

How to: Programmatically Format Text in Documents

You can use the [Range](#) object to format text in a Microsoft Office Word document.

Applies to: The information in this topic applies to document-level projects and VSTO add-in projects for Word. For more information, see [Features Available by Office Application and Project Type](#).

The following example selects the first paragraph in the document and changes the font size, the font name, and the alignment. It then selects the range and displays a message box to pause before executing the next section of code. The next section calls the Undo method of the [Microsoft.Office.Tools.Word.Document](#) host item (for a document-level customization) or the [Microsoft.Office.Interop.Word.Document](#) class (for a VSTO Add-in) three times. It applies the Normal Indent style and displays a message box to pause the code. Then the code calls the [Undo](#) method once, and displays a message box.

Document-Level Customization Example

To format text using a document-level customization

1. The following example can be used in a document-level customization. To use this code, run it from the `ThisDocument` class in your project.

C#

```
private void RangeFormat()
{
    // Set the Range to the first paragraph.
    Word.Range rng = this.Paragraphs[1].Range;

    // Change the formatting. To change the font size for a right-to-left
    language,
    // such as Arabic or Hebrew, use the Font.SizeBi property instead of
    Font.Size.
    rng.Font.Size = 14;
    rng.Font.Name = "Arial";
    rng.ParagraphFormat.Alignment =
    Word.WdParagraphAlignment.wdAlignParagraphCenter;

    rng.Select();
    MessageBox.Show("Formatted Range");

    // Undo the three previous actions.
    object numTimes3 = 3;
    this.Undo(ref numTimes3);

    rng.Select();
    MessageBox.Show("Undo 3 actions");
}
```

```
// Apply the Normal Indent style.
object indentStyle = "Normal Indent";
rng.setStyle(ref indentStyle);

rng.Select();
MessageBox.Show("Normal Indent style applied");

// Undo a single action.
object numTimes1 = 1;
this.Undo(ref numTimes1);

rng.Select();
MessageBox.Show("Undo 1 action");
}
```

VSTO Add-in Example

To format text using a VSTO Add-in

1. The following example can be used in a VSTO Add-in. This example uses the active document. To use this code, run it from the ThisAddIn class in your project.

C#

```
private void RangeFormat()
{
    // Set the Range to the first paragraph.
    Word.Document document = this.Application.ActiveDocument;
    Word.Range rng = document.Paragraphs[1].Range;

    // Change the formatting. To change the font size for a right-to-left
    language,
    // such as Arabic or Hebrew, use the Font.SizeBi property instead of
    Font.Size.
    rng.Font.Size = 14;
    rng.Font.Name = "Arial";
    rng.ParagraphFormat.Alignment =
    Word.WdParagraphAlignment.wdAlignParagraphCenter;

    rng.Select();
    MessageBox.Show("Formatted Range");

    // Undo the three previous actions.
    object numTimes3 = 3;
    document.Undo(ref numTimes3);

    rng.Select();
    MessageBox.Show("Undo 3 actions");

    // Apply the Normal Indent style.
}
```

```
object indentStyle = "Normal Indent";
rng.setStyle(ref indentStyle);

rng.Select();
MessageBox.Show("Normal Indent style applied");

// Undo a single action.
object numTimes1 = 1;
document.Undo(ref numTimes1);

rng.Select();
MessageBox.Show("Undo 1 action");
}
```

See Also

[How to: Programmatically Define and Select Ranges in Documents](#)

[How to: Programmatically Insert Text into Word Documents](#)

[How to: Programmatically Search for and Replace Text in Documents](#)

© 2017 Microsoft