SpringBoot 二维码生成（复制即用）

果粒橙
我是一个普通的程序员...

+关注他

9 人赞同了该文章

二维码生成

基础环境

SpringBoot、Maven

代码

依赖

```
<!--     二维码生成   -->
<dependency>
   <groupId>com.google.zxing</groupId>
   <artifactId>core</artifactId>
   <version>3.3.3</version>
</dependency>
<dependency>
   <groupId>com.google.zxing</groupId>
   <artifactId>javase</artifactId>
   <version>3.3.3</version>
</dependency>
```

工具类

```
import cn.hutool.core.codec.Base64;
import cn.hutool.core.util.StrUtil;
import com.google.zxing.BarcodeFormat;
import com.google.zxing.EncodeHintType;
import com.google.zxing.common.BitMatrix;
import com.google.zxing.qrcode.QRCodeWriter;
import com.google.zxing.qrcode.decoder.ErrorCorrectionLevel;
import lombok.experimental.UtilityClass;
import lombok.extern.slf4j.Slf4j;

import javax.imageio.ImageIO;
import javax.servlet.ServletOutputStream;
```

```java
import java.awt.*;
import java.awt.geom.RoundRectangle2D;
import java.awt.image.BufferedImage;
import java.io.ByteArrayOutputStream;
import java.io.IOException;
import java.io.OutputStream;
import java.net.URL;
import java.util.HashMap;

/**
 * <p>QRCodeUtil</p>
 *
 * @author lvhaosir6
 * @version 1.0.0
 * @date 2020/7/8
 */
@Slf4j
@UtilityClass
public class QRCodeUtil {
    /**
     * 默认宽度
     */
    private static final Integer WIDTH = 140;
    /**
     * 默认高度
     */
    private static final Integer HEIGHT = 140;

    /**
     * LOGO 默认宽度
     */
    private static final Integer LOGO_WIDTH = 22;
    /**
     * LOGO 默认高度
     */
    private static final Integer LOGO_HEIGHT = 22;

    /**
     * 图片格式
     */
    private static final String IMAGE_FORMAT = "png";
    private static final String CHARSET = "utf-8";
    /**
     * 原生转码前面没有 data:image/png;base64 这些字段，返回给前端是无法被解析
     */
```

```
private static final String BASE64_IMAGE = "data:image/png;base64,%s";

/**
 * 生成二维码，使用默认尺寸
 *
 * @param content 内容
 * @return
 */
public String getBase64QRCode(String content) {
    return getBase64Image(content, WIDTH, HEIGHT, null, null, null);
}

/**
 * 生成二维码，使用默认尺寸二维码，插入默认尺寸logo
 *
 * @param content 内容
 * @param logoUrl logo地址
 * @return
 */
public String getBase64QRCode(String content, String logoUrl) {
    return getBase64Image(content, WIDTH, HEIGHT, logoUrl, LOGO_WIDTH, LOGO_HEIGHT);
}

/**
 * 生成二维码
 *
 * @param content    内容
 * @param width      二维码宽度
 * @param height     二维码高度
 * @param logoUrl    logo 在线地址
 * @param logoWidth  logo 宽度
 * @param logoHeight logo 高度
 * @return
 */
public String getBase64QRCode(String content, Integer width, Integer height, String
    return getBase64Image(content, width, height, logoUrl, logoWidth, logoHeight);
}

private String getBase64Image(String content, Integer width, Integer height, String
    ByteArrayOutputStream os = new ByteArrayOutputStream();
    BufferedImage bufferedImage = crateQRCode(content, width, height, logoUrl, logoW
    try {
        ImageIO.write(bufferedImage, IMAGE_FORMAT, os);
    } catch (IOException e) {
        log.error("[生成二维码，错误{}]", e);
```

```java
   }
   // 转出即可直接使用
   return String.format(BASE64_IMAGE, Base64.encode(os.toByteArray()));
}


/**
 * 生成二维码
 *
 * @param content    内容
 * @param width      二维码宽度
 * @param height     二维码高度
 * @param logoUrl    logo 在线地址
 * @param logoWidth  logo 宽度
 * @param logoHeight logo 高度
 * @return
 */
private BufferedImage crateQRCode(String content, Integer width, Integer height, St
   if (StrUtil.isNotBlank(content)) {
      ServletOutputStream stream = null;
      HashMap<EncodeHintType, Comparable> hints = new HashMap<>(4);
      // 指定字符编码为utf-8
      hints.put(EncodeHintType.CHARACTER_SET, CHARSET);
      // 指定二维码的纠错等级为中级
      hints.put(EncodeHintType.ERROR_CORRECTION, ErrorCorrectionLevel.M);
      // 设置图片的边距
      hints.put(EncodeHintType.MARGIN, 2);
      try {
         QRCodeWriter writer = new QRCodeWriter();
         BitMatrix bitMatrix = writer.encode(content, BarcodeFormat.QR_CODE, width,
         BufferedImage bufferedImage = new BufferedImage(width, height, BufferedIma
         for (int x = 0; x < width; x++) {
            for (int y = 0; y < height; y++) {
               bufferedImage.setRGB(x, y, bitMatrix.get(x, y) ? 0xFF000000 : 0xFFFF
            }
         }
         if (StrUtil.isNotBlank(logoUrl)) {
            insertLogo(bufferedImage, width, height, logoUrl, logoWidth, logoHeight
         }
         return bufferedImage;
      } catch (Exception e) {
         e.printStackTrace();
      } finally {
         if (stream != null) {
            try {
                    stream.flush();
                    stream.close();
                } catch (IOException e) {
                    e.printStackTrace();
                }
            }
        }
    }
    return null;
}

/**
 * 二维码插入logo
 *
 * @param source     二维码
 * @param width      二维码宽度
 * @param height     二维码高度
 * @param logoUrl    logo 在线地址
 * @param logoWidth  logo 宽度
 * @param logoHeight logo 高度
 * @throws Exception
 */
private void insertLogo(BufferedImage source, Integer width, Integer height, String
    // logo 源可为 File/InputStream/URL
    Image src = ImageIO.read(new URL(logoUrl));
    // 插入LOGO
    Graphics2D graph = source.createGraphics();
    int x = (width - logoWidth) / 2;
    int y = (height - logoHeight) / 2;
    graph.drawImage(src, x, y, logoWidth, logoHeight, null);
    Shape shape = new RoundRectangle2D.Float(x, y, logoWidth, logoHeight, 6, 6);
    graph.setStroke(new BasicStroke(3f));
    graph.draw(shape);
    graph.dispose();
}


/**
 * 获取二维码
 *
 * @param content 内容
 * @param output  输出流
 * @throws IOException
 */
public void getQRCode(String content, OutputStream output) throws IOException {
        BufferedImage image = crateQRCode(content, WIDTH, HEIGHT, null, null, null);
        ImageIO.write(image, IMAGE_FORMAT, output);
    }

    /**
     * 获取二维码
     *
     * @param content 内容
     * @param logoUrl logo资源
     * @param output  输出流
     * @throws Exception
     */
    public void getQRCode(String content, String logoUrl, OutputStream output) throws E
        BufferedImage image = crateQRCode(content, WIDTH, HEIGHT, logoUrl, LOGO_WIDTH, L
        ImageIO.write(image, IMAGE_FORMAT, output);
    }

}
```

