

```
1 <%@ Page language="c#" Src="Invoice.aspx.cs" Inherits="Web_CSharp.GeneratorExamples.Invoice" %>
2 <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN" >
3
4 <html>
5   <head>
6     <title>DynamicPDF Generator v10.0 for .NET - Invoice Example</title>
7     <link rel="stylesheet" type="text/css" href="../DynamicPDF.css">
8   </head>
9   <body MS_POSITIONING="GridLayout">
10    <div id="banner">
11      
12      <div id="header"><i>Dynamic</i><b>PDF</b> Generator v10.0 for .NET Examples</div>
13      <h1>Invoice Example</h1>
14    </div>
15    <div id="content">
16      <form id="FormFill" method="get" target="_blank" runat="server">
17        <h1>Select the invoices to include in the PDF.</h1>
18        <asp:CheckBoxList id="chkInvoices" runat="server">
19          <asp:ListItem Value="10248">10248</asp:ListItem>
20          <asp:ListItem Value="10249">10249</asp:ListItem>
21          <asp:ListItem Value="10250">10250</asp:ListItem>
22          <asp:ListItem Value="10251">10251</asp:ListItem>
23          <asp:ListItem Value="10252">10252</asp:ListItem>
24          <asp:ListItem Selected=True Value="10360">10360</asp:ListItem>
25          <asp:ListItem Selected=True Value="10979">10979</asp:ListItem>
26          <asp:ListItem Value="11077">11077</asp:ListItem>
27        </asp:CheckBoxList>
28        <asp:Button id="btnCreateInvoices" Text="Create Invoices" runat="server" />
29      </form>
30    </div>
31  </body>
32 </html>
```

```
1 using System;
2 using System.Data;
3 using System.Data.SqlClient;
4 using ceTe.DynamicPDF;
5 using ceTe.DynamicPDF.PageElements;
6 using ceTe.DynamicPDF.PageElements.BarCoding;
7
8 namespace Web_CSharp.GeneratorExamples
9 {
10     public class Invoice : System.Web.UI.Page
11     {
12         #region Member Variables
13
14         protected System.Web.UI.WebControls.CheckBoxList chkInvoices;
15         protected System.Web.UI.WebControls.Button btnCreateInvoices;
16         protected System.Web.UI.HtmlControls.HtmlForm FormFill;
17
18         // Connection string to the Northwind database
19         static string dBConnString =
20             System.Configuration.ConfigurationSettings.AppSettings["NorthWindConnectionString"];
21
22         #endregion
23
24         #region Page_Load Method
25
26         private void Page_Load(object sender, System.EventArgs e)
27         {
28             if( IsPostBack )
29             {
30                 CreateInvoices();
31             }
32         }
33
34         #endregion
35
36         #region Private Methods
37
38         private void CreateInvoices()
39         {
40             // Create a document and set it's properties
41             Document document = new Document();
42             document.Creator = "Invoice.aspx";
43             document.Author = "ceTe Software";
44             document.Title = "Invoice";
45
46             //Add the template to the document
47             document.Template = MyInvoice.GetTemplate;
48
49             // Establishes connection to the database
50             SqlConnection connection = GetOpenDBConn();
```

```
51
52     // Add Invoices to the document
53     foreach( System.Web.UI.WebControls.ListItem item in chkInvoices.Items )
54     {
55         if( item.Selected )
56         {
57             MyInvoice invoice = new MyInvoice( this.Server );
58             invoice.Draw( connection, document, int.Parse( item.Value ) );
59         }
60     }
61
62     // Cleans up database connections
63     connection.Close();
64
65     // Outputs the Invoices to the current web page
66     try
67     {
68         document.DrawToWeb( "Invoice.pdf" );
69     }
70     catch( EmptyDocumentException )
71     {
72         // Show an empty document error
