1. 引入zxing jar包
2. 引入QRCodeFactory
3. public class QRCodeFactory {  
     
    private static Logger *logger* = LoggerFactory.*getLogger*(QRCodeFactory.class);  
     
    */\*\*  
    \* 获取二维码，以BufferedImage的形式返回  
    \** ***@param*** *width  
    \** ***@param*** *height  
    \** ***@param*** *margin  
    \** ***@param*** *encoding  
    \** ***@param*** *contents  
    \** ***@return*** *{****@link*** *BufferedImage}  
    \*/* public static BufferedImage getQRCode(  
    int width, int height, int margin, String encoding, String contents) {  
    Map<EncodeHintType, Object> hints = new HashMap<>();  
    hints.put(EncodeHintType.*CHARACTER\_SET*, encoding); //编码格式  
    hints.put(EncodeHintType.*ERROR\_CORRECTION*, ErrorCorrectionLevel.*M*); //容错等级  
    hints.put(EncodeHintType.*MARGIN*, margin); //边框宽度  
     
    BitMatrix bitMatrix = null;  
    try {  
    bitMatrix = new MultiFormatWriter()  
    .encode(contents, BarcodeFormat.*QR\_CODE*, width, height, hints);  
    } catch (WriterException e) {  
    *logger*.error("生成二维码出错", e);  
    }  
    return MatrixToImageWriter.*toBufferedImage*(bitMatrix);  
    }  
     
    */\*\*  
    \* 获取二维码信息，通过Result  
    \** ***@param*** *encoding  
    \** ***@param*** *image  
    \** ***@return*** *{****@link*** *Result}  
    \*/* public static Result decodeQRCode(  
    String encoding, BufferedImage image) {  
    MultiFormatReader multiFormatReader = new MultiFormatReader();  
    Result result = null;  
    try {  
    BinaryBitmap binaryBitmap =  
    new BinaryBitmap(  
    new HybridBinarizer(  
    new BufferedImageLuminanceSource(image)));  
    Map hints = new HashMap();  
    hints.put(EncodeHintType.*CHARACTER\_SET*, encoding);  
    result = multiFormatReader.decode(binaryBitmap, hints);  
    } catch (NotFoundException e) {  
    e.printStackTrace();  
    }  
    return result;  
    }  
     
   }
4. 在需要的地方使用

@RestController  
public class UserController {  
 @RequestMapping(value = "/greeting")  
 public User greeting(@RequestParam(value="name", defaultValue="World") String name) throws IOException {  
 User user = new User();  
 user.setName(name);  
 user.setId(333);  
  
 //获取二维码  
 BufferedImage bufferedImage = QRCodeFactory.*getQRCode*(300, 300, 2, "utf-8", "www.wwww.xxx.sss");  
 ImageIO.*write*(bufferedImage, "jpg", new File("D:\\二维码所需资源\\qqq.jpg"));  
  
 return user;  
 }  
  
}

1. Result 类 //存储二维码信息

result.getBarcodeFormat() //获取二维码类型

result.getText() //获取二维码内容