学生课程成绩管理系统

——中间件实现技术课程项目报告

学号：2018104167

姓名：曲芳

1. 项目内容

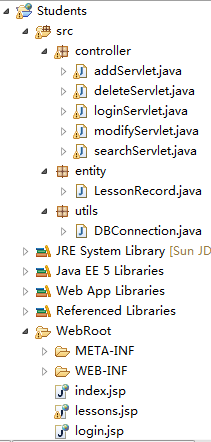
开发一个简易的学生课程成绩管理系统，能够查看学生对应的课程成绩，实现增删改查功能

2. 实验平台

* Java Web项目使用MyEclipse 10 + mysql 5
* 使用Servlet的doPost()方法实现数据库的增删改查过程
* 使用jsp展示管理系统的页面

3. 源代码分析

项目名为Student，简化了MVC的开发框架，在utils包中定义了数据库连接方法，在entity包中定义了课程记录的属性，controller中的Servlet执行对数据库的增删改查以及与jsp的数据传输。



3.1 DBConnection.java 数据库连接

String DBDRIVER = "com.mysql.jdbc.Driver";

String DBURL = "jdbc:mysql://localhost:3306/student?characterEncoding=gbk";

String DBUSER = "root";

String DBPWD = "root";

Connection conn = **null**;

**public** DBConnection() **throws** Exception{

**try**{

Class.*forName*(DBDRIVER);

**this**.conn = DriverManager.*getConnection*(DBURL, DBUSER, DBPWD);

System.*out*.println("yes");

}**catch**(Exception e){

**throw** e;

}

}

本项目使用低版本的mysql驱动 com.mysql.jdbc.Driver

使用的是mysql的student数据库中的student表，用户名和密码都是root，注意增加gbk的编码格式

3.2 LessonRecord.java 课程记录

**public** **class** LessonRecord {

**private** String id;

**private** String name;

**private** String less\_name;

**private** String grade;

**public** String getId(){

**return** id;

}

**public** String getName(){

**return** name;

}

**public** String getLess\_name(){

**return** less\_name;

}

**public** String getGrade(){

**return** grade;

}

**public** **void** setId(String id){

**this**.id = id;

}

**public** **void** setName(String name){

**this**.name = name;

}

**public** **void** setLess\_name(String less\_name){

**this**.less\_name = less\_name;

}

**public** **void** setGrade(String grade){

**this**.grade = grade;

}

}

定义了ID，学生姓名，课程名，课程成绩这4个属性，以及对应的set()和get()方法，对应数据库中的4列：



这里为了显示与传输方便，都定义为String类型

3.3 login.jsp 登录页面

<form name=*"form0"* action=*"/Students/loginServlet"* method=*"post"*>

账号：<input type=*"text"* name=*"user"* style="width: *106px*; "><br>

密码：<input type=*"password"* name=*"passwd"* style="width: *106px*; "/><br>

<input type=*"submit"* value=*"登录并查看课程成绩"* style="width: *140px*; color: *Black*"/>

</form>

使用表单，参数action定义了提交表单后的跳转地址，将user和passwd的值传递到loginServlet中

3.4 loginServlet.java 核对用户名和密码，查找并返回全部记录

String usr = request.getParameter("user");

String pw = request.getParameter("passwd");

**if**(usr.equals("123") && pw.equals("123")){

**try** {

Class.*forName*("com.mysql.jdbc.Driver");

Connection conn = DriverManager.*getConnection*("jdbc:mysql://localhost:3306/student?characterEncoding=gbk", "root", "root");

Statement stmt = conn.createStatement();

ResultSet rs = stmt.executeQuery("select \* from lesson");

List<LessonRecord> LessonList = **new** ArrayList<LessonRecord>();

**while**(rs.next()){

String id = rs.getString(1);

String name = rs.getString(2);

String less\_name = rs.getString(3);

String grade = rs.getString(4);

LessonRecord lr = **new** LessonRecord();

lr.setId(id);

lr.setName(name);

lr.setLess\_name(less\_name);

lr.setGrade(grade);

LessonList.add(lr);

System.*out*.println("ok");

}

conn.close();

request.setCharacterEncoding("utf-8");

response.setCharacterEncoding("utf-8");

request.setAttribute("LessonList", LessonList);

request.getRequestDispatcher("/lessons.jsp").forward(request,response);

} **catch** (Exception e) {

e.printStackTrace();

}

}**else**{

System.*out*.println("Wrong User !");

request.getRequestDispatcher("/login.jsp").forward(request,response);

}

接收login.jsp传来的user和passwd，当用户名和密码都为123（这里只是简单的字符串匹配）时，连接数据库，执行select \* from lesson查询操作，并将查询结果以list的形式返回给lessons.jsp；否则跳转回login.jsp页面，重新输入用户名和密码

