

PROFESSIONAL DOCUMENT WRITING IN MICROSOFT WORD

ADVANCED TOOLS FOR THESES, REPORTS AND PUBLICATIONS

(Version September 2026)

Author: Prof. dr. ir. Wim VAN PAEPEGEM

Ghent University
Faculty of Engineering and Architecture
Department of Materials, Textiles and Chemical Engineering
Technologiepark-Zwijnaarde 46
9052 Zwijnaarde
Belgium
Tel. : +32-(0)9/331.04.32
E-mail: Wim.VanPaepegem@UGent.be

Preface

Microsoft Word is frequently criticized for its unstable formatting of large documents, the clunky typesetting of mathematical equations and the poor reference management.

Unfortunately, these are exactly the features that we need to write dissertations (e.g. MSc and PhD theses), extensive reports and collaborative project proposals.

One could argue that LaTeX is superior in all these aspects, but LaTeX is not used at all in industry and is almost never used for collaborative editing of documents, even in academia.

So despite these criticisms, Microsoft Word is used by more than a billion people worldwide and it is the industry standard. In recent years, Word is used even more as the standard choice for collaborative editing of documents in a Teams or Sharepoint environment.

Therefore, instead of only criticizing the shortcomings of Word, it seems wise to learn how these shortcomings can be resolved, by using more advanced features of Word. You can benefit from these learnings during your whole professional life, and it can save you a lot of hours of frustration, because your efficiency in writing large or complex documents can be increased a lot.

I believe it is possible to solve almost all shortcomings of Word by making smart use of advanced document-engineering features such as field codes, bookmarks, cross-references, and Visual Basic macros. It is correct to argue that after so many years, Microsoft should have implemented user-friendly ways to solve these shortcomings themselves, and not force users to dive into advanced features of Word, but that's the way it is.

This tutorial was originally written for my students and my researchers, and every time I learnt a new trick myself, I added it to the document. This version is a complete upgrade for the most recent version of Microsoft Word 365 (version 2026), running on a Windows 11 operating system with a university campus license.

I am very much open to any suggestions/criticisms/comments on this tutorial. If you have any tips or tricks yourself, or you have a beautiful style template or very useful macros that you have programmed, do not hesitate to share your input by e-mailing to wim.vanpaepegem@ugent.be. I will include those in future updates of the document.

This document is written without any commercial purpose, and can be freely shared.

Prof. Wim VAN PAEPEGEM

wim.vanpaepegem@ugent.be

Ghent, Belgium

July 2026

Table of contents

1. INTRODUCTION.....	1
1.1. SCOPE OF THIS TUTORIAL	1
1.2. PREPARING YOUR WORD VERSION FOR THIS TUTORIAL	1
2. USING PARAGRAPH STYLES.....	4
2.1. PARAGRAPH STYLE FOR CENTERED EQUATION.....	13
2.2. OUTLINE LEVELS OF PARAGRAPH STYLES.....	18
2.3. PARAGRAPH STYLE FOR LEFT-BORDERED PARAGRAPH.....	22
2.4. PARAGRAPH STYLE WITH BORDERS AND SHADING	25
3. USING FIELD CODES AND BOOKMARKS.....	28
3.1. FIELD CODES	28
3.2. THE SEQ FIELD CODE	36
3.3. THE STYLEREF FIELD CODE	39
3.4. COMBINING STYLEREF AND SEQ FIELD CODES	39
3.5. BOOKMARKS.....	43
3.6. CALCULATING WITH FIELD CODES.....	48
4. USING CAPTIONS AND CROSS-REFERENCING FOR FIGURES AND TABLES	52
4.1. NUMBERING FIGURES AND TABLES.....	52
4.2. NUMBERING FIGURES AND TABLES PER CHAPTER	56
4.3. EQUATION NUMBERING WITH CAPTIONS.....	60
5. INSERTING AND FORMATTING FIGURES.....	63
5.1. INSERTING FIGURES	63
5.2. FORMATTING FIGURES	69
6. USING PAGE BREAKS AND SECTION BREAKS	72
7. USING HEADERS, FOOTERS AND PAGE NUMBERING	78
8. USING NUMBERED CHAPTERS AND SECTIONS	87
8.1. DEFINING MULTI-LEVEL LISTS	88
8.2. DEFINING MULTI-LEVEL LISTS WITH CUSTOM PARAGRAPH STYLES	106
8.3. CROSS-REFERENCING TO NUMBERED SECTIONS	109
9. AUTOMATICALLY GENERATING A TABLE OF CONTENTS	112
9.1. TABLE OF CONTENTS WITH BUILT-IN HEADING STYLES.....	112
9.2. TABLE OF CONTENTS WITH CUSTOM HEADING STYLES.....	118
9.3. ADDING FRONT MATTER TO THE TABLE OF CONTENTS	120
10. LATEX LIKE CHAPTER TITLES.....	124
11. USING MACROS.....	128
11.1. MACRO FOR INSERTING A CENTERED AND NUMBERED EQUATION.....	128
11.2. MACRO FOR INSERTING A CENTERED AND NUMBERED EQUATION PER CHAPTER	141
11.3. MACRO FOR INSERTING NUMBERED EXAMPLES PER CHAPTER	144
12. INTEGRATING USER INPUT IN MACROS	146
12.1. EXTENSION OF MACRO FOR EQUATION NUMBERING WITH BOOKMARKING	146
12.2. REAL-TIME CHECKING OF USER INPUT IN A DIALOGUE BOX	150
12.3. FIND & REPLACE IN FIELD CODES	157
13. FORMATTING EQUATIONS.....	163
13.1. ALIGNING EQUATIONS.....	164

13.2.	USING MATH COMMANDS.....	168
14.	MANAGING REFERENCES AND CITATIONS	172
14.1.	IMPORTANCE OF MANAGING REFERENCES	172
14.2.	REFERENCE MANAGEMENT SOFTWARE	174
14.3.	MANAGING REFERENCES AND CITATIONS WITH ONLY WORD.....	175
15.	SOLVING LAYOUT ISSUES.....	178
15.1.	ALIGNING LISTS AND PARAGRAPHS.....	178
15.2.	FORMATTING TABULAR DATA.....	183
15.3.	GROUPING OR SPLITTING WORDS	183
16.	IMPORTANCE OF WORD TEMPLATES.....	185
17.	WORD DATA STRUCTURE	187
17.1.	XML DATA STRUCTURE	187
17.2.	SELECTION PANE.....	191
17.3.	REVEAL FORMATTING	192
17.4.	SAVE AS PDF	193
18.	CONCLUSIONS	198

1

Introduction

1.1. SCOPE OF THIS TUTORIAL

This tutorial assumes that the reader knows the basics of Microsoft Word. The focus is on the use of more advanced features of Word, that can make the writing and editing of larger documents more efficient. We will show that the use of bookmarks, field codes, numbered equations, reference management, etc. are within everybody's reach.

Since this tutorial was originally written for my engineering students and researchers, there may be a certain bias towards tools that are needed for our engineering and scientific field, such as typesetting mathematical equations, automatic numbering of equations, structuring large documents, generating a table of contents, layouting figures and equations, etc.

But many tips and tricks can equally apply to documents in industry, being it quality reports, management reports, collaborative documents, legal documents, etc.

All features in this tutorial have been tested with Word for Microsoft 365 MSO (Version 2606) on a Windows 11 64-bit Operating System, dated July 2026.

1.2. PREPARING YOUR WORD VERSION FOR THIS TUTORIAL

Before you proceed with this tutorial, make sure that you set the following options in MS Word. Click on File > Options... to open the *Options* dialogue box. Select the *Display* tab and check the options *Object anchors* and *Show all formatting marks*. Then you can see the anchors of figures and all formatting marks (paragraphs, tabs, tables, page breaks, etc.). The check boxes are indicated in Figure 1.

