from Wangcz_Reports  
group by wcz_cid
```



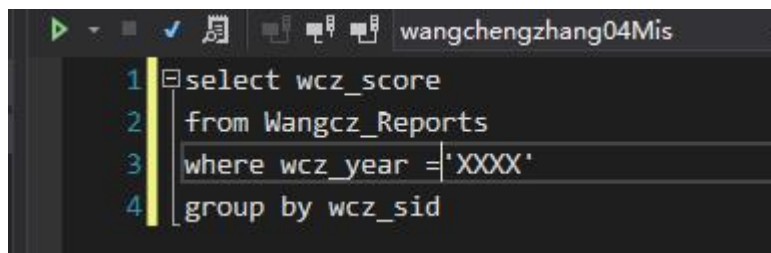
5.3 索引的建立

5.4 触发器建立

5.5 存储过程建立

5.6 业务处理和查询功能 SQL 语句

1) 学生成绩按每学年进行成绩统计的 SQL 语句:

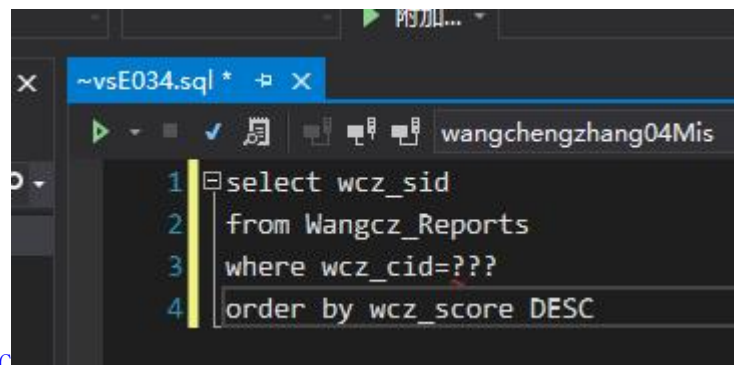


```
1 select wcz_score
2 from Wangcz_Reports
3 where wcz_year = 'XXXX'
4 group by wcz_sid
```

```
select wcz_score
from Wangcz_Reports
where wcz_year = 'XXXX'
group by wcz_sid
```

2) 学生成绩名次排定 SQL 语句:

```
select wcz_sid
from Wangcz_Reports
where wcz_cid=???
```

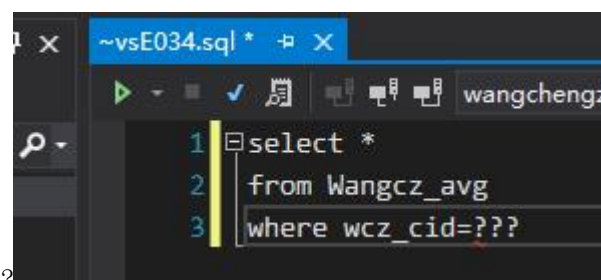


```
1 select wcz_sid
2 from Wangcz_Reports
3 where wcz_cid=???
4 order by wcz_score DESC
```

```
order by wcz_score DESC
```

3) 每门课程平均成绩统计 SQL 语句:

