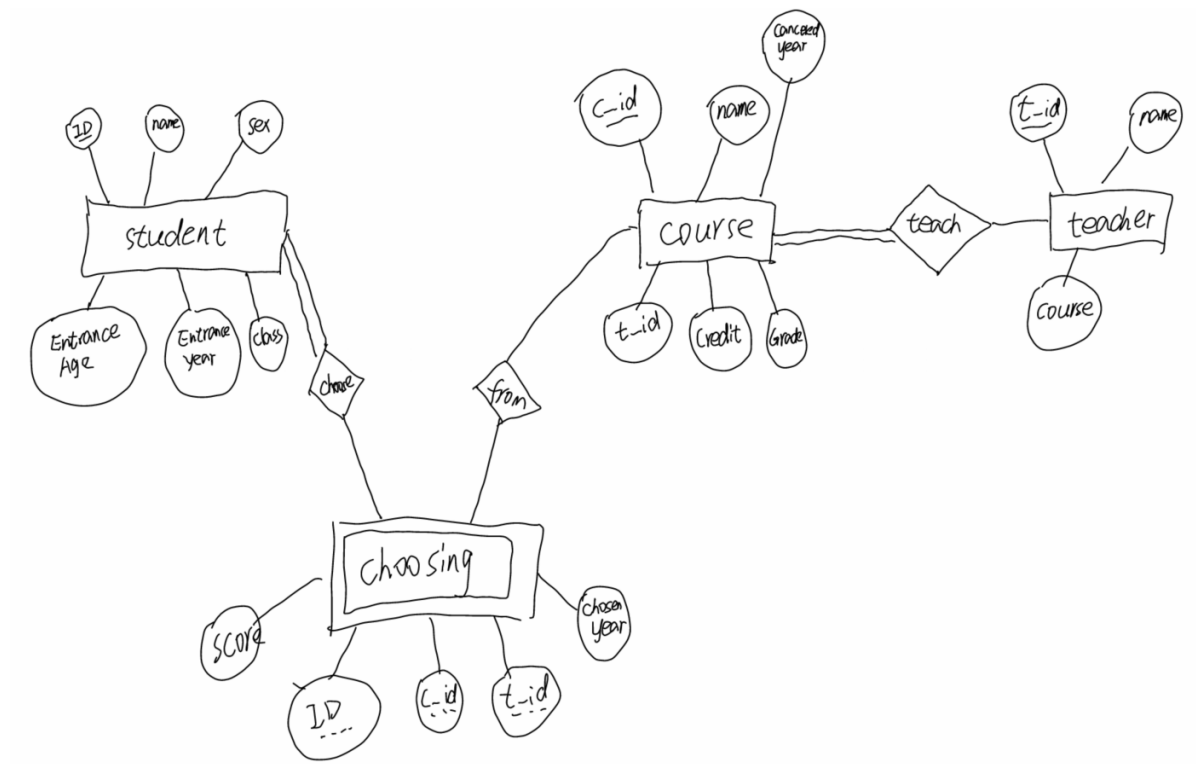E-R
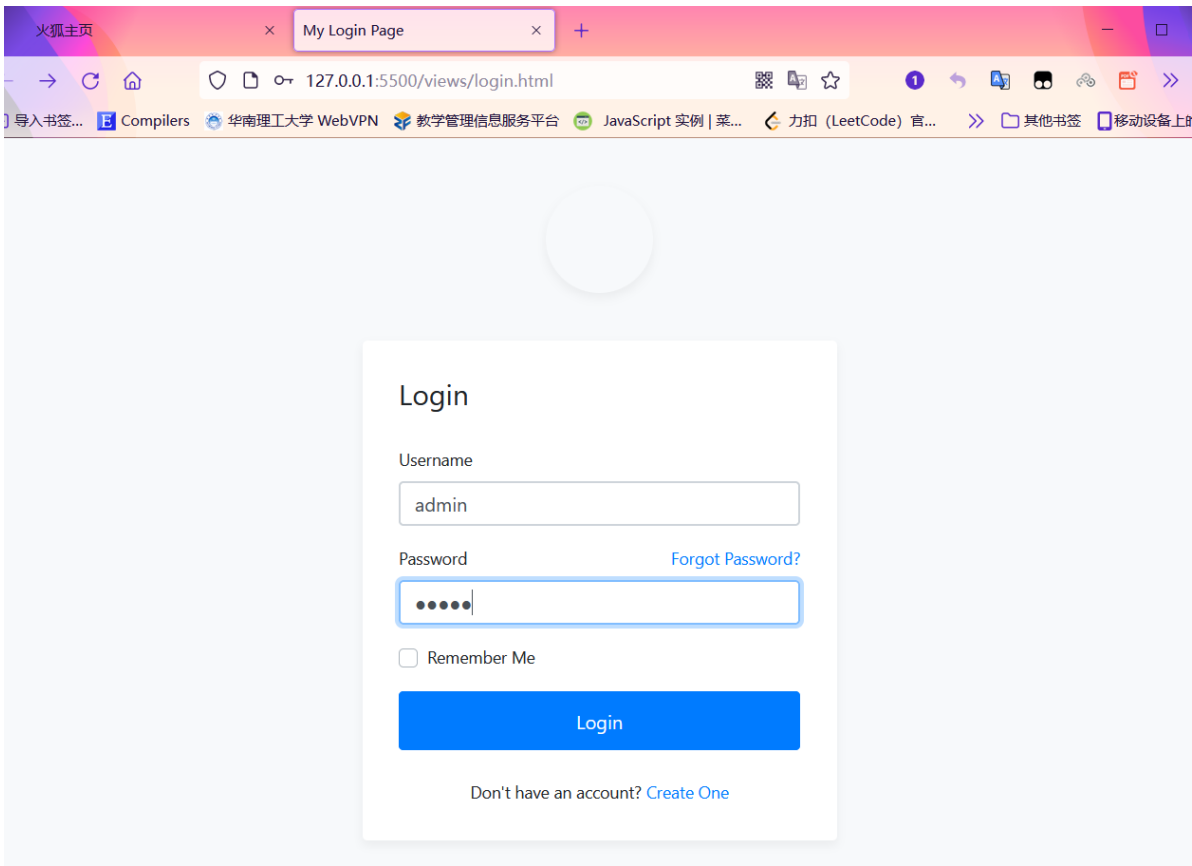

