实验五 综合性程序设计—简单学生信息管理系统

实验日期： 2017 年 12 月 1 日 班级： 软件 1502

学号（后四位）：\_\_0411\_\_\_ 姓名：\_\_\_\_陈德玺\_\_\_\_\_\_\_\_\_\_ 成绩：

一.实验目的

1.培养学生进行简单网站的整体开发能力；

2.掌握网站开发所需要的基本知识。

二．实验内容

以学生信息管理系统为例，编写能够满足如下条件的程序：

a. 管理员可以对学生、老师和课程进行增加、删除、查询。

b. 学生登录只能查看、修改自己的个人信息，查看自己所有课程成绩。

c. 班级任课教师登录后，可浏览对应课程学生基本信息和相应的成绩信息。如是java、os、math任课老师，可修改、登记相应课程的信息

**备注：建议大家将《面向对象程序设计》的课程设计题目作为本次综合实验，有三种身份的人群，不同身份的人群拥有不同的权限。**

三．实验总结

**需求分析：**

1. 管理员可以对学生、老师和课程进行增加、删除、查询。

管理员拥有系统的最高权限，在系统中可以进行对于系统中基础数据的修改。

1. 学生登录只能查看、修改自己的个人信息，查看自己所有课程成绩。

学生相当于系统的普通用户，在系统中进行最基本的查询操作、修改个人信息。

1. 班级任课教师登录后，可浏览对应课程学生基本信息和相应的成绩信息。如是java、os、math任课老师，可修改、登记相应课程的信息

教师对于系统中的个人班级信息具有修改权力，对于课程下的学生也可进行基本的查询操作。Java、os、math可以修改相应课程的信息。

**开发环境：**

Java版本：jdk 9.0.1

Sql server：windows

Tomcat：tomcat 9

Eclipse：Eclipse Jee Oxygen

**概要设计：**

1. 系统中三种角色的存在使用js控制用户的角色权限，为相应的用户自动生成不同的操作页面。
2. 数据库设计中对于三种角色的基本信息表，以及教师的课程信息表，每种课程的选课信息表，选课成绩表，需要实现对于成绩的录入功能等等。
3. Jsp项目的模式使用mvc模式，需要连接数据库方面使用JDBC第三方包进行连接，需要使用权限控制功能时，使用js和session的方法实现。
4. 项目的界面使用bootshtrap中文模板，生成相应的网页css、js、jqury模板，引入项目中。
5. 数据库的建立和初始化需要先检测数据库在系统中是否已经存在，如果不存在时则执行sql初始化文件，存在则启动程序。

**详细设计：**

1. 登陆页面(login.jsp)
2. 注册页面(registration.jsp)
3. 欢迎页面(welcome.jsp)
4. 查询页面(view.jsp)
5. 添加页面(add.jsp)
6. 管理员查看用户信息(view\_user.jsp)
7. 管理员修改用户信息(admin\_modify.jsp)
8. 教师查看课程信息(view\_course.jsp)
9. 教师录入学生成绩(mark\_grade.jsp)

10、

**数据库简单设计;**

1、用户表（user）：

|  |  |  |  |
| --- | --- | --- | --- |
| 用户ID | 用户名 | 用户密码 | 用户角色 |
| Userid char(20) | Username char(40) | Userpasswd char(20) | Userrole char(20) |

2、班级表（classes）：

|  |  |  |  |
| --- | --- | --- | --- |
| 用户ID | 学生姓名 | 学生年龄 | 学生性别 |
| Usernumber char(20) | Username char(40) | Userage smallint | Usersex char(20) |

3、课程表（course）：

|  |  |  |
| --- | --- | --- |
| 课程ID | 课程姓名 | 教师ID |
| Coursenumber char(20) | Coursename char(40) | TeacherID char(20) |

4、选课信息表（choose\_course）：

|  |  |  |
| --- | --- | --- |
| 学生ID | 选修课程 | 课程成绩 |
| Studentnumber char(20) | choose\_courseid char(20) | course\_grade char(20) |

数据库初始化代码：

create database student

go

use student

create table userinfo /\* 用户表 \*/

(Userid char(20) primary key,

Username char(20) unique,

Userpasswd char(20),

Userrole char(20),

);

create table classes /\* 软件班级信息表 \*/

(Usernumber char(20) primary key,

Username char(40) not null, /\* 列级完整性约束条件 Cname不能取空值 \*/

Userage smallint,

Usersex char(20),

Foreign key(Usernumber) references userinfo(Userid) /\* Cpno 是外码,被参照表是Course, 被参照列是Cno \*/

);

create table course /\* 课程表 \*/

(Coursenumber char(20) primary key,

Coursename char(40) not null, /\* 列级完整性约束条件 Coursename不能取空值 \*/

TeacherID char(20),

Foreign key(TeacherID) references userinfo(Userid) /\* Cpno 是外码,被参照表是Course, 被参照列是Cno \*/

);

create table choose\_course /\* 选课信息表 \*/

(Studentnumber char(20) not null,

choose\_courseid char(20) not null, /\* 列级完整性约束条件 Coursename不能取空值 \*/

course\_grade char(20),

primary key(Studentnumber,choose\_courseid),

Foreign key(choose\_courseid) references course(Coursenumber), /\* Cpno 是外码,被参照表是Course, 被参照列是Cno \*/

Foreign key(Studentnumber) references userinfo(Userid)

);

insert into userinfo values ('2015001', '王雪涛', '123456', 'teacher');

insert into userinfo values ('2015009', '郑丽萍', '123456', 'teacher');

insert into userinfo values ('2015010', '刘於勋', '123456', 'teacher');

insert into userinfo values ('2015011', '于景新', '123456', 'teacher');

insert into userinfo values ('2015002', '陈德玺', '123456', 'admin');

insert into userinfo values ('2015003', '不二陈', '123456', 'admin');

insert into userinfo values ('2015004', '马超', '123456', 'student');

insert into userinfo values ('2015005', '黄忠', '123456', 'student');

insert into userinfo values ('2015006', '张飞', '123456', 'student');

insert into userinfo values ('2015007', '赵云', '123456', 'student');

insert into userinfo values ('2015008', '关羽', '123456', 'student');

insert into classes values ('2015001', '王雪涛', '22', 'man');

insert into classes values ('2015004', '马超', '22', 'man');

insert into classes values ('2015005', '黄忠', '22', 'man');

insert into classes values ('2015006', '张飞', '22', 'man');

insert into classes values ('2015007', '赵云', '22', 'man');

insert into classes values ('2015008', '关羽', '22', 'man');

insert into course values ('001', 'java', '2015009');

insert into course values ('002', 'os', '2015010');

insert into course values ('003', 'math', '2015011');

insert into course values ('004', 'test', '2015001');

insert into choose\_course values ('2015004', '001', '100');

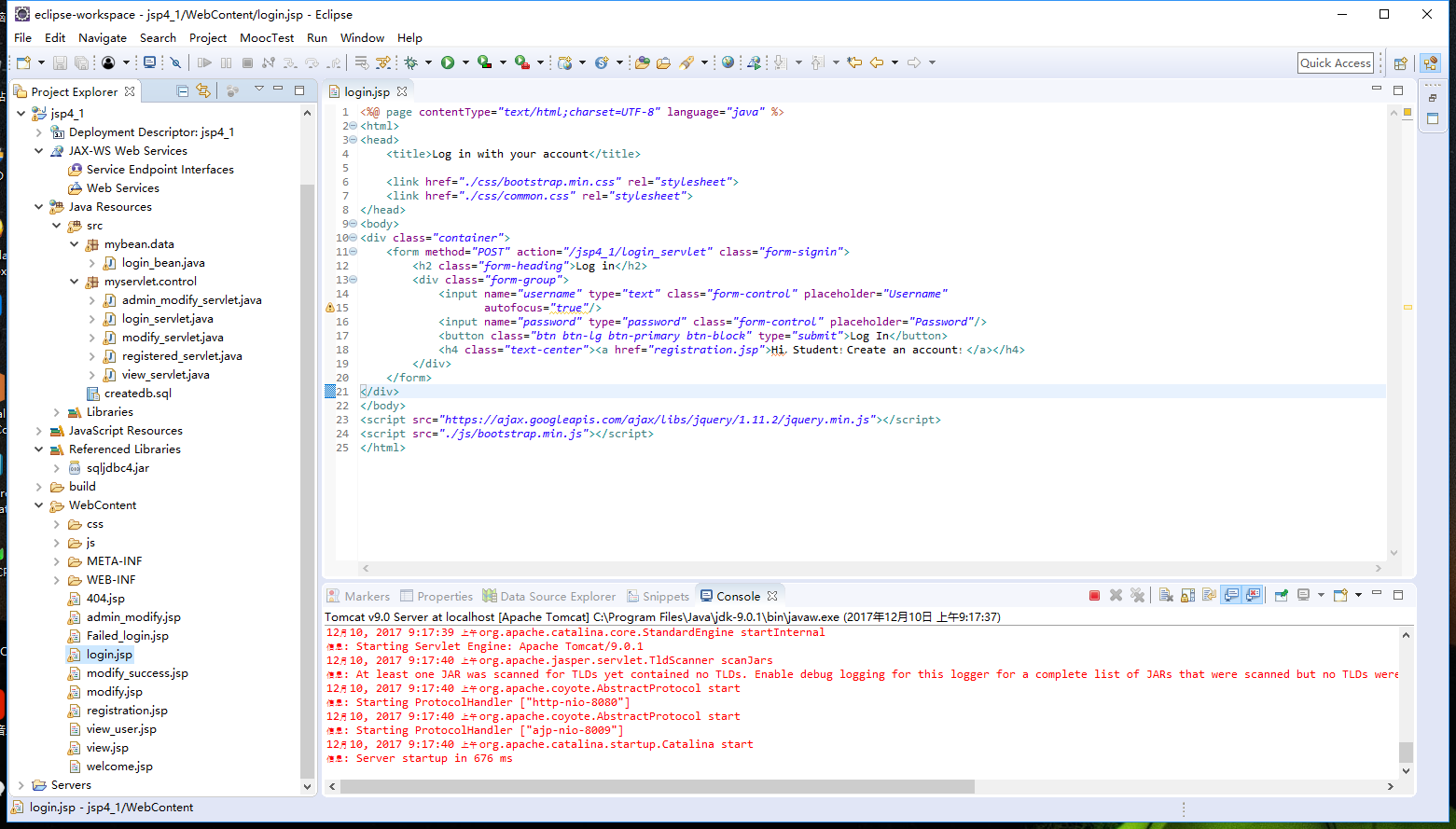
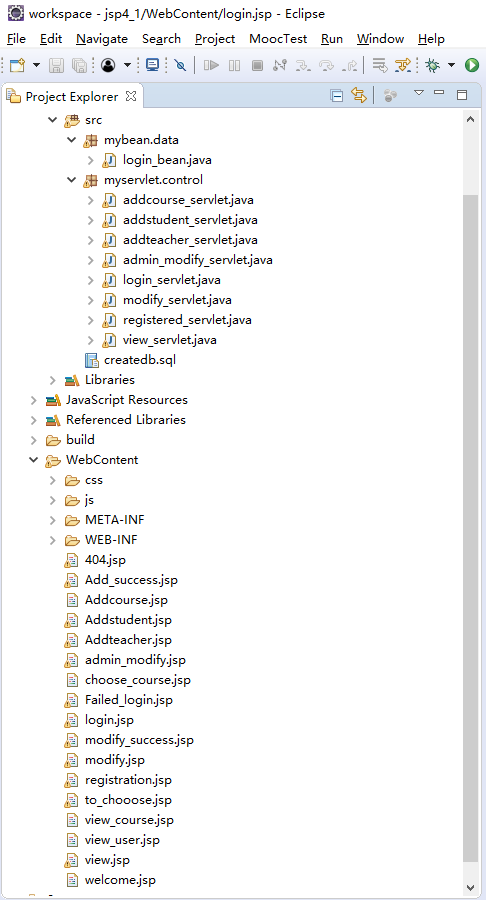
insert into choose\_course values ('2015004', '002', '100');

insert into choose\_course values ('2015005', '003', '100');

insert into choose\_course values ('2015005', '004', '100');

