

ExcelTemplate.BindData(System.Collections.Generic.IEnumerable(T), String, DataBindingProperties)

Introduced in build 8.6.1

Description

Sets an IEnumerable<T> as a data source to bind to template data markers.

C#

```
public void BindData(System.Collections.Generic.IEnumerable<T> dataSource,
    System.String dataSourceName, DataBindingProperties property)
```

vb.net

```
Public Sub BindData(ByVal dataSource As System.Collections.Generic.IEnumerable(Of T),
    ByVal dataSourceName As String, ByVal [property] As DataBindingProperties)
```

Parameters

dataSource

The IEnumerable<T> collection to use as the data source.

dataSourceName

The name of the set of data markers at which to insert the values imported from the data source. *dataSourceName* must be specified, but can be left as null or an empty string if this is the first data source bound AND the data markers in the template use the [short data marker syntax](#) or refer to the datasource by number rather than name. Note: *dataSourceName* does not include a data marker's column name, for example, the *dataSourceName* for `%%=Products.ProductID` is "Products."

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

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

SARuntimeException

`BindData` will throw this exception if the data source contains more rows than the worksheet can hold.

If there is more than one data marker referring to a data source and the data source is forward only, the exception will be thrown only if the source is larger than all bindings can hold.

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();

    interface IOrder
    {
        int OrderID;
        string Customer;
        DateTime OrderDate;
        float OrdersTotal;
    }

    IEnumerable<IOrder> orders = GetOrders();
    xlt.BindData(orders, "Orders", xlt.CreateDataImportProperties());
```

vb.net

```
Dim xlt As New ExcelTemplate()
    Interface IOrder
        Property OrderID() as Int32
        Property Customer() as String
        Property OrderDate() as DateTime
        Property OrdersTotal() as Single
    End Interface

    Dim orders as IEnumerable(Of Order) = GetOrders()
    xlt.BindData(orders, "Orders", xlt.CreateDataImportProperties())
```