```
select *
from Wangcz_avg
```

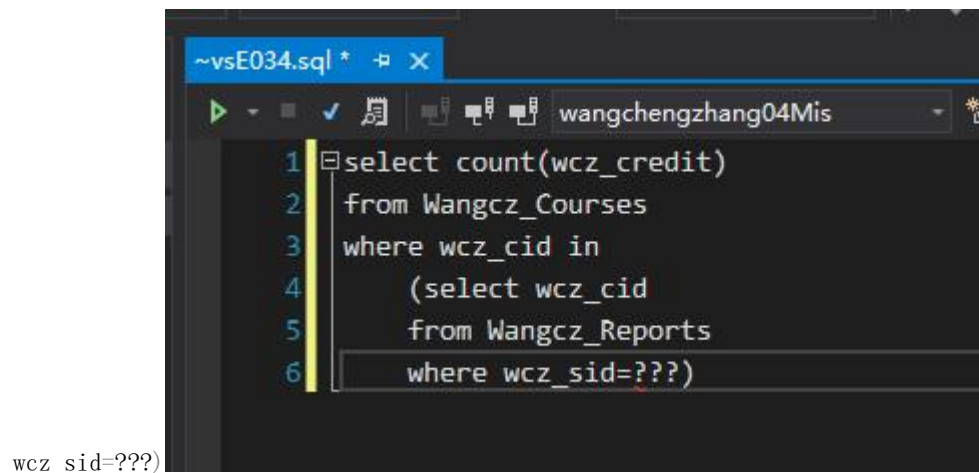


```
1 select *
2 from Wangcz_avg
3 where wcz_cid=???
```

```
where wcz_cid=???
```

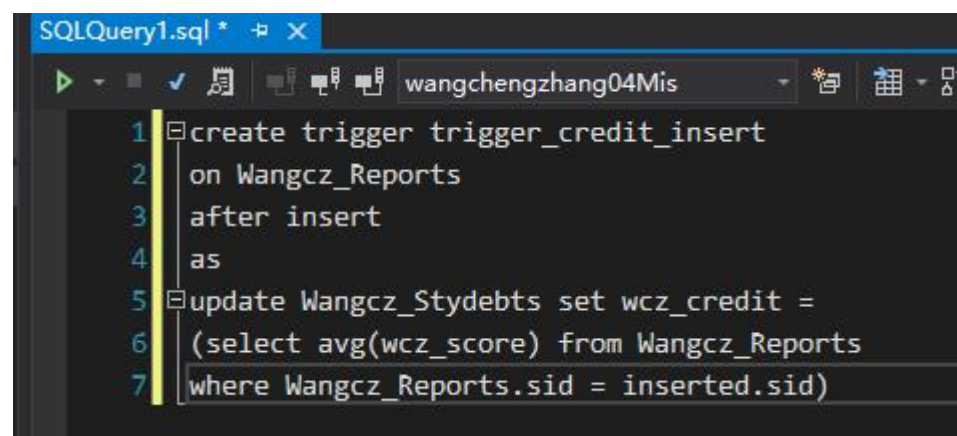
4)学生所学课程及学分统计 SQL 语句:

```
select count(wcz_credit)
from Wangcz_Courses
where wcz_cid in
      (select wcz_cid
       from Wangcz_Reports
        where
```



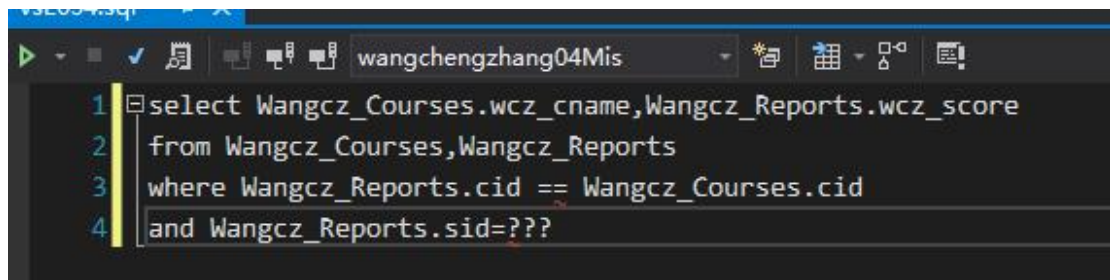
5)输入每个学生成绩时，自动生成该学生已修总学分 SQL 语句:

```
create trigger trigger_credit_insert
on Wangcz_Reports
after insert
as
update Wangcz_Stydebts set wcz_credit =
(select avg(wcz_score) from
Wangcz_Reports
```



6)where Wangcz_Reports.sid = inserted.sid)学生成绩查询 SQL 语句:

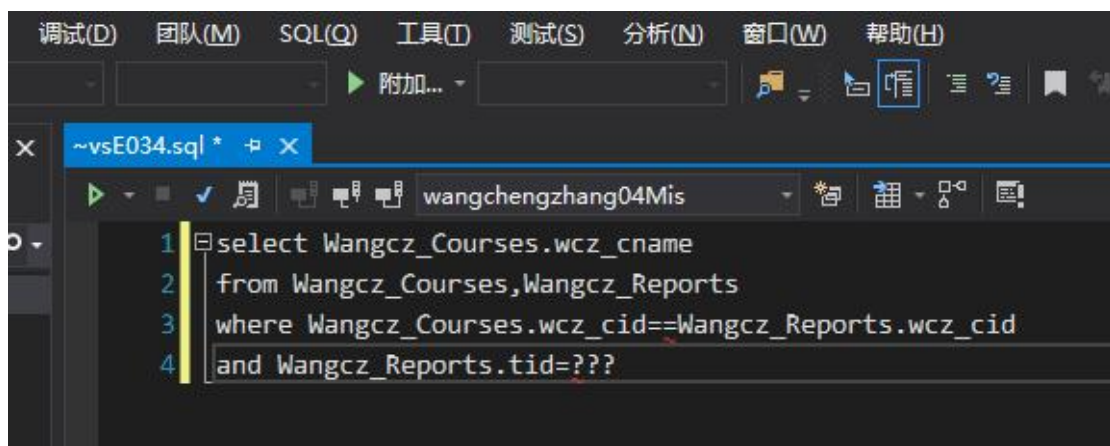
```
select Wangcz_Courses.wcz_cname, Wangcz_Reports.wcz_score
from Wangcz_Courses, Wangcz_Reports
where Wangcz_Reports.cid == Wangcz_Courses.cid
and Wangcz_Reports.sid=???)
```



