Git link

Developing a Backend Admin for Learner's Academy.

Dr. Jyothi N.M.

Source code

Struts.xml

```xml
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE struts SYSTEM "http://struts.apache.org/dtds/struts-2.1.dtd" PUBLIC "-//Apache Software Foundation//DTD Struts Configuration 2.1//EN">
<struts><constant value="true" name="struts.devMode"/><package name="default" extends="struts-default"><action name="how-to-use"><result>form.jsp</result></action><action name="form" method="loginCheck" class="front.User"><result name="success">form.jsp</result><result name="error">login.jsp</result></action><action name="main" method="login" class="front.User"><result name="success">form.jsp</result><result name="error">login-error.jsp</result></action><action name="register" method="register" class="front.User"><result name="success">register-success.jsp</result><result name="error">register-error.jsp</result></action><action name="view_timetable" method="fromForm" class="front.TimetableAction"><result name="success">view.jsp</result></action><action name="view-timetable" method="fromFile" class="front.TimetableAction"><result name="success">view.jsp</result></action><action name="logout" method="logout" class="front.User"><result>login.jsp</result></action></package></struts>
```

Scheduler.java

package scheduler;

import java.io.File;

import java.util.*;

```java
public class inputdata {

        public static StudentGroup[] studentgroup;

        public static Teacher[] teacher;

        public static double crossoverrate = 1.0, mutationrate = 0.1;

        public static int nostudentgroup, noteacher;

        public static int hoursperday, daysperweek;


        public inputdata() {

                studentgroup = new StudentGroup[100];

                teacher =   new Teacher[100];

        }


        boolean classformat(String l) {

                StringTokenizer st = new StringTokenizer(l, " ");

                if (st.countTokens() == 3)

                        return (true);

                else

                        return (false);

        }


        public void takeinput()// takes input from file input.txt

        {

                //this method of taking input through file is only for development purpose so hours
and days are hard coded

                hoursperday = 7;

                daysperweek = 5;

                try {

                        File file = new File("c:\\test\\input.txt");

                        // File file = new File(System.getProperty("user.dir") +
```

```java
                    // "/input.txt");

    Scanner scanner = new Scanner(file);

    while (scanner.hasNextLine()) {
        String line = scanner.nextLine();

        // input student groups
        if (line.equalsIgnoreCase("studentgroups")) {
            int i = 0, j;
            while (!(line =
scanner.nextLine()).equalsIgnoreCase("teachers")) {
                studentgroup[i] = new StudentGroup();
                StringTokenizer st = new StringTokenizer(line, " ");
                studentgroup[i].id = i;
                studentgroup[i].name = st.nextToken();
                studentgroup[i].nosubject = 0;
                j = 0;
                while (st.hasMoreTokens()) {
                    studentgroup[i].subject[j] = st.nextToken();
                    studentgroup[i].hours[j++] =
Integer.parseInt(st.nextToken());
                    studentgroup[i].nosubject++;
                }
                i++;
            }
            nostudentgroup = i;
        }

        //input teachers
        if (line.equalsIgnoreCase("teachers")) {
            int i = 0;
```

```java
                                        while (!(line = scanner.nextLine()).equalsIgnoreCase("end"))
{

                                                teacher[i] = new Teacher();

                                                StringTokenizer st = new StringTokenizer(line, " ");

                                                teacher[i].id = i;

                                                teacher[i].name = st.nextToken();

                                                teacher[i].subject = st.nextToken();


                                                i++;
                                        }
                                        noteacher = i;
                                }


                        }
                        scanner.close();
                } catch (Exception e) {
                        e.printStackTrace();
                }


                assignTeacher();


        }


        // assigning a teacher for each subject for every studentgroup
        public void assignTeacher() {


                // looping through every studentgroup
                for (int i = 0; i < nostudentgroup; i++) {


                        // looping through every subject of a student group
                        for (int j = 0; j < studentgroup[i].nosubject; j++) {
```

```java
                    int teacherid = -1;

                    int assignedmin = -1;


                    String subject = studentgroup[i].subject[j];


                    // looping through every teacher to find which teacher teaches the
current subject

                    for (int k = 0; k < noteacher; k++) {


                            // if such teacher found,checking if he should be assigned
the subject or some other teacher based on prior assignments
                            if (teacher[k].subject.equalsIgnoreCase(subject)) {


                                    // if first teacher found for this subject
                                    if (assignedmin == -1) {
                                            assignedmin = teacher[k].assigned;

                                            teacherid = k;

                                    }


                                    // if teacher found has less no of pre assignments
than the teacher assigned for this subject
                                    else if (assignedmin > teacher[k].assigned) {
                                            assignedmin = teacher[k].assigned;

                                            teacherid = k;

                                    }
                            }
                    }


                    // 'assigned' variable for selected teacher incremented
                    teacher[teacherid].assigned++;
```

```
                                studentgroup[i].teacherid[j]= teacherid;
                        }
                }
        }
}



Subject.java

package scheduler;


public class Subject {
        int id;
        String name;
        Teacher[] teacher;
        int noteachers;
        Subject(){
                teacher=new Teacher[20];
        }
}



package scheduler;


public class Teacher {
        int id;
        String name;
        String subject;
        int assigned;
        public int getId() {
                return id;
        }
}
```

```java
        public void setId(int id) {

                this.id = id;

        }

        public String getName() {

                return name;

        }

        public void setName(String name) {

                this.name = name;

        }

        public String getSubject() {

                return subject;

        }

        public void setSubject(String subject) {

                this.subject = subject;

        }


}


package scheduler;


//through this class slots are generated to match minimum hours criteria for each subject

//also this class holds slots in order

public class TimeTable {


        public static Slot[] slot;


        TimeTable() {


                int k = 0;

                int subjectno = 0;

                int hourcount = 1;
```

```java
int days = inputdata.daysperweek;

int hours = inputdata.hoursperday;

int nostgrp = inputdata.nostudentgroup;


// creating as many slots as the no of blocks in overall timetable

slot = new Slot[hours * days * nostgrp];


// looping for every student group

for (int i = 0; i < nostgrp; i++) {


        subjectno = 0;
        // for every slot in a week for a student group
        for (int j = 0; j < hours * days; j++) {


                StudentGroup sg = inputdata.studentgroup[i];


                // if all subjects have been assigned required hours we give
                // free periods
                if (subjectno >= sg.nosubject) {
                        slot[k++] = null;
                }


                // if not we create new slots
                else {


                        slot[k++] = new Slot(sg, sg.teacherid[subjectno],
sg.subject[subjectno]);


                        // suppose java has to be taught for 5 hours then we make 5
                        // slots for java, we keep track through hourcount
                        if (hourcount < sg.hours[subjectno]) {
```

```java
                                        hourcount++;
                        } else {
                                hourcount = 1;
                                subjectno++;
                        }

                }

                } // end inner for

        } // end outer for

    }// end constructor

    public static Slot[] returnSlots() {
            return slot;
    }
}
```

