ExcelTemplate

Description

The `ExcelTemplate` object represents an ExcelWriter template spreadsheet.

C#

```
public class ExcelTemplate
```

vb.net

```
Public Class ExcelTemplate
```

Remarks

An ExcelWriter template is a file created in Microsoft Excel that contains data markers.

A data marker specifies a database column, variable, or array to insert in the spreadsheet column containing the marker.

The `ExcelTemplate` object is in the `SoftArtisans.OfficeWriter.ExcelWriter` namespace. The object can be referenced as `SoftArtisans.OfficeWriter.ExcelWriter.ExcelTemplate`. To minimize typing and errors, use an `Import` directive to import the namespace to the aspx page, and reference the object as `ExcelTemplate`, without the namespace prefix.

Examples

If you are coding directly in the .aspx page, following the Page directive, include:

```
<%@ Import Namespace="SoftArtisans.OfficeWriter.ExcelWriter" %>
```

If you are coding in the "code behind" page (.aspx.vb or .aspx.cs), include an `Imports` or `using` statement at the top of the "code behind" page:

C#

```
using SoftArtisans.OfficeWriter.ExcelWriter;
```

vb.net

```
Imports SoftArtisans.OfficeWriter.ExcelWriter
```

To create an instance of the `ExcelTemplate` object use:

C#

```
ExcelTemplate oExcelTemplate = new ExcelTemplate();
```

vb.net

```
Dim oExcelTemplate As New ExcelTemplate()
```

Fields

ALL_ROWS	Used as DataBindingProperties.MaxRows, to indicate that as much of the data source should be imported as possible.

Properties

Name	Description
ContentType	Sets or returns the generated spreadsheet's MIME content type.
DecryptPassword	To decrypt an encrypted template, set DecryptPassword to the password set using EncryptPassword or in Microsoft Excel (Tools -> Options... -> Security).
EncryptPassword	If you set EncryptPassword, ExcelWriter will encrypt the generated spreadsheet - when Process is called - using RC4 encryption.
ExcludeMacros	Sets whether ExcelWriter should remove template macros from the generated spreadsheet or include them.
LicenseKey	Sets or returns the 23-character license key that enables ExcelWriter for the current instance.
PreserveStrings	If you set PreserveStrings to `true`, ExcelWriter will preserve numeric strings as strings.
RemoveExtraDataMarkers	Sets whether ExcelWriter should remove data markers in the template that do not bind to data sources in code.
StretchCellReferencesInCharts	Sets or returns whether to stretch cell references in charts.
Version	Returns the exact version of ExcelWriter.

Methods

Name	Description
BindCellData(Object, String, DataBindingProperties)	Sets an object as a data source for a single cell in a template.
BindColumnData(Object(), String, DataBindingProperties)	Sets an array of objects as a data source for a single column in the template.
BindColumnData(System.Collections.IDictionary, String, DataBindingProperties)	Sets a column's data source to an `IDictionary`. An `IDictionary` represents a collection of key-and-value pairs.
BindColumnData(System.Collections.IEnumerable, String, DataBindingProperties)	Sets a column's data source to an `IEnumerable` collection. The `IEnumerable` interface supports a simple iteration over a collection.
BindData(Object()(), String(), String, DataBindingProperties)	Sets a two-dimensional - possibly jagged - array of objects as a template data source.
BindData(Object(,), String(), String, DataBindingProperties)	Sets a rectangular array of objects as a template data source.
BindData(System.Data.IDataReader, String, DataBindingProperties)	Sets an ADO.NET IDataReader as a data source to bind to template data markers.
BindData(System.Data.DataSet, String, DataBindingProperties)	Sets an ADO.NET DataSet as a data source to bind to template data markers.