**部分代码：**

1、项目结构



2、Mvc文件名

表1 功能实现对应表

|  |  |  |
| --- | --- | --- |
| 相关代码  功能 | Jsp页面 | Servlet |
| 登陆页面 | Login.jsp | Login\_servlet.java |
| 注册页面 | Registration.jsp | Registered\_servlcet.java |
| 欢迎页面 | Welcome.jsp |  |
| 查看信息 | View.jsp | View\_servlet.java |
| 修改个人信息 | Modify.jsp | Modify\_servlet.java |
| 管理员查看信息 | View\_user.jsp | View\_servlet.java |
| 管理员修改用户信息 | Admin\_modify.jsp | Modify\_servlet.java |
| 登陆失败页面 | Failed\_login.jsp |  |
| url异常页面 | 404.jsp |  |
| 修改信息成功页面 | Modify\_Success.jsp |  |
| 添加学生页面 | Add\_student.jsp | Addstudent\_servlet.java |
| 添加教师页面 | Add\_teacher.jsp | Addteacher\_servlet.java |
| 添加课程页面 | Add\_course.jsp | Addcourse\_servlet.java |
| 添加成功页面 | Add\_success.jsp |  |
| 学生选择课程页面 | Choose\_course.jsp |  |
| 选择课程成功页面 | To\_choose.jsp |  |
| 教师查看课程信息页面 | View\_course.jsp |  |
| 教师录入成绩 | Mark\_grade.jsp | Markgrade\_servlet.java |

3、Js+session实现权限控制（代码实例）

<script type=*"text/javascript"*>

**var** userrole="<%=session.getAttribute("role")%>";

**if** ("admin" == userrole){

document.write("| <a href="+"view.jsp"+">View User</a>");

}

</script>

<script type=*"text/javascript"*>

**var** userrole="<%=session.getAttribute("role")%>";

**if** ("admin" == userrole){

document.write("| <a href="+"view.jsp"+">View User</a>");

document.write("| <a href="+"Addteacher.jsp"+">Add Teacher</a>");

document.write("| <a href="+"Addcourse.jsp"+">Add Course</a>");

document.write("| <a href="+"Addstudent.jsp"+">Add Student</a>");

}

</script>

<script type=*"text/javascript"*>

**var** userrole="<%=session.getAttribute("role")%>";

**if** ("teacher" == userrole){

document.write("| <a href="+"view\_course.jsp"+">course\_info</a>");

document.write("| <a href="+"mark\_grade.jsp"+">mark\_grade</a>");

}

</script>

<script type=*"text/javascript"*>

**var** userrole="<%=session.getAttribute("role")%>";

**if** ("student" == userrole){

document.write("| <a href="+"choose\_course.jsp"+">choose course</a>");

}

</script>

4、bootstrap模板（部分代码）

**if** ("undefined" == **typeof** jQuery)**throw** **new** Error("Bootstrap's JavaScript requires jQuery");

+**function** (a) {

"use strict";

**var** b = a.fn.jquery.split(" ")[0].split(".");

**if** (b[0] < 2 && b[1] < 9 || 1 == b[0] && 9 == b[1] && b[2] < 1)**throw** **new** Error("Bootstrap's JavaScript requires jQuery version 1.9.1 or higher")

}(jQuery), +**function** (a) {

"use strict";

**function** b() {

**var** a = document.createElement("bootstrap"), b = {

WebkitTransition: "webkitTransitionEnd",

MozTransition: "transitionend",

OTransition: "oTransitionEnd otransitionend",

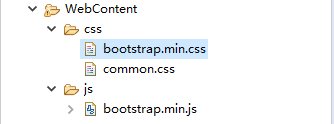
transition: "transitionend"

};

**for** (**var** c **in** b)**if** (void 0 !== a.style[c])**return** {end: b[c]};

**return** !1

}



5、连接数据库

**try** {

Class.*forName*("com.microsoft\*\*.jdbc.sqlserver.\*\*SQLServerDriver");

}

**catch**(Exception e) {}

String DB\_Url="jdbc:sqlserver://localhost:1433;DataName=student?useUnicode=true&characterEncoding=UTF-8";

String user="sa";//登陆数据库

String password="111111";

**try** {

Class.*forName*("com.microsoft.sqlserver.jdbc.SQLServerDriver");

} **catch** (ClassNotFoundException e) {

e.printStackTrace();

}

Connection connection= **null**;

6、css文件实现html元素控制（部分代码）

*.form-signin* *.checkbox* {

margin-bottom: *10px*;

}

*.form-signin* *.checkbox* {

font-weight: *normal*;

}

*.form-signin* *.form-control* {

position: *relative*;

height: *auto*;

-webkit-box-sizing: *border-box*;

-moz-box-sizing: *border-box*;

box-sizing: *border-box*;

padding: *10px*;

font-size: *16px*;

}

*.form-signin* *.form-control:focus* {

z-index: *2*;

}

*.form-signin* **input** {

margin-top: *10px*;

border-bottom-right-radius: *0*;

border-bottom-left-radius: *0*;

}

*.form-signin* **button** {

margin-top: *10px*;