1. login

for administrator, username is admin and password is admin

for student, username is studen tand password is student

for teacher, username is teacher and password is teacher

2. GUI

2.1admin.html

student

get all

Teacher

Course

Choose

Score

id	name	sex	entrance_age	entrance_year	class	operations
2019123458	Cooper	Male	19	2019	class1	delete/update
2020143456	Easton	Male	20	2020	class1	delete/update
2020123457	Bryson	Male	20	2020	class1	delete/update
2020123456	Sebastian	Male	18	2020	class1	delete/update
2020123056	Jordan	Male	20	2020	class2	delete/update
2019123459	Luke	Male	20	2019	class1	delete/update

id	name	sex	entrance_age	entrance_year	class	course_id	teacher_id	chosen_year	score

insert

getAll insert

id	name

id	name	s	er_id	chosen_y

Insert Student

ID

name

sex

entrance_age

entrance_year

class

Insert   Cancel

search

	getAll	insert								

Search Student

ID

2019123458

name

Search	Cancel

id						rations
2019123458						te/update
2020143456						te/update
2020123457						te/update
2020123456						te/update
2020123056	Jordan	Male	20	2020	class2	delete/update
2019123459	Luke	Male	20	2019	class1	delete/update

id	name	sex	entrance_age	entrance_year	class	course_id	teacher_id	chosen_year
2019123458	Cooper	Male	19	2019	class1	0000001	00001	2020
2019123458	Cooper	Male	19	2019	class1	0000002	00002	2020
2019123458	Cooper	Male	19	2019	class1	0000003	00003	2020

update

Update Student

insert

ID

2019123458

					erations

name

23458 | | | | | | ete/update

sex

43456 | | | | | | ete/update

23457 | | | | | | ete/update

entrance_age

23456 | | | | | | ete/update

23056 | | | | | | ete/update

entrance_year

23459 | | | | | | ete/update

class

name					er_id	chos

Update	Cancel

delete

teacher

get all

teacher_id	name	courses
00001	Robert	A
00002	Myles	B
00003	Wyatt	C
00004	Bennett	D
00005	Landon	E
00006	Theodore	F
00007	Aiden	Q
00008	Carson	R
00009	Noah	T
00010	Aaron	Y

Student | Teacher | Course | Choose | Score

getAll | search

search

Student | Teacher | Course | Choose | Score

getAll | search

teacher_id

00001

Search

ID

00001

name

Search   Cancel

course and choose are similar to student

score

find a student scores based on id or name

[{"score":90,"course_id":"0000001"},
{"score":85,"course_id":"0000002"},
{"score":85,"course_id":"0000003"},
{"score":91,"course_id":"0000004"}]

Student | ID:
Teacher | 2019123458
Course | Name:
Choose
Score | Search

Search Results

A Student AVG By Id

find a student average score

ID:

Name:

Search

A Student AVG By Id

2019123458

Search

Search Results

[{"avg":"87.7500000000000000"}]

find all students average score

ID:

Name:

Search

A Student AVG By Id

Search

All Students AVG | Search

Search Results

[{"avg":"87.0000000000000000"}]

find avg score in the same course & class

ID:

Name:

