**Java Web开发**

**实验三**

**作业报告**

|  |  |
| --- | --- |
| 学 院 | 计算机 |
| 所在系 班 级 | 2017软件工程系11班 |
| 学 号 姓 名 | 黄恺 |
| 指 导 教 师 | 郭晓燕 |
| 完 成 时 间 | 2019 年3月20日- |

# 实验题目

必做，以下三题任选，但必须完成一题。

\* 将第二章实训题的第二题，登录页面，提交用户名、密码，若用户名不为空字符串且长度在6-8位间，密码不为，空长度为6位，则登录成功，显示“\*\*\*（用户名）登录成功！”，否则显示“\*\*\*登录失败”。

\* 猜数页面，猜错猜对及输入不正确的情况分别处理，后台随机生成一个0-100的数，让用户猜。最多猜五次，每次猜完后要对用户提示“猜大了”或“猜小了”，如果相等，提示“猜对了，游戏结束！”

\* 上一周做的注册页面也可，验证条件可以自行设定，注册之后根据用户输入情况进行相应的提示

# 分析与设计

做第一个 总共有两个类文件 一个是scccess.Java 一个是注册的html

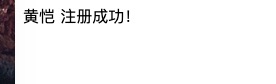
# 关键源码（部分）

1. import java.io.IOException;  
    import javax.servlet.ServletException;  
    import javax.servlet.http.HttpServlet;  
    import javax.servlet.http.HttpServletRequest;  
    import javax.servlet.http.HttpServletResponse;  
     
   public class Success extends HttpServlet {  
    private static final long *serialVersionUID* = 1L;  
     
    public Success() {  
    super();  
     
    }  
    protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {  
    response.setContentType("text/html");  
    request.setCharacterEncoding("UTF-8");  
    response.setContentType("text/html;charset=utf-8");  
    String z=request.getParameter("zh");  
    String m=request.getParameter("mima");  
    String nm=request.getParameter("xingming");  
    String s=request.getParameter("sex");  
    String a=request.getParameter("age");  
    String n=request.getParameter("num");  
    if(nm!=""&&s!=""&&a!=""&&n!=""&&z!=""&&m!="" && (z.length()>5 && z.length() < 9 )&& m.length()==6)  
    {  
    response.getWriter().write(nm+" 注册成功！");  
     
    }  
    else  
    {  
    response.getWriter().write("智商不够请充值，请看清楚要求！");  
     
    }  
    //response.getWriter().append("Served at: ").append(request.getContextPath());  
    }  
     
     
    protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {  
     
    doGet(request, response);  
    }  
     
   }

注册界面

<html>  
  
<head>  
  
 <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">  
  
 <title>Insert title here</title>  
  
</head>  
  
<body>  
  
<form action="Success" method="post">  
  
 <table align="center" border="1" weight="200" height="200">  
  
 <tr align="center"><td>请填写注册信息</td></tr>  
 <tr align="center"><td> 请填写六位数的密码及长度为6-8内的账户名</td></tr>  
 <tr align="center"><td>帐号：<input type="text" name="zh"></td></tr>  
 <tr align="center"><td>密码：<input type="text" name="mima"></td></tr>  
 <tr align="center"><td>姓名：<input type="text" name="xingming"></td></tr>  
  
 <tr align="center"><td>性别：<input type="text" name="sex"></td></tr>  
  
 <tr align="center"><td>年龄：<input type="text" name="age"></td></tr>  
  
 <tr align="center"><td>手机号码：<input type="text" name="num"></td></tr>  
  
 <tr align="center"><td><input type="submit" value="提交信息"></td></tr>  
  
 </table>  
  
</form>  
  
</body>  
  
</html>

# 运行效果图WeChat21c5722ddea69cd5515c4860d98d12ed.png





# 收获与问题

无，没啥新知识点

附加题：

验证码：



代码：两个文件 一个jsp一个servlet

jsp：<%@ **page** language="java" import="java.util.\*" pageEncoding="UTF-8"%>  
  
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">  
<html>  
<head>  
 <title>验证码</title>  
 <script language="javascript">  
 **function** myReload() {  
 document.getElementById("CreateCheckCode").src = document  
 .getElementById("CreateCheckCode").src  
 + "?nocache=" + **new** Date().getTime();  
 }  
 </script>  
</head>  
  
<body>  
<form action="Check.jsp" method="post">  
 <input name="checkCode" type="text" id="checkCode" title="验证码区分大小写"  
 size="8" ,maxlength="4" />  
 <img src="PictureCheckCode" id="CreateCheckCode" align="middle">  
 <a href="" onclick="myReload()"> 看不清,换一个</a>  
 <input type="submit" value="提交" />  
</form>  
</body>  
</html>

servlet

import javax.servlet.ServletException;  
import javax.servlet.annotation.WebServlet;  
import javax.servlet.http.HttpServlet;  
import javax.servlet.http.HttpServletRequest;  
import javax.servlet.http.HttpServletResponse;  
import java.io.IOException;  
import java.awt.BasicStroke;  
import java.awt.Color;  
import java.awt.Font;  
import java.awt.Graphics;  
import java.awt.Graphics2D;  
import java.awt.geom.AffineTransform;  
import java.awt.geom.Line2D;  
import java.awt.image.BufferedImage;  
import java.util.Random;  
import javax.imageio.ImageIO;  
import javax.servlet.http.HttpSession;  
  