}

*.has-error* {

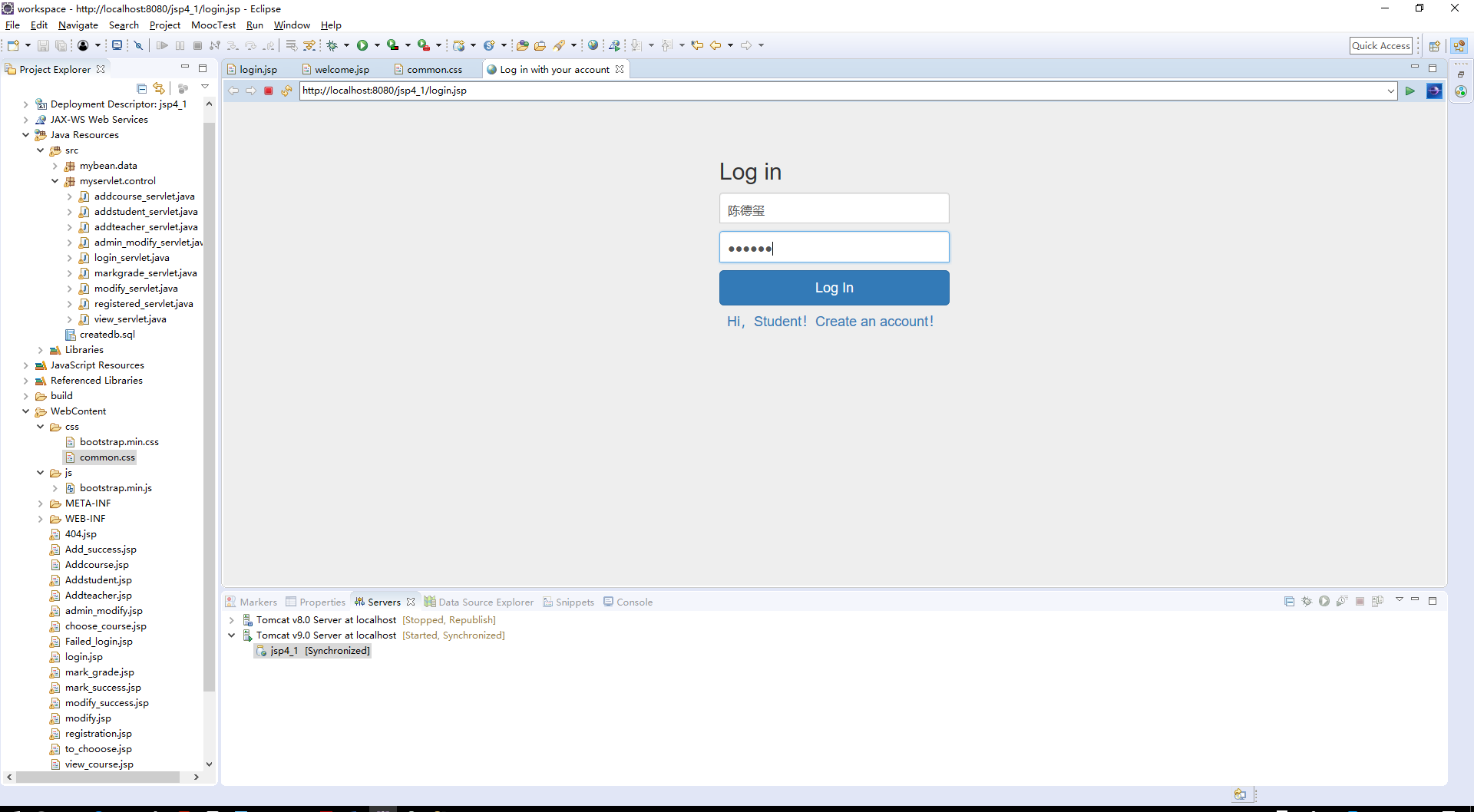
color: *red*

}

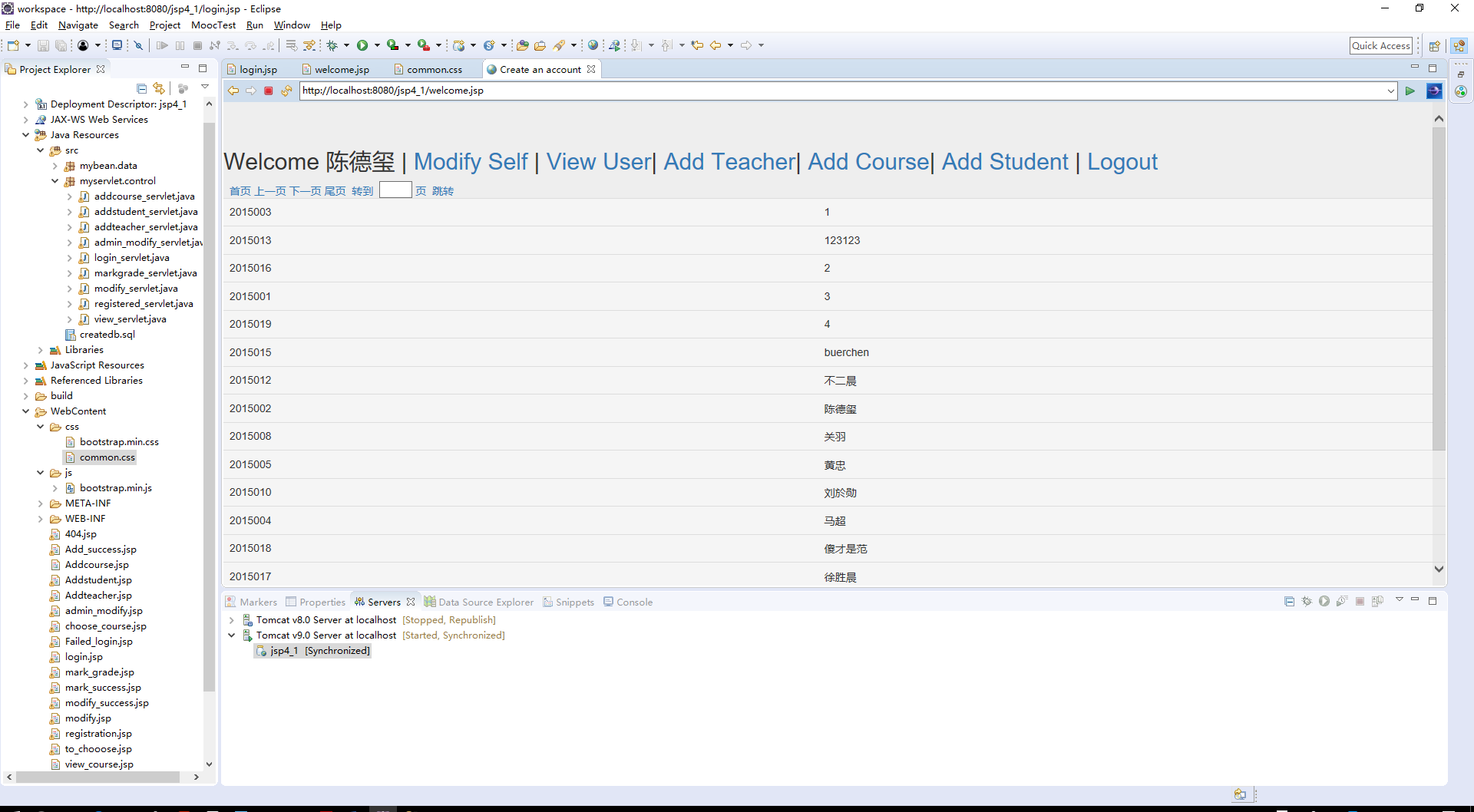
四．运行结果

A、管理员功能部分

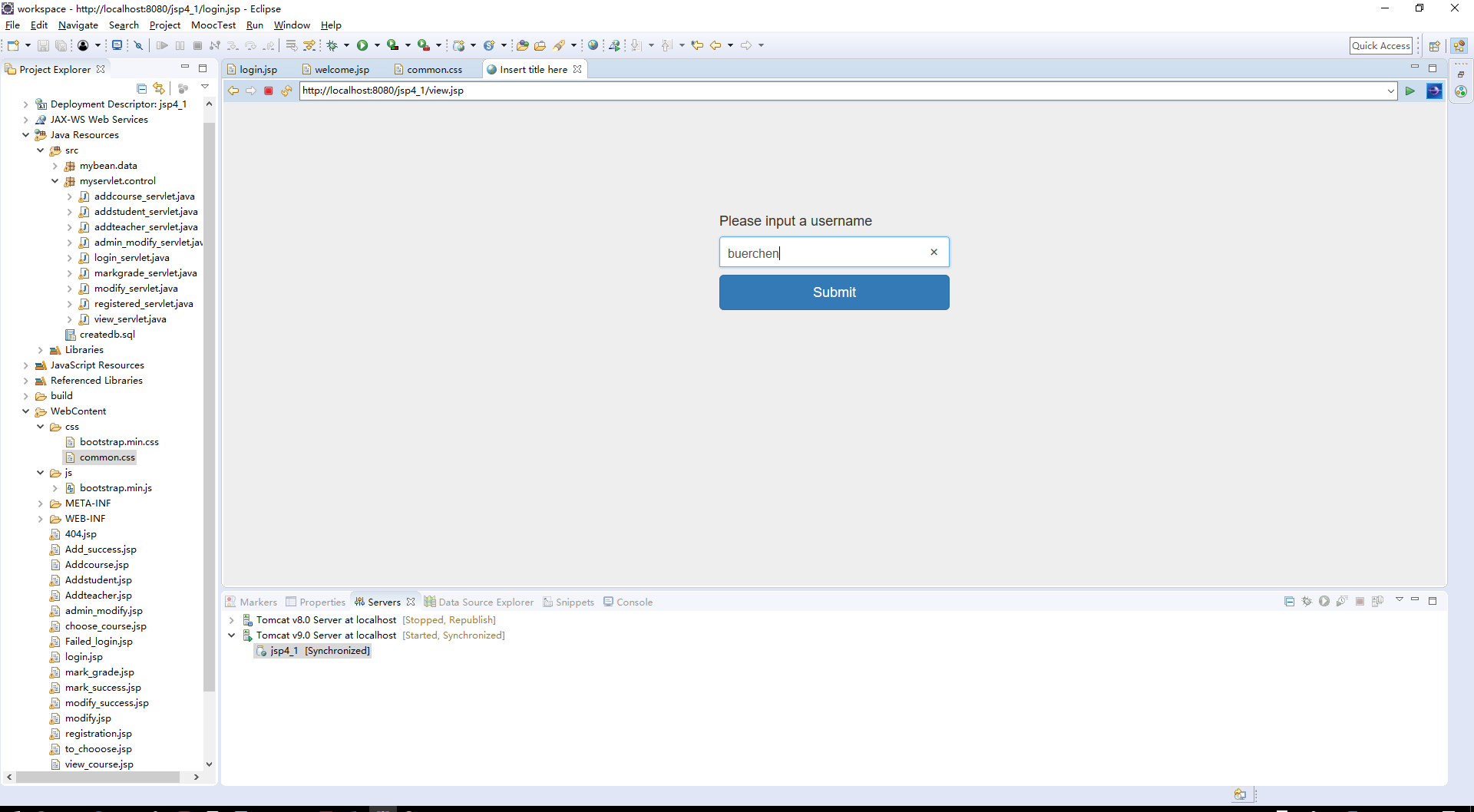
1、登陆页面

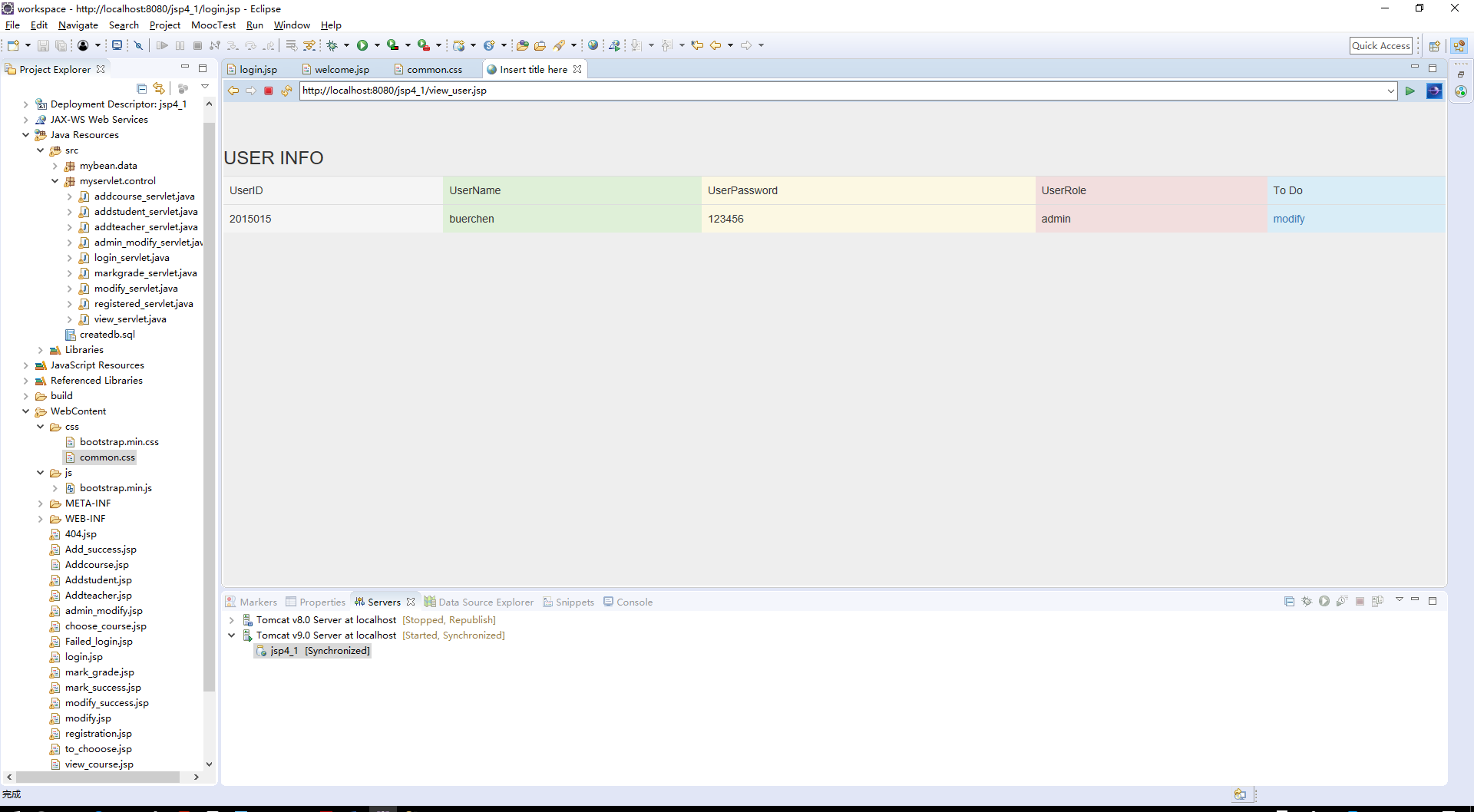


2、管理员欢迎界面

