

WordApplication

Description

[WordApplication](#) is the main class for pure code-based Word document generation (for information about template-based workbook generation, see [WordTemplate](#)). This class is an engine used to open, create, and write (save to disk or stream to a browser) Word documents. A single instance of [WordApplication](#) can generate multiple documents.

C#

```
public sealed class WordApplication
```

vb.net

```
Public NotInheritable Class WordApplication
```

Remarks

WordApplication is only available in **WordWriter Enterprise Edition**.

A generated document is represented by the [Document](#) class.

The [WordApplication](#) class is in the `SoftArtisans.OfficeWriter.WordWriter` namespace. The class can be referenced as `SoftArtisans.OfficeWriter.WordWriter`. To minimize typing and errors, import the namespace to the aspx page, and reference the class as `WordApplication`, without the namespace prefix. If you are coding directly in the .aspx page, following the `Page` directive, include:

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

C#

```
using SoftArtisans.OfficeWriter.WordWriter;
```

vb.net

```
Imports SoftArtisans.OfficeWriter.WordWriter
```

Examples

C#

```
//--- Create an instance of the WordApplication class
WordApplication oWordApplication = new WordApplication();
```

vb.net

```
'--- Create an instance of the WordApplication class
Dim oWordApplication As New WordApplication()
```

Properties

Name	Description
AlwaysUseUnicode	Returns a boolean that represents whether or not WordWriter will write all text using unicode. A value of <code>false</code> can result in smaller file sizes but text that is not part of cp1252 (Windows-1252) will be unreadable. This is set to <code>true</code> by default. It is recommended that you should set this to <code>false</code> if it is known that only characters supported by Windows-1252 are in the document, especially when creating large documents.
LicenseKey	Sets or returns the license key used for this instance of WordApplication . This property is mainly used for debugging purposes. For managing your license keys, please use the License Key Manager that is installed with OfficeWriter.
Preserve	Sets or returns a boolean that represents whether or not WordWriter will preserve feature of an existing document that it does not natively support. Setting this to <code>false</code> can result in smaller file sizes, but features that are not natively supported by WordWriter will be stripped out. This is set to <code>true</code> by default.
Version	Returns the exact version of WordWriter . The format of the version string returned is:

Methods

Name	Description
Create()	Creates and returns a new Document based on the Normal style.
Open(String)	Opens and returns an existing Document from disk.
Open(System.IO.Stream)	Opens and returns an existing Document from the specified Stream .
Open(WordTemplate)	Takes a WordTemplate object representing a template Word file, and opens the file as a Document object.
Save(Document, String)	Writes a Document to a file.
Save(Document, System.IO.Stream)	Writes a Document to a stream in the form of a Microsoft Word document.
Save(Document, System.Web.HttpResponse, String, Boolean)	Streams a Document to an HttpResponse with the option of having it open in the user's web browser. The <code>fileName</code> specified will appear in the browser window. In addition, the content-type to use when streaming the file is specified.
Save(Document, System.Web.HttpResponse, String, Boolean, String)	Streams a Document to an HttpResponse with the option of having it open in the user's web browser. The <code>fileName</code> specified will appear in the browser window. In addition, the content-type to use when streaming the file is specified.
WordApplication()	Creates a new WordApplication object

Extension Methods

Overload	Description
<code>Open(Microsoft.SharePoint.SPDocumentLibrary, String)</code>	Opens and returns an existing Document from a SharePoint Document Library.
<code>Open(Microsoft.SharePoint.SPListItem, String)</code>	Opens and returns an existing Document from a SharePoint List Item.
<code>Save(Document, Microsoft.SharePoint.SPDocumentLibrary, String, Boolean)</code>	Writes a Document to a file and saves it to a SharePoint Document Library.
<code>Save(Document, Microsoft.SharePoint.SPListItem, String)</code>	Writes a Document to a file and saves it to a SharePoint List Item.