

```
1
2 import com.cete.dynamicpdf.*;
3 import com.cete.dynamicpdf.pageelements.Image;
4 import com.cete.dynamicpdf.pageelements.Label;
5 import com.cete.dynamicpdf.pageelements.TextArea;
6 import com.cete.dynamicpdf.pageelements.barcoding.InvalidValueBarCodeException;
7 import com.cete.dynamicpdf.pageelements.barcoding.Postnet;
8 import java.io.FileNotFoundException;
9 import java.io.IOException;
10 import javax.servlet.ServletConfig;
11 import javax.servlet.ServletException;
12 import javax.servlet.ServletOutputStream;
13 import javax.servlet.http.HttpServlet;
14 import javax.servlet.http.HttpServletRequest;
15 import javax.servlet.http.HttpServletResponse;
16
17 public class USEnvelope extends HttpServlet {
18
19     ServletOutputStream sOut;
20
21     public void init(ServletConfig config) throws ServletException {
22         super.init(config);
23     }
24
25     public void doGet(HttpServletRequest req, HttpServletResponse res)
26         throws IOException, ServletException {
27
28         sOut = res.getOutputStream();
29         // Create a document and set it's properties
30         Document objDocument = new Document();
31         objDocument.setCreator("USEnvelope.html");
32         objDocument.setAuthor("ceTe Software");
33         objDocument.setTitle("US Envelope");
34
35         // Create a page and add it to the document.
36         Page objPage = new Page(PageSize.ENVELOPE_10, PageOrientation.LANDSCAPE,
37             18);
38         objDocument.getPages().add(objPage);
39
40         // Add page elements to the page
41         objPage.getElements().add(new Image(getServletContext().
42             .getRealPath("/images/DPDFLogo.png"), 0, 2, 0.16f));
43
44         objPage.getElements().add(new TextArea(getFromAddress(req), 50, 0, 350,
45             80, Font.getHelvetica(), 10));
46         objPage.getElements().add(new TextArea(getToAddress(req), 300, 140, 360,
47             100, Font.getHelvetica(), 12));
48         objPage.getElements().add(new Postnet(req.getParameter("txtToZip"), 300,
49             248));
```

```
50 // Outputs the document to the current web page
51 objDocument.drawToWeb(req, res, sOut, "USEnvelope.pdf");
52
53 sOut.close();
54
55 }
56
57
58
59 private String getFromAddress(HttpServletRequest req) {
60     // Returns a formatted From Address
61     return getAddress(req.getParameter("txtFromName"),
62                       req.getParameter("txtFromAddress"),
63                       req.getParameter("txtFromAddress2"),
64                       req.getParameter("txtFromCity"),
65                       req.getParameter("txtFromState"),
66                       req.getParameter("txtFromZip"));
67 }
68
69 private String getToAddress(HttpServletRequest req) {
70     // Returns a formatted To Address
71     return getAddress(req.getParameter("txtToName"),
72                       req.getParameter("txtToAddress"),
73                       req.getParameter("txtToAddress2"),
74                       req.getParameter("txtToCity"),
75                       req.getParameter("txtToState"),
76                       req.getParameter("txtToZip"));
77 }
78
79 private String getAddress(String name, String address, String address2,
80                           String city, String state, String zip) {
81     // Formats a US address
82     String strAddress = name.trim() + "\n";
83     strAddress += address.trim() + "\n";
84     if (address2.trim().length() > 0 ) {
85         strAddress += address2.trim() + "\n";
86     }
87     strAddress += city.trim() + ", " + state.trim() + " " + zip.trim();
88
89     return strAddress;
90 }
91 }
```