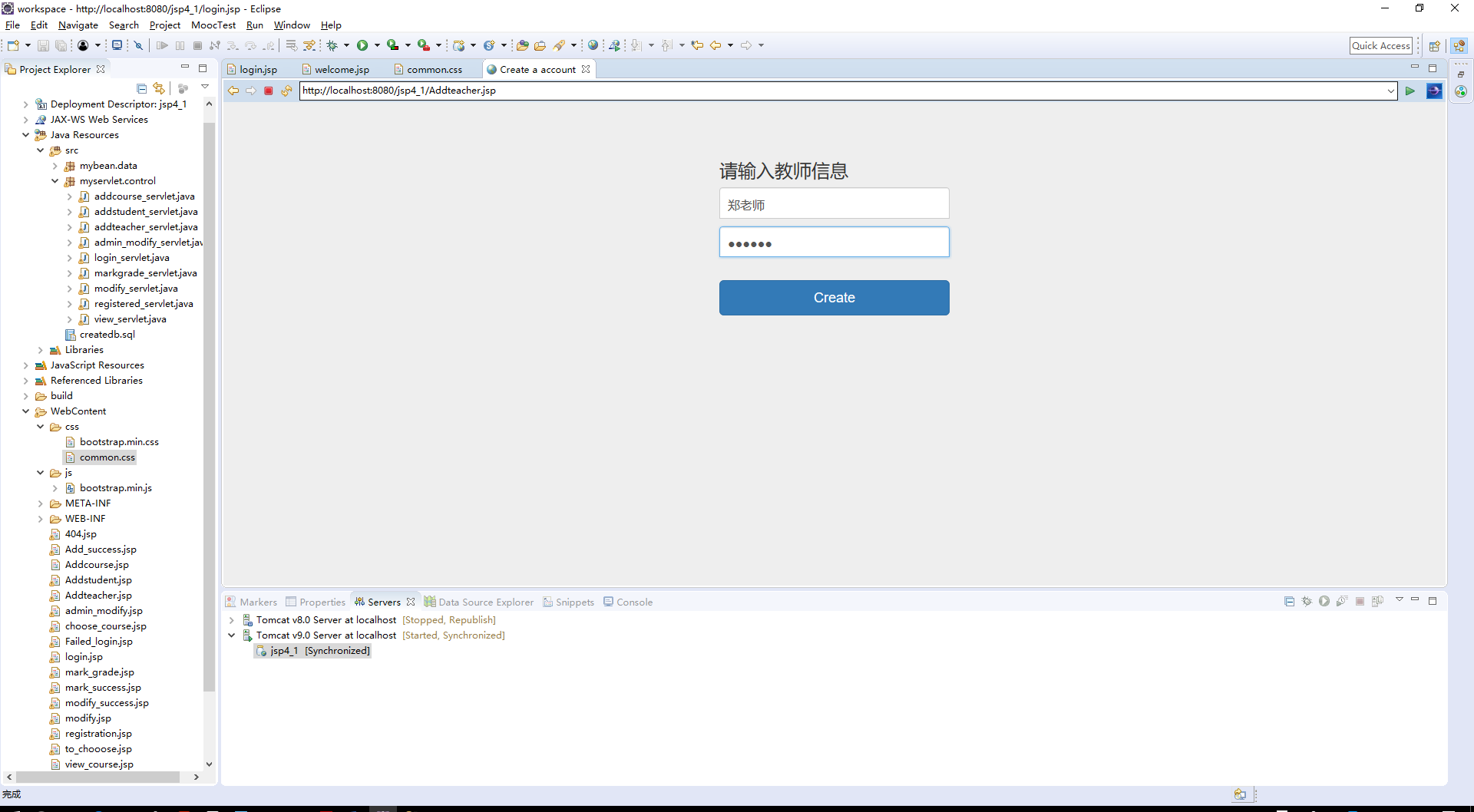


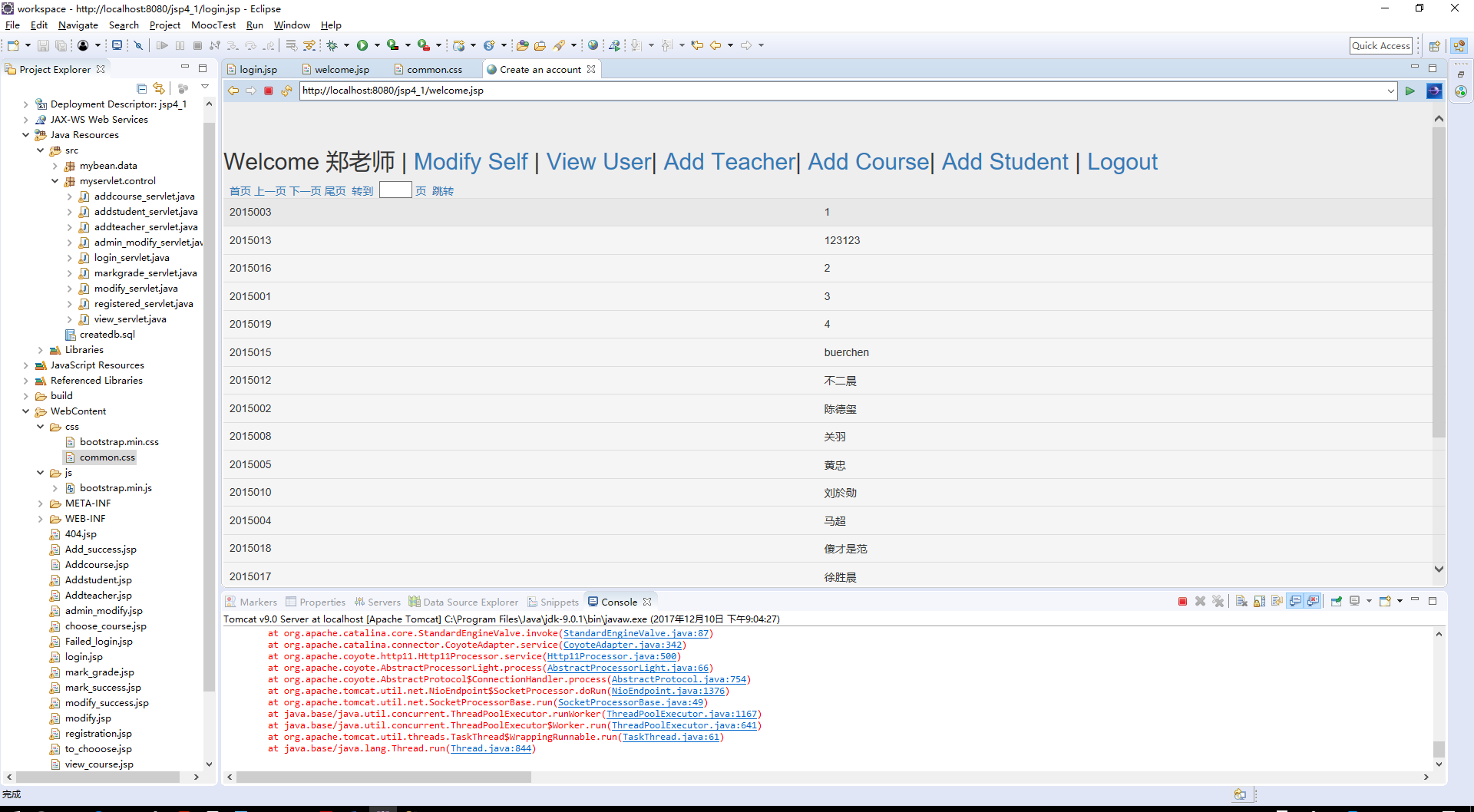
3、管理员查看用户信息





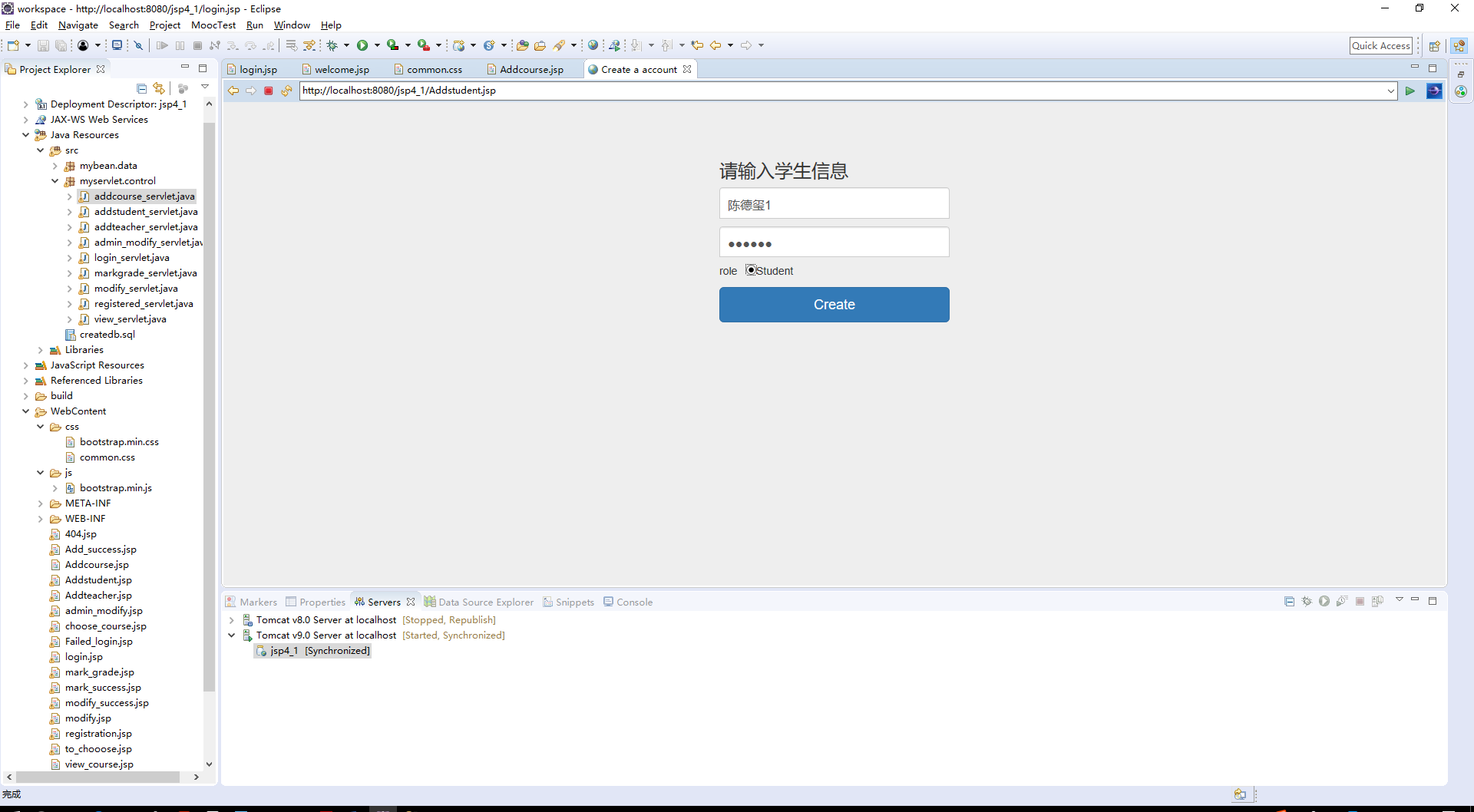
4、管理员添加教师

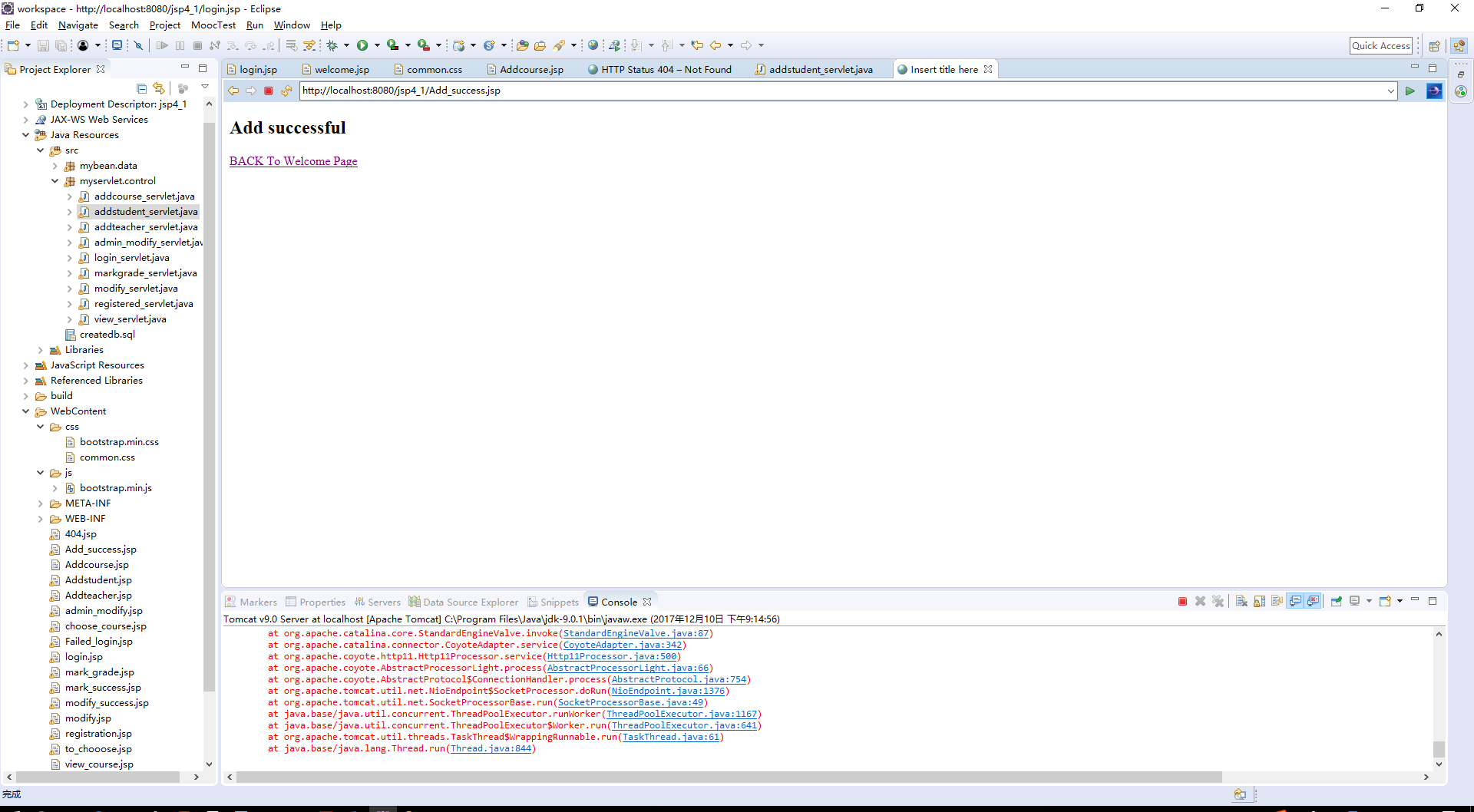




（此错误是由于数据库插入函数没有返回结果集，java9加大了对于异常的处理级别，对于这样的错误会打印到控制台）

