

```
1 <%@ Page language="c#" Src="TimeMachine.aspx.cs" Inherits="Web_CSharp.GeneratorExamples.TimeMachine" %>
```

```
1 using System;
2 using System.IO;
3 using ceTe.DynamicPDF;
4 using ceTe.DynamicPDF.PageElements;
5 using ceTe.DynamicPDF.PageElements.BarCoding;
6
7 namespace Web_CSharp.GeneratorExamples
8 {
9     public class TimeMachine : System.Web.UI.Page
10    {
11        // Templates for header and footer elements
12        private Template footerTemplate = new Template();
13        private EvenOddTemplate headerTemplate = new EvenOddTemplate();
14
15        private void Page_Load(object sender, System.EventArgs e)
16        {
17            // Create a document and set it's properties
18            Document document = new Document();
19            document.Creator = "TimeMachine.aspx";
20            document.Author = "H. G. Wells";
21            document.Title = "The Time Machine";
22            document.Template = headerTemplate;
23
24            // Adds elements to the header and footer groups
25            SetPageHeaderTemplate();
26            SetPageFooterTemplate();
27
28            // Sets up outline hierarchy
29            Outline titlePageOutline = document.Outlines.Add( "The Time Machine", new XYDestination( 1, 0, 0 ) );
30            Outline chapterOutline = titlePageOutline.ChildOutlines.Add( "Chapters" );
31
32            // Builds the report
33            BuildDocument( document, titlePageOutline, chapterOutline );
34
35            // Outputs the ContactList to the current web page
36            document.DrawToWeb( "TimeMachine.pdf" );
37        }
38
39        private void BuildDocument( Document document, Outline titlePage, Outline chapters )
40        {
41            // Adds Title page to document
42            document.Sections.Begin( NumberingStyle.None, "Cover" );
43            AddTitlePage( document );
44            document.Sections.Begin( NumberingStyle.Numeric, "Page ", footerTemplate );
45
46            // Adds Chapters to the document
47            AddChapter( document, MapPath( "../Text/TimeMachine_Chapter1.txt" ), "1", "Chapter 1", chapters );
48            AddChapter( document, MapPath( "../Text/TimeMachine_Chapter2.txt" ), "2", "Chapter 2", chapters );
49            AddChapter( document, MapPath( "../Text/TimeMachine_Chapter3.txt" ), "3", "Chapter 3", chapters );
50            AddChapter( document, MapPath( "../Text/TimeMachine_Chapter4.txt" ), "4", "Chapter 4", chapters );
```

```
51     AddChapter( document, MapPath( "../Text/TimeMachine_Chapter5.txt" ), "5", "Chapter 5", chapters );
52     AddChapter( document, MapPath( "../Text/TimeMachine_Chapter6.txt" ), "6", "Chapter 6", chapters );
53     AddChapter( document, MapPath( "../Text/TimeMachine_Chapter7.txt" ), "7", "Chapter 7", chapters );
54     AddChapter( document, MapPath( "../Text/TimeMachine_Chapter8.txt" ), "8", "Chapter 8", chapters );
55     AddChapter( document, MapPath( "../Text/TimeMachine_Chapter9.txt" ), "9", "Chapter 9", chapters );
56     AddChapter( document, MapPath( "../Text/TimeMachine_Chapter10.txt" ), "10", "Chapter 10", chapters );
57     AddChapter( document, MapPath( "../Text/TimeMachine_Chapter11.txt" ), "11", "Chapter 11", chapters );
58     AddChapter( document, MapPath( "../Text/TimeMachine_Chapter12.txt" ), "12", "Chapter 12", chapters );
59
60     // Add the Epilogue to the document
61     AddChapter( document, MapPath( "../Text/TimeMachine_Epilogue.txt" ), "Epilogue", "Epilogue", titlePage );
62 }
63
64 private void SetPageHeaderTemplate()
65 {
66     // Uncomment the line below to add a layout grid to the title header page
67     //headerTemplate.Elements.Add( new LayoutGrid() );
68
69     // Adds elements to the header template
70     headerTemplate.OddElements.Add( new Label( "The Time Machine", -18, -18, 324, 11, Font.TimesRoman, 11,
TextAlign.Left ) );
71     headerTemplate.OddElements.Add( new Label( "H. G. Wells", 18, -18, 324, 11, Font.TimesRoman, 11, TextAlign.Right )
);
72     headerTemplate.EvenElements.Add( new Label( "H. G. Wells", -18, -18, 324, 11, Font.TimesRoman, 11, TextAlign.Left )
);
73     headerTemplate.EvenElements.Add( new Label( "The Time Machine", 18, -18, 324, 11, Font.TimesRoman, 11,
TextAlign.Right ) );
74 }
75
76 private void SetPageFooterTemplate()
77 {
78     // Adds elements to the footer template
79     PageNumberingLabel pageNumberLabel = new PageNumberingLabel( "- %CP%" , 0, 478, 324, 11, Font.TimesRoman, 11,
