



链滴

JAVA 生成验证码

作者: [Single](#)

原文链接: <https://ld246.com/article/1499420667768>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

```

import java.awt.Color;
import java.awt.Font;
import java.awt.Graphics;
import java.awt.image.BufferedImage;
import java.util.Random;

import javax.imageio.ImageIO;
import javax.servlet.ServletOutputStream;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.servlet.http.HttpSession;

import org.springframework.stereotype.Controller;
import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.servlet.ModelAndView;

import com.google.code.kaptcha.Constants;

@Controller
public class CaptchaController {

    @RequestMapping("/kaptcha")
    public ModelAndView handleRequest(HttpServletRequest request, HttpServletResponse response) throws Exception {
        response.setDateHeader("Expires", 0);
        response.setHeader("Cache-Control", "no-store, no-cache, must-revalidate");
        response.addHeader("Cache-Control", "post-check=0, pre-check=0");
        response.setHeader("Pragma", "no-cache");
        response.setContentType("image/jpeg");
        HttpSession session = request.getSession();

        BufferedImage buImage = new BufferedImage(80, 30, BufferedImage.TYPE_3BYTE_BGR); //
        创建BufferedImage对象
        Graphics gra = buImage.getGraphics(); // 使用BufferedImage对象得到Graphics对象
        Random ran = new Random(); // 随机数对象
        gra.setColor(Color.white); // 设置画布背景色
        gra.setFont(new Font("Times New Roman", Font.ITALIC, 20)); // 设置字体
        gra.fillRect(0, 0, 80, 30); // 此部为绘制一个矩形，其中x,y坐标都是从0开始，宽度为80，高度为30
        String vliCode = "";
        while (vliCode.length() < 4) {
            gra.setColor(new Color(0, 68, 141)); // 随机得到颜色用于绘制内容
            int tmp;
            // the style of the code
            switch (ran.nextInt(3)) {
                case 0:
                    tmp = ran.nextInt(10); // 得到数字0-9
                    break;
                case 1:
                    tmp = ran.nextInt(10); // 得到数字0-9
                    break;
                default:
                    tmp = ran.nextInt(10); // 得到数字0-9
                    break;
            }
            vliCode += tmp + " ";
        }
    }
}

```

```

    }
    gra.drawString(tmp + "", 10 + vliCode.length() * 15, 20); // 此步为在x坐标为 (10+vliCode.
    ength()*20) y坐标为20的地方绘制内容为tmp的字符图。
    vliCode += tmp;
}
// 绘制干扰线
gra.setColor(new Color(0, 68, 141)); // 设置干扰线的颜色
gra.drawLine(0, 15, 100, 5); // 设置干扰线的坐标

//设置超时时间
session.setMaxInactiveInterval(60*10);
session.setAttribute(Constants.KAPTCHA_SESSION_KEY, vliCode);
ServletOutputStream out = response.getOutputStream();
ImageIO.write(bufImage, "jpg", out);
try {
    out.flush();
} finally {
    out.close();
}
return null;
}
}

```