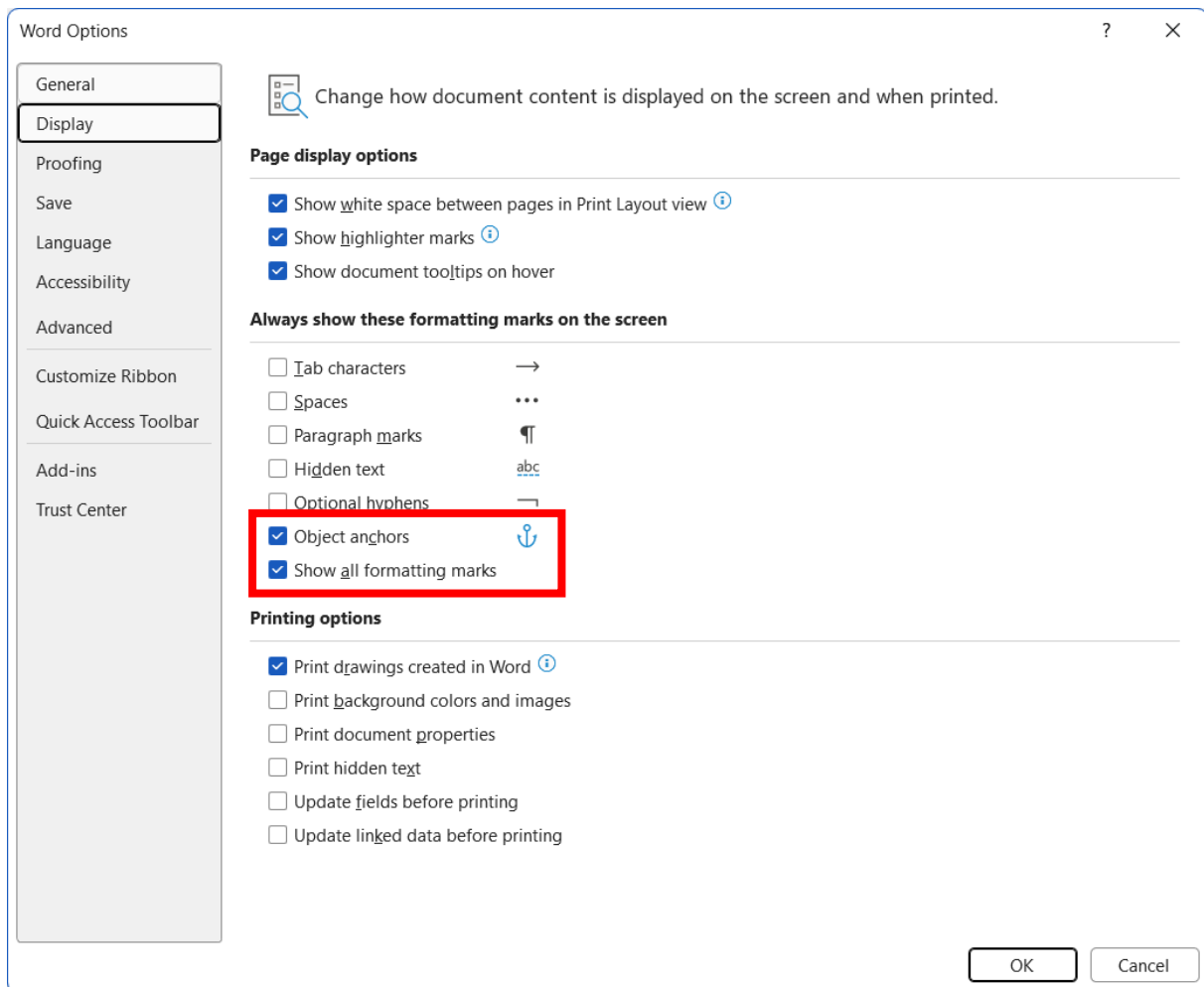


Figure 1 Settings in the *File > Options > Display* dialogue box.

Further, go to the *Advanced* tab in the left menu, scroll down to the section *Show document content* and check the option *Show bookmarks*. Also select *Always* in the drop-down list of *Field shading*, as shown in Figure 2. These options make sure that also bookmarks and field codes are clearly visible.

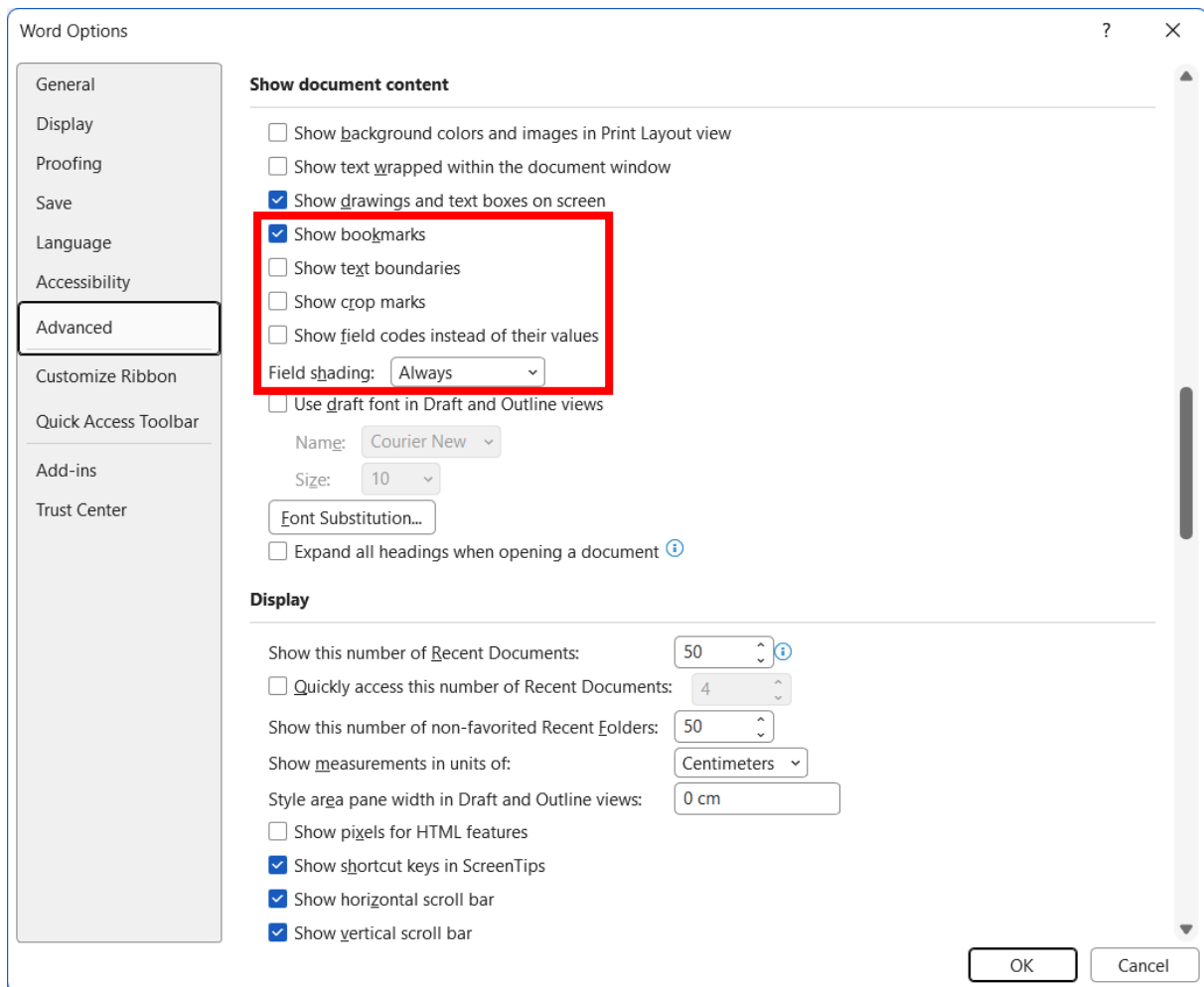


Figure 2 Settings in the *File > Options > Advanced* dialogue box.

We will introduce all these concepts in the next chapters. Therefore visualizing them, will be helpful if you implement the tips and tricks of the following chapters.

2

Using paragraph styles

A style is a set of formatting characteristics that you can apply to text, tables and lists in your document to quickly change their appearance. When you apply a style, you apply a whole group of formats in one simple task.

A lot of styles are pre-defined in MS Word. The default style in MS Word is called *Normal*. This style defines the standard font type, font size, paragraph alignment, tab positions, etc. that you use in every paragraph. Further, there are other pre-defined styles like *Title*, *Heading 1*, *Heading 2*, *Heading 3*,...

If you select a part of text and assign one of these styles by selecting the appropriate style in the drop-down list of the *Styles* toolbar, you will notice that the layout and style of this part of text is changed, as you can see in Figure 3, where different built-in heading styles are assigned to different lines of text. Notice that due to the view settings of the previous chapter, you can see individual spaces and paragraph marks. Those will not be printed, but are a visual aid to see the layout.

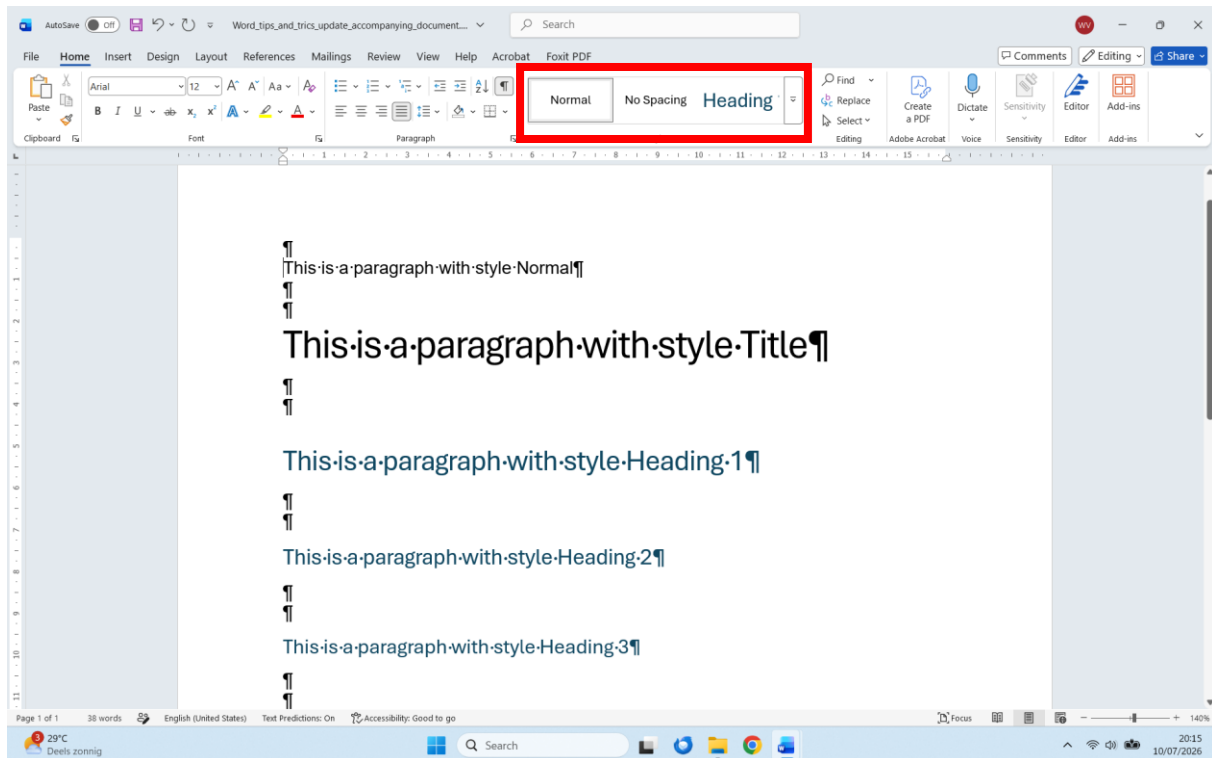


Figure 3 Applying paragraph styles in MS Word.