TextAlign.Center );
80     pageNumberLabel.PageOffset = -1;
81     footerTemplate.Elements.Add( pageNumberLabel );
82 }
83
84 private void AddTitlePage( Document document )
85 {
86     //Adds a title page to the document
87     ceTe.DynamicPDF.Page page = new ceTe.DynamicPDF.Page( 396, 540, 36 );
88
89     // Uncomment the line below to add a layout grid to the title page
90     //page.Elements.Add( new LayoutGrid() );
91
92     string disclaimer = "This document is in the public domain. Permission to use, copy, modify, and distribute this
document for any purpose and without fee is hereby granted, without any conditions or restrictions. The barcode below is for
demonstration purposes only.";
```

```
93     string generated = "Generated by\nDynamicPDF Generator for .NET\non " + DateTime.Now.ToLongDateString() + ",\nat " +
94     page.Elements.Add( new Label( "The Time Machine", 36, 36, 252, 30, Font.TimesBold, 30, TextAlign.Center ) );
95     page.Elements.Add( new Label( "by H. G. Wells", 36, 96, 252, 22, Font.TimesBold, 22, TextAlign.Center ) );
96     page.Elements.Add( new Label( "1895", 36, 148, 252, 22, Font.TimesBold, 22, TextAlign.Center ) );
97     page.Elements.Add( new Image( MapPath( "../Images/DPDFLogo.png" ), 62, 208, 0.21f ) );
98     page.Elements.Add( new Label( generated, 132, 206, 182, 55, Font.TimesRoman, 11 ) );
99     page.Elements.Add( new Label( disclaimer, 36, 274, 252, 66, Font.TimesRoman, 11, TextAlign.Justify ) );
100    page.Elements.Add( new Ean13Sup5( "201234567890", "90000", 82, 360 ) );
101    page.ApplyDocumentTemplate = false;
102    document.Pages.Add( page );
103    }
104
105    private void AddChapter( Document document, string filePath, string title, string bookmarkText, Outline parentOutline )
106    {
107        // Retrieves the text from the sections file
108        string sectionText = GetTextFromFile( filePath );
109
110        // Adds the first page of the section
111        ceTe.DynamicPDF.Page page = AddSectionHeaderPage( title, bookmarkText, parentOutline );
112        page.ApplyDocumentTemplate = false;
113
114        // Creates a TextArea for the sections text
115        TextArea textArea = new TextArea( sectionText, 0, 146, 324, 322, Font.TimesRoman, 11 );
116        textArea.Leading = 14;
117        textArea.ParagraphSpacing = 20;
118        page.Elements.Add( textArea );
119        document.Pages.Add( page );
120
121        // Creates a TextArea for the overflow text
122        textArea = textArea.GetOverflowTextArea( 0, 0, 324, 468 );
123
124        // Loops until no overflow text is found.
125        while( textArea != null )
126        {
127            // Adds a new page to the document
128            page = new ceTe.DynamicPDF.Page( 396, 540, 36 );
129            page.Elements.Add( textArea );
130
131            // Adds new page to the document
132            document.Pages.Add( page );
133
134            // Creates a TextArea for the overflow text
135            textArea = textArea.GetOverflowTextArea();
136        }
137    }
138
139    private ceTe.DynamicPDF.Page AddSectionHeaderPage( string title, string bookmarkText, Outline parentOutline )
140    {
141        // Adds the first page of a section to the document
```

```
142         ceTe.DynamicPDF.Page page = new ceTe.DynamicPDF.Page( 396, 540, 36 );
143
144         // Uncomment the line below to add a layout grid to the section header pages
145         //page.Elements.Add( new LayoutGrid() );
146
147         page.Elements.Add( new Bookmark( bookmarkText, 0, 0, parentOutline ) );
148         page.Elements.Add( new Label( "The Time Machine", 0, 36, 324, 30, Font.TimesBold, 30, TextAlign.Center ) );
149         page.Elements.Add( new Label( title, 0, 96, 324, 22, Font.TimesBold, 22, TextAlign.Center ) );
150         page.Elements.Add( new Line( 120, 128, 204, 128 ) );
151         PageNumberingLabel pageNumberLabel = new PageNumberingLabel( "- %%CP%% -", 0, 478, 324, 11, Font.TimesRoman, 11,
TextAlign.Center );
152         pageNumberLabel.PageOffset = -1;
153         page.Elements.Add( pageNumberLabel );
154         return page;
155     }
156
157     private string GetTextFromFile( string filePath )
158     {
159         // Opens a text file and returns the text from it.
160         StreamReader textFile = File.OpenText( filePath );
161         string sectionText = textFile.ReadToEnd();
162         textFile.Close();
163         return sectionText;
164     }
165 }
166 }
```