73         ShowErrorDocument( "Document is empty. No invoices were selected.", "" );
74     }
75     catch( Exception excGeneral )
76     {
77         // Show a general error
78         ShowErrorDocument( excGeneral.Message, excGeneral.StackTrace );
79     }
80     finally
81     {
82         Visible = false;
83     }
84 }
85
86 private void ShowErrorDocument( string message, string stackTrace )
87 {
88     // Shows the error in a PDF document
89     Response.Clear();
90     Document document = new Document();
91     ceTe.DynamicPDF.Page page = new ceTe.DynamicPDF.Page( PageSize.Letter, PageOrientation.Portrait );
92
93     TextArea messageArea = new TextArea( message, 0, 0, 512, 36, Font.Helvetica, 18, RgbColor.Red );
94     messageArea.Height = messageArea.GetRequiredHeight();
95     page.Elements.Add( messageArea );
96
97     if( stackTrace.Trim().Length > 0 )
98     {
99         float stackTraceTop = messageArea.Y + messageArea.Height + 20;
100
```

```
101         TextArea stackTraceArea = new TextArea( stackTrace, 0, stackTraceTop, 512, 692 - stackTraceTop, Font.Helvetica,
102
103         page.Elements.Add( new ceTe.DynamicPDF.PageElements.Label( "StackTrace:", 0, stackTraceTop - 12, 512, 12,
Font.HelveticaBold, 10 ) );
104         page.Elements.Add( stackTraceArea );
105     }
106     document.Pages.Add( page );
107     document.DrawToWeb();
108 }
109
110 private SqlConnection GetOpenDBConn()
111 {
112     // Creates and opens a database connection
113     SqlConnection connection = new SqlConnection( dBConnString );
114     connection.Open();
115     return connection;
116 }
117
118 #endregion
119
120 #region MyInvoice Class
121
122 private class MyInvoice
123 {
124     #region Member Variables
125
126     private decimal subTotal = 0;
127     private decimal freight = 0;
128     private static Template template = new Template();
129     private static bool pageTemplateImagesSet = false;
130     private static WebColor backgroundColor = new WebColor( "#E0E0FF" );
131     private static WebColor totalBGColor = new WebColor( "#FFC0C0" );
132     private static WebColor thankYouText = new WebColor( "#000080" );
133
134     #endregion
135
136     #region Constructors
137
138     static MyInvoice()
139     {
140         // Uncomment the line below to add a layout grid to the pages
141         //template.Elements.Add( new LayoutGrid() );
142
143         // Top part of Invoice
144         template.Elements.Add( new Label( "Northwind Traders\n1234 International Drive\nAnywhere, Earth ABC123", 56, 0,
200, 44, Font.Helvetica ) );
145         template.Elements.Add( new Label( "Invoice", 0, 0, 540, 18, Font.HelveticaBold, 18, TextAlign.Right ) );
146         template.Elements.Add( new PageNumberingLabel( "Page %%SP%% of %%ST%% ", 450, 253, 90, 20, Font.HelveticaBold,
12, TextAlign.Center ) );
```

```
147
148     // Add Invoice Details Template
149     template.Elements.Add( GetDetailsGroup() );
150
151     // Add BillTo Template
152     template.Elements.Add( GetBillToGroup() );
153
154     // Add ShipTo Template
155     template.Elements.Add( GetShipToGroup() );
156
157     // Add Line Item Template
158     template.Elements.Add( GetLineItemGroup() );
159 }
160
161 public MyInvoice( System.Web.HttpServerUtility server )
162 {
163     // Sets the image to the page template
164     // This is done in the constructor because MapPath is not available in the static methods
165     SetPageTemplateImage( server );
166 }
167
168 #endregion
169
170 public static Template GetTemplate
171 {
172     get { return template; }
173 }
174
175 #region Static Methods
176
177 private static Group GetDetailsGroup()
178 {
179     // Returns a group containing the details template