Why would you use such styles ? There are several reasons:

- the built-in *Heading* styles allow you to automatically number chapters, subchapters, sections, subsections, subsubsections, paragraphs, subparagraphs,... to a maximum of nine levels. If you cut and paste a certain section to another location in your document, all sections and dependent heading levels are renumbered automatically,
- if you want to change the style of a certain heading level (for example from 16 pt., Arial, Bold to 16 pt., Times New Roman, Italic), you can apply this change to all headings with the same style, only by changing the formats of that particular style,
- you can automatically generate a table of contents, based on the built-in *Heading* styles, as we will show later,
- the power of styles is not limited to paragraphs. The following are the types of styles you can create and apply:
 - a paragraph style controls all aspects of a paragraph's appearance, such as text alignment, tab stops, line spacing, and borders, and it can include character formatting,
 - a character style affects selected text within a paragraph, such as the font type and font size of text, and bold and italic formats,
 - a table style provides a consistent look to borders, shading, alignment and fonts in tables,
 - a list style applies similar alignment, numbering or bullet characters, and fonts to lists.

If you want to modify existing styles, the easiest way is to right-click on the style in the Style list in the Home Ribbon in the top menu, and then choose *Modify...* in the dropdown list, as illustrated in Figure 4.

Then you can modify all aspects of the style in the *Modify Style* dialogue box, as shown in Figure 5.

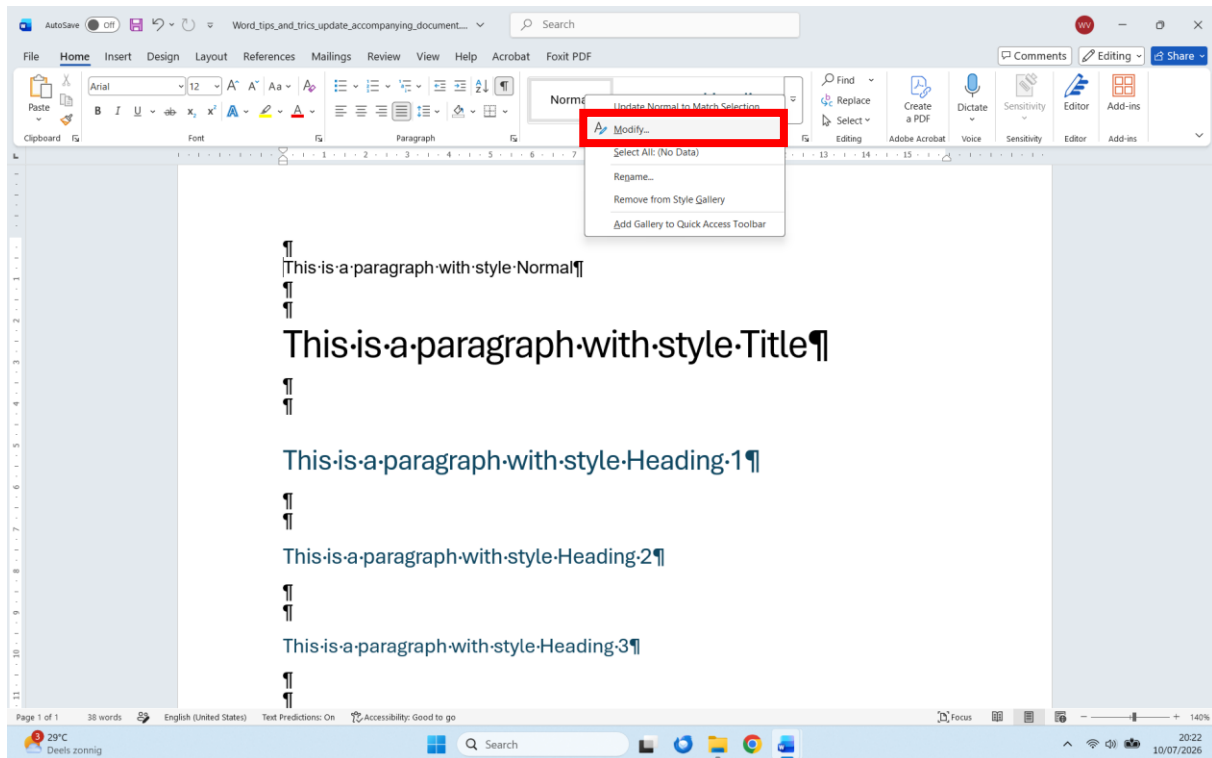


Figure 4 Modifying an existing style throughout the document by right-clicking the style.

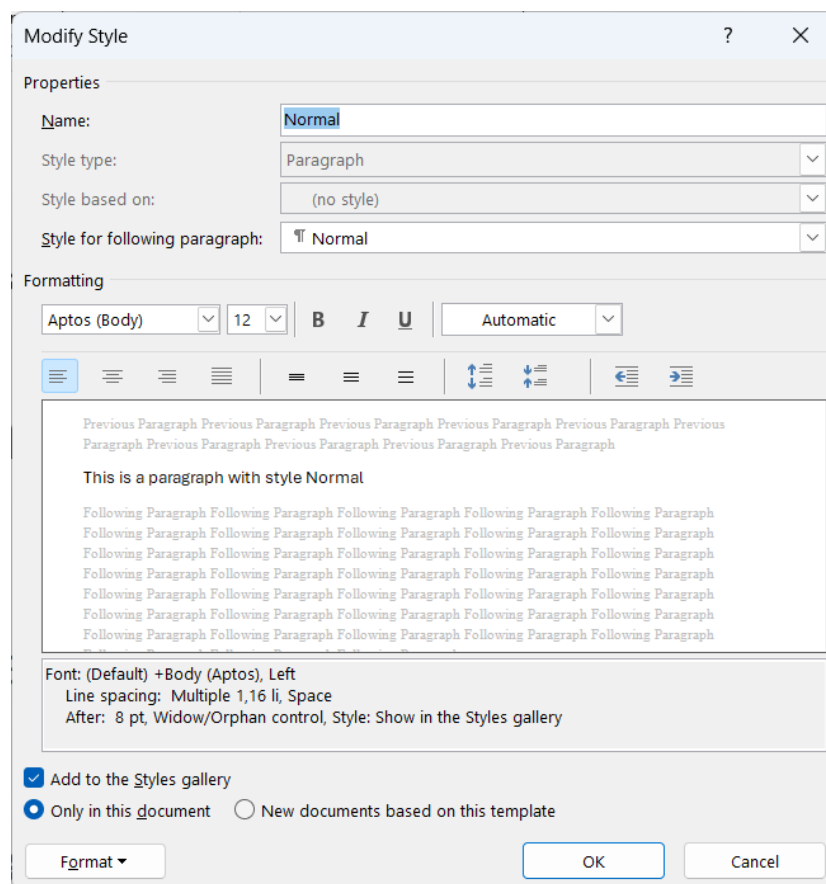


Figure 5 Dialogue box to modify an existing style.

If you want to create a new style, click on the button at the right side of the *Styles* toolbar, and the *Styles* toolbar expands. In the expanded *Styles* pane, there are 3 commands at the bottom. You select *Create a Style...*, as shown in Figure 6.

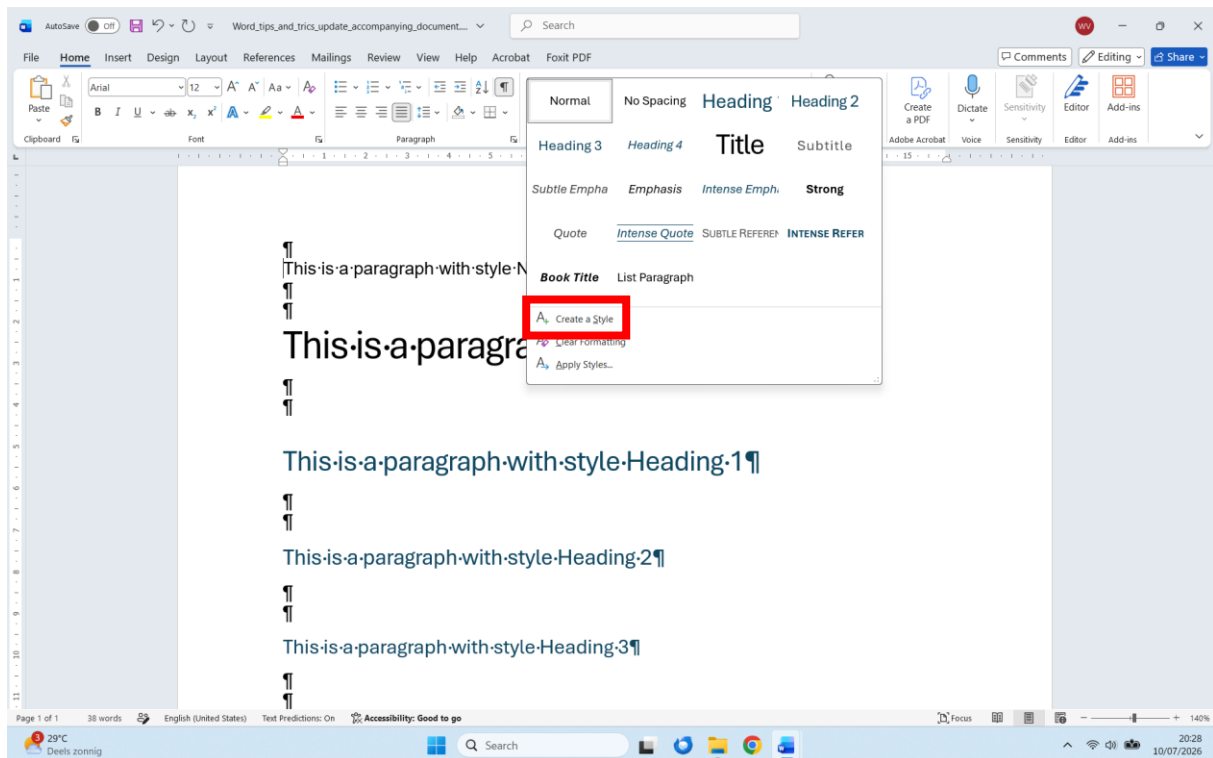


Figure 6 Open the *Styles and Formatting* task pane in MS Word.