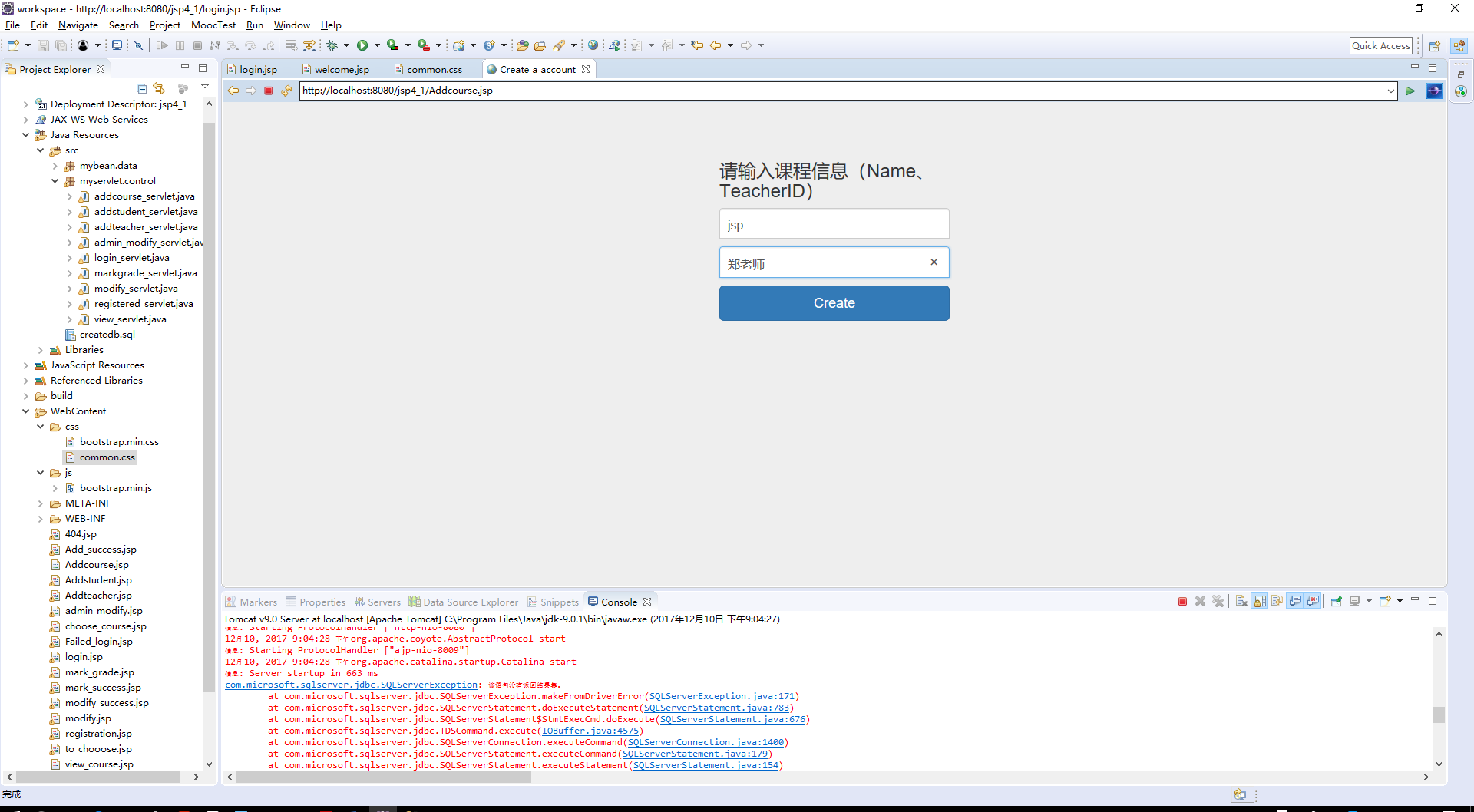
5、管理员添加学生





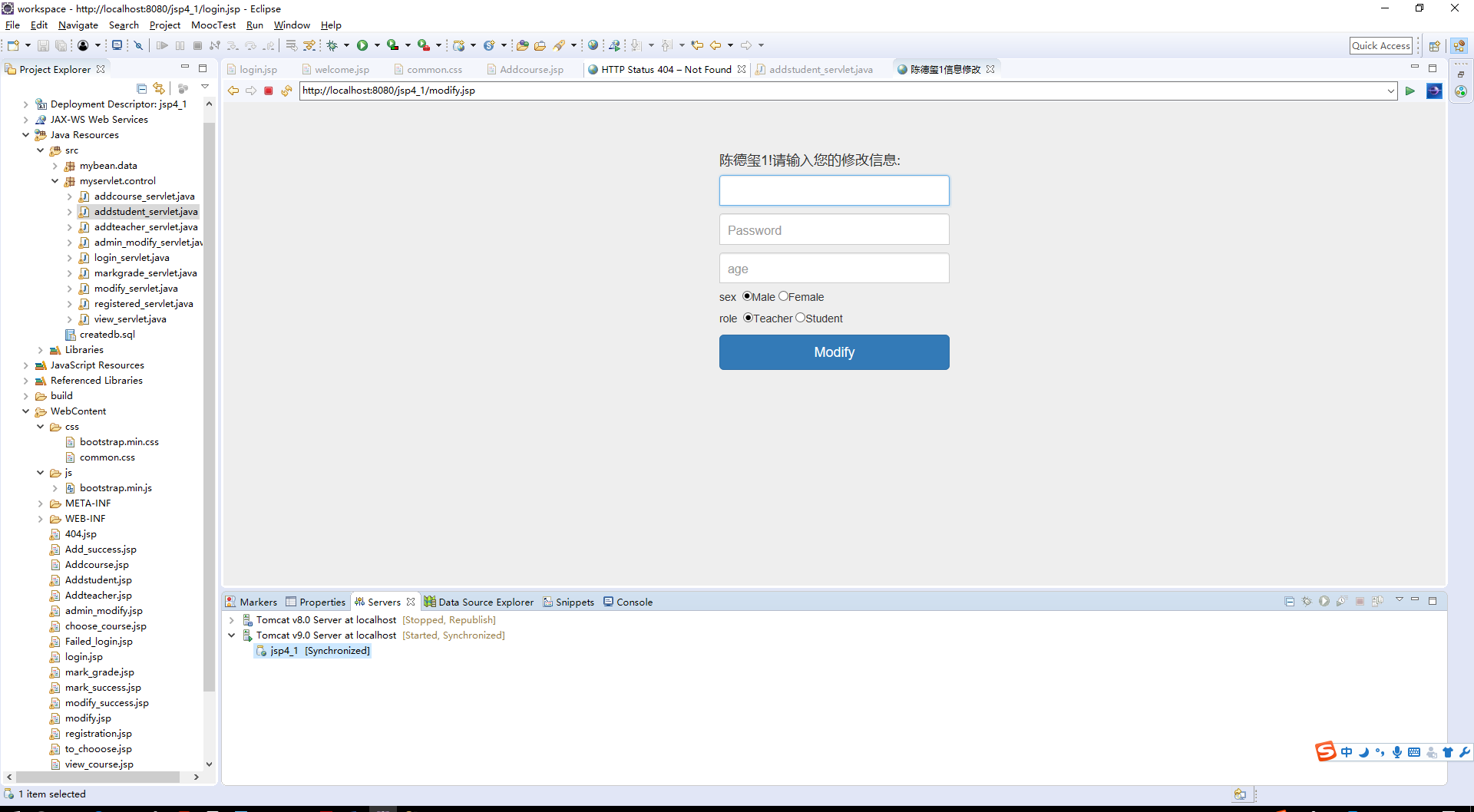
（此错误是由于数据库插入函数没有返回结果集，java9加大了对于异常的处理级别，对于这样的错误会打印到控制台）

6、管理员添加课程



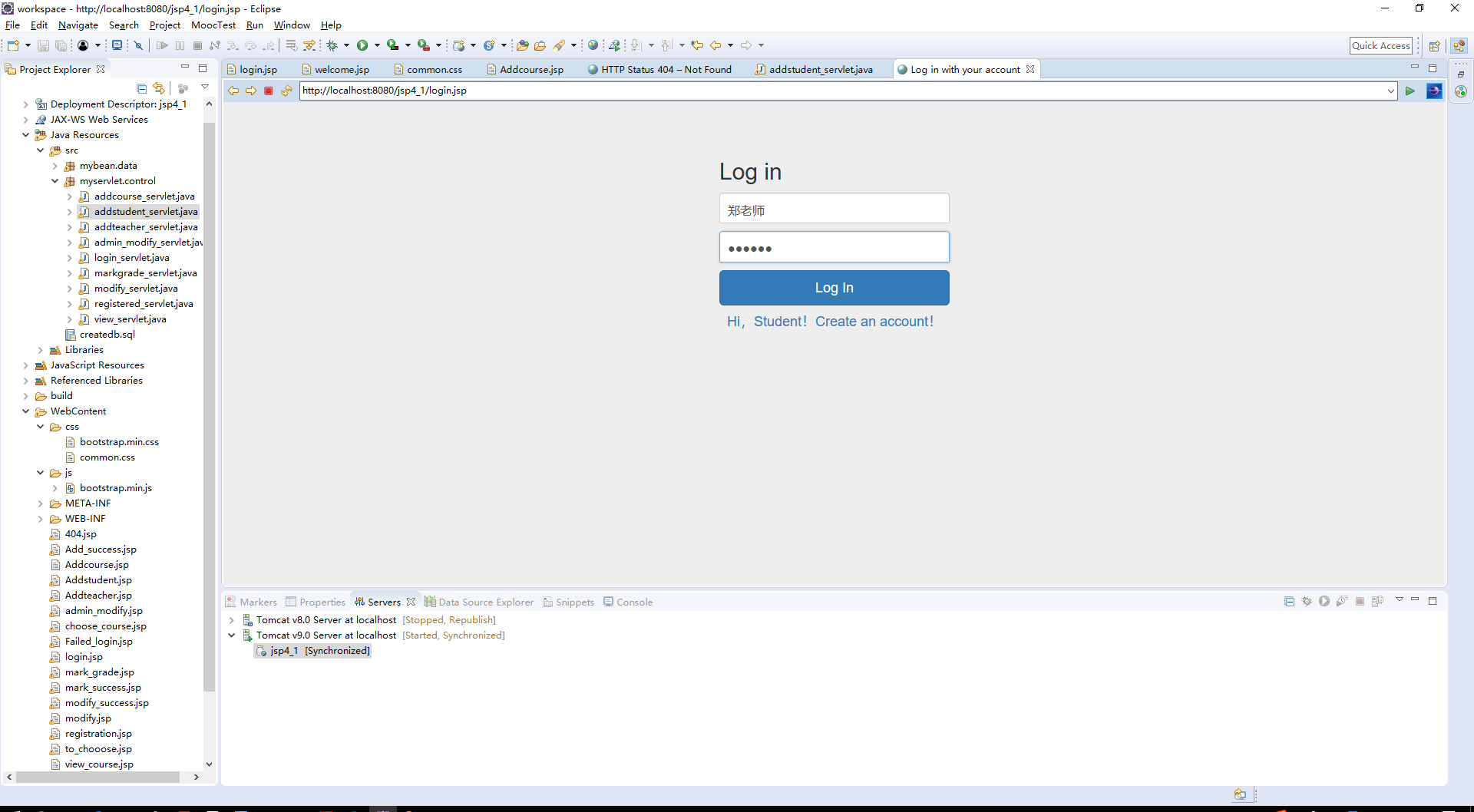
（此错误是由于数据库插入函数没有返回结果集，java9加大了对于异常的处理级别，对于这样的错误会打印到控制台）

