

ExcelTemplate.BindRowData(System.Collections.IEnumerable, String(), String, DataBindingProperties)

Description

Sets a row's data source to an [IEnumerable](#) collection. The [IEnumerable](#) 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.

C#

```
public virtual void BindRowData(System.Collections.IEnumerable dataSource,
System.String[] columnNames, System.String dataSourceName, DataBindingProperties
property)
```

vb.net

```
Public Overridable Sub BindRowData(ByVal dataSource As System.Collections.IEnumerable,
ByVal columnNames As String(), ByVal dataSourceName As String, ByVal [property] As
DataBindingProperties)
```

Parameters

dataSource

An [IEnumerable](#) collection to use as the data source.

colNames

The names of the columns to get from the data source. If the `columnNames` parameter is null, field binding can only be performed by ordinal (for example, `%%=DSN.#1` or `%%=$DSN`). If `colNames` is specified, both ordinal field binding and named field binding can be used.

dataSourceName

The name of the data marker at which to insert the values imported from the data source. For example, to bind a data source to the data marker `%%=Orders.OrderId`, the value of `dataSourceName` should be "Orders". The string passed must begin with a letter.

property

The [DataBindingProperties](#) object which contains information about how the data should be bound to the template. `property` Must be specified, but the [DataBindingProperties](#) need not be set beforehand. To bind data to a template with the default [DataBindingProperties](#), pass in `ExcelTemplate.CreateDataBindingProperties()` as the `property` value. Otherwise, use the `ExcelTemplate.CreateDataBindingProperties()` method to generate a new [DataBindingProperties](#) object and set the [DataBindingProperties.MaxRows](#), [DataBindingProperties.Transpose](#), and/or [DataBindingProperties.WorksheetName](#) properties for the workbook.

Exceptions

ArgumentNullException

`BindRowData` will throw this exception if `null` (C#) or `Nothing` (VB.NET) is passed to the method.

ArgumentException

Remarks

You can set several data sources for a single template. Use the following methods to set template data sources: [BindCellData](#), [BindColumnData](#), [BindRowData](#), and [BindData](#).

Examples

C#

```
ExcelTemplate xlt = new ExcelTemplate();
xlt.Open(@"C:\ExcelWriter\EmployeeOrdersTemplate.xls");

//--- Bind the IEnumerable collection to the
//--- %=Orders.[Field] data marker.
xlt.BindRowData(OrdersColl,
    null,
    "Orders",
    xlt.CreateDataBindingProperties());
xlt.Process();
xlt.Save(Page.Response, "EmployeeOrders.xls", false);
```

vb.net

```
Dim xlt As New ExcelTemplate();
xlt.Open("C:\ExcelWriter\EmployeeOrdersTemplate.xls")

'--- Bind the IEnumerable collection to the
'--- %=Orders.[Field] data marker.
xlt.BindRowData(OrdersColl, _
    null, _
    "Orders", _
    xlt.CreateDataBindingProperties())
xlt.Process()
xlt.Save(Page.Response, "EmployeeOrders.xls", False)
```