

GemBox Support Center

Portal > Knowledgebase > GemBox.Document > Keep table on same page

Keep table on same page

Mario - GemBox - 2017-05-08 - 0 Comments - in GemBox.Document

To prevent or avoid a [Table](#) element from being split across pages, we need to format [Paragraph](#) elements inside that [Table](#) element so that their [KeepLinesTogether](#) and [KeepWithNext](#) properties are set to [True](#).

For the demonstration, we will create a new document with a [Table](#) element that will not fit on a single page:

C# code

```
DocumentModel document = new DocumentModel();
Section section = new Section(document);
document.Sections.Add(section);

Paragraph paragraph = new Paragraph(document,
    new Run(document, "Keep Table on same page.")
    { CharacterFormat = { Size = 48 } })
{ ParagraphFormat = { SpaceAfter = 200, SpaceBefore = 200 } };
section.Blocks.Add(paragraph);

Table table = new Table(document, 30, 5);
table.TableFormat.PreferredWidth = new TableWidth(100, TableWidthUnit
.Percentage);
section.Blocks.Add(table);

document.Save("TableOnTwoPages.docx");
```

VB.NET code

```
Dim document As New DocumentModel()
Dim section As New Section(document)
document.Sections.Add(section)

Dim paragraph As New Paragraph(document,
    New Run(document, "Keep Table on same page.") With {.CharacterFor
mat = New CharacterFormat With {.Size = 48}}) _
```

```

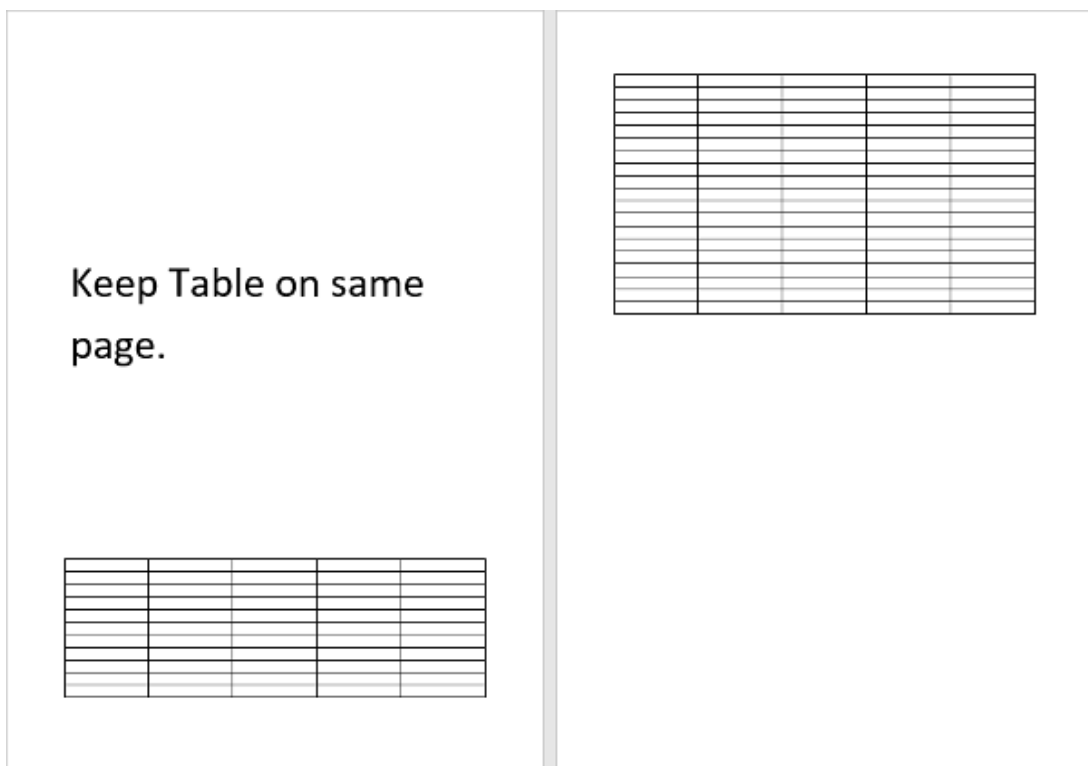
With {.ParagraphFormat = New ParagraphFormat With {.SpaceAfter = 200,
    .SpaceBefore = 200}}
section.Blocks.Add(paragraph)

Dim table As New Table(document, 30, 5)
table.TableFormat.PreferredWidth = New TableWidth(100, TableWidthUnit
    .Percentage)
section.Blocks.Add(table)

document.Save("TableOnTwoPages.docx")

```

This is the resulting `TableOnTwoPages.docx` file:



Now, to keep the whole `Table` element on the same page, before saving the `DocumentModel` we'll iterate through the `Table` element's child `Paragraph` elements and set their `ParagraphFormat.KeepLinesTogether` and `ParagraphFormat.KeepWithNext` properties to `True`:

C# code

```

// Get all Paragraph formats in a Table element.
IEnumerable<ParagraphFormat> formats = table.GetChildElements(true, E
    lementType.TableCell)
    .Cast<TableCell>()

```

```

        .SelectMany(cell =>
        {
            if (cell.Blocks.Count == 0)
                cell.Blocks.Add(new Paragraph(cell.Document));
            return cell.GetChildElements(true, ElementType.Paragraph);
        })
        .Cast<ParagraphFormat>()
        .Select(p => p.ParagraphFormat);

// Set KeepLinesTogether and KeepWithNext properties.
foreach (ParagraphFormat format in formats)
{
    format.KeepLinesTogether = true;
    format.KeepWithNext = true;
}

document.Save("TableOnOnePage.docx");

```

VB.NET code

```

' Get all Paragraph formats in a Table element.
Dim formats As IEnumerable(Of ParagraphFormat) = table.GetChildElements(
    True, ElementType.TableCell) _
    .Cast(Of TableCell)() _
    .SelectMany(
        Function(cell)
            If cell.Blocks.Count = 0 Then
                cell.Blocks.Add(New Paragraph(cell.Document))
            End If
            Return cell.GetChildElements(True, ElementType.Paragraph)
        End Function) _
    .Cast(Of ParagraphFormat)() _
    .Select(Function(p) p.ParagraphFormat)

' Set KeepLinesTogether and KeepWithNext properties.
For Each format As ParagraphFormat In formats
    format.KeepLinesTogether = True
    format.KeepWithNext = True
Next

```