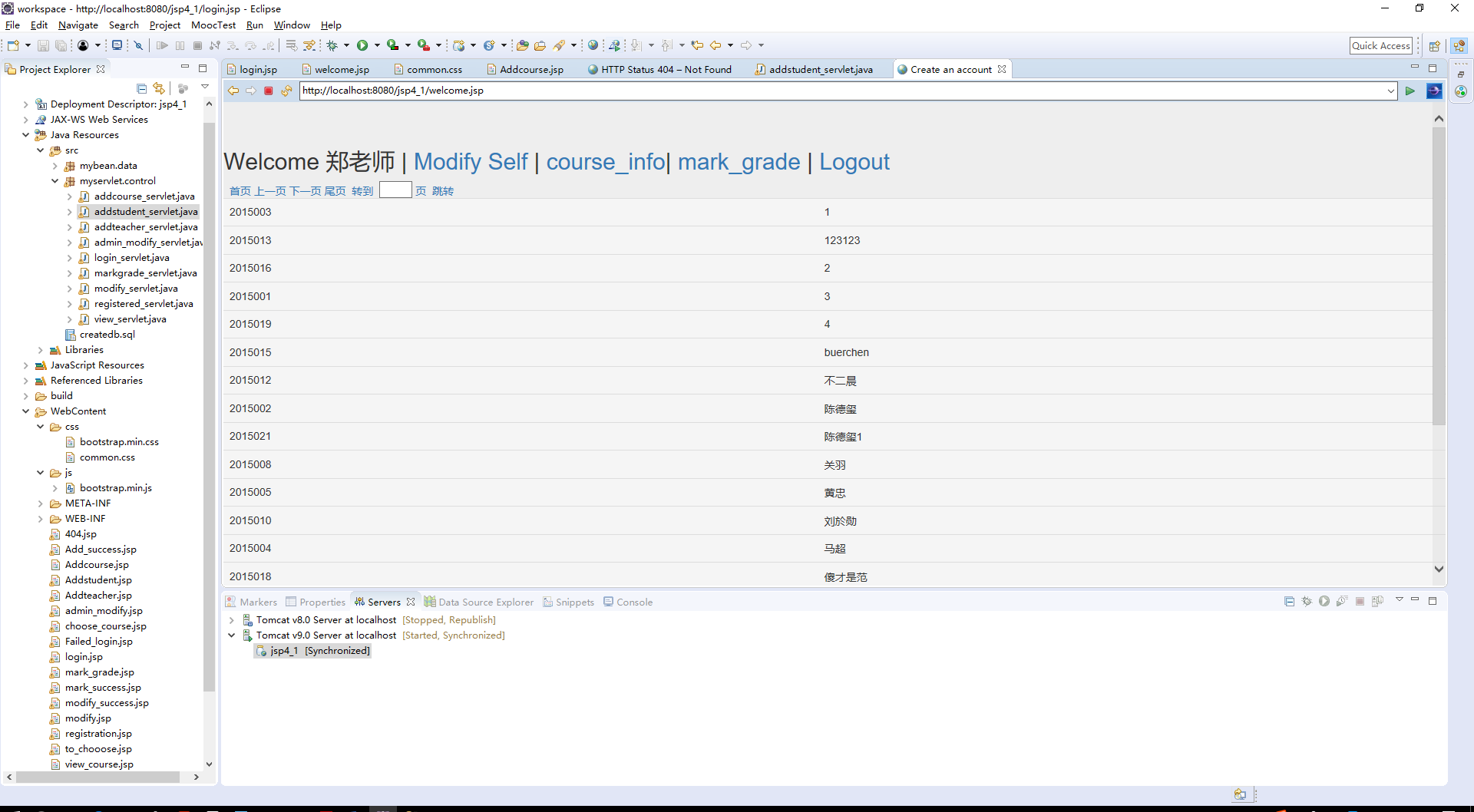
7、管理员修改个人信息



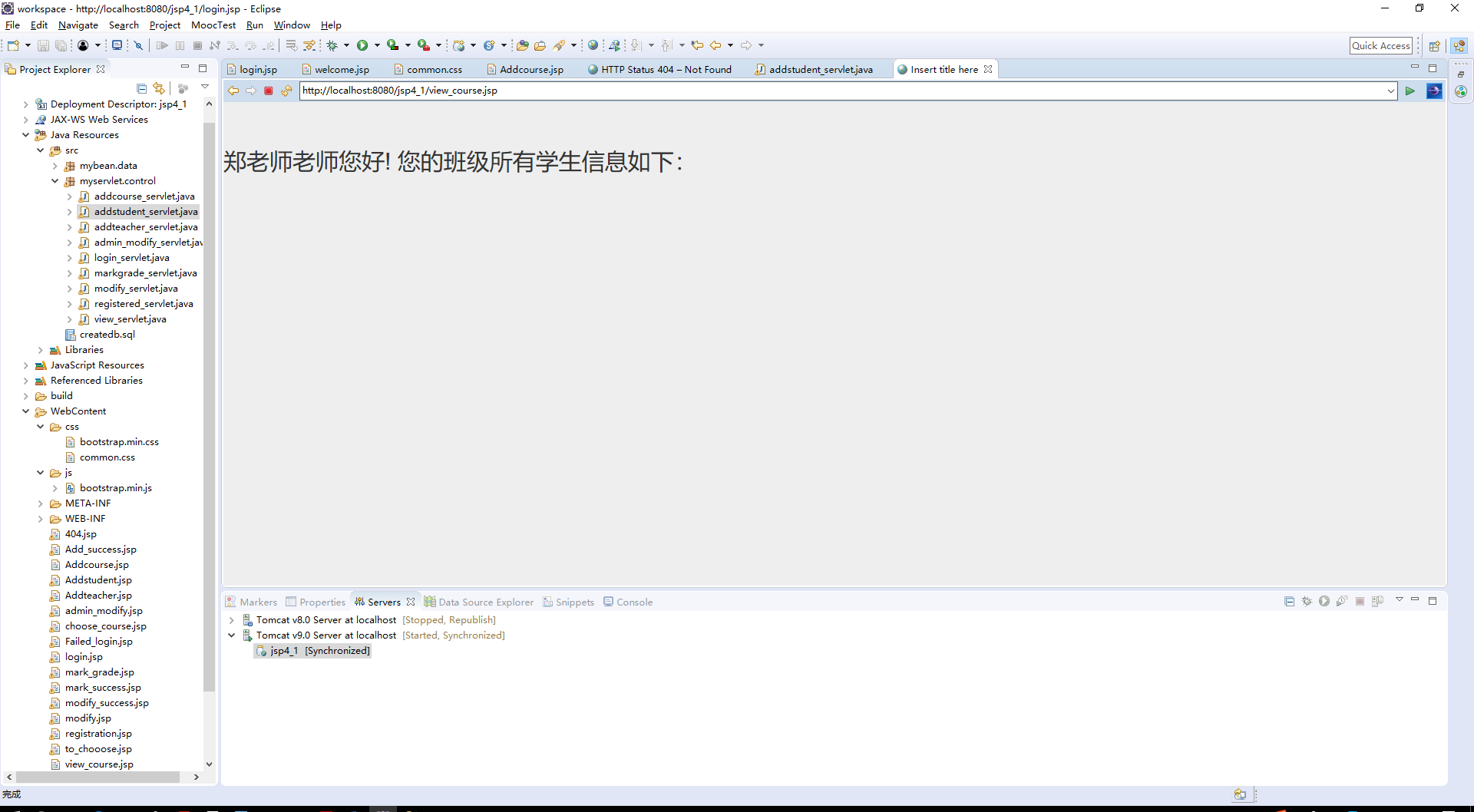
B、教师功能部分

1、教师登录

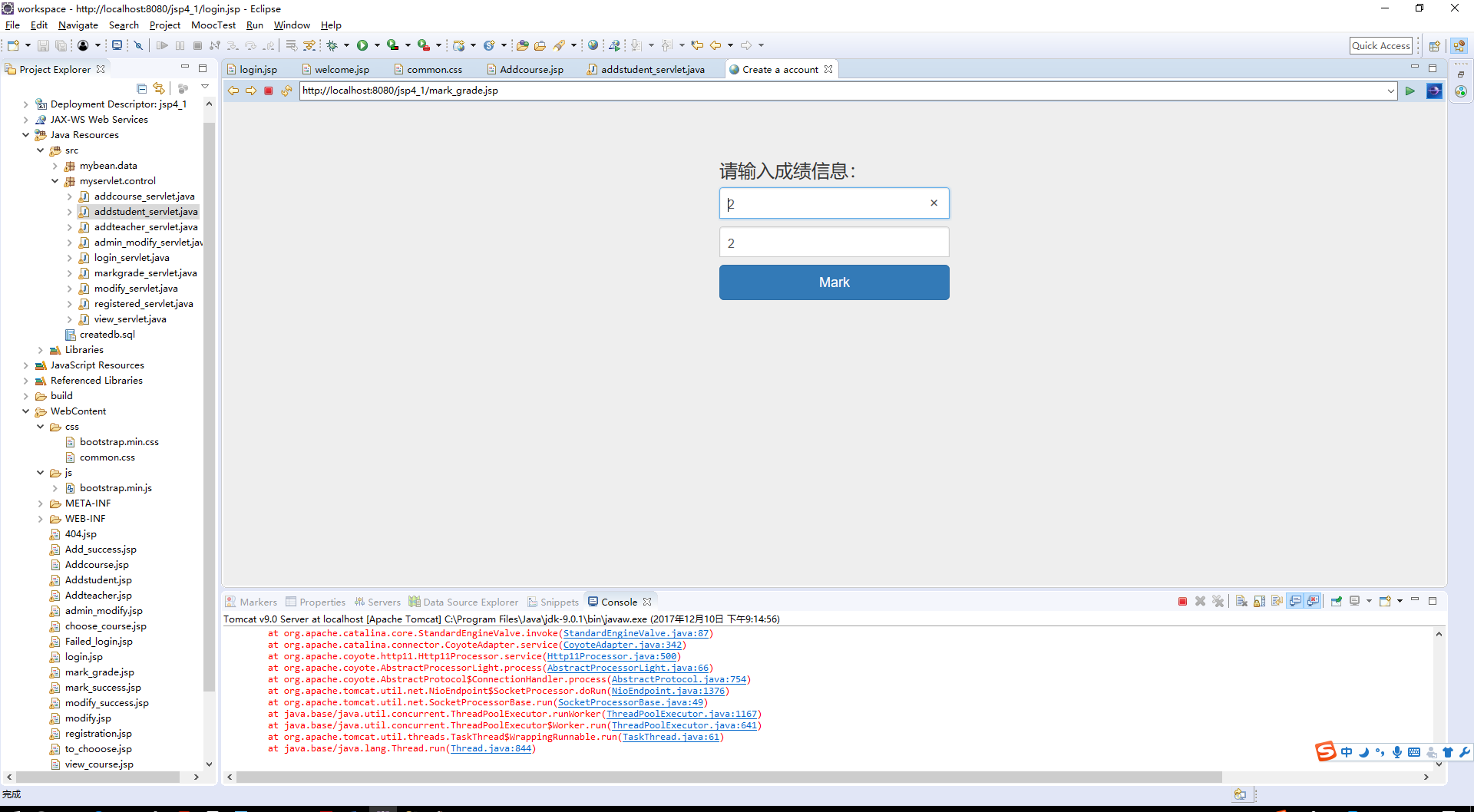




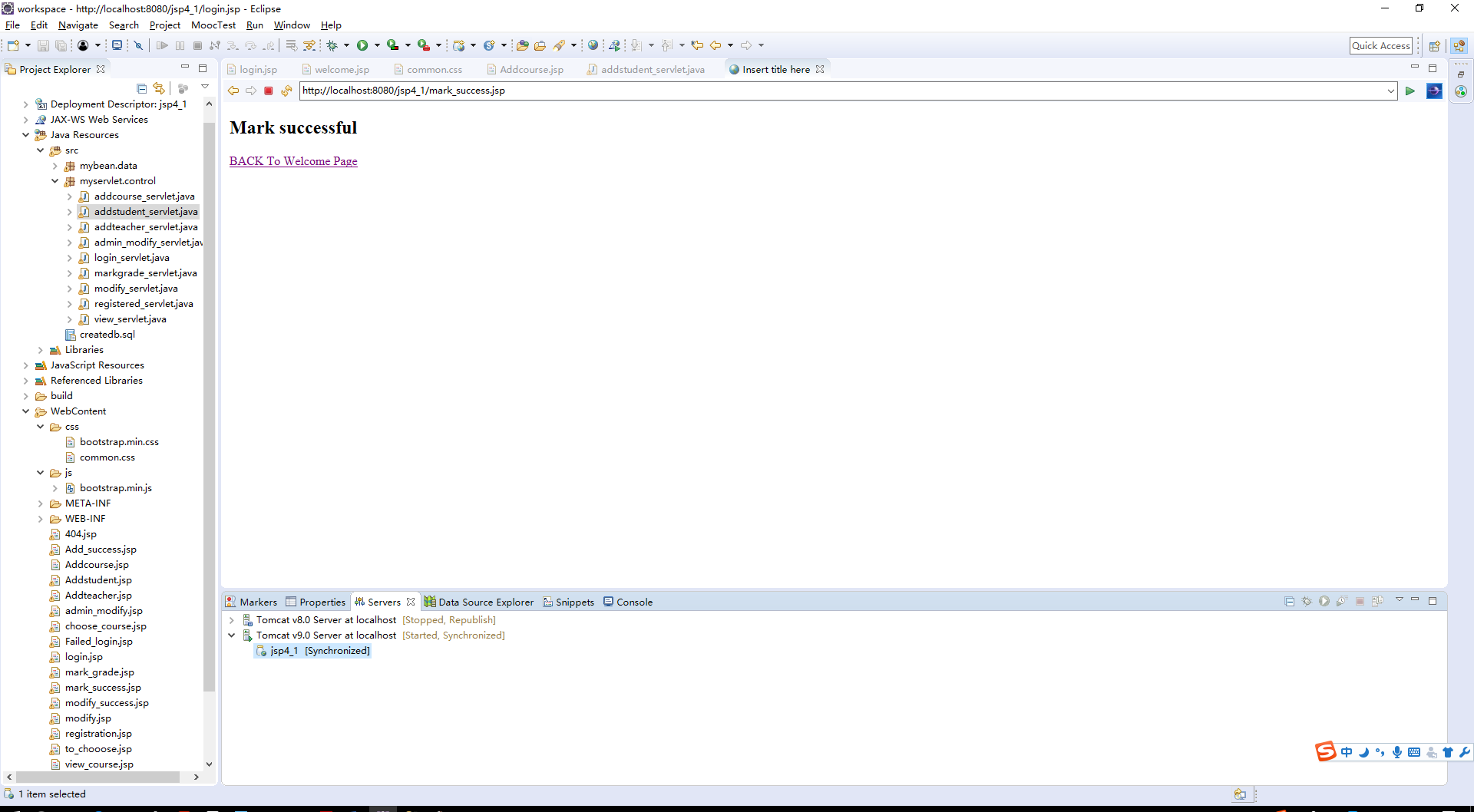