180     Group group = new Group();
181     group.Add( new Rectangle( 340, 24, 200, 72, RgbColor.Black, backgroundColor, 0.5f ) );
182     group.Add( new Label( "Order ID:", 343, 25, 85, 12, Font.HelveticaBold, 12, TextAlign.Right ) );
183     group.Add( new Label( "Order Date:", 343, 39, 85, 12, Font.HelveticaBold, 12, TextAlign.Right ) );
184     group.Add( new Label( "Customer ID:", 343, 53, 85, 12, Font.HelveticaBold, 12, TextAlign.Right ) );
185     group.Add( new Label( "Shipped Date:", 343, 67, 85, 12, Font.HelveticaBold, 12, TextAlign.Right ) );
186     group.Add( new Label( "Order ID:", 343, 25, 85, 12, Font.HelveticaBold, 12, TextAlign.Right ) );
187     group.Add( new Label( "Shipped Via:", 343, 81, 85, 12, Font.HelveticaBold, 12, TextAlign.Right ) );
188     return group;
189 }
190
191 private static Group GetBillToGroup()
192 {
193     // Returns a group containing the bill to template
194     Group group = new Group();
195     group.Add( new Rectangle( 25, 120, 200, 90, 0.5f ) );
196     group.Add( new Line( 25, 136, 225, 136, 0.5f ) );
```

```
197         group.Add( new Label( "Bill To:", 28, 121, 200, 12, Font.HelveticaBold, 12 ) );
198         return group;
199     }
200
201     private static Group GetShipToGroup()
202     {
203         // Returns a group containing the ship to template
204         Group group = new Group();
205         group.Add( new Rectangle( 315, 120, 200, 90, 0.5f ) );
206         group.Add( new Line( 315, 136, 515, 136, 0.5f ) );
207         group.Add( new Label( "Ship To:", 318, 121, 200, 12, Font.HelveticaBold, 12 ) );
208         return group;
209     }
210
211     private static Group GetLineItemGroup()
212     {
213         // Returns a group containing the line items template
214         Group group = new Group();
215
216         for( int i = 0; i < 10; i++ )
217         {
218             group.Add( new Rectangle( 0, 306 + i * 36, 540, 18, RgbColor.Black, backgroundColor, 0.0F ) );
219         }
220
221         group.Add( new Rectangle( 450, 250, 90, 20, 0.5f ) );
222         group.Add( new Rectangle( 450, 702, 90, 18, RgbColor.Black, totalBGCColor, 0.0F ) );
223         group.Add( new Rectangle( 0, 270, 540, 450, 0.5f ) );
224         group.Add( new Line( 0, 288, 540, 288, 0.5f ) );
225         group.Add( new Line( 0, 666, 540, 666, 0.5f ) );
226         group.Add( new Line( 60, 270, 60, 666, 0.5f ) );
227         group.Add( new Line( 360, 270, 360, 720, 0.5f ) );
228         group.Add( new Line( 450, 270, 450, 720, 0.5f ) );
229         group.Add( new Line( 450, 702, 540, 702, 0.5f ) );
230         group.Add( new Label( "Quantity", 0, 272, 60, 12, Font.HelveticaBold, 12, TextAlign.Center ) );
231         group.Add( new Label( "Description", 60, 272, 300, 12, Font.HelveticaBold, 12, TextAlign.Center ) );
232         group.Add( new Label( "Unit Price", 360, 272, 90, 12, Font.HelveticaBold, 12, TextAlign.Center ) );
233         group.Add( new Label( "Price", 450, 272, 90, 12, Font.HelveticaBold, 12, TextAlign.Center ) );
234         group.Add( new Label( "Thank you for your purchase.\nWe appreciate your business.", 5, 672, 350, 54,
Font.HelveticaBold, 18, thankYouText ) );
235         group.Add( new Label( "Sub Total", 364, 668, 82, 12, Font.HelveticaBold, 12, TextAlign.Right ) );
236         group.Add( new Label( "Freight", 364, 686, 82, 12, Font.HelveticaBold, 12, TextAlign.Right ) );
237         group.Add( new Label( "Total", 364, 704, 82, 12, Font.HelveticaBold, 12, TextAlign.Right ) );
238
239         return group;
240     }
241
242     #endregion
243
244     #region Private Methods
245
```

```
246 private void SetPageTemplateImage( System.Web.HttpServerUtility server )
247 {
248     // Adds the image to page template if it is not already added
