

在服务器中默认的编码为ISO-8859-1. 它不支持中文，Tomcat规定的
浏览器默认使用GBK编码
Ctrl+shift+O导包

```
printwriter out = response.getWriter (); 得到一个字符输出流
```

在doget中解决中文乱码的问题：

①对于字符输出流

//告诉服务器应用utf-8解析文本

```
response.setCharacterEncoding("utf-8");
```

//告诉客户端使用什么进行编码

```
response.setHeader("Content-type","text/html;charset=utf-8");
```

或者

//做了两件事：告诉服务器应用utf-8解析文本和告诉客户端使用什么进行编码

```
response.setContentType("text/html;charset=UTF-8");
```

②对于字节输出流

//做了两件事：告诉服务器应用utf-8解析文本和告诉客户端使用什么进行编码

```
response.setContentType("text/html;charset=utf-8");
```

```
ServletOutputStream out = response.getOutputStream();
```

//getBytes默认编码为本地编码，即gbk

```
out.write("你好，world!".getBytes("UTF-8"));
```

文件下载：

//通过路径得到一个输入流

```
String path = this.getServletContext().getRealPath("/WEB-INF/classes/te.gif");
```

```
FileInputStream fis = new FileInputStream(path);
```

//创建一个字节输出流

```
ServletOutputStream sos = response.getOutputStream();
```

//告知客户端要下载文件

//得到要下载的文件名

```
String filename = path.substring(path.lastIndexOf("\\")+1);
```

//设置文件名的编码, 将不安全的文件名改为utf-8格式

```

        filename = URLEncoder.encode(filename, "utf-8");
        response.setHeader("content-disposition",
"attachement;filename="+filename);
        response.setHeader("content-type", "image/gif");
//执行输出操作
        int len = 1;
        byte []b = new byte[1024];
        while((len=fis.read(b))!=-1)
        {
            sos.write(b, 0, len);
        }
        sos.close();
        fis.close();

```

将一段代码抽取出来作为一个公共的方法：

选中需要抽取的代码---->右键refactor--->external method---->输入抽取出来的方法名

验证码：

实现方式1：

servlet代码：

```

private void test1(HttpServletResponse response) throws IOException {
    int width = 150;
    int height = 25;
    //在内存中创建一个图片对象
    BufferedImage img = new BufferedImage(width, height,
BufferedImage.TYPE_3BYTE_BGR);

    //创建一个画笔
    Graphics g = img.getGraphics();
    //给图片添加背景色
    g.setColor(Color.white);
    //填充颜色
    g.fillRect(1, 1, width-2, height-2);
    //给边框一个颜色

```

```

g.setColor(Color.RED);
//设置边框的显示坐标
g.drawRect(0, 0, width-1, height-1);
//设置文本样式
g.setColor(Color.cyan);
g.setFont(new Font("宋体",Font.BOLD|Font.ITALIC,15));
//给图片添加内容
Random rd = new Random();
int position = 20;
for(int i = 0;i<6;i++){
    //给图片添加文本
    g.drawString(rd.nextInt(10)+"", position, 20);
    position = position+20;
}
//添加九条干扰线
for (int i = 0; i < 9; i++) {
    g.drawLine(rd.nextInt(width),rd.nextInt(height),
rd.nextInt(width), rd.nextInt(height));
}
//将图片对象以流的方式输出到客户端
ImageIO.write(img, "jpg", response.getOutputStream());
}

```

HTML代码:

```

<html>
<head>
<meta charset="UTF-8">
<title>Insert title here</title>
<script type="text/javascript">

```

```

function changecode() {
    var img = document.getElementsByTagName("img")[0];
    img.src = "/newS/servlet/test4?time="+new Date().getTime();
}

```

```

</script>
</head>

```

```

<body>
<form action="#" method = "post">
用户名<input type = "text" name = "use"/><br/>
密 码<input type = "password" name = "pass"/><br/>
验证码<input type = "text" name = "yzm"/>

<a href = "javascript:changeCode()">看不清，换一张</a>
<br/>
<input type = "submit" value = "提交"/><br/>
</form>
</body>
</html>

```

实现方式2:

使用验证码架包，将该包放在lib文件夹下：在doGet方法下写：

```
ValidateCode vc = new ValidateCode(宽度，高度，需要的验证码个数，需要的验证码
干扰线数);
```

```
vc.write(response.getOutputStream());
```

得到生成的验证码：

```
string code = vc.getCode();
```



ValidateCode.jar
30.37KB

//告诉客户端不使用缓存，处理不同浏览器都不实现缓存

```
response.setHeader("pragma", "no-cache");
```

```
response.setHeader("cache-control", "no-cache");
```

```
response.setHeader("expires", "0");
```

//实现一秒钟刷新一次

```
response.setIntHeader("refresh", 1);
```

//实现三秒钟跳转到另一个页面

```
response.setContentType("text/html; charset=utf-8");
```

```
response.getWriter().write("注册成功！3秒后跳转");
```

```
response.setHeader("refresh", "3;url=/newS/login.html");
```