2、教师查看班级信息



3、教师录入成绩

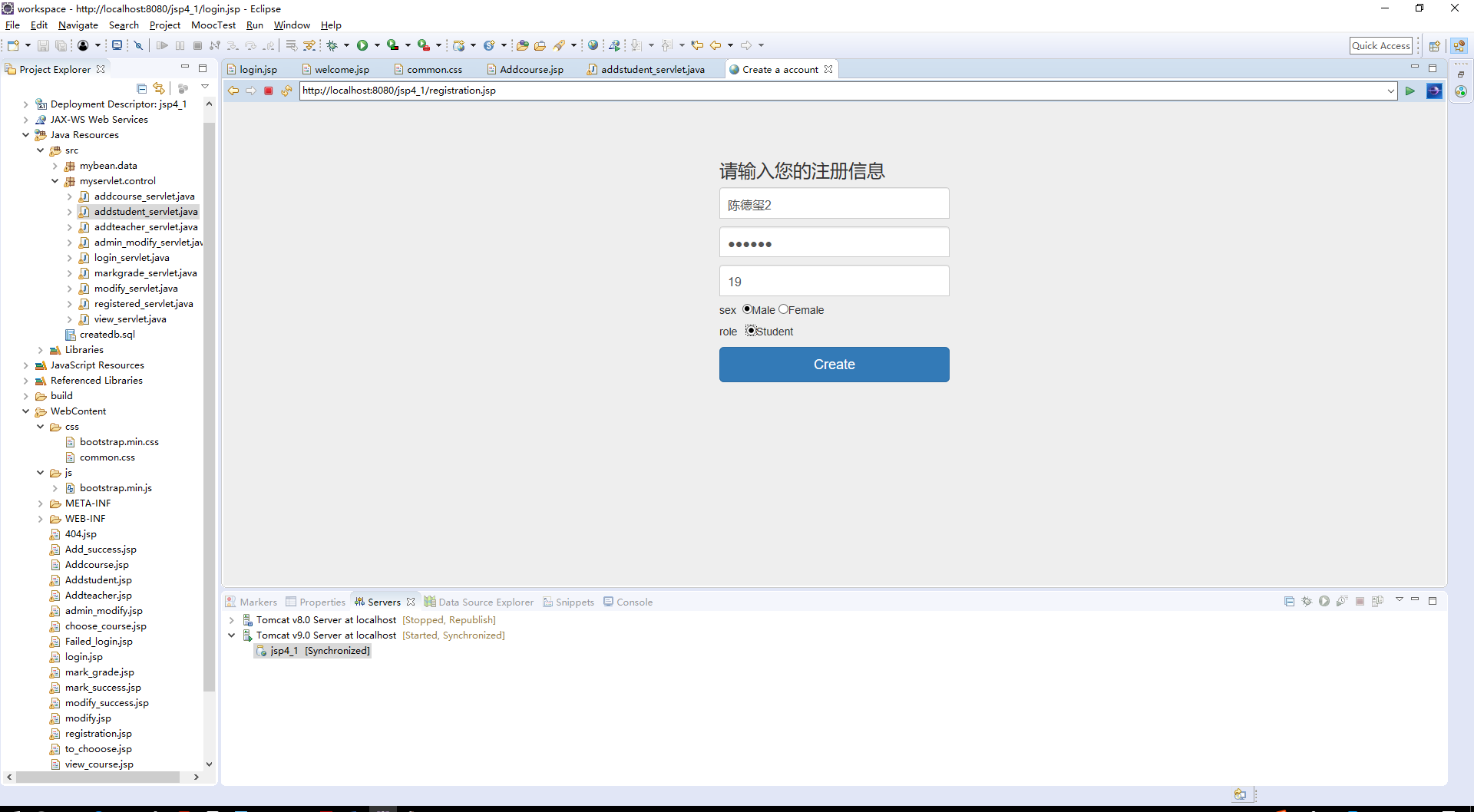


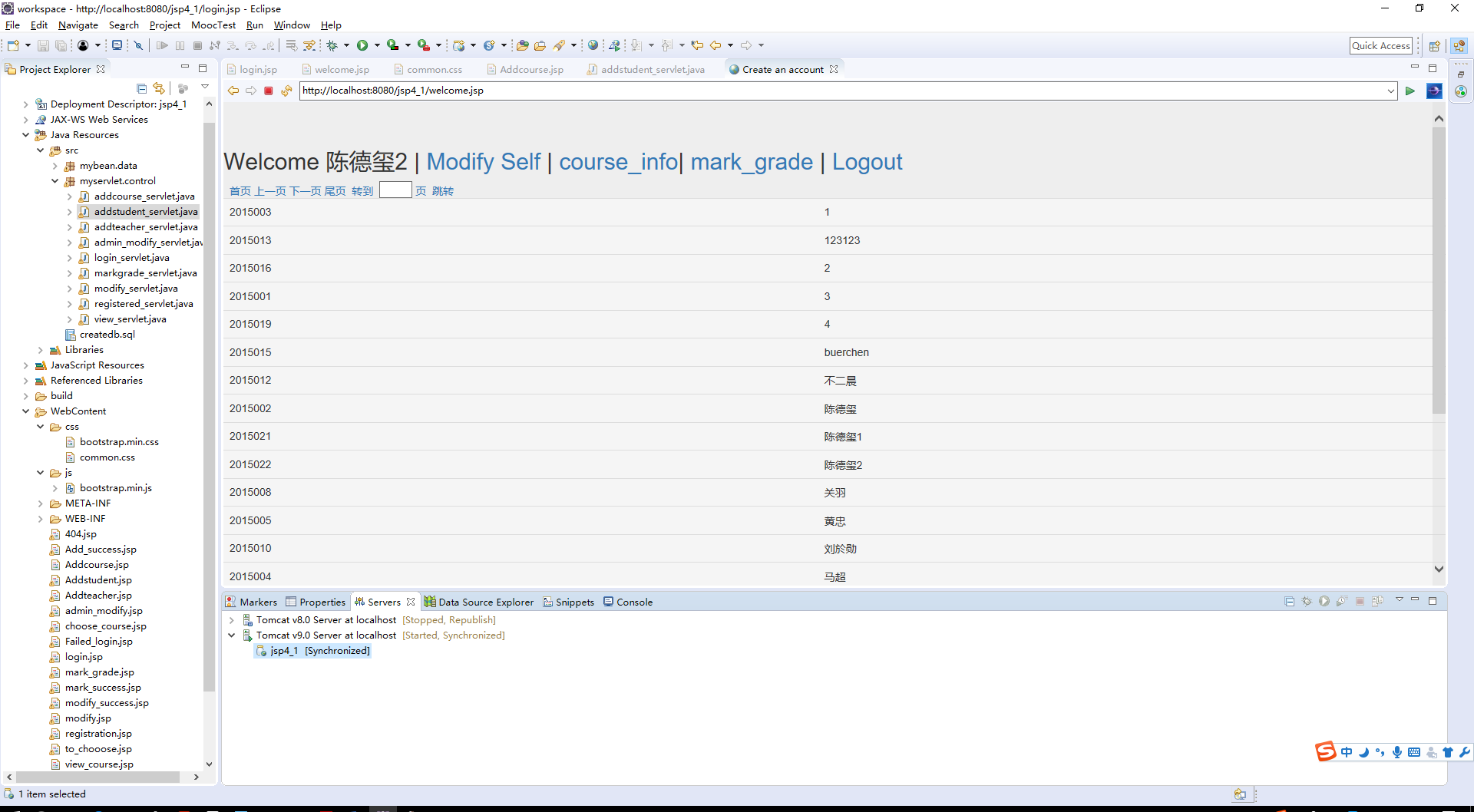
（此错误是由于数据库插入函数没有返回结果集，java9加大了对于异常的处理级别，对于这样的错误会打印到控制台）



