# Comparison Utility

Table of Contents

[Comparison Utility 1](#__RefHeading___Toc105_2417130723)

[Working with Text 3](#__RefHeading___Toc107_2417130723)

[Block-level vs. inline text objects 3](#__RefHeading___Toc109_2417130723)

[Paragraph properties 4](#__RefHeading___Toc111_2417130723)

[Paragraph properties2 5](#__RefHeading___Toc111_24171307231)

# Working with Text

To work effectively with text we have to check it’s api content, it’s important to first understand a little about block-level elements like paragraphs and inline-level objects like runs.

## Block-level vs. inline text objects

The paragraph is the primary block-level object in Word.

A block-level item flows the text it contains between its left and right edges, adding an additional line each time the text extends beyond its right boundary. For a paragraph, the boundaries are generally the page margins, but they can also be column boundaries if the page is laid out in columns, or cell boundaries if the paragraph occurs inside a table cell.

A table is also a block-level object.

An inline object is a portion of the content that occurs inside a block-level item. An example would be a word that appears in bold or a sentence in all-caps. The most common inline object is a run. All content within a block container is inside of an inline object. Typically, a paragraph contains one or more runs, each of which contain some part of the paragraph’s text.

The attributes of a block-level item specify its placement on the page, such items as indentation and space before and after a paragraph. The attributes of an inline item generally specify the font in which the content appears, things like typeface, font size, bold, and italic.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Month | Data 1 | Data 2 | Data 3 | Data 4 |
| JAN | 58 | 45 | 41 |  |
| FEB | 78 | 89 | 66 | 45 |
| MAR | 78 | 552 | 65 | 66 |
| APR | 12 | 265 | 665 | 44 |

## Paragraph properties

A paragraph has a variety of properties that specify its placement within its container (typically a page) and the way it divides its content into separate lines.

In general, it’s best to define a paragraph style collecting these attributes into a meaningful group and apply the appropriate style to each delted , rather than repeatedly apply those properties directly to each paragraph. This is analogous to how Cascading Style Sheets (CSS) work with HTML. All the paragraph properties described here can be set using a style as well as applied directly to a paragraph.

The formatting properties of a paragraph are accessed using the [ParagraphFormat](https://python-docx.readthedocs.io/en/latest/api/text.html" \l "docx.text.parfmt.ParagraphFormat) object available using the paragraph’s [paragraph\_format](https://python-docx.readthedocs.io/en/latest/api/text.html" \l "docx.text.paragraph.Paragraph.paragraph_format) property.

## Paragraph properties2

A paragraph has a variety of properties that specify its placement within its container (typically a page) and the way it divides its content into separate lines.

In general, it’s best to define a this is additional paragraph to test the comparison feature between the two documents./ as well as applied directly to a paragraph.

The formatting properties of a paragraph are accessed using the [ParagraphFormat](https://python-docx.readthedocs.io/en/latest/api/text.html" \l "docx.text.parfmt.ParagraphFormat) object available using the paragraph’s [paragraph\_format](https://python-docx.readthedocs.io/en/latest/api/text.html" \l "docx.text.paragraph.Paragraph.paragraph_format) property.