Controller

```
import cn.lvhaosir.common.core.util.QRCodeUtil;
import cn.lvhaosir.common.core.util.R;
import org.springframework.web.bind.annotation.GetMapping;
import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.bind.annotation.RequestParam;
import org.springframework.web.bind.annotation.RestController;

import javax.servlet.ServletOutputStream;
import javax.servlet.http.HttpServletResponse;

/**
 * <p>QRCodeController</p>
 *
 * @author lvhaosir6
 * @version 1.0.0
 * @date 2020/7/10
 */
@RestController
@RequestMapping("/qrcode")
public class QRCodeController {

    /**
```

```
     * 根据 content 生成二维码
     *
     * @param content
     * @param width
     * @param height
     * @return
     */
    @GetMapping("/getQRCodeBase64")
    public R getQRCode(@RequestParam("content") String content,
                       @RequestParam(value = "logoUrl", required = false) String logoU
                       @RequestParam(value = "width", required = false) Integer width,
                       @RequestParam(value = "height", required = false) Integer heigh
        return R.ok(QRCodeUtil.getBase64QRCode(content, logoUrl));
    }

    /**
     * 根据 content 生成二维码
     */
    @GetMapping(value = "/getQRCode")
    public void getQRCode(HttpServletResponse response,
                          @RequestParam("content") String content,
                          @RequestParam(value = "logoUrl", required = false) String lo
        ServletOutputStream stream = null;
        try {
            stream = response.getOutputStream();
            QRCodeUtil.getQRCode(content, logoUrl, stream);
        } catch (Exception e) {
            e.getStackTrace();
        } finally {
            if (stream != null) {
                stream.flush();
                stream.close();
            }
        }
    }

}
```