The *Create new style from Formatting* task pane appears and you can enter a name for the new style, as shown in Figure 7.

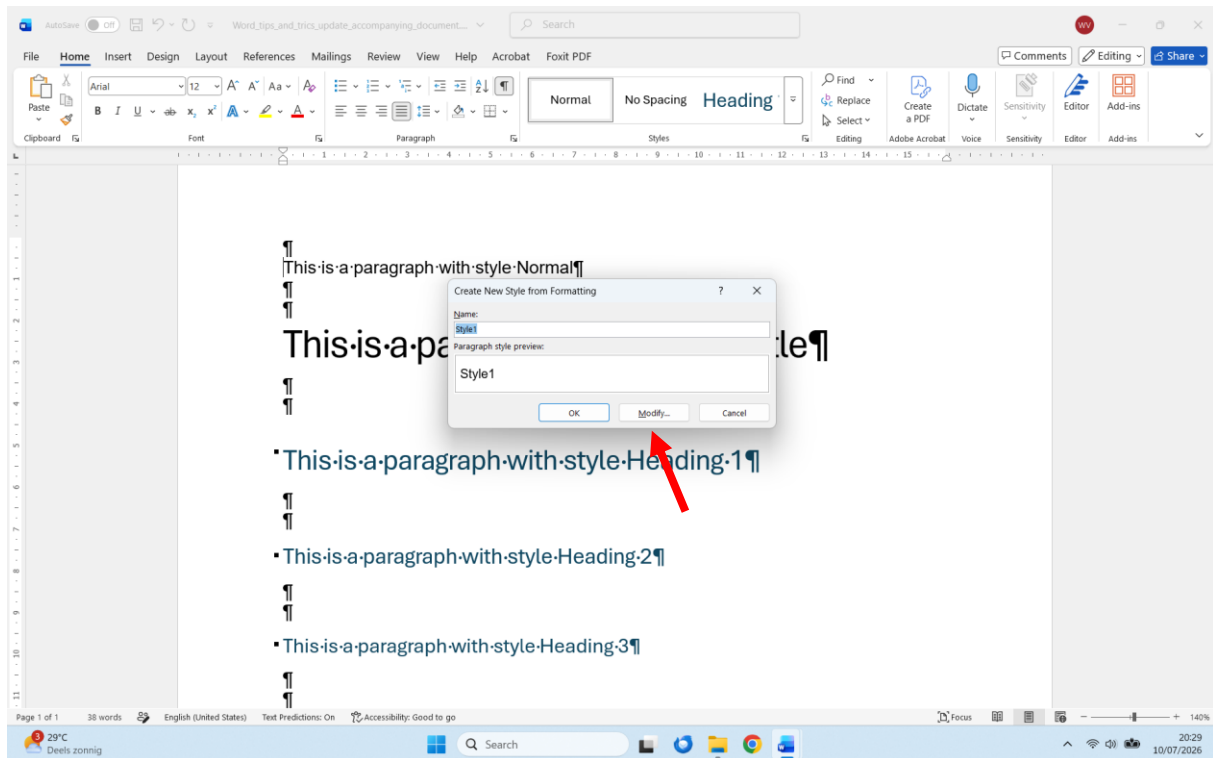


Figure 7 Creating a new paragraph style in MS Word.

If you select the option Modify... in Figure 7, the *Modify style* dialogue box in Figure 8 appears. This dialogue box describes the current group of formats that is applied for the *Style1* style.

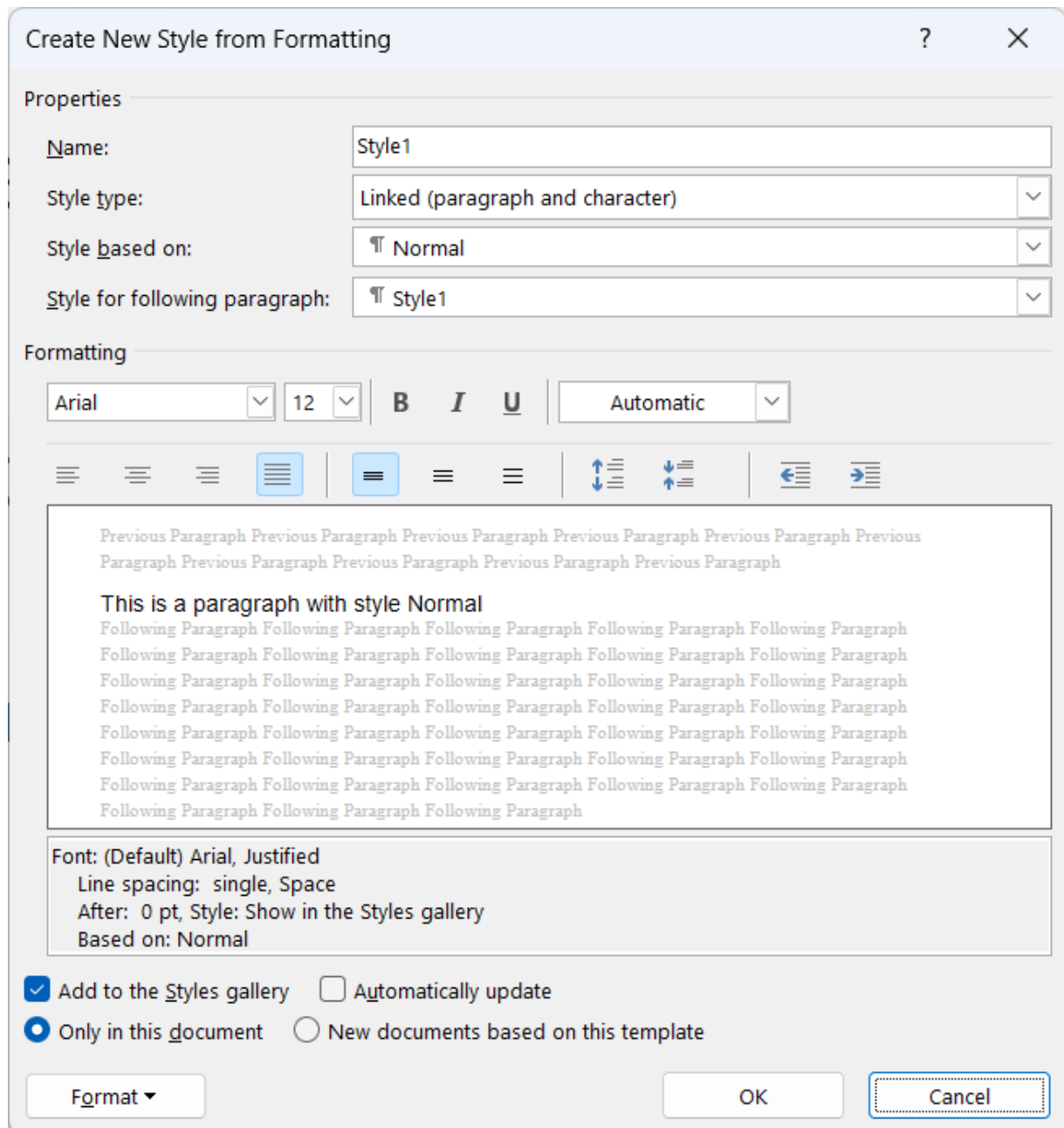


Figure 8 **Creating a new paragraph style.**

When you click the *Format* button in the bottom left corner, as illustrated in Figure 9, you can change all formatting you like:

- the font type and font size,
- the alignment of the paragraph and the white space before and after the paragraph,
- the numbering style: bullets, arabic numbers (1, 2, 3, ...), lower case Roman numerals (i, ii, iii, ...), upper case Roman numerals (I, II, III, ...), lower case alphabetical characters (a, b, c, ...), or upper case alphabetical characters (A, B, C, ...),
- the position of the tab stops,
- ...

