

WordTemplate.SetRepeatBlock(System.Data.IDataReader, String, Int32)

Description

Sets the specified repeat block's data source to an IDataReader interface, which may be either a SqlDataReader, OleDbDataReader, or an AdomdDataReader.

C#

```
public void SetRepeatBlock(System.Data.IDataReader dr, System.String bookmark, int maxRows)
```

vb.net

```
Public Sub SetRepeatBlock(ByVal dr As System.Data.IDataReader, ByVal bookmark As String, ByVal maxRows As Integer)
```

Parameters

dr

a SqlDataReader, OleDbDataReader or and AdomdDataReader to use as the data source.

bookmark

The bookmark name of the template repeat block. The bookmark passed to SetRepeatBlock must exist in the template Word file. To see a list of template bookmark names:

1. Open the template in Microsoft Word.
2. Open the **Edit** menu.
3. Select **Go To...**
4. Select **Bookmark**.
5. Under **Enter bookmark name**, open the drop-down list of bookmark names.

maxRows

Specifies the maximum number of repetitions to write to the file. If Next fields are enabled (see [EnableNEXTFields](#)), then the number of rows imported from the data source is $\text{maxRows} * (\text{numberOfNextFields} + 1)$. If Next fields are not enabled the number of rows imported is equal to the number of repetitions written out. To write the maximum rows available, use the constant [WordTemplate.ALL_ROWS](#).

Exceptions

ArgumentNullException

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

ArgumentException

Introduced in build 8.4

SAException

SetRepeatBlock will throw this exception if the bookmark has been set to be removed.

Remarks

A repeat block is a fragment in the template document that will be repeated for each row in a data source. In the template document, repeat blocks are defined by Word bookmarks that contain merge fields.

You can call [SetRepeatBlock](#) several times for a single instance of [WordTemplate](#). The repeat block specified by the parameter bookmark must exist in the template.

If you pass an empty `DataReader` to `SetRepeatBlock` - by either setting `maxRows` to 0 or passing an empty `DataReader` - `WordWriter` will not include the specified repeat block in the generated Word file.

Not all Word features can be included in a repeat block:

\$body

Supported in Repeat Blocks	Not Supported in Repeat Blocks
• Character Formatting (everywhere) • Paragraph Formatting (Alignment, Outlines, Indent, Spacing, Page Break before, Keep Together) • Multiple Columns (2, 3, Column Breaks) • Borders and Shading • Tabs • Bullets • Numbering (numbering not reset, it is continued from one repeat block to another) • Page Breaks • Section Breaks • Auto Text Field • Hyperlink • Pictures • Table Row Repeat (entire row only) • Fields (with some restrictions)	• Nested repeat blocks (Only one data source may be assigned to a single repeat block.) • Overlapping bookmarks • Comments • Drawing Objects • Text Boxes • Footnote and endnote references • Table and picture indexes • Single cell in a table • Smart tags (smart tags will be removed from the document) • Repeat blocks in headers, footers, footnotes, comments, text boxes, etc.

\$body

Examples

C#

```
//--- Get a DataReader
//--- WordWriter accepts OleDbDataReader, SqlDataReader, and AdomdDataReader
OleDbDataReader dr = GetOleDbDataReader();
WordTempalte oWW = new WordTemplate();
oWW.Open(Page.MapPath("./template.doc"));

//--- Set a data source for the repeat block
//--- defined by the bookmark "Page"
//--- "dr" is the DataReader containing information to be
//--- imported and repeated in the repeat block
//--- Set MaxRows to 10. Only 10 rows from the DataReader
//--- will be imported
oWW.SetRepeatBlock(dr, "Page", 10);
oWW.Process();
oWW.Save(Page.Response, "EmployeeOutput.doc", false);

//--- Close the DataReader in a finally block
dr.Close();

private OleDbDataReader GetOleDbDataReader()
{
    OleDbConnection Conn = new OleDbConnection(connString);

    //--- SQL Query for categories.
    string CategorySQL = "SELECT CategoryID, CategoryName," +
        "Description FROM Categories";
    Conn.Open();
    return new OleDbCommand(CategorySQL, Conn).ExecuteReader(
        CommandBehavior.CloseConnection);
}
```

```
'--- Get a DataReader
'--- WordWriter accepts OleDbDataReader, SqlDataReader, and AdomdDataReader
Dim dr As OleDbDataReader = GetOleDbDataReader()
Dim oWW As New WordTemplate()
oWW.Open(Page.MapPath("./template.doc"))

'--- Set a data source for the repeat block
'--- defined by the bookmark "Page"
'--- "dr" is the DataReader containing information to be
'--- imported and repeated in the repeat block
'--- Set MaxRows to 10. Only 10 rows from the DataReader
'--- will be imported
oWW.SetRepeatBlock(dr, "Page", 10)
oWW.Process()
oWW.Save(Page.Response, "EmployeeOutput.doc", False)

'--- Close the DataReader in a finally block
dr.Close()

Private Function GetOleDbDataReader() As OleDbDataReader
    Dim Conn As New OleDbConnection(connString)

    '--- SQL Query for categories.
    Dim CategorySQL As String = "SELECT CategoryID, CategoryName," & _
        "Description FROM Categories"
    Conn.Open()
    Return New OleDbCommand(CategorySQL, Conn).ExecuteReader( _
        CommandBehavior.CloseConnection)
End Function
```