

# ExcelTemplate.BindRowData(System.Data.DataSet, String, DataBindingProperties)

## Description

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.

### C#

```
public void BindRowData(System.Data.DataSet dataSource, System.String dataSourceName,
    DataBindingProperties property)
```

### vb.net

```
Public Sub BindRowData(ByVal dataSource As System.Data.DataSet, ByVal dataSourceName
    As String, ByVal [property] As DataBindingProperties)
```

## Parameters

### **source**

The DataSet to use as the data source.

### **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

Only the first row of the DataSet will be used. This method is commonly used to create a set of key-value pairs via a single row of data.

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 DataSet to the
    ///--- %=Orders.[Field] data marker.
    xlt.BindRowData(OrdersDS,
        "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 DataSet to the
    '--- %=Orders.[Field] data marker.
    xlt.BindRowData(OrdersDS, _
        "Orders", _
        xlt.CreateDataBindingProperties())
    xlt.Process()
    xlt.Save(Page.Response, "EmployeeOrders.xls", False)
```