TaggedPdfWithStructureElements.java

```
1
2  import com.cete.dynamicpdf.*;
3  import com.cete.dynamicpdf.pageelements.Label;
4  import com.cete.dynamicpdf.pageelements.Image;
5  import com.cete.dynamicpdf.pageelements.TextArea;
6  import com.cete.dynamicpdf.pageelements.OrderedList;
7  import com.cete.dynamicpdf.pageelements.OrderedSubList;
8  import com.cete.dynamicpdf.pageelements.PageNumberingLabel;
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 import java.io.FileNotFoundException;
17
18
19 public class TaggedPdfWithStructureElements extends HttpServlet {
20
21     ServletOutputStream sOut;
22
23     public void init(ServletConfig config) throws ServletException {
24         super.init(config);
25     }
26
27     public void doGet(HttpServletRequest req, HttpServletResponse res)
28                     throws IOException,ServletException {
29
30         sOut = res.getOutputStream();
31
32         // Create a PDF document
33         Document document = new Document();
34         document.setCreator("TaggedPdfWithStructureElements.aspx");
35         document.setAuthor("ceTe Software");
36         document.setTitle("TaggedPdfWithStructureElements");
37
38         // Specify document as a tagged PDF
39         document.setTag(new TagOptions());
40
41         // Create a page and add it to the document
42         Page page = new Page();
43         document.getPages().add(page);
44
45         // Create a text area
46         TextArea textArea1 = new TextArea("This is Paragraph one", 0, 0, 512, 100, Font.getHelvetica(), 15f);
47         textArea1.setAlign(TextAlign.CENTER);
48
49         // Create a structure element of tag type section
```

TaggedPdfWithStructureElements.java

```
50          StructureElement parentStrucutreElement = new StructureElement(TagType.getSection());
51
52          // Create a structure element
53          StructureElement structureElement1 = new StructureElement(TagType.getParagraph());
54          structureElement1.setIncludeDefaultAttributes(true);
55
56          // Set structure element parent
57          structureElement1.setParent(parentStrucutreElement);
58
59          // Set strucute element order
60          structureElement1.setOrder(2);
61
62          // Set structure element to the text area
63          textArea1.setTag(structureElement1);
64
65          // Create a text area
66          TextArea textArea2 = new TextArea("This is Paragraph two", 0, 50, 512, 100, Font.getHelvetica(), 15f);
67          textArea2.setAlign(TextAlign.CENTER);
68
69          // Create a structure element
70          StructureElement structureElement2 = new StructureElement(TagType.getParagraph());
71          structureElement2.setIncludeDefaultAttributes(true);
72
73          // Set structure element parent
74          structureElement2.setParent(parentStrucutreElement);
75
76          // Set strucute element order
77          structureElement2.setOrder(3);
78
79          // Set structure element to the text area
80          textArea2.setTag(structureElement2);
81
82          // Create a text area
83          TextArea textArea3 = new TextArea("This is Paragraph three", 0, 100, 512, 100, Font.getHelvetica(), 15f);
84          textArea3.setAlign(TextAlign.CENTER);
85
86          // Create a structure element
87          StructureElement structureElement3 = new StructureElement(TagType.getParagraph());
88          structureElement3.setIncludeDefaultAttributes(true);
89
90          // Set structure element parent
91          structureElement3.setParent(parentStrucutreElement);
92
93          // Set strucute element order
94          structureElement3.setOrder(1);
95
96          // Set structure element to the text area
97          textArea3.setTag(structureElement3);
98
```

TaggedPdfWithStructureElements.java

```
99          // Add text areas to the page
100         page.getElements().add(textArea1);
101         page.getElements().add(textArea2);
102         page.getElements().add(textArea3);
103          try {
104         // Create an image
105         Image image = new Image(getServletContext().getRealPath("/images/DPDFLogo.png"), 180f, 150f, 0.24f);
106         image.setHeight(200);
107         image.setWidth(200);
108
109         // Create a structure element
110         StructureElement imageStructureElement = new StructureElement(TagType.getFigure());
111         imageStructureElement.setIncludeDefaultAttributes(true);
112         imageStructureElement.setAlternateText("DynamicPDF Logo");
113