Search

A Student AVG By Id

Search

All Students AVG   Search

The Same Course AVG   Search

Search Results

[{"course_name":"D","avg":"90.0000000000000000"},
{"course_name":"A","avg":"87.3333333333333333"},
{"course_name":"C","avg":"86.0000000000000000"},
{"course_name":"B","avg":"84.5000000000000000"}]

rs 🌐 华南理工大学 WebVPN   教学管理信息服务平台   JavaScript 实例 | 菜...   力扣 (LeetCode) 官...   ≫   其他书签

:

ame:

Search

Search Results

[{"class":"class2","avg":"86.0000000000000000"},
{"class":"class1","avg":"87.1250000000000000"}]

2.2 teacher.html

Choose	getAll	search

s_id	course_id	teacher_id	chosen_year	score
2019123458	0000001	00001	2020	90
2019123458	0000002	00002	2020	85
2019123458	0000003	00003	2020	85
2019123458	0000004	00004	2020	91
2019123459	0000004	00004	2020	89
2019123459	0000003	00003	2020	87
2019123459	0000002	00002	2020	84
2019123459	0000001	00001	2020	86
2020123056	0000001	00001	2020	86

2.3 student.html

Choose	getAll	insert	search

s_id	course_id	teacher_id	chosen_year	score	
2019123458	0000001	00001	2020	90	delete/update
2019123458	0000002	00002	2020	85	delete/update
2019123458	0000003	00003	2020	85	delete/update
2019123458	0000004	00004	2020	91	delete/update
2019123459	0000004	00004	2020	89	delete/update
2019123459	0000003	00003	2020	87	delete/update
2019123459	0000002	00002	2020	84	delete/update
2019123459	0000001	00001	2020	86	delete/update
2020123056	0000001	00001	2020	86	delete/update

3.CODE

In this file, there are main sql statements

insert

```
export function insert(obj,table)
{
    var sql;
```

```
    if(table=='student')
    {
        sql="insert into student(id,name,sex,entrance_age,entrance_year,class)
values('"+obj.id+"','"+obj.name+"','"+obj.sex+"',"
        +obj.entrance_age+","+obj.entrance_year+",'"+obj.clas+"')";
    }
    else if(table=="teacher")
    {
        sql="insert into teacher(teacher_id,name,courses)
values('"+obj.t_id+"','"+obj.name+"','"+obj.courses+"')";
    }
    else if(table=='choose')
    {
        sql="insert into choose(id,course_id,teacher_id,chosen_year,score)
values('"+obj.id+"', '"+obj.c_id+"' ,'"+obj.t_id+"',"
        +obj.chosen_year+","+obj.score+")";
    }
    else if(table=='course')
    {
        sql="insert into
course(course_id,teacher_id,credit,grade,canceled_year,course_name)
values('"+obj.c_id+"', '"+obj.t_id+"' ,'"+obj.credit+"','"
        +obj.grade+"',"+obj.canceled+",'"+obj.name+"')";
    }
    pool.connect(function(err,client,done){
        if(err){console.error("Error connecting");}
        client.query(sql,function(err,result){
            if(err){console.error("Error",err);}
            console.log(result.rows);
            done();
        })
    })
}
```

delete

```
export function del(obj,table)
{
    var sql;

    if(table=='student')
    {
        sql="delete from student where id = '"+obj.id+"'";
    }
    else if(table=="teacher")
    {
        sql="delete from teacher where teacher_id = '"+obj.t_id+"'";
    }
    else if(table=='choose')
    {
        sql="delete from choose where id='"+obj.id+"' and
course_id='"+obj.c_id+"'";
    }
    else if(table=='course')
    {
        sql="delete from course where course_id = '"+obj.c_id+"'";
    }
```

```
    pool.connect(function(err,client,done){
        if(err){console.error("Error connecting");}
        client.query(sql,function(err,result){
            if(err){console.error("Error",err);}
            done();
        })
    })
}
```

update

```
export function update(obj,table)
{
    var sql;

    if(table=='student')
    {
        sql="update student set
name='"+obj.name+"',sex='"+obj.sex+"',entrance_year="+obj.entrance_year+",entran
ce_age="+obj.entrance_age+",class='"+obj.clas+"' where id='"+obj.id+"'";
    }
    else if(table=="teacher")
    {
        sql="update teacher set
name='"+obj.name+"',courses='"+obj.courses+"'"+"where
teacher_id='"+obj.t_id+"'";
    }
    else if(table=='choose')
    {
        sql="update choose set score = "+obj.score+" where id='"+obj.id+"'and
course_id= '"+obj.c_id+"'";
    }
    else if(table=='course')
    {
        sql="update course set teacher_id = '"+obj.t_id+"',
credit='"+obj.credit+"', grade='"+obj.grade+"',
canceled_year="+obj.canceled+",course_name='"+obj.name+"'"+" where
course_id='"+obj.c_id+"'";
        console.log(sql);
    }
    pool.connect(function(err,client,done){
        if(err){console.error("Error connecting");}
        client.query(sql,function(err,result){
            if(err){console.error("Error",err);}
            done();
        })
    })
}
```