3.5 lessons.jsp 课程成绩展示与操作

<table border=*"1"* charset=*"UTF-8"*>

<tr>

<td>ID</td>

<td>NAME</td>

<td>LESSON</td>

<td>GRADE</td>

</tr>

<%

List<LessonRecord> LR = (List<LessonRecord>)request.getAttribute("LessonList");

**if**(LR != **null**){

**for**(LessonRecord lr: LR){

%>

<tr>

<td><%=lr.getId() %></td>

<td><%=lr.getName() %></td>

<td><%=lr.getLess\_name() %></td>

<td><%=lr.getGrade() %></td>

</tr>

<%

}

}

%>

</table>

<br>

<br>

<form name=*"form0"* action=*"/Students/addServlet"* method=*"post"*>

<input type=*"submit"* value=*"ADD"* style="width: *100px*; color: *Black*"/>

ID:<input type=*"text"* name=*"id"* style="width: *106px*; "/>

NAME:<input type=*"text"* name=*"name"* style="width: *106px*; "/>

LESSON:<input type=*"text"* name=*"less\_name"* style="width: *106px*; "/>

Grade:<input type=*"text"* name=*"grade"* style="width: *106px*; "/><br>

</form>

<form name=*"form1"* action=*"/Students/modifyServlet"* method=*"post"*>

<input type=*"submit"* value=*"MODIFY"* style="width: *100px*; color: *Black*"/>

ID:<input type=*"text"* name=*"id"* style="width: *106px*; "/>

Grade:<input type=*"text"* name=*"grade"* style="width: *106px*; "/><br>

</form>

<form name=*"form2"* action=*"/Students/deleteServlet"* method=*"post"*>

<input type=*"submit"* value=*"DELETE"* style="width: *100px*; color: *Black*"/>

ID:<input type=*"text"* name=*"id"* style="width: *106px*; "/>

</form>

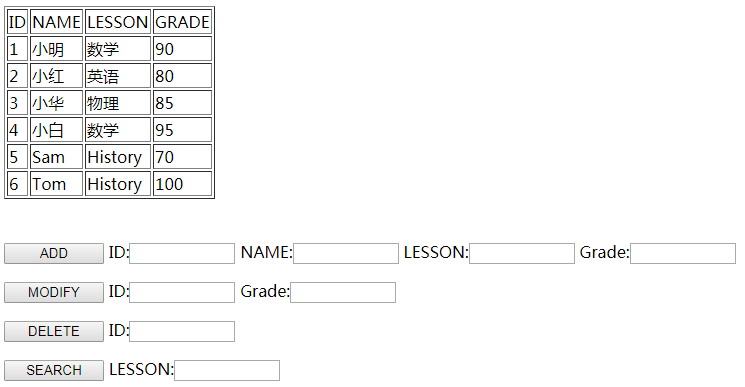
<form name=*"form3"* action=*"/Students/searchServlet"* method=*"post"*>

<input type=*"submit"* value=*"SEARCH"* style="width: *100px*; color: *Black*"/>

LESSON:<input type=*"text"* name=*"less\_name"* style="width: *106px*; "/>

</form>

以table的形式展示所有课程成绩记录和增删改查操作



在table中接收loginServlet传来的LessonList参数，在for循环中获取id/name/less\_name/grade并展示出来。在table下方，增加了ADD/MODIFY/DELETE/SEARCH增删改查4个表单，对应了addServlet/modifyServlet/deleteServlet/searchServlet操作。

3.6 addServlet/modifyServlet/deleteServlet/searchServlet.java 增删改查

* 接收参数，转换成sql

add: insert into lesson values(?,?,?,?)

modify: update lesson set grade=? where id=?

delete: delete from lesson where id=?

search: select \* from lesson where lesson=?

* 连接数据库
* 执行sql
* 返回结果给lessons.jsp

3.6.1 addServlet.java

**int** id = Integer.*parseInt*(request.getParameter("id"));

String name = request.getParameter("name");

String less\_name = request.getParameter("less\_name");

**float** grade = Float.*parseFloat*(request.getParameter("grade"));

String sql = "insert into lesson values("+id+",'"+name+"','"+less\_name+"',"+grade+")";

3.6.2 modifyServlet.java

**int** id = Integer.*parseInt*(request.getParameter("id"));

**float** grade = Float.*parseFloat*(request.getParameter("grade"));

String sql = "update lesson set grade = " + grade + " where id = " + id;

3.6.3 deleteServlet.java

**int** id = Integer.*parseInt*(request.getParameter("id"));

String sql = "delete from lesson where id = " + id;

3.6.4 searchServlet.java

String less\_name = request.getParameter("less\_name");

String sql = "select \* from lesson where less\_name = '" + less\_name + "'";