114         // Set structure element to the image
115         image.setTag(imageStructureElement);
116
117         // Add image to the page
118         page.getElements().add(image);
119          }
120          catch (FileNotFoundException ex1) {
121             System.err.println("Image file not found :"+ex1);
122         }
123         // Create an ordered list
124         OrderedList orderedList = new OrderedList(165, 400, 300, 300, Font.getHelvetica(), 12);
125
126         // Add items to the ordered list
127         orderedList.getListItems().add("Fruits");
128         orderedList.getListItems().add("Vegetables");
129
130         // Tag the ordered list with the structure element
131         orderedList.setTag(new StructureElement(TagType.getList()));
132
133         // Create an ordered sublist and add items to it
134         OrderedSubList orderedSubList =
orderedList.getListItems().getItem(0).getSubLists().addOrderedSubList(NumberingStyle.ROMAN_UPPERCASE);
135         orderedSubList.getListItems().add("Citrus");
136         orderedSubList.getListItems().add("Non-Citrus");
137
138         // Tag the item one child lists with the structure element
139         orderedList.getListItems().getItem(0).setTag(new StructureElement(TagType.getList(), true));
140
141         // Tag the item one body with the structure element
142         orderedList.getListItems().getItem(0).setBodyTag(new StructureElement(TagType.getListBody(), true));
143
144         // Tag the item one bullet with the structure element
145         orderedList.getListItems().getItem(0).setBulletTag (new StructureElement(TagType.getLabel(), true));
146
```

```
147         // Tag the item one  with the structure element
148         orderedList.getListItems().getItem(0).setListItemTag(new StructureElement(TagType.getListItem(), true));
149 
150         // Tag the sublist with the structure element
151         orderedSubList.setTag(new StructureElement(TagType.getList(), true));
152 
153         // Create an ordered sublist and add items to it
154         OrderedSubList orderedSubList2 =
orderedList.getListItems().getItem(1).getSubLists().addOrderedSubList(NumberingStyle.ROMAN_UPPERCASE);
155         orderedSubList2.getListItems().add("Potato");
156         orderedSubList2.getListItems().add("Beans");
157 
158         // Create an ordered sublist and add items to it
159         OrderedSubList subOrderedSubList =
orderedSubList.getListItems().getItem(0).getSubLists().addOrderedSubList(NumberingStyle.ROMAN_LOWERCASE);
160         subOrderedSubList.getListItems().add("Lime");
161         subOrderedSubList.getListItems().add("Orange");
162 
163         // Create an ordered sublist and add items to it
164         OrderedSubList subOrderedSubList2 =
orderedSubList.getListItems().getItem(1).getSubLists().addOrderedSubList(NumberingStyle.ROMAN_LOWERCASE);
165         subOrderedSubList2.getListItems().add("Mango");
166         subOrderedSubList2.getListItems().add("Banana");
167 
168         // Create an ordered sublist and add items to it
169         OrderedSubList subOrderedSubList3 =
orderedSubList2.getListItems().getItem(0).getSubLists().addOrderedSubList(NumberingStyle.ROMAN_LOWERCASE);
170         subOrderedSubList3.getListItems().add("Sweet Potato");
171 
172         // Create an ordered sublist and add items to it
173         OrderedSubList subOrderedSubList4 =
orderedSubList2.getListItems().getItem(1).getSubLists().addOrderedSubList(NumberingStyle.ROMAN_LOWERCASE);
174         subOrderedSubList4.getListItems().add("String Bean");
175         subOrderedSubList4.getListItems().add("Lima Bean");
176 
177         // Add ordered list to the page
178         page.getElements().add(orderedList);
179 
180         // Create a Page numbering label
181         PageNumberingLabel pageNumberingLabel = new PageNumberingLabel("Page %%CP%% of %%TP%%", 0, 680, 512, 12,
Font.getHelvetica(),
182                 12, TextAlign.CENTER);
183 
184         // Create a artifact and add type
185         Artifact artifact = new Artifact();
186         artifact.setType(ArtifactType.PAGINATION);
187 
188         // Set artifact to the page numbering label
189         pageNumberingLabel.setTag(artifact);
```

```
190
191         // Add page numbering label to the page
192         page.getElements().add(pageNumberingLabel);
193
194         // Outputs the document to current web page.
195         document.drawToWeb(req, res , sOut, "TaggedPdfWithStructureElements.pdf");
196         sOut.close();
197     }
198
199 }
```