249     if(!pageTemplateImagesSet )
250     {
251         template.Elements.Add(new Image(server.MapPath( "../Images/DPDFLogo.png" ), 0, 0, 0.16f));
252         pageTemplateImagesSet = true;
253     }
254 }
255
256 public void Draw( SqlConnection connection, Document document, int invoiceNumber )
257 {
258     // Gets the Invoice data
259     SqlDataReader data = GetInvoiceData( connection, invoiceNumber );
260
261     // Adds the invoice if there is data
262     if( data.Read() )
263     {
264         // Sets the Freight amount
265         freight = (decimal)data[ "Freight" ];
266
267         // Draws the invoice data, returns a page object if it is the last page
268         Page lastPage = DrawInvoiceData( data, document );
269
270         // Draws additional pages if necessary
271         while( lastPage == null )
272         {
273             lastPage = DrawInvoiceData( data, document );
274         }
275
276         // Draws the totals to the bottom of the last page of the Invoice
277         DrawTotals( data, lastPage );
278     }
279
280     // Close DataReader
281     data.Close();
282 }
283
284 private Page DrawInvoiceData( SqlDataReader data, Document document )
285 {
286     // Tracks if the invoice is finished
287     bool invoiceFinished = true;
288
289     // Tracks the y position on the page
290     float yOffset = 288;
291
292     // Create a page for the Invoice
293     Page page = new Page( PageSize.Letter, PageOrientation.Portrait, 36.0f );
294
295     // Add Details to the Invoice
```

```
296 DrawInvoiceDetails( data, page );
297
298 // Add bill to address
299 DrawBillTo( data, page );
300
301 // Add ship to address
302 DrawShipTo( data, page );
303
304 // Add line items to the Invoice
305 while( DrawLineItem( data, page, ref yOffset ) )
306 {
307     // Break if at the bottom of the page
308     if( yOffset >= 666 )
309     {
310         invoiceFinished = false;
311         break;
312     }
313 }
314
315 // Add the page to the document
316 document.Pages.Add( page );
317
318 // If Invoice is finished return the page else return null so another page will be added
319 if( invoiceFinished )
320 {
321     return page;
322 }
323 else
324 {
325     page.Elements.Add( new Label( "Continued...", 454, 704, 82, 12, Font.Helvetica, 12, TextAlign.Right ) );
326     return null;
327 }
328 }
329
330 private void DrawInvoiceDetails( SqlDataReader data, Page page )
331 {
332     // Adds Invoice details to the page
333     page.Elements.Add( new Label( data[ "OrderID" ].ToString(), 437, 25, 100, 12, Font.Helvetica, 12 ) );
334     page.Elements.Add( new Label( ((DateTime)data[ "OrderDate" ]).ToString( "d MMM yyyy" ), 437, 39, 100, 12,
Font.Helvetica, 12 ) );
335     page.Elements.Add( new Label( data[ "CustomerID" ].ToString(), 437, 53, 100, 12, Font.Helvetica, 12 ) );
336
337     if( !( data[ "ShippedDate" ] is DBNull ) )
338     {
339         page.Elements.Add( new Label( ((DateTime)data[ "ShippedDate" ]).ToString( "d MMM yyyy" ), 437, 67, 100, 12,
Font.Helvetica, 12 ) );
340     }
341
342     page.Elements.Add( new Label( data[ "ShipperName" ].ToString(), 437, 81, 100, 12, Font.Helvetica, 12 ) );
343     page.Elements.Add( new Interleaved25( data[ "OrderID" ].ToString(), 380, 4, 18, false ) );
```



```
344     }
345
346     private void DrawBillTo( SqlDataReader data, Page page )
347     {
348         // Adds bill to address
349         string billToAddress =
350             data[ "BillName" ].ToString() + "\n" +
351             data[ "BillAddress" ].ToString() + "\n" +
352             data[ "BillCity" ].ToString() + ", " +
353             data[ "BillPostalCode" ].ToString() + "\n" +
354             data[ "BillCountry" ].ToString();