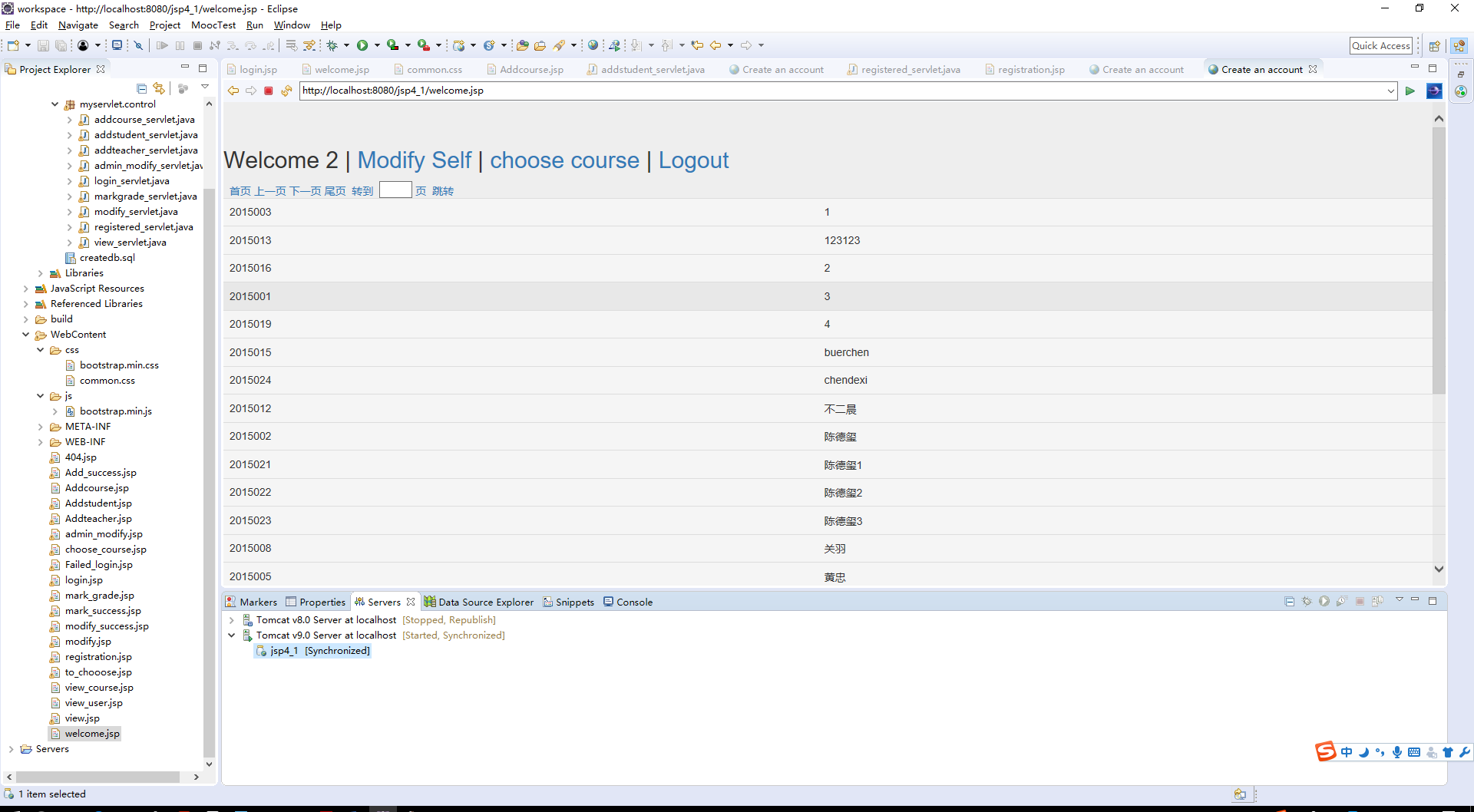
C、教师功能部分

1、学生创建账号

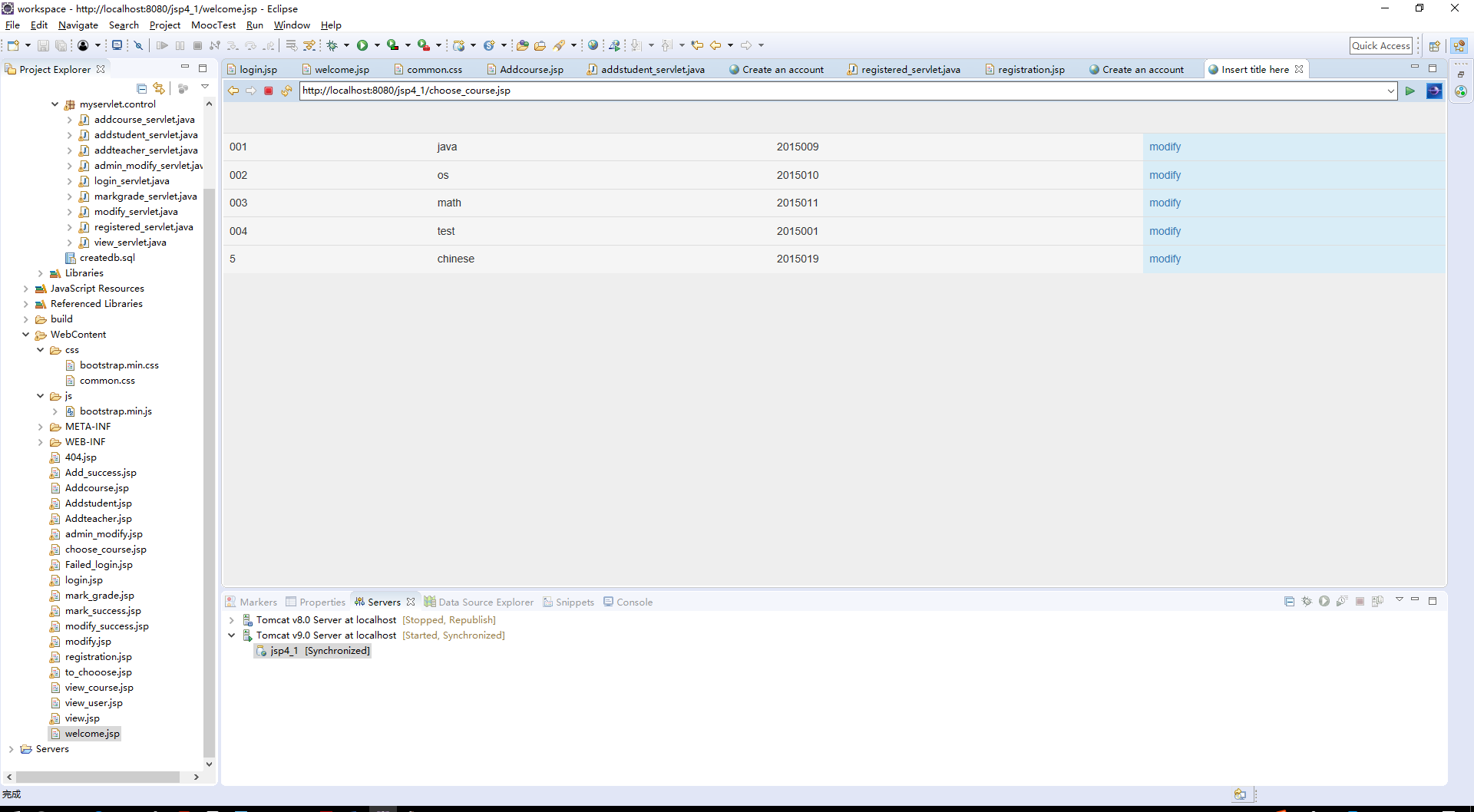


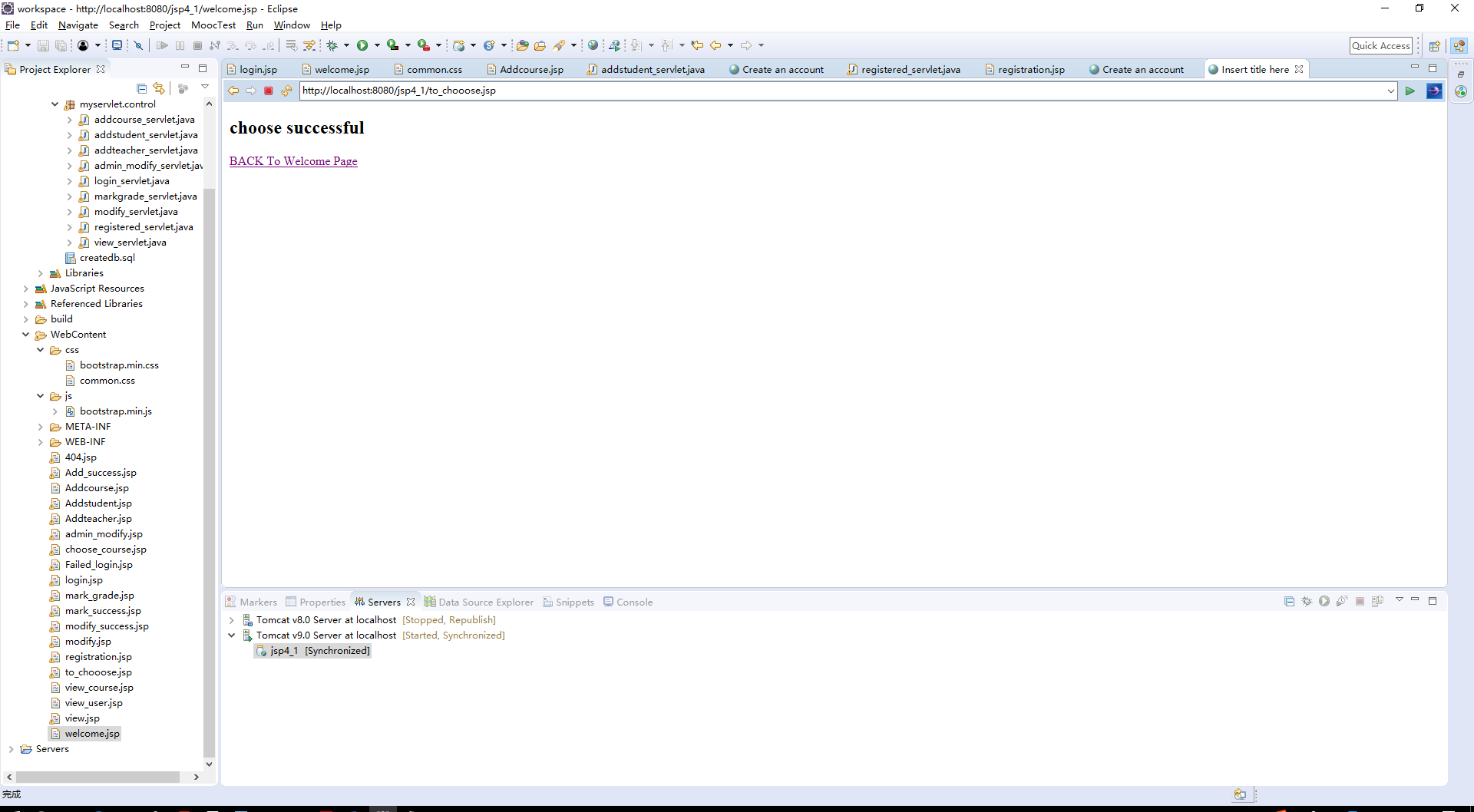


2、学生登陆成功



3、学生选修课程





五．心得体会

通过本次实验从中学到了很多，感触也很多，像这样比较小的网站程序就耗费了很大的力气，不敢想象以后参加工作的话会是一番情景。所以来说还是要好好的练就自己的本领，只有自己强大了，才能够在自己的人生中找到自信。