效果

Base64 字符串

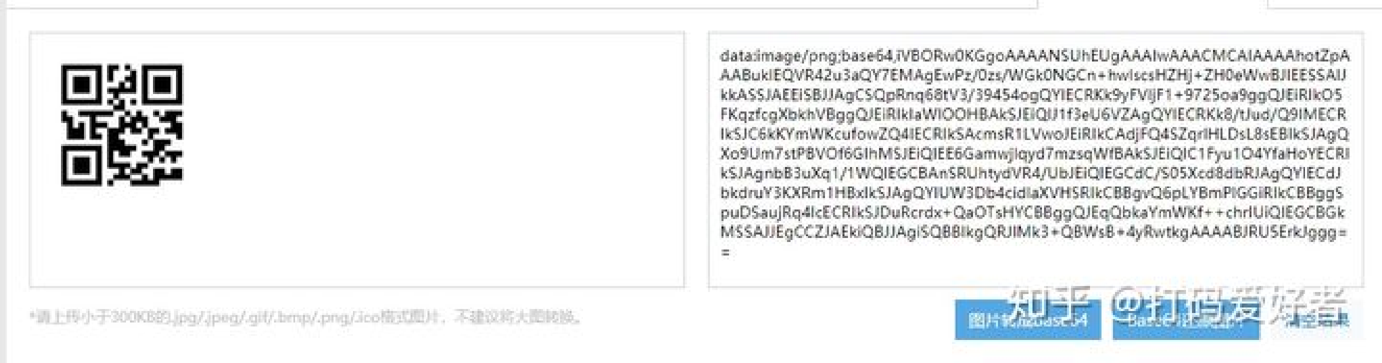
base64 转换为图片在线工具

tool.chinaz.com/tools/i

不带 Logo 的二维码：

```
http://localhost:8080/qrcode/getQRCodeBase64?content=www.baidu.com

{
  "code": 0,
  "msg": "data:image/png;base64,iVBORw0KGgoAAAANSUhEUgAAAIwAAACMCAIAAAAhotZpAAABuklEQV
  "data": null
}
```



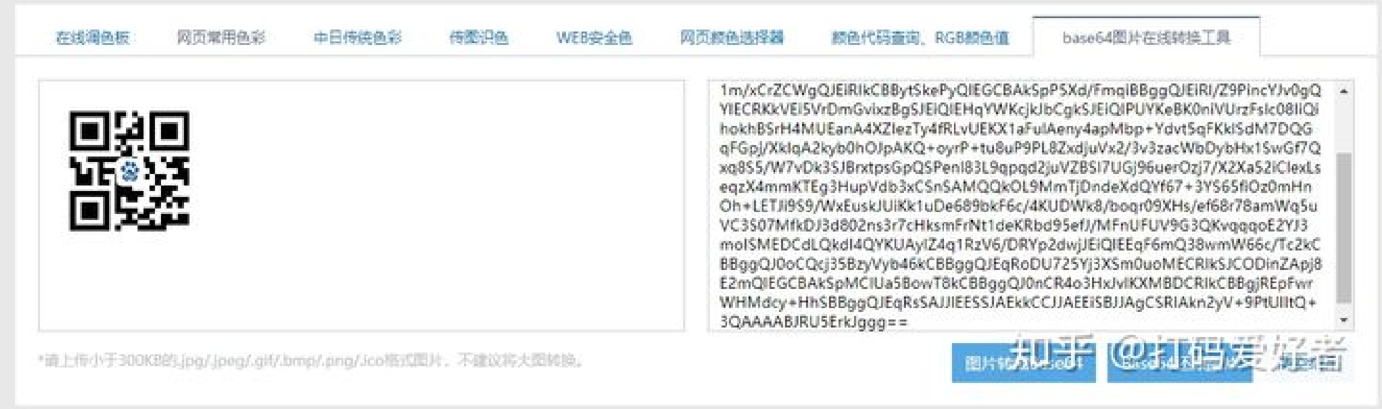
data:image/png;base64,iVBORw0KGgoAAAANSUhEUgAAAIwAAACMCAIAAAAhotZpA
AABuklEQVR42u3aQY7EMAgEwPz/0zs/WGk0NGCn+hwIscsHZHj+ZH0eWwBJIEESSAIJ
kkASSJAEEISBJJAgCSQpRnq68tV3/39454ogQYIECRKk9yFVIjF1+9725oa9ggQJEiRIkO5
FKqzfcgXbkhVBggQJEiRIklaWIOOHBAkSJEiQIJ1f3eU6VZAgQYIECRKk8/tJud/Q9IMECR
IkSJC6kKYmWKcufowZQ4IECRIkSAcmsR1LVwoJEiRIkCAdjFQ4SZqrlHLDsL8sEBIkSJAgQ
Xo9Um7stPBVOf6GIhMSJEiQIEE6GamwjIqyd7mzsqWfBAkSJEiQIC1Fyu1O4YfaHoYECRI
kSJAgnbB3uXq1/1WQIEGCBAnSRUhtydVR4/UbJEiQIEGCdC/S05Xcd8dbRJAgQYIECdJ
bkdruY3KXRm1HBxIkSJAgQYIUW3Db4cidlaXVHSRIkCBBgvQ6pLYBmPIGGiRIkCBBggS
puDSaujRq4lcECRIkSJDuRcrdx+QaOTsHYCBBggQJEqQbkaYmWKf++chrIUiQIEGCBGk
MSSAJJEgCCZJAEkiQBJJAgISQBBIkgQRJIMk3+QBWsB+4yRwtkgAAAABJRU5ErkJggg=
=