get all

```
function getAll(table,callback)
{
    var sql;
    if(table=="student"){
```

```
            sql="select * from student";
        }
        else if(table=="teacher")
        {
            sql ="select * from teacher";
        }
        else if(table=="course")
        {
            sql="select * from course";
        }
        else if(table=="choose")
        {
            sql ="select * from choose";
        }
        pool.connect(function(err,client,done){
            if(err){console.error("Error connecting");}
            client.query(sql,function(err,result){
                if(err){
                    console.error("Error",err);
                    return false;
                }
                else
                {
                    callback(result);
                }
                done();
                //return result.rows;
            })
        })
    }
```

find student and the course he/she chooses by id

```
export function findStuCourseById(id,callback)
{

    var sql ="select * from student as s, choose as c where s.id = c.id and s.id
= '"+id+"'";

    pool.connect(function(err,client,done){
        if(err){console.error("Error connecting");}
        client.query(sql,function(err,result){
            if(err){
                console.error("Error",err);
                return false;
            }
            else
            {
                callback(result);
            }
            done();
            //return result.rows;

        })
    })
}
```

find student and course he/she chooses by name

```
export function findStuCourseByName(name,callback)
{
    var sql ="select * from student as s, choose as c where s.id = c.id and s.id
= '"+name+"'";
    pool.connect(function(err,client,done){
        if(err){console.error("Error connecting");}
        client.query(sql,function(err,result){
            if(err){console.error("Error",err);}
            else
            {
                callback(result);
            }
            console.log(result);
            done();
        })
    })
}
```

find all about students and course chosen

```
export function findStuCourseAll(callback)
{
    var sql ="select * from student as s, choose as c where s.id = c.id ";
    pool.connect(function(err,client,done){
        if(err){console.error("Error connecting");}
        client.query(sql,function(err,result){
            if(err){console.error("Error",err);}
            else
            {
                callback(result);
            }
            console.log(result);
            done();
        })
    })
}
```

find score

```
export function findScore(para,kind,callback)
{
    var sql;
    //console.log("aa",para,kind);
    if(kind =="id")
    {
        console.log("111");
        sql ="select score,course_id from choose where id='"+para+"'";
        console.log(sql);
    }
    else if(kind =="name")
    {
        sql ="select score,course_id from choose,student where student.name=
'"+para+"'";
```

```
    }
    else
    {
        sql="select s.id,c.score,course_id from student as s, choose as c where
s.id = c.id";
    }
}
```

find course information

```
export function findCourse(para,kind,callback)
{
    var sql;
    if(kind =="id")
    {
        sql ="select * from course where course_id='"+para+"'";
    }
    else if(kind =="name")
    {
        sql ="select * from course where course_name='"+para+"'";
    }
    else
    {
        sql ="select * from course";
    }
}
```

##

find choose information

```
export function findChoose(para,kind,callback)
{
    var sql;
    if(kind =="id")
    {
        sql ="select * from choose where course_id='"+para+"'";
    }
    else if(kind =="name")
    {
        sql ="select * from choose where course_name='"+para+"'";
    }
    else
    {
        sql ="select * from choose";
    }
}
```

find teacher information

```
export function findTeacher(para,kind,callback)
```

```
{
    var sql;
    if(kind =="id")
    {
        sql ="select * from teacher where teacher_id='"+para+"'";
    }
    else if(kind =="name")
    {
        sql ="select * from teacher where name='"+para+"'";
    }
    else
    {
        sql ="select * from teacher";
    }
}
```

find teacher and course he/she teaches

```
export function findTeachCourse(para,kind,callback)
{
    var sql;
    if(kind =="id")
    {
        sql ="select * from course where teacher_id='"+para+"'";
    }
    else if(kind =="name")
    {
        sql ="select * from course as c,teacher as t where
t.teacher_id=c.teacher_id and t.name='"+para+"'";
    }
    else
    {
        sql ="select * from course";
    }}
```

average score

```
export function findAvgScore(para,kind,callback)
{
    var sql;
    if(kind=="stu")//a student
    {
        sql ="select avg(score) from choose where choose.id='"+para+"'";
    }
    else if(kind=="all")//all student
    {
        sql = "select avg(score) from choose";
    }
    else if(kind=="class")//in the same class
    {
        sql = "select class, avg(c.score) from choose as c, student as s where
c.id=s.id group by class";
```

```
        }
    else if(kind=="course")//in the same course
    {
        sql = "select course_name, avg(choose.score) from choose, course where
course.course_id=choose.course_id group by course_name";
    }}
```