Web content

Login .jsp

```jsp
<%
response.setHeader("Cache-Control","no-store");
response.setHeader("Pragma","no-cache");
response.setHeader ("Expires", "0"); //prevents caching at the proxy server
%>

<!DOCTYPE html>
<html lang="en">
```

```html
<head>

<meta charset="utf-8">
<meta http-equiv="X-UA-Compatible" content="IE=edge">
<meta name="viewport" content="width=device-width, initial-scale=1">
<meta name="description" content="">
<meta name="author" content="">


<title>Login | Time Table Generator</title>


<!-- Bootstrap Core CSS -->
<link href="vendor/bootstrap/css/bootstrap.min.css" rel="stylesheet">


<!-- Custom Fonts -->
<link href="vendor/font-awesome/css/font-awesome.min.css"
      rel="stylesheet" type="text/css">
<link href="https://fonts.googleapis.com/css?family=Montserrat:400,700"
      rel="stylesheet" type="text/css">
<link href='https://fonts.googleapis.com/css?family=Kaushan+Script'
      rel='stylesheet' type='text/css'>
<link
      href='https://fonts.googleapis.com/css?family=Droid+Serif:400,700,400italic,700italic'
      rel='stylesheet' type='text/css'>
<link
      href='https://fonts.googleapis.com/css?family=Roboto+Slab:400,100,300,700'
      rel='stylesheet' type='text/css'>


<!-- Theme CSS -->
<link href="css/agency.min.css" rel="stylesheet">
```

```html
<!-- HTML5 Shim and Respond.js IE8 support of HTML5 elements and media queries -->
<!-- WARNING: Respond.js doesn't work if you view the page via file:// -->
<!--[if lt IE 9]>
    <script src="https://oss.maxcdn.com/libs/html5shiv/3.7.0/html5shiv.js"></script>
    <script src="https://oss.maxcdn.com/libs/respond.js/1.4.2/respond.min.js"></script>
  <![endif]-->


</head>


<body style="background-image: url('img/lib.jpg'); background-attachment: fixed;">
        <!-- Navigation -->
        <nav id="mainNav"
                class="navbar navbar-default navbar-custom navbar-fixed-top"
                style="background-color: black">
                <div class="container">
                        <!-- Brand and toggle get grouped for better mobile display -->
                        <div class="navbar-header page-scroll">
                                <a class="navbar-brand page-scroll" href="#page-top">Time Table
                                        Generator</a>
                        </div>
                </div>
        </nav>



        <div class=fluid-container>
                <br />
                <br />
                <br />
        </div>
        <div class=row>
```

```html
<div class="col-sm-1"></div>


<div class="container col-sm-4 ">
        <div class=jumbotron>
                <h2 class=page-header>Login</h2>
                <form role="form" action="main" method=post>
                        <div class="form-group">
                                <label for="un">Username:</label> <input
type="text"
                                        class="form-control" id="un"
name=username>
                        </div>
                        <div class="form-group">
                                <label for="pwd">Password:</label> <input
type="password"
                                        class="form-control" id="pwd"
name=password>
                        </div>
                        <div class="checkbox">
                                <label><input type="checkbox"> Remember
me</label>
                        </div>
                        <button type="submit" class="btn btn-
primary">Login</button>
                </form>
        </div>
</div>


<div class="col-sm-1"></div>



<div class="container col-sm-5">
        <div class=jumbotron>
```

```html
<h2 class=page-header>Signup</h2>
<form role="form" action=register method=post>
    <div class="form-group">
        <label for="un">Username:</label> <input type="text"
            class="form-control" id="un" name=username>
    </div>

    <div class="form-group">
        <label for="email">Email address:</label> <input type="email"
            class="form-control" id="email" name=email>
    </div>

    <div class="form-group">
        <label for="country">Country:</label> <input type="text"
            class="form-control" id="country" name=country>
    </div>

    <div class="form-group">
        <label for="pwd">Password:</label> <input type="password"
            class="form-control" id="pwd" name=password>
    </div>

    <div class="form-group">
        <label for="cpwd">Confirm Password:</label> <input type="password"
```

```
                                        class="form-control" id="pwd"
name=confirm>

                                </div>


                                <button type="submit" class="btn btn-
primary">Signup</button>

                        </form>

                </div>

        </div>



        <div class="col-sm-1"></div>

</div>


<!-- jQuery -->
<script src="vendor/jquery/jquery.min.js"></script>


<!-- Bootstrap Core JavaScript -->
<script src="vendor/bootstrap/js/bootstrap.min.js"></script>


<!-- Plugin JavaScript -->
<script
        src="http://cdnjs.cloudflare.com/ajax/libs/jquery-
easing/1.3/jquery.easing.min.js"></script>


<!-- Contact Form JavaScript -->
<script src="js/jqBootstrapValidation.js"></script>
<script src="js/contact_me.js"></script>


<!-- Theme JavaScript -->
<script src="js/agency.min.js"></script>
```

