

SharePointIntegration API

Sections

SharePointIntegration

The `SharePointIntegration` class contains the extension methods that allow users to bind SharePoint data sources to Word documents as well as open and save files in SharePoint lists and document libraries.

- `SharePointIntegration.ImportData(Element, Microsoft.SharePoint.SPList)` — Imports the data from a SharePoint List into a Table at the point of the current element in the document.
- `SharePointIntegration.ImportData(Element, Microsoft.SharePoint.SPView, Microsoft.SharePoint.SPList, DataImportProperties)` — Imports the data from a SharePoint View into a Table at the point of the current element in the document. Also accepts a SharePoint List to which the View belongs and a `DataImportProperties` object with the import options.
- `SharePointIntegration.Open(WordApplication, Microsoft.SharePoint.SPDocumentLibrary, String)` — Opens and returns an existing Document from a SharePoint Document Library.
- `SharePointIntegration.Open(WordApplication, Microsoft.SharePoint.SPListItem, String)` — Opens and returns an existing Document from a SharePoint List Item.
- `SharePointIntegration.Open(WordTemplate, Microsoft.SharePoint.SPDocumentLibrary, String)` — Opens the `WordWriter` template specified by the `templateFileName` parameter.
- `SharePointIntegration.Open(WordTemplate, Microsoft.SharePoint.SPListItem, String)` — Opens the `WordWriter` template specified by the `templateFileName` parameter.
- `SharePointIntegration.Save(WordApplication, Document, Microsoft.SharePoint.SPDocumentLibrary, String, Boolean)` — Writes a Document to a file and saves it to a SharePoint Document Library.
- `SharePointIntegration.Save(WordApplication, Document, Microsoft.SharePoint.SPListItem, String)` — Writes a Document to a file and saves it to a SharePoint List Item.
- `SharePointIntegration.Save(WordTemplate, Microsoft.SharePoint.SPDocumentLibrary, String, Boolean)` — Saves the generated Word file to a SharePoint Document Library.
- `SharePointIntegration.Save(WordTemplate, Microsoft.SharePoint.SPListItem, String)` — Saves the generated Word file to a SharePoint List Item.
- `SharePointIntegration.SetDataSource(WordTemplate, Microsoft.SharePoint.SPList, String)` — Sets a main document data source to a SharePoint List. If the specified List contains more than one row, `WordWriter` will use the first row as the data source.
- `SharePointIntegration.SetDataSource(WordTemplate, Microsoft.SharePoint.SPList)` — Sets a main document data source to a SharePoint List. If the specified List contains more than one row, `WordWriter` will use the first row as the data source.
- `SharePointIntegration.SetDataSource(WordTemplate, Microsoft.SharePoint.SPView, Microsoft.SharePoint.SPList, String)` — Sets a main document data source to a SharePoint View. If the specified View contains more than one row, `WordWriter` will use the first row as the data source.
- `SharePointIntegration.SetDataSource(WordTemplate, Microsoft.SharePoint.SPView, Microsoft.SharePoint.SPList)` — Sets a main document data source to a SharePoint View. If the specified View contains more than one row, `WordWriter` will use the first row as the data source.
- `SharePointIntegration.SetMailMerge(WordTemplate, Microsoft.SharePoint.SPList, Int32)` — Sets the `MailMerge`'s data source to a SharePoint View. A `MailMerge` by default treats the page content as a repeat block, repeating the page content for each row unless `NEXT` fields are used. If the `WordTemplate.EnableNEXTFields` property is set to true, the `NEXT` field can also be used to indicate that the next row of data should be inserted instead of the current row at the next occurrence of the merge fields.
- `SharePointIntegration.SetMailMerge(WordTemplate, Microsoft.SharePoint.SPList)` — Sets the `MailMerge`'s data source to a SharePoint View. A `MailMerge` by default treats the page content as a repeat block, repeating the page content for each row unless `NEXT` fields are used. If the `WordTemplate.EnableNEXTFields` property is set to true, the `NEXT` field can also be used to indicate that the next row of data should be inserted instead of the current row at the next occurrence of the merge fields.
- `SharePointIntegration.SetMailMerge(WordTemplate, Microsoft.SharePoint.SPView, Microsoft.SharePoint.SPList, Int32)` — Sets the `MailMerge`'s data source to a SharePoint View. A `MailMerge` by default treats the page content as a repeat block, repeating the page content for each row unless `NEXT` fields are used. If the `WordTemplate.EnableNEXTFields` property is set to true, the `NEXT` field can also be used to indicate that the next row of data should be inserted instead of the current row at the next occurrence of the merge fields.
- `SharePointIntegration.SetMailMerge(WordTemplate, Microsoft.SharePoint.SPView, Microsoft.SharePoint.SPList)` — Sets the `MailMerge`'s data source to a SharePoint View. A `MailMerge` by default treats the page content as a repeat block, repeating the page content for each row unless `NEXT` fields are used. If the `WordTemplate.EnableNEXTFields` property is set to true, the `NEXT` field can also be used to indicate that the next row of data should be inserted instead of the current row at the next occurrence of the merge fields.
- `SharePointIntegration.SetRepeatBlock(WordTemplate, Microsoft.SharePoint.SPList, String, Int32)` — Sets the specified repeat block's data source to a SharePoint List.

- `SharePointIntegration.SetRepeatBlock(WordTemplate, Microsoft.SharePoint.SPList, String)` — Sets the specified repeat block's data source to a SharePoint List.
- `SharePointIntegration.SetRepeatBlock(WordTemplate, Microsoft.SharePoint.SPView, Microsoft.SharePoint.SPList, String, Int32)` — Sets the specified repeat block's data source to a SharePoint View.
- `SharePointIntegration.SetRepeatBlock(WordTemplate, Microsoft.SharePoint.SPView, Microsoft.SharePoint.SPList, String)` — Sets the specified repeat block's data source to a SharePoint View.

Introduced in build 4.1.0.1379