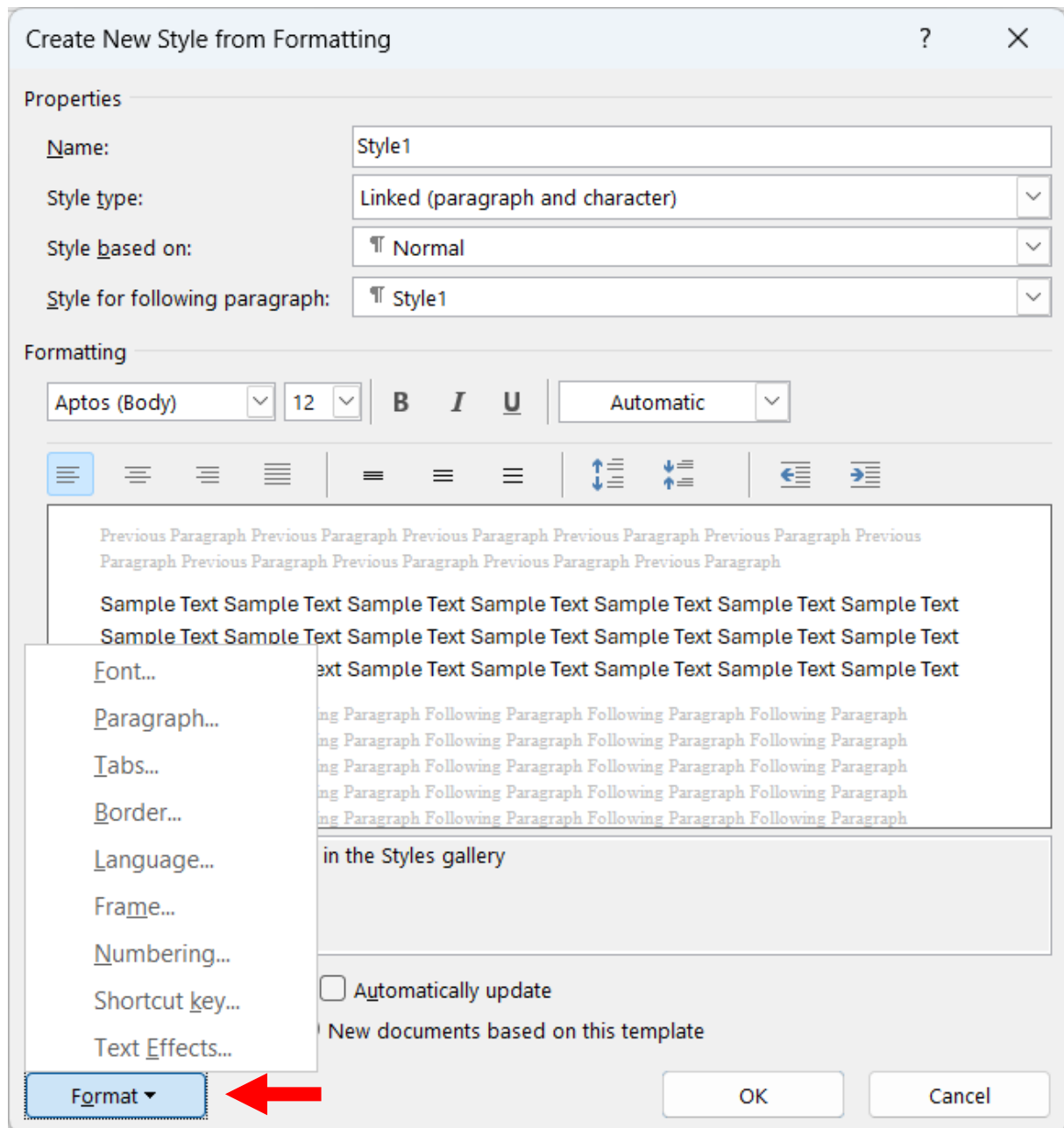


Figure 9 All formatting options for a new paragraph style.

I mentioned that styles can be defined for paragraphs, characters, tables and lists, but you noticed in the previous figures that the newly created style seems to be implicitly chosen as a paragraph style, starting from the “Normal” paragraph style.

This is correct. If you choose the option “Create a new style” from the dropdown menu in Figure 6, Word assumes a paragraph style. You can deduce that from the dialogue box that is opened then, where it mentions “*Paragraph style preview:*”, as shown in Figure 10.

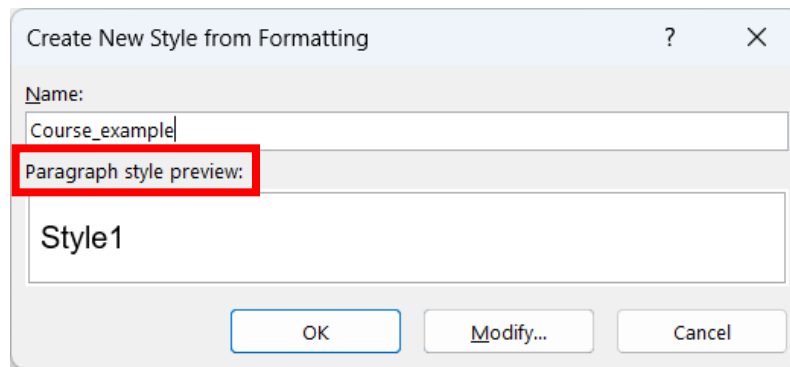


Figure 10 Shortcut to create a new paragraph style.

To be able to choose the style type explicitly yourself, you have to open the *Style Manager*. You open that dialogue box by clicking on the small arrow in the bottom right corner of the *Styles* toolbar, as indicated in Figure 11. If you hover over that small arrow, you see the popup text for the *Styles Manager*. The keyboard shortcut to open this menu, is **Ctrl+Alt+Shift+S**.

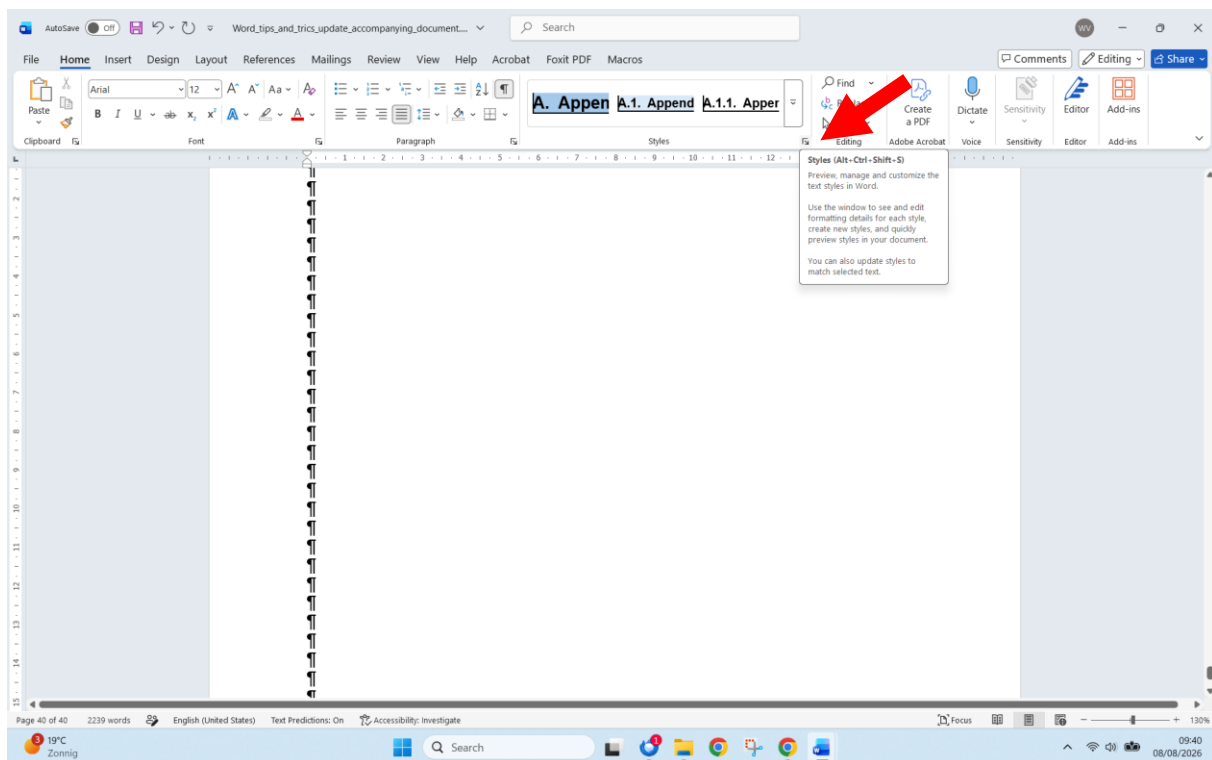


Figure 11 How to open the *Style Manager* from the *Styles* toolbar.

Then the *Styles* pane in Figure 12 opens. You can create a new style by clicking the first icon in the bottom row of the *Styles* pane, indicated by a letter A with a subscript + sign.

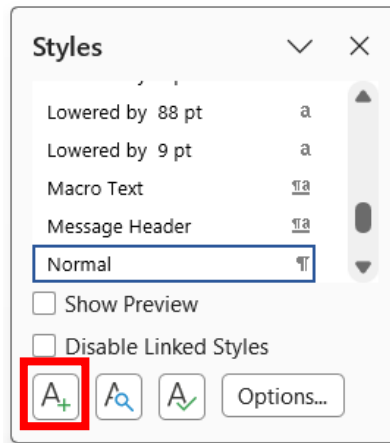


Figure 12 Creating a new style from the Styles manager.

If you click on this button to create a new style, the window in Figure 13 opens. It looks very similar to the window in Figure 8, but this time, you can change the Style type in the dropdown list next to the label “*Style type:*”, as illustrated in Figure 13. Here, you can choose the style types, as indicated in Figure 13:

- Paragraph
- Character
- Linked (paragraph and character)
- Table
- List

Word assumes the default style “Linked (paragraph and character)” when you create a new style, but even in Figure 8, you see that you could have still changed the style to another type. We will now show a few examples of paragraph styles, that can make your formatting more efficient.