BindData(System.Data.DataTable, String, DataBindingProperties)	Sets an ADO.NET DataTable as a data source to bind to template data markers.
BindRowData(Object(), String(), String, DataBindingProperties)	Sets an array of objects as a data source for a single row in a template.? When binding an array horizontally, you must insert a datamarker for each element you want displayed.
BindRowData(System.Collections.IDictionary, String, DataBindingProperties)	Sets a row's data source to an `IDictionary`. An `IDictionary` rep resents a collection of key-and-value pairs. When binding an IDictionary horizontally, you must insert a datamarker for each value you want displayed.
BindRowData(System.Collections.IEnumerable, String(), String, DataBindingProperties)	Sets a row's data source to an `IEnumerable` collection. The `IEnum erable` interface supports a simple iteration over a collection. This method will return only one row of data for the column bound by the datamarker. You must insert a datamarker into the template for each column of data you want shown.
BindRowData(System.Data.DataSet, String, DataBindingProperties)	Sets a DataSet as a data source to bind to a row in the template. This method will import only the first row of the first DataTable of the DataSet. You must insert a datamarker into the template for each column of data you want shown.
BindRowData(System.Data.DataTable, String, DataBindingProperties)	Sets a DataTable as a data source to bind to a row in the template. This method will import only the first row of the DataTable. You must insert a datamarker into the template for each column of data you want shown.
BindRowData(System.Data.DataView, String, DataBindingProperties)	Sets a DataView as a data source to bind to a row in the template. This method will import only the first row of the DataView. You must insert a datamarker into the template for each column of data you want shown.
CreateDataBindingProperties()	Creates a DataBindingProperties object for assigning the property values associated with one or more data binding calls.
ExcelTemplate()	Creates a new ExcelTemplate object.
Open(ExcelApplication, Workbook)	Passes a spreadsheet from ExcelApplication to ExcelTemplate.
Open(String)	Opens an ExcelWriter template spreadsheet. A template is a file created in Microsoft Excel that contains data markers where data source values will be inserted.
Open(System.IO.Stream)	Opens an ExcelWriter template from a `System.IO.Stream`. An ExcelWriter template is a file created in Microsoft Excel that contains data markers where data source values will be inserted.
Process()	The Process method enters data source values in a template's data markers, and creates the output file (the new spreadsheet) in memory. The Save method can then save the output file to disk, stream it to the browser, or both.
Save(String)	Saves the generated Excel file on the server. The ASPNET account (IIS5), NETWORK SERVICE account (IIS6 or IIS7), or the authenticated user must have Write access to the destination directory specified by the `outputFileName` parameter.
Save(System.IO.Stream)	Generates an Excel binary file and streams it to the specified `System .IO.Stream` or, a class derived from `System.IO.Stream` (for example, `System.IO.FileStream`). If you pass Save a `System.I O.FileStream`, ExcelWriter will save the generated file on the server. If you pass `Save Response.OutputStream`, ExcelWriter will stream the the generated file to the client.
Save(System.Web.HttpResponse)	If you pass Save an `HttpResponse` object object, ExcelWriter will stream the generated file to the client.
Save(System.Web.HttpResponse, String, Boolean)	If you pass Save an HttpResponse object, ExcelWriter will stream the generated file to the client. This method allows you to specify a default client-side file name, and whether the file should be opened in the browser window or in Microsoft Excel.

Extension Methods

Name	Description
BindData(Microsoft.SharePoint.SPView, Microsoft.SharePoint.SPList, String, DataBindingProperties)	Sets a SharePoint View as a template data source.
BindData(Microsoft.SharePoint.SPList, String, DataBindingProperties)	Sets a SharePoint List as a template data source.
BindRowData(Microsoft.SharePoint.SPView, Microsoft.SharePoint.SPList, String, DataBindingProperties)	Sets a SharePoint View as a data source for a single row in a template.
BindRowData(Microsoft.SharePoint.SPList, String, DataBindingProperties)	Sets a SharePoint List as a data source for a single row in a template.
Open(Microsoft.SharePoint.SPDocumentLibrary, String)	Opens a spreadsheet from a SharePoint Document Library.
Open(Microsoft.SharePoint.SPListItem, String)	Opens a spreadsheet from a SharePoint List Item.
Save(Microsoft.SharePoint.SPDocumentLibrary, String, Boolean)	Generates an Excel binary or OOXML file and saves it to a SharePoint Document Library.
Save(Microsoft.SharePoint.SPListItem, String)	Generates an Excel binary or OOXML file and saves it as an attachment to a SharePoint list item.