```
1 select Wangcz_Courses.wcz_cname,Wangcz_Reports.wcz_score
2 from Wangcz_Courses,Wangcz_Reports
3 where Wangcz_Reports.cid == Wangcz_Courses.cid
4 and Wangcz_Reports.sid=???
```

7)教师任课查询 SQL 语句:

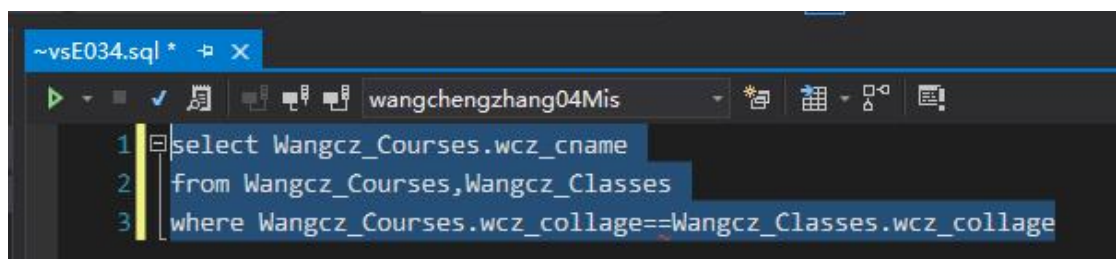
```
select Wangcz_Courses.wcz_cname
from Wangcz_Courses,Wangcz_Reports
where Wangcz_Courses.wcz_cid==Wangcz_Reports.wcz_cid
and Wangcz_Reports.tid=???
```



```
1 select Wangcz_Courses.wcz_cname
2 from Wangcz_Courses,Wangcz_Reports
3 where Wangcz_Courses.wcz_cid==Wangcz_Reports.wcz_cid
4 and Wangcz_Reports.tid=???
```

8)班级课程开设查询 SQL 语句:

```
select Wangcz_Courses.wcz_cname
from Wangcz_Courses,Wangcz_Classes
where Wangcz_Courses.wcz_collage==Wangcz_Classes.wcz_collage
```



```
1 select Wangcz_Courses.wcz_cname
2 from Wangcz_Courses,Wangcz_Classes
3 where Wangcz_Courses.wcz_collage==Wangcz_Classes.wcz_collage
```

6、应用系统开发与试运行

6.1 开发平台和开发环境介绍

开发平台 Windows

开发环境 Python3\Node.js

6.2 系统功能图

6.3 系统功能界面截图

1) 登陆界面

在正式进入系统前，系统会要求用户输入账号和密码进行登录。

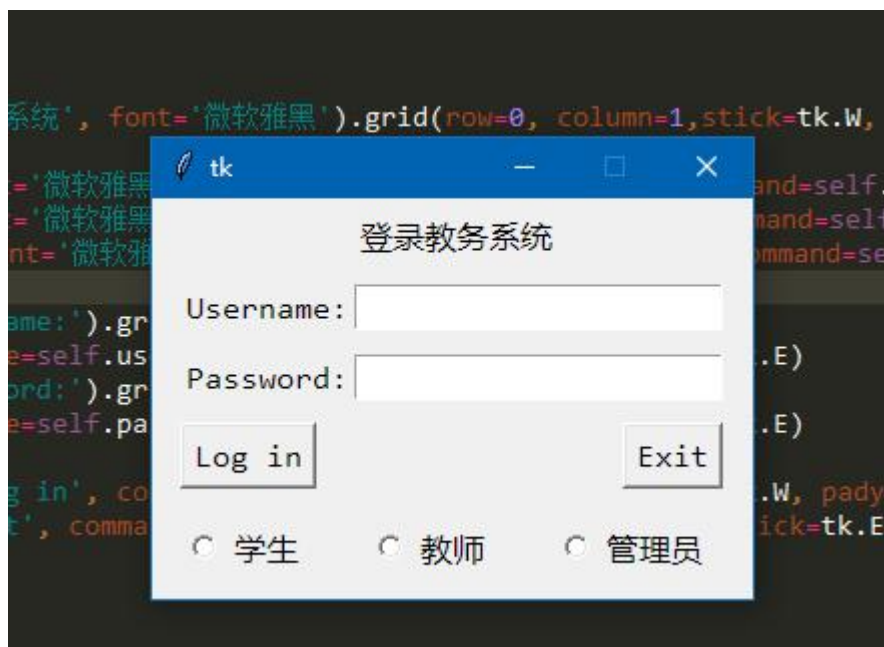


图 6-1 系统登录界面

2) 学生信息录入与修改功能