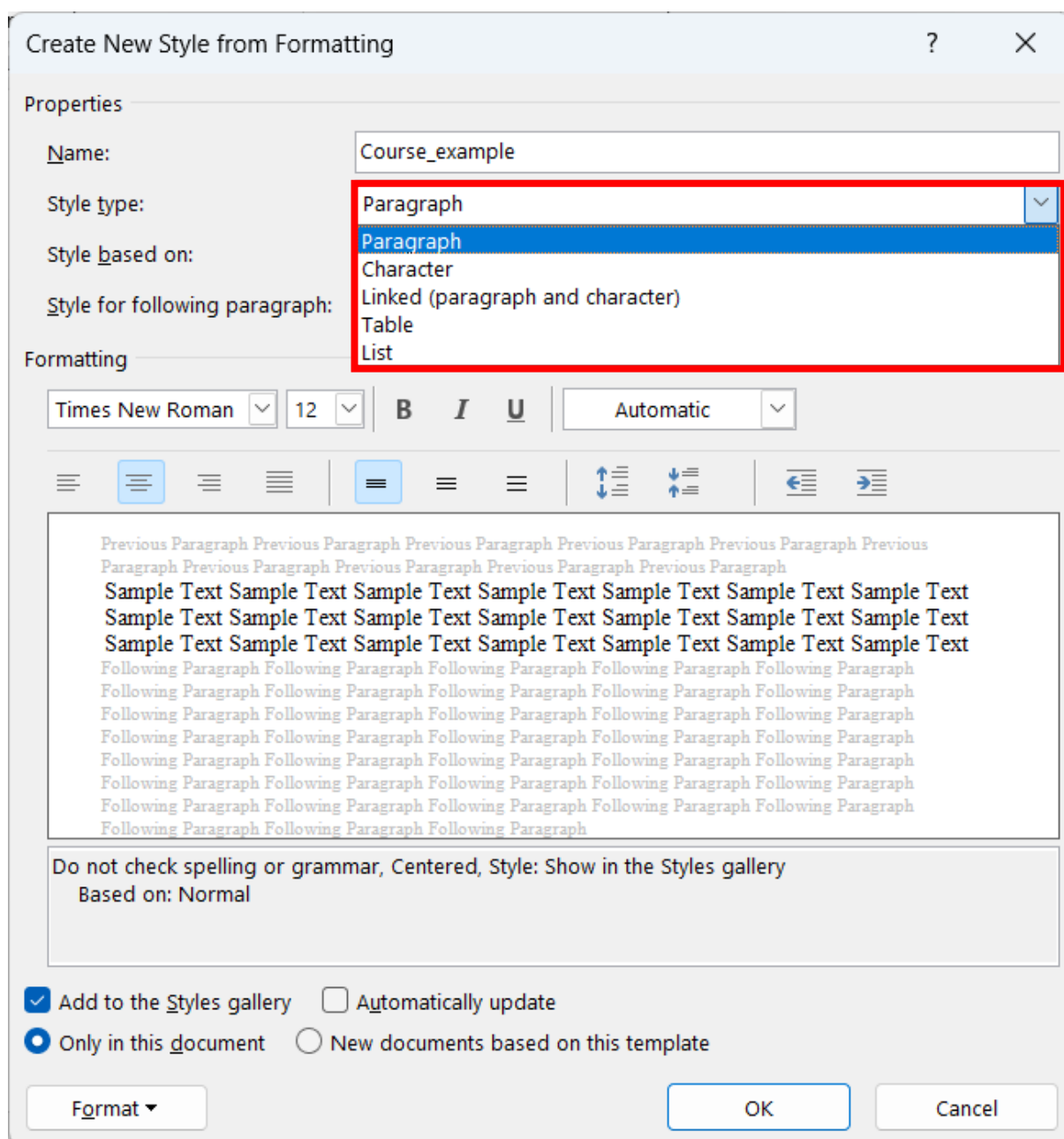


Figure 13 Different style types.

2.1. PARAGRAPH STYLE FOR CENTERED EQUATION

Now, let us for example define a paragraph style for equations. Normally, an equation is centered on the line and the equation number is right aligned. Therefore, if we use A4 paper and the left and right margins are set to 2.5 cm, we need a center tab at 8 cm and a right aligning tab at 16 cm. The paragraph style is for example called *Centered_equation*.

To define this paragraph style, you open again the *New Style* dialogue box in Figure 14. You fill in the name of the new style *Centered_equation*, and you indicate on the fourth line that the style for the next paragraph is again *Normal*. The meaning of this latter option is that, if you hit the <Enter> key to insert a new paragraph after the *Centered_equation* paragraph, this new

paragraph has not the paragraph style *Centered_equation* again, but the default paragraph style *Normal*.

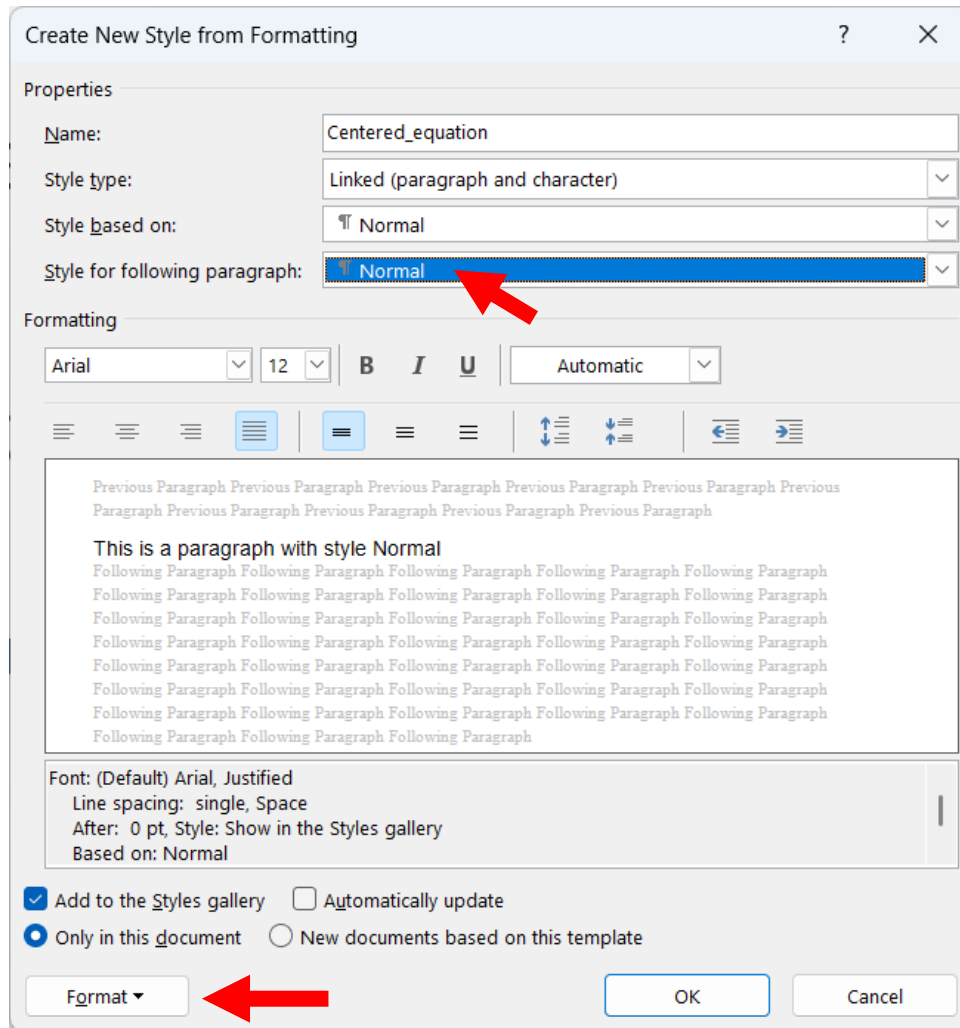


Figure 14 Create the new paragraph style *Centered_equation*.

Next, you click on the *Format* button in the left bottom corner to set the tab stop positions. The *Tabs* dialogue box in Figure 15 appears. You set a center tab at 8 cm and a right aligned tab at 16 cm. Of course, if the margins are set differently in your document, you have to change these distances.

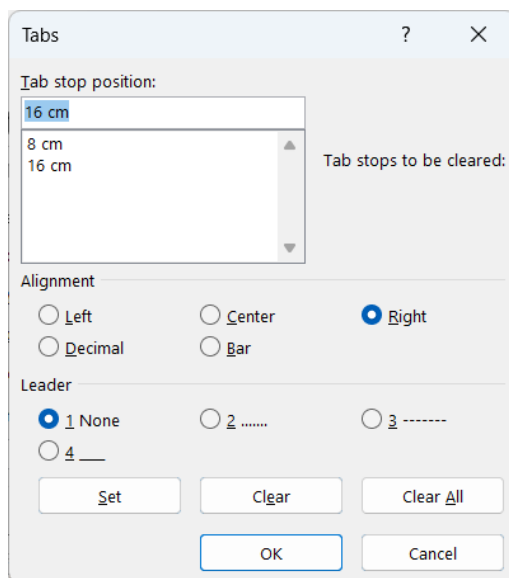


Figure 15 Setting the tab stop positions for the new paragraph style *Centered_equation*.

If you now select a paragraph and you apply the *Centered_equation* paragraph style, you will see that a center tab and right aligned tab are set for this paragraph.

If you want, you can also add white spacing before and after the *Centered_equation* paragraphs. Therefore, you modify the *Centered_equation* paragraph style, you click on the same *Format* button in the left bottom corner (see Figure 14) and you select *Paragraph...* The *Paragraph* dialogue box in Figure 16 appears. You can choose now the spacing before and after this paragraph (e.g. 12 pt).

The power of using styles is in the automatic updating. Assume that you have inserted a lot of equations, and that then the margins have to change from 2.5 cm to 3 cm. That means that your equations are no longer centered and that the equation number is outside the margins. If you change the tab stop positions for the *Centered_equation* paragraph style, the tab stops will be changed for every paragraph with this style.

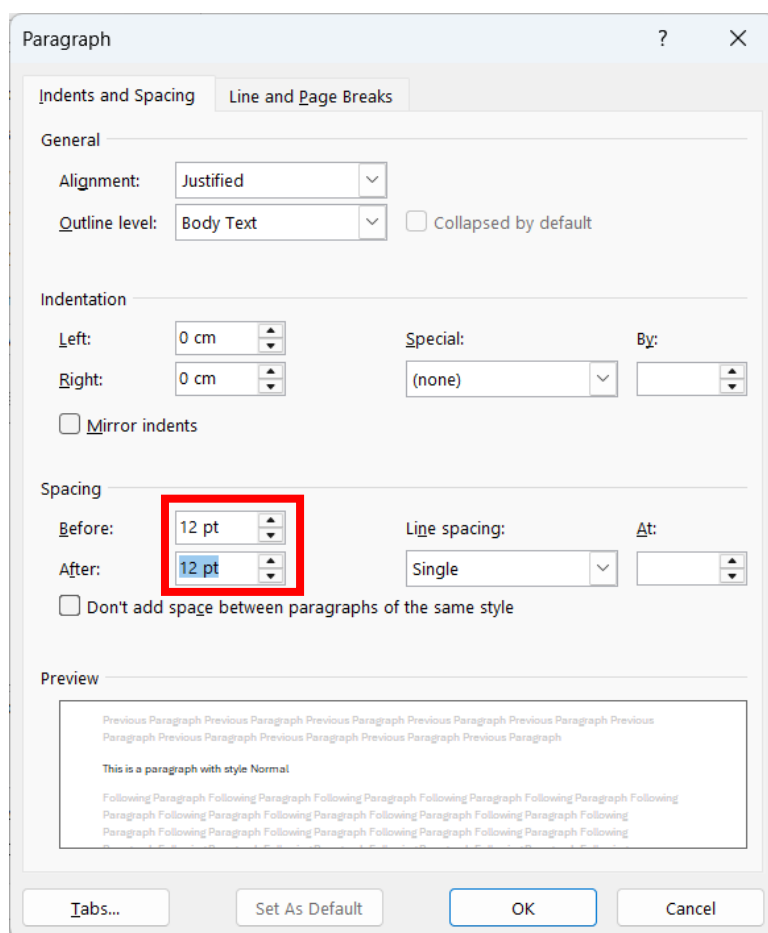


Figure 16 Setting the leading and trailing white space for the *Centered_equation* paragraph style.