@WebServlet(name = "Servlet2")  
public class PictureCheckCode extends HttpServlet {  
  
 private static final long *serialVersionUID* = 1L;  
  
 public PictureCheckCode() {  
 super();  
 }  
  
 public void destroy() {  
 super.destroy();  
 }  
  
 public void init() throws ServletException {  
 super.init();  
 }  
 /\*该方法主要作用是获得随机生成的颜色\*/  
 public Color getRandColor(int s,int e){  
 Random random=new Random ();  
 if(s>255) s=255;  
 if(e>255) e=255;  
 int r,g,b;  
 r=s+random.nextInt(e-s); //随机生成RGB颜色中的r值  
 g=s+random.nextInt(e-s); //随机生成RGB颜色中的g值  
 b=s+random.nextInt(e-s); //随机生成RGB颜色中的b值  
 return new Color(r,g,b);  
 }  
  
 @Override  
 public void service(HttpServletRequest request, HttpServletResponse response)  
 throws ServletException, IOException {  
 //设置不缓存图片  
 response.setHeader("Pragma", "No-cache");  
 response.setHeader("Cache-Control", "No-cache");  
 response.setDateHeader("Expires", 0);  
 //指定生成的响应图片,一定不能缺少这句话,否则错误.  
 response.setContentType("jpeg");  
 int width=86,height=22; //指定生成验证码的宽度和高度  
 BufferedImage image=new BufferedImage(width,height,BufferedImage.*TYPE\_INT\_RGB*); //创建BufferedImage对象,其作用相当于一图片  
 Graphics g=image.getGraphics(); //创建Graphics对象,其作用相当于画笔  
 Graphics2D g2d=(Graphics2D)g; //创建Grapchics2D对象  
 Random random=new Random();  
 Font mfont=new Font("楷体",Font.*BOLD*,16); //定义字体样式  
 g.setColor(getRandColor(200,250));  
 g.fillRect(0, 0, width, height); //绘制背景  
 g.setFont(mfont); //设置字体  
 g.setColor(getRandColor(180,200));  
  
 //绘制100条颜色和位置全部为随机产生的线条,该线条为2f  
 for(int i=0;i<100;i++){  
 int x=random.nextInt(width-1);  
 int y=random.nextInt(height-1);  
 int x1=random.nextInt(6)+1;  
 int y1=random.nextInt(12)+1;  
 BasicStroke bs=new BasicStroke(2f,BasicStroke.*CAP\_BUTT*,BasicStroke.*JOIN\_BEVEL*); //定制线条样式  
 Line2D line=new Line2D.Double(x,y,x+x1,y+y1);  
 g2d.setStroke(bs);  
 g2d.draw(line); //绘制直线  
 }  
 //输出由英文，数字，和中文随机组成的验证文字，具体的组合方式根据生成随机数确定。  
 String sRand="";  
 String ctmp;  
 int itmp=0;  
 //制定输出的验证码为四位  
 for(int i=0;i<4;i++){  
 switch(random.nextInt(3)){  
 case 1: //生成A-Z的字母  
 itmp=random.nextInt(26)+65;  
 ctmp=String.*valueOf*((char)itmp);  
 break;  
 case 2: //生成汉字  
 String[] rBase={"0","1","2","3","4","5","6","7","8","9","a","b","c","d","e","f"};  
 //生成第一位区码  
 int r1=random.nextInt(3)+11;  
 String str\_r1=rBase[r1];  
 //生成第二位区码  
 int r2;  
 if(r1==13){  
 r2=random.nextInt(7);  
 }else{  
 r2=random.nextInt(16);  
 }  
 String str\_r2=rBase[r2];  
 //生成第一位位码  
 int r3=random.nextInt(6)+10;  
 String str\_r3=rBase[r3];  
 //生成第二位位码  
 int r4;  
 if(r3==10){  
 r4=random.nextInt(15)+1;  
 }else if(r3==15){  
 r4=random.nextInt(15);  
 }else{  
 r4=random.nextInt(16);  
 }  
 String str\_r4=rBase[r4];  
 //将生成的机内码转换为汉字  
 byte[] bytes=new byte[2];  
 //将生成的区码保存到字节数组的第一个元素中  
 String str\_12=str\_r1+str\_r2;  
 int tempLow=Integer.*parseInt*(str\_12, 16);  
 bytes[0]=(byte) tempLow;  
 //将生成的位码保存到字节数组的第二个元素中  
 String str\_34=str\_r3+str\_r4;  
 int tempHigh=Integer.*parseInt*(str\_34, 16);  
 bytes[1]=(byte)tempHigh;  
 */\*\*  
 \* 汉字显示  
 \*/* // ctmp=new String(bytes);  
 ctmp = new String(bytes,"gb2312");  
 break;  
 default:  
 itmp=random.nextInt(10)+48;  
 ctmp=String.*valueOf*((char)itmp);  
 break;  
 }  
 sRand+=ctmp;  
 Color color=new Color(20+random.nextInt(110),20+random.nextInt(110),random.nextInt(110));  
 g.setColor(color);  
 //将生成的随机数进行随机缩放并旋转制定角度 PS.建议不要对文字进行缩放与旋转,因为这样图片可能不正常显示  
 /\*将文字旋转制定角度\*/  
 Graphics2D g2d\_word=(Graphics2D)g;  
 AffineTransform trans=new AffineTransform();  
 trans.rotate((45)\*3.14/180,15\*i+8,7);  
 /\*缩放文字\*/  
 float scaleSize=random.nextFloat()+0.8f;  
 if(scaleSize>1f) scaleSize=1f;  
 trans.scale(scaleSize, scaleSize);  
 g2d\_word.setTransform(trans);  
 g.drawString(ctmp, 15\*i+18, 10);  
 }  
 HttpSession session=request.getSession(true);  
 session.setAttribute("randCheckCode", sRand);  
 g.dispose(); //释放g所占用的系统资源  
 ImageIO.*write*(image,"JPEG",response.getOutputStream()); //输出图片  
 }  
  
}