```
def find_db(sql):  
    import pymysql  
    conn = pymysql.connect(  
        host='127.0.0.1',  
        user='Administator',  
        password='',  
        database='Wangchengzhang04Mis',  
        charset="utf8")  
    cur = conn.cursor()  
    cur.execute(sql)  
    result = cur.fetchall()  
    return result
```

3)

4)

```

self.TabStrip1.add(self.TabStrip1__Tab3, text='插入数据')

def change_db(self, mode):
    info = self.tree_1.get_children()
    data = {
        'mode': mode,
        'info': info
    }
    sock = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
    sock.connect(address)
    bytedata = repr(data).encode()
    sock.send(bytedata)

```

5)

```

self.insertButton = tk.Button(self.subFrame, text='插入数据', font='微软雅黑', width=self.buttonwidth, command=Lambda : self.change_db(mode='insert'))
self.insertButton.pack(anchor=tk.N, side=tk.LEFT, padx=10, pady=2)
self.updateButton = tk.Button(self.subFrame, text='更新数据', font='微软雅黑', width=self.buttonwidth, command=Lambda : self.change_db(mode='update'))
self.updateButton.pack(anchor=tk.N, side=tk.LEFT, padx=10, pady=2)
self.deleteButton = tk.Button(self.subFrame, text='删除数据', font='微软雅黑', width=self.buttonwidth, command=Lambda : self.change_db(mode='delete'))
self.deleteButton.pack(anchor=tk.N, side=tk.LEFT, padx=10, pady=2)

```

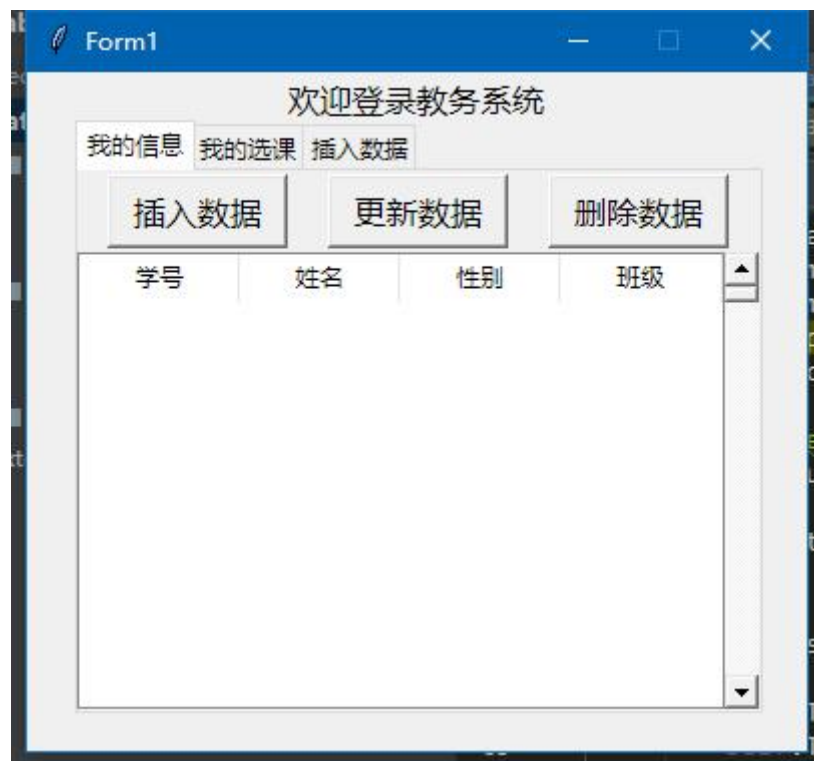


图 6-2 学生信息

7、实验总结

7.1 遇到的问题和解决的办法

7.2 系统设计的不足

7.3 进一步改进思路和体会