**Java Web开发**

**实验四**

**作业报告**

|  |  |
| --- | --- |
| 学 院 | 计算机 |
| 所在系 班 级 | 2017软件工程系08班 |
| 学 号 姓 名 | 04171603鲍稚明 |
| 指 导 教 师 | 郭晓燕 |
| 完 成 时 间 | 2018年3月30日 |

# 实验题目

将实验三的三题进行改写，在Servlet控制器中获取信息，验证后根据具体情况跳转到对应页面，成功请求转发到成功页面，否则重定向跳转到失败页面（可将动态验证码添加在页面上）

\* 登录页面，提交用户名、密码及身份

改写一：支持中文，确保两种表单提交方式（get、post）都支持中文，另外对程序的空对象情况进行处理。

改写二：比如登录页面除了之前的用户名、密码另外再加上所在专业（计算机、软件工程及网络工程，单选框或下拉列表框皆可），登录成功后根据身份进行跳转，进入对应的页面，页面需显示\*\*\*\*专业学号姓名。登录失败转向失败页面，并进行报错！

\* 猜数功能，猜错猜对及输入不正确的情况分别处理

\* 注册功能、发帖功能等皆可

# 分析与设计

添加动态验证码需要在外部类中生成验证码图片，其要素在于随机字符(数字，字母)、随机大小、随机字体和背景颜色，我采用白色。生成后在servlet的dopost中把生成验证码的值传递给登录页面，并在登录页面作比较给出用户提示。

# 关键源码（部分）

----------------------------------login.jsp

<p>请选择您的专业：<select name=*'zy'*><option value =*"rg"*>软件工程</option><option value =*"jw"*>计算机网络</option>

<option value =*"xk"*>电子信息科学技术</option><option value =*"zd"*>自动化</option></select>

<p>验证码:</font><input type=*"text"* name=*"vcod"*>

<p><img src=*"/Logintest/servlet/Vcode"*>

<p><input type=*"submit"* value=*"提交"* name=*"submit"*>

---------vscode.servlet

import java.awt.BasicStroke;

import java.awt.Color;

import java.awt.Font;

import java.awt.Graphics;

import java.awt.Graphics2D;

import java.awt.image.BufferedImage;

import java.io.IOException;

import java.io.OutputStream;

import java.io.PrintWriter;

import java.util.Random;

import javax.imageio.ImageIO;

import javax.servlet.ServletException;

import javax.servlet.http.HttpServlet;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

public class Vcode extends HttpServlet{

public void doGet(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

doPost(request,response);

}

public void doPost(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

Vcode vc = new Vcode();

//获取图片对象

BufferedImage bi = vc.getImage();

request.getSession().setAttribute("vscode", vc.getText());

System.out.println("&&&&&&&"+vc.getText());

vc.output(bi, response.getOutputStream());

}

private int w = 70;

private int h = 35;

private Random r = new Random();

// {"宋体", "华文楷体", "黑体", "华文新魏", "华文隶书", "微软雅黑", "楷体\_GB2312"}

private String[] fontNames = {"宋体", "华文楷体", "黑体", "微软雅黑", "楷体\_GB2312"};

// 可选字符

private String codes = "0123456789abcdefghjkmnopqrstuvwxyzABCDEFGHJKMNPQRSTUVWXYZ";

// 背景色

private Color bgColor = new Color(255, 255, 255);

// 验证码上的文本

private String text ;

// 生成随机的颜色

private Color randomColor () {

int red = r.nextInt(150);

int green = r.nextInt(150);

int blue = r.nextInt(150);

return new Color(red, green, blue);

}

// 生成随机的字体

private Font randomFont () {

int index = r.nextInt(fontNames.length);

String fontName = fontNames[index];//生成随机的字体名称

int style = r.nextInt(4);//生成随机的样式, 0(无样式), 1(粗体), 2(斜体), 3(粗体+斜体)

int size = r.nextInt(5) + 24; //生成随机字号, 24 ~ 28

return new Font(fontName, style, size);

}

// 画干扰线

private void drawLine (BufferedImage image) {

int num = 3;//一共画3条

Graphics2D g2 = (Graphics2D)image.getGraphics();

for(int i = 0; i < num; i++) {//生成两个点的坐标，即4个值

int x1 = r.nextInt(w);

int y1 = r.nextInt(h);

int x2 = r.nextInt(w);

int y2 = r.nextInt(h);

g2.setStroke(new BasicStroke(1.5F));

g2.setColor(Color.BLUE); //干扰线是蓝色

g2.drawLine(x1, y1, x2, y2);//画线

}

}

// 随机生成一个字符

private char randomChar () {

int index = r.nextInt(codes.length());

return codes.charAt(index);

}

// 创建BufferedImage

private BufferedImage createImage () {

BufferedImage image = new BufferedImage(w, h, BufferedImage.TYPE\_INT\_RGB);

Graphics2D g2 = (Graphics2D)image.getGraphics();

g2.setColor(this.bgColor);

g2.fillRect(0, 0, w, h);

return image;