3.7 web.xml

<servlet-mapping>

<servlet-name>loginServlet</servlet-name>

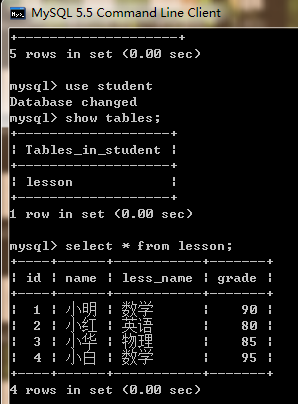
<url-pattern>/loginServlet</url-pattern>

</servlet-mapping>

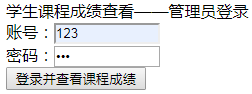
记录了各Servlet的路径

4. 实验结果

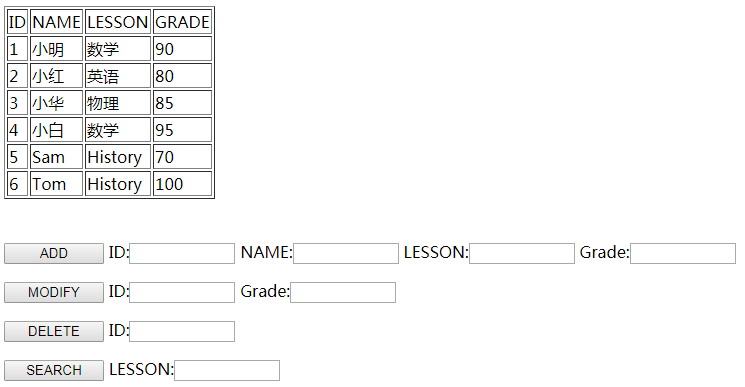
4.1 原有的数据库记录



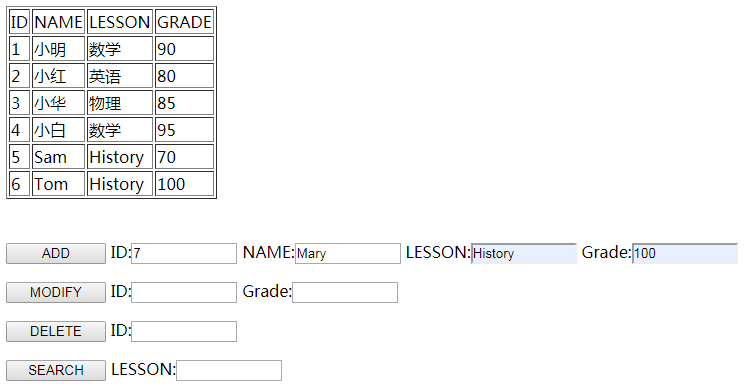
4.2 在浏览器中登录 http://qufang-pc:8080/Students/login.jsp 用户名和密码为123