User-defined paragraph styles do not necessarily show up in the *Paragraph Styles* gallery on the Home Ribbon. To enforce this, you first click on the small button in the bottom right corner of the *Paragraph Styles* gallery, to open the *Style Manager* (Figure 17).

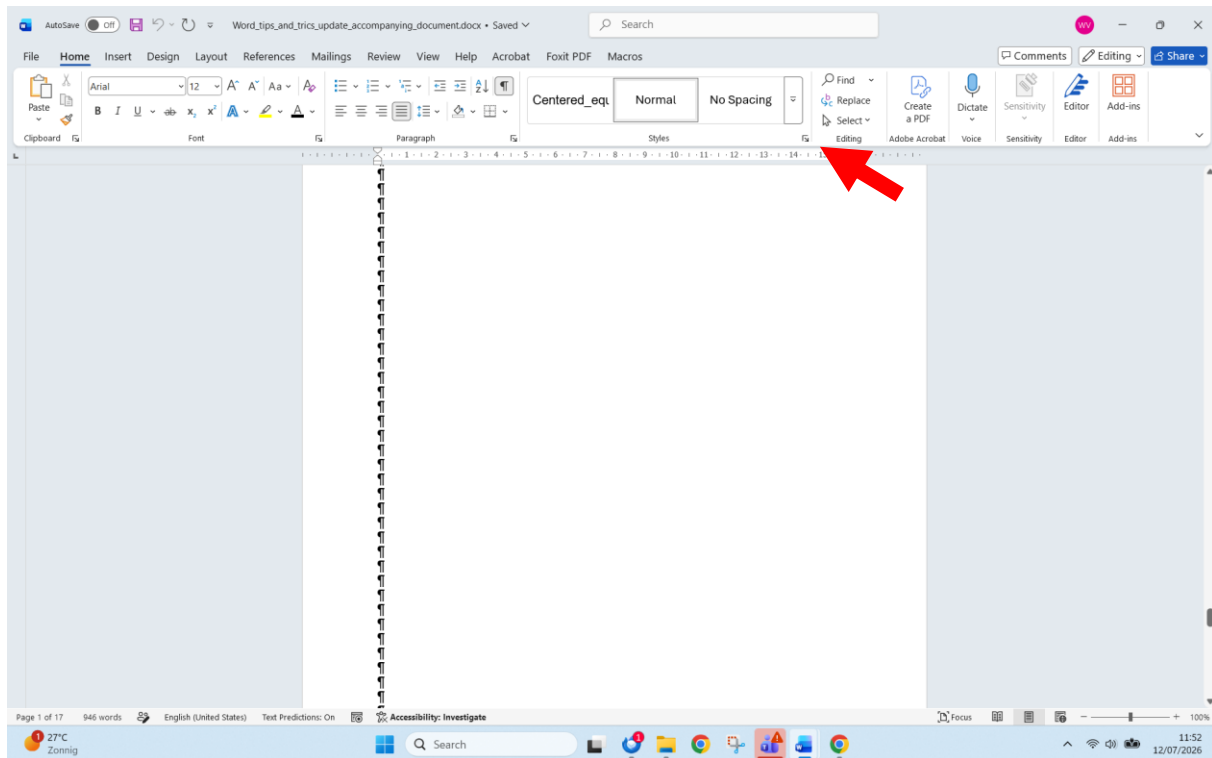


Figure 17 Opening the Style manager.

In the *Style Manager*, you see all built-in paragraph styles, and also your own style “*Centered equation*”. When you right-click on this style (Figure 18), you discover several powerful options:

- Add the paragraph style to the *Paragraph Style Gallery*,
- Select all instances that have this paragraph style adopted (that is also a very useful option sometimes, for example to copy and paste all equations from one document into another one, if all equations have the “*Centered equation*” paragraph style)
- Clear the formatting of all instances
- Delete the paragraph style

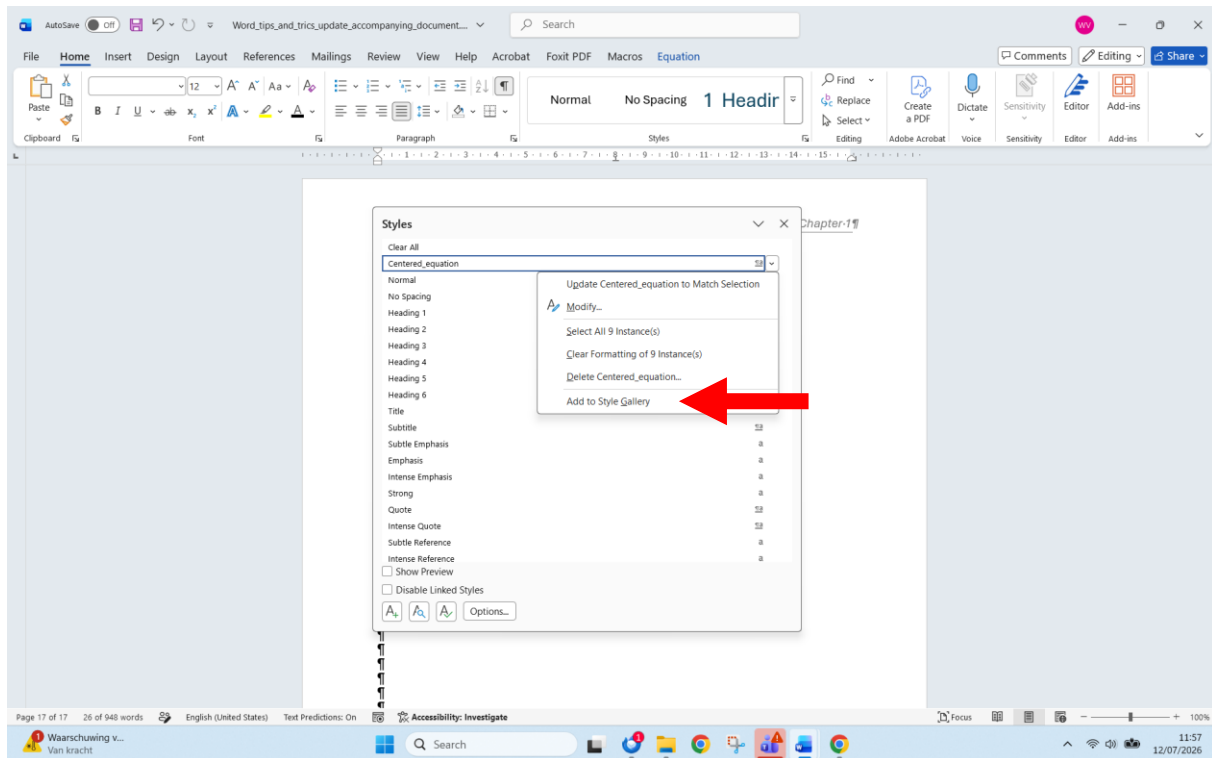


Figure 18 Adding the *Centered_equation* paragraph style to the *Style Gallery*.

2.2. OUTLINE LEVELS OF PARAGRAPH STYLES

When you create a new paragraph style, the *Format* button in the bottom left corner of the dialogue box gives access to formatting the font, paragraph, tabs, etc., as illustrated in Figure 19.

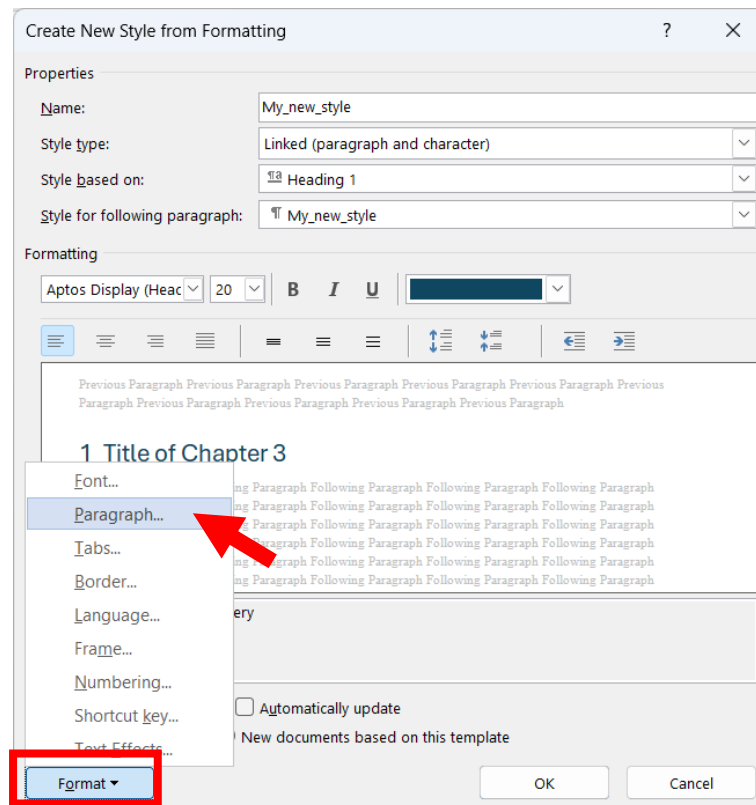


Figure 19 Selecting the “Format paragraph” pane for a new paragraph style.