}

// 调用这个方法得到验证码

public BufferedImage getImage () {

BufferedImage image = createImage();//创建图片缓冲区

Graphics2D g2 = (Graphics2D)image.getGraphics();//得到绘制环境

StringBuilder sb = new StringBuilder();//用来装载生成的验证码文本

// 向图片中画4个字符

for(int i = 0; i < 4; i++) {//循环四次，每次生成一个字符

String s = randomChar() + "";//随机生成一个字母

sb.append(s); //把字母添加到sb中

float x = i \* 1.0F \* w / 4; //设置当前字符的x轴坐标

g2.setFont(randomFont()); //设置随机字体

g2.setColor(randomColor()); //设置随机颜色

g2.drawString(s, x, h-5); //画图

}

this.text = sb.toString(); //把生成的字符串赋给了this.text

drawLine(image); //添加干扰线

return image;

}

// 返回验证码图片上的文本

public String getText () {

return text;

}

// 保存图片到指定的输出流

public void output (BufferedImage image, OutputStream out)

throws IOException {

ImageIO.write(image, "JPEG", out);

}

}

----------------------------------login.java servlet

**if**(!request.getSession().getAttribute("vscode").equals(request.getParameter("vcod")))

{out.println("<script language='javascript'>alert('验证码输入错误！3秒跳转后，请重新输入！')</script>");

response.setHeader("refresh","3;/Logintest/login.jsp");

}

**else**

**if**(length<6

||length>8

||request.getParameter("key").length()!=6)

request.getRequestDispatcher("failed\_page").forward(request, response);

**else**

{

request.getRequestDispatcher("success\_page" ).forward(request, response);

System.*out*.println(request.getParameter("name")+"同学："+request.getParameter("key"));

}

}

此处对用户输入与验证码值进行判断

第二题----------------猜数游戏

**if**(time==4)//超过5次判定失败

{

out.println("<script language='javascript'>alert('您错误5次了，游戏失败！请耐心等待3秒进入开始页面！')</script>");

response.setHeader("refresh","3;Inputnum"); //通过setHeader转发页面请求

}

**else**

**if**(request.getParameter("num")==**null**)//第一次传递时我未传递num

{

out.println("<input type=\"hidden\" value=\""+

time+"\" name=\"times\" >");

out.println("<input type=\"hidden\" value=\"yes"+"\" name=\"write\" >");

System.*out*.println("TIMES!!!!!!!!!!!"+time);

//request.getRequestDispatcher("compare").forward(request, response);

}

**else**

**if**(!*isInteger*(request.getParameter("num"))||request.getParameter("num").equals(""))//判断是否为合法数字，否则Integer包含的函数无法使用

{out.println("<script language='javascript'>alert('请重新输入')</script>");

out.println("<input type=\"hidden\" value=\""+

time+"\" name=\"times\" >");

}

**else**

{ **int** realnum=Integer.*parseInt*(request.getParameter("secert"));

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

System.*out*.println("TIMES!!!!!!!!!!!"+newnum+realnum+time+"++++++"+request.getParameter("write"));

**if**(newnum>realnum)

{

out.println("<input type=\"hidden\" value=\""+

(time+1)+"\" name=\"times\" >");

out.println("<script language='javascript'>alert('大了')</script>");

}

**else**

**if**(newnum<realnum)

{out.println("<input type=\"hidden\" value=\""+

(time+1)+"\" name=\"times\" >");

out.println("<script language='javascript'>alert('小了')</script>");

}

**else**

{

out.println("<script language='javascript'>alert('对了,您"+(time+1)+"次就猜对了，请耐心等待3秒进入开始页面！')</script>");

response.setHeader("refresh","3;Inputnum");

}

}

以上为判定过程。

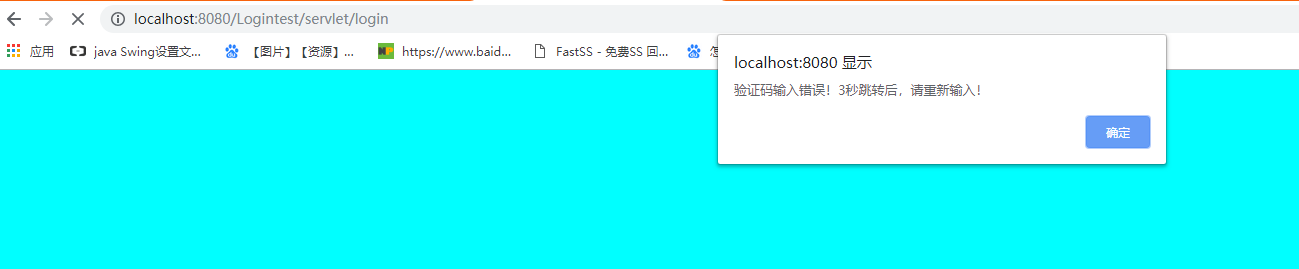
# 运行效果图

第一题





验证码输入错误时：

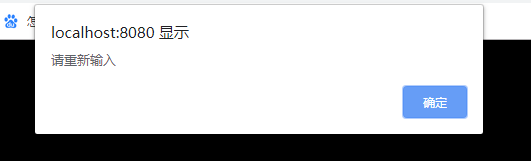


用户名密码输错时：



猜数字游戏

输入字母，非数字



# 收获与问题

在开始做验证码的时候，每次第一次启动页面总会报错，错误名字为“空指针错误”，我看了后发现我在一个java文件里写用out.print提供页面效果会产生顺序上的冲突，应该是先有输入再有获取，而执行servlet就一定会读到底产生不必要的顺序上的错误。后来我把servlet和jsp分开，逻辑和代码都清晰很多。