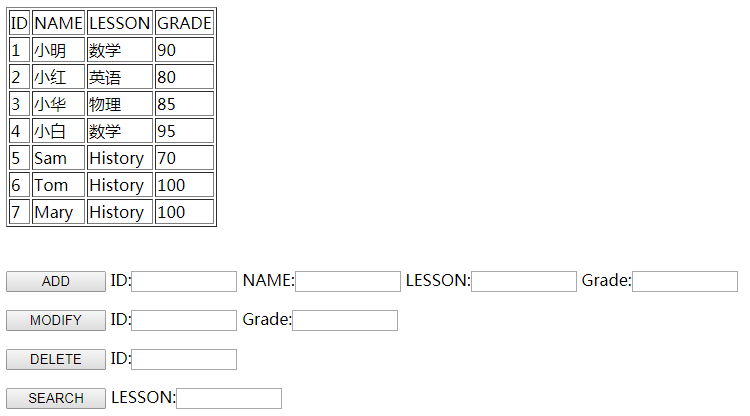
4.3 课程成绩记录与操作页面



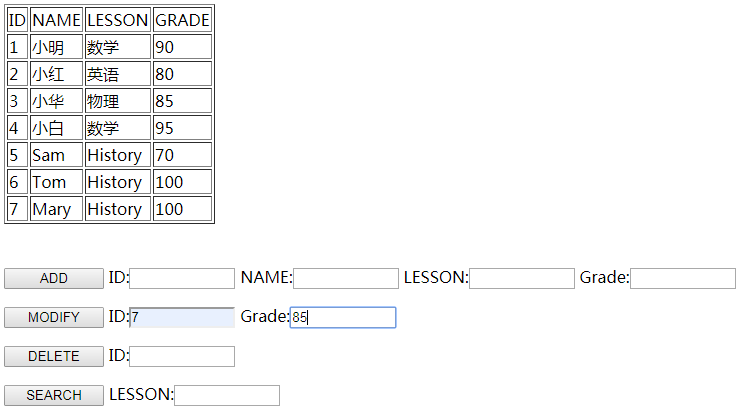
4.4 增加



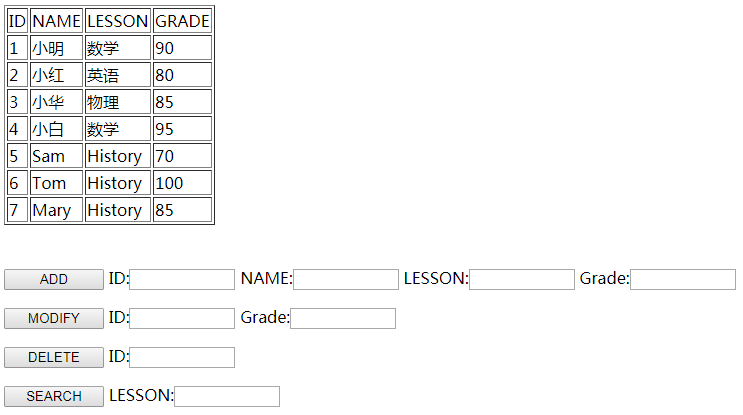
结果



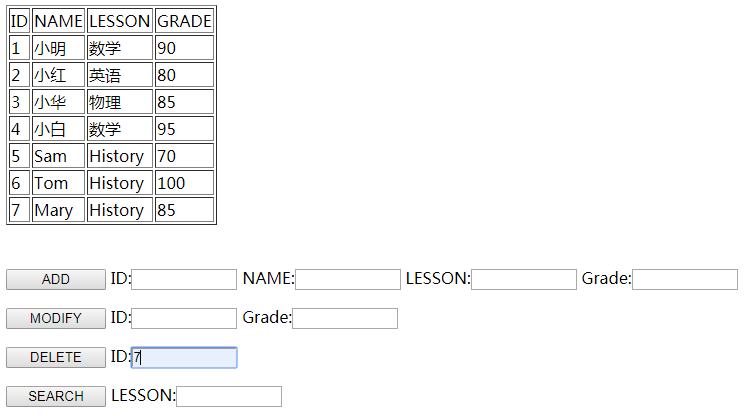
4.5 修改



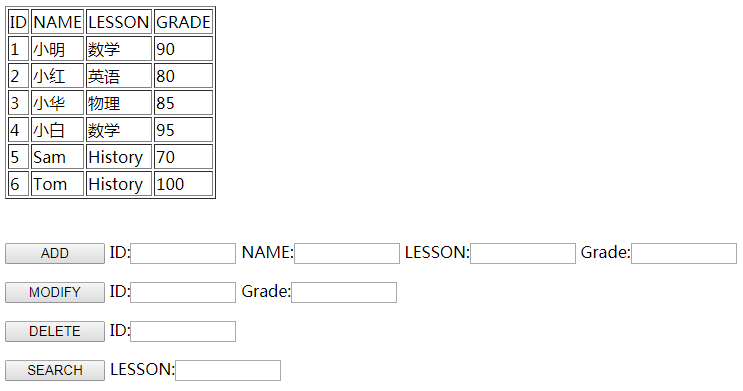
结果



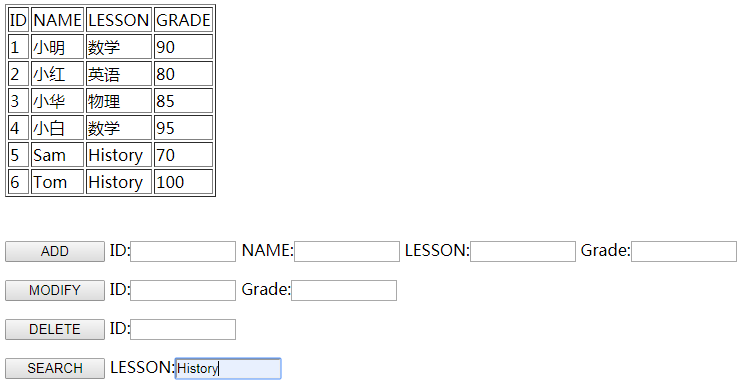
4.6 删除



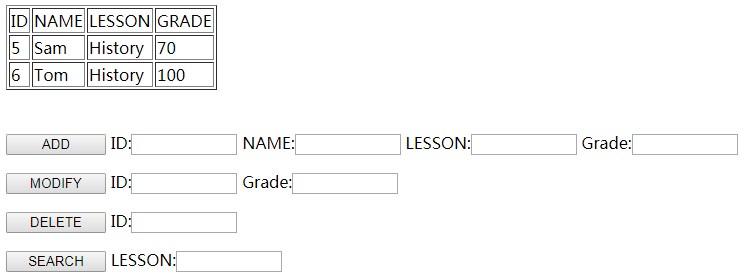
结果



4.7 查找



结果



Trick：在查找时，若LESSON中不输入任何值，即不指定LESSON，可以查找全部记录