355         page.Elements.Add( new TextArea( billToAddress, 28, 139, 194, 70, Font.Helvetica, 12 ) );
356     }
357
358     private void DrawShipTo( SqlDataReader data, Page page )
359     {
360         // Adds ship to address
361         string shipToAddress =
362             data[ "ShipName" ].ToString() + "\n" +
363             data[ "ShipAddress" ].ToString() + "\n" +
364             data[ "ShipCity" ].ToString() + ", " +
365             data[ "ShipPostalCode" ].ToString() + "\n" +
366             data[ "ShipCountry" ].ToString();
367         page.Elements.Add( new TextArea( shipToAddress, 318, 139, 194, 70, Font.Helvetica, 12 ) );
368     }
369
370     private bool DrawLineItem( SqlDataReader data, Page page, ref float yOffset )
371     {
372         // Adds a line item to the invoice
373         Int16 quantity = (Int16)data[ "Quantity" ];
374         decimal unitPrice = (decimal)data[ "UnitPrice" ];
375         decimal lineTotal = unitPrice * quantity;
376         subTotal += lineTotal;
377
378         page.Elements.Add( new Label( quantity.ToString(), 4, 3 + yOffset, 52, 12, Font.Helvetica, 12, TextAlign.Right )
379 );
380         page.Elements.Add( new Label( data[ "ProductName" ].ToString(), 64, 3 + yOffset, 292, 12, Font.Helvetica, 12 )
381 );
382         page.Elements.Add( new Label( unitPrice.ToString( "#,##0.00" ), 364, 3 + yOffset, 82, 12, Font.Helvetica, 12,
383 TextAlign.Right ) );
384         page.Elements.Add( new Label( lineTotal.ToString( "#,##0.00" ), 454, 3 + yOffset, 82, 12, Font.Helvetica, 12,
385 TextAlign.Right ) );
386         yOffset += 18;
387
388         // Advances the recordset and returns if there are additional records
389         return data.Read();
390     }
391
392     private void DrawTotals( SqlDataReader data, Page page )
393     {
```

```
390         // Add totals to the bottom of the Invoice
391         decimal grandTotal = subTotal + freight;
392         page.Elements.Add( new Label( subTotal.ToString( "#,##0.00" ), 454, 668, 82, 12, Font.Helvetica, 12,
TextAlign.Right ) );
393         page.Elements.Add( new Label( freight.ToString( "#,##0.00" ), 454, 686, 82, 12, Font.Helvetica, 12,
TextAlign.Right ) );
394         page.Elements.Add( new Label( grandTotal.ToString( "#,##0.00" ), 454, 704, 82, 12, Font.Helvetica, 12,
TextAlign.Right ) );
395     }
396
397     private SqlDataReader GetInvoiceData( SqlConnection connection, int invoiceNumber )
398     {
399         // Creates a DataReader for the Invoice
400         SqlCommand command = connection.CreateCommand();
401         command.CommandText =
402             "SELECT o.OrderID, o.CustomerID, o.OrderDate, o.ShippedDate, Freight, " +
403             "      o.ShipName, o.ShipAddress, o.ShipCity, o.ShipPostalCode, o.ShipCountry, " +
404             "      c.CompanyName as BillName, c.Address as BillAddress, " +
405             "      c.City as BillCity, c.PostalCode as BillPostalCode, " +
406             "      c.Country as BillCountry, s.CompanyName as ShipperName, " +
407             "      p.ProductName, od.UnitPrice, od.Quantity " +
408             "FROM    Orders o " +
409             "JOIN    Customers c ON " +
410             "      o.CustomerID = c.CustomerID " +
411             "JOIN    Shippers s ON " +
412             "      o.ShipVia = s.ShipperID " +
413             "JOIN    [Order Details] od ON " +
414             "      o.OrderID = od.OrderID " +
415             "JOIN    Products p ON " +
416             "      od.ProductID = p.ProductID " +
417             "WHERE   o.OrderID = " + invoiceNumber;
418         SqlDataReader dataReader = command.ExecuteReader();
419         return dataReader;
420     }
421     #endregion
422 }
423 #endregion
424 }
425 }
```