*请上传小于300KB的.jpg/.jpeg/.gif/.bmp/.png/.ico格式图片，不建议将大图转换。

带 Logo 的二维码：

```
http://localhost:8080/qrcode/getQRCodeBase64?content=www.baidu.com&logoUrl=http://pic.

{
  "code": 0,
  "msg": "data:image/png;base64,iVBORw0KGgoAAAANSUhEUgAAAIwAAACMCAIAAAAhotZpAAAC40lEQV
  "data": null
}
```



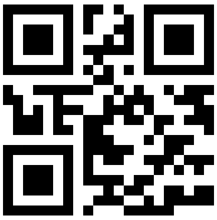
1m/xCrZCWgQJEiRIkCBBytSkePyQIEGCBAkSpP5Xd/FmqIBBggQJEiRI/Z9PincYJv0gQ
YIECRKkVEi5VrDmGvixzBgSJEiQIEHqYWKcjkJbCgkSJEiQIPUYKeBK0nlVUrzFslc08IiQi
hokhBSrH4MUEanA4XZIezTy4fRLvUEKX1aFulAeny4apMbp+Ydvt5qFKkISdM7DQG
qFGpJ/XkIqA2kyb0hOJpAKQ+oyrP+tu8uP9PL8ZxdjuVx2/3v3zacWbDybHx1SwGf7Q
xq855/W7vDk3SJBrxtpsGpQSPenI83L9qpqd2juVZBSI7UGj96uerOzj7/X2XaS2ICIexLs
eqzX4mmKTEg3HupVdb3xCSnSAMQQkOL9MmTjDndeXdQYf67+3YS65fiOz0mHn
Oh+LETJi9S9/WxEuskJUiKk1uDe689bkF6c/4KUDWk8/boqr09XHs/ef68r78amWq5u
VC3S07MfkDJ3d802ns3r7cHksmFrNt1deKRbd95efJ/MFnUFUV9G3QKvqqqoE2YJ3
moISMEDCdLQkdI4QYKUAyIZ4q1RzV6/DRYp2dwjJEiQIEEqF6mQ38wmW66c/Tc2kC
BBggQJ0oCQcj35BzyVyb46kCBBggQJEqRoDU725Yj3XSm0uoMECRIkSJCODinZApj8
E2mQIEGCBAkSpMCIUa5BowT8kCBBggQJ0nCR4o3HxJvIKXMBDCRIkCBBgjREpFwr
WHMdcy+HhSBBggQJEqRsSAJJIEESSJAEkkCCJJAEEiSBJJAgCSRIAkn2yV+9PtUIItQ+
3QAAAABJRU5ErkJggg==

*请上传小于300KB的.jpg/.jpeg/.gif/.bmp/.png/.ico格式图片，不建议将大图转换。

输出流

不带 Logo 的二维码：

localhost:8080/qrcode/g



带 Logo 的二维码：

localhost:8080/qrcode/g