</body>

</html>

Showdays.jsp

```jsp
<script type="text/javascript" src="vendor/bootstrap/js/custom.js"></script>

<%
        String si=(String)request.getParameter("val");
        int i=7;
        if(si!=""){
                i=Integer.parseInt(si);
        }
%>
                                <div class="checkbox">
                                 <label><input type="checkbox" value="Monday" name="day0" onClick=
<%="'return KeepCount("+i+")'" %>>Monday</label>
                                </div>
                                <div class="checkbox">
                                 <label><input type="checkbox" value="Tuesday" name="day1"
onClick=<%="'return KeepCount("+i+")'" %>>Tuesday</label>
                                </div>


                                <div class="checkbox">
                                 <label><input type="checkbox" value="Wednesday" name="day2"
onClick=<%="'return KeepCount("+i+")'" %>>Wednesday</label>
                                </div>


                                <div class="checkbox">
```

```jsp
            <label><input type="checkbox" value="Thursday" name="day3"
onClick=<%="'return KeepCount("+i+")'" %>>Thursday</label>

        </div>


        <div class="checkbox">

            <label><input type="checkbox" value="Friday" name="day4"
onClick=<%="'return KeepCount("+i+")'" %>>Friday</label>

        </div>

        <div class="checkbox">

            <label><input type="checkbox" value="Saturday" name="day5"
onClick=<%="'return KeepCount("+i+")'" %>>Saturday</label>

        </div>

        <div class="checkbox">

            <label><input type="checkbox" value="Sunday" name="day6"
onClick=<%="'return KeepCount("+i+")'" %>>Sunday</label>

        </div><br>


<script type="text/javascript" src="vendor/bootstrap/js/custom.js"></script>


<%

    String si=(String)request.getParameter("val");

    int i=7;

    if(si!=""){

            i=Integer.parseInt(si);

    }
%>
        <div class="checkbox">

            <label><input type="checkbox" value="Monday" name="day0" onClick=
<%="'return KeepCount("+i+")'" %>>Monday</label>

        </div>

        <div class="checkbox">

            <label><input type="checkbox" value="Tuesday" name="day1"
onClick=<%="'return KeepCount("+i+")'" %>>Tuesday</label>
```

```
                    </div>


          <div class="checkbox">

            <label><input type="checkbox" value="Wednesday" name="day2"
onClick=<%="'return KeepCount("+i+")'" %>>Wednesday</label>

          </div>


          <div class="checkbox">

            <label><input type="checkbox" value="Thursday" name="day3"
onClick=<%="'return KeepCount("+i+")'" %>>Thursday</label>

          </div>


          <div class="checkbox">

            <label><input type="checkbox" value="Friday" name="day4"
onClick=<%="'return KeepCount("+i+")'" %>>Friday</label>

          </div>

          <div class="checkbox">

            <label><input type="checkbox" value="Saturday" name="day5"
onClick=<%="'return KeepCount("+i+")'" %>>Saturday</label>

          </div>

          <div class="checkbox">

            <label><input type="checkbox" value="Sunday" name="day6"
onClick=<%="'return KeepCount("+i+")'" %>>Sunday</label>

          </div><br>
```

Similary ther required jsp and html xml file were created