If you select the formatting of the paragraph, the dialogue box in Figure 20 shows up. We already used this dialogue box in the previous example to format the leading and trailing white space of the paragraph style “*Centered equation*” (see Figure 16).

However, there is also a setting called “*Outline level*:”. Many people do not know what this is about, but the outline level defines the level of your new paragraph style in the *Outline* view.

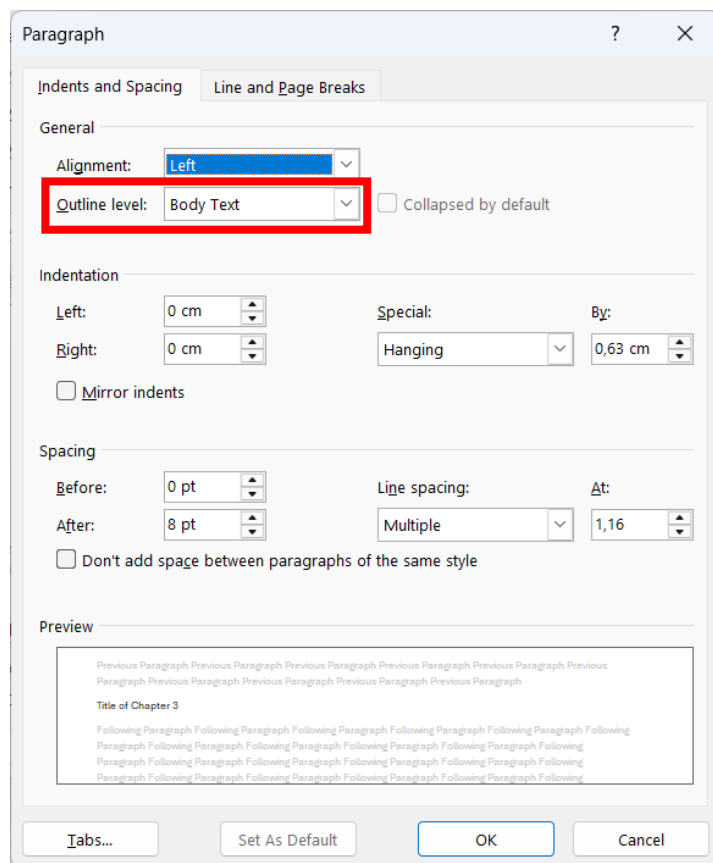


Figure 20 Definition of the outline level of the new paragraph style.

This *Outline* view can be accessed by clicking in the top menu on *View > Outline*, as shown in Figure 21. This particular view of the document shows you a collapsed outline of the document. If you look carefully, a new tab “*Outlining*” has appeared in the top menu of your document window.

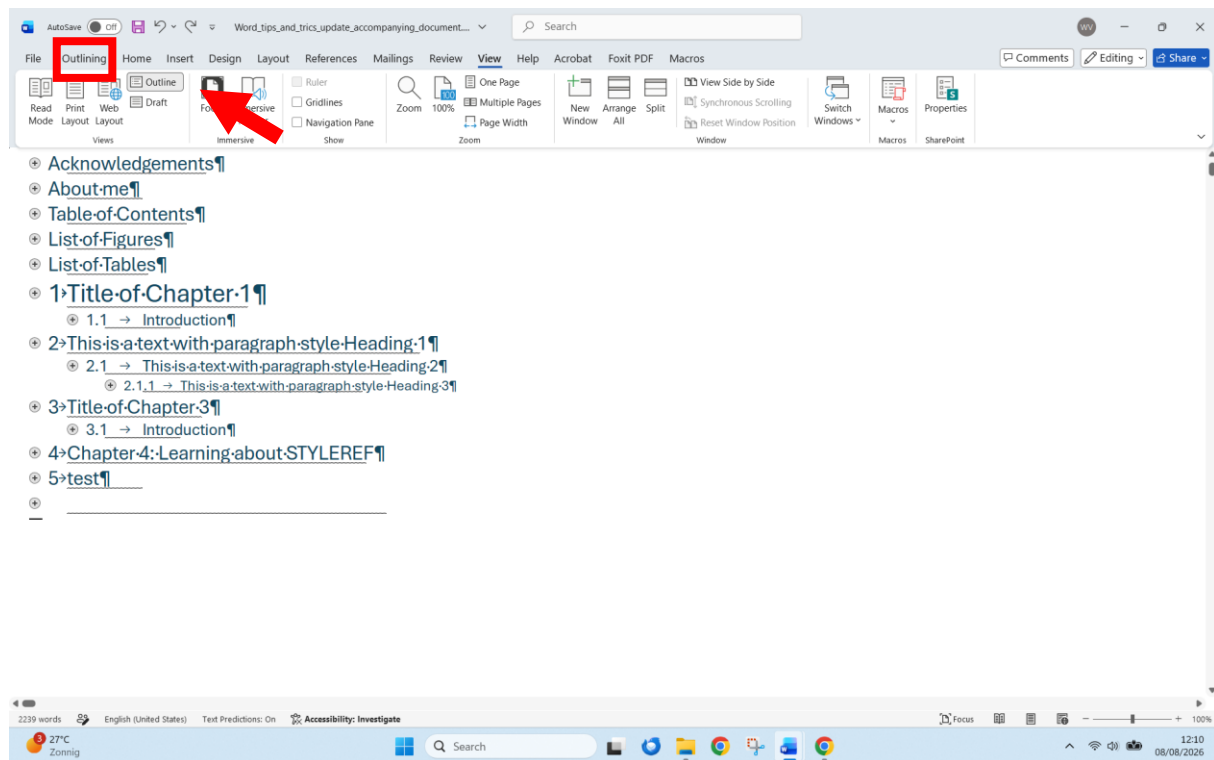


Figure 21 Switching to the *Outline* view of your document.

As shown in Figure 22, you can now define which outline levels are shown in the document. You can select for example only the Level 1 headings, or also Level 2 headings, etc.

The outline level that you have defined for your new paragraph style in Figure 20, defines to which level this paragraph style corresponds in the *Outline* view. So the built-in heading style *Heading 1* has the outline level 1, the *Heading 2* style has the outline level 2, etc. If you define your own heading styles, you can define at which level they appear in the *Outline* view.

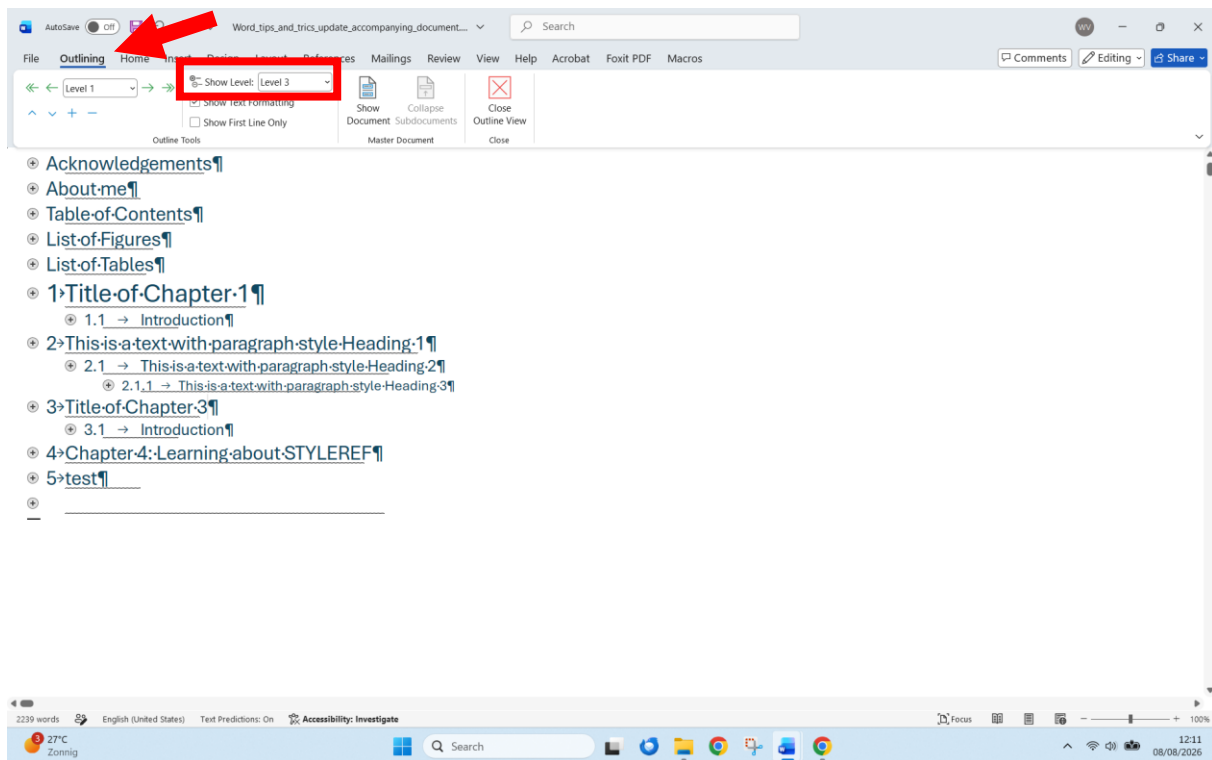


Figure 22 Selecting the outline levels to show in the Outline view.

2.3. PARAGRAPH STYLE FOR LEFT-BORDERED PARAGRAPH

In my courses, I introduce examples that illustrate the theory. They have a paragraph style with a left border. It looks like this:

Example 1.1

Text example 1

Figure example 1

I can manually insert this left border with the *Borders* style item in the *Paragraph* toolbar on the Home Ribbon, but if I want to later change for example the line thickness of this border, I would have to manually select all paragraphs and change their border properties. It is much better to define a new paragraph style.

So we open again the *New Style* dialogue box (see Figure 14) and we create a new paragraph style “*Course_example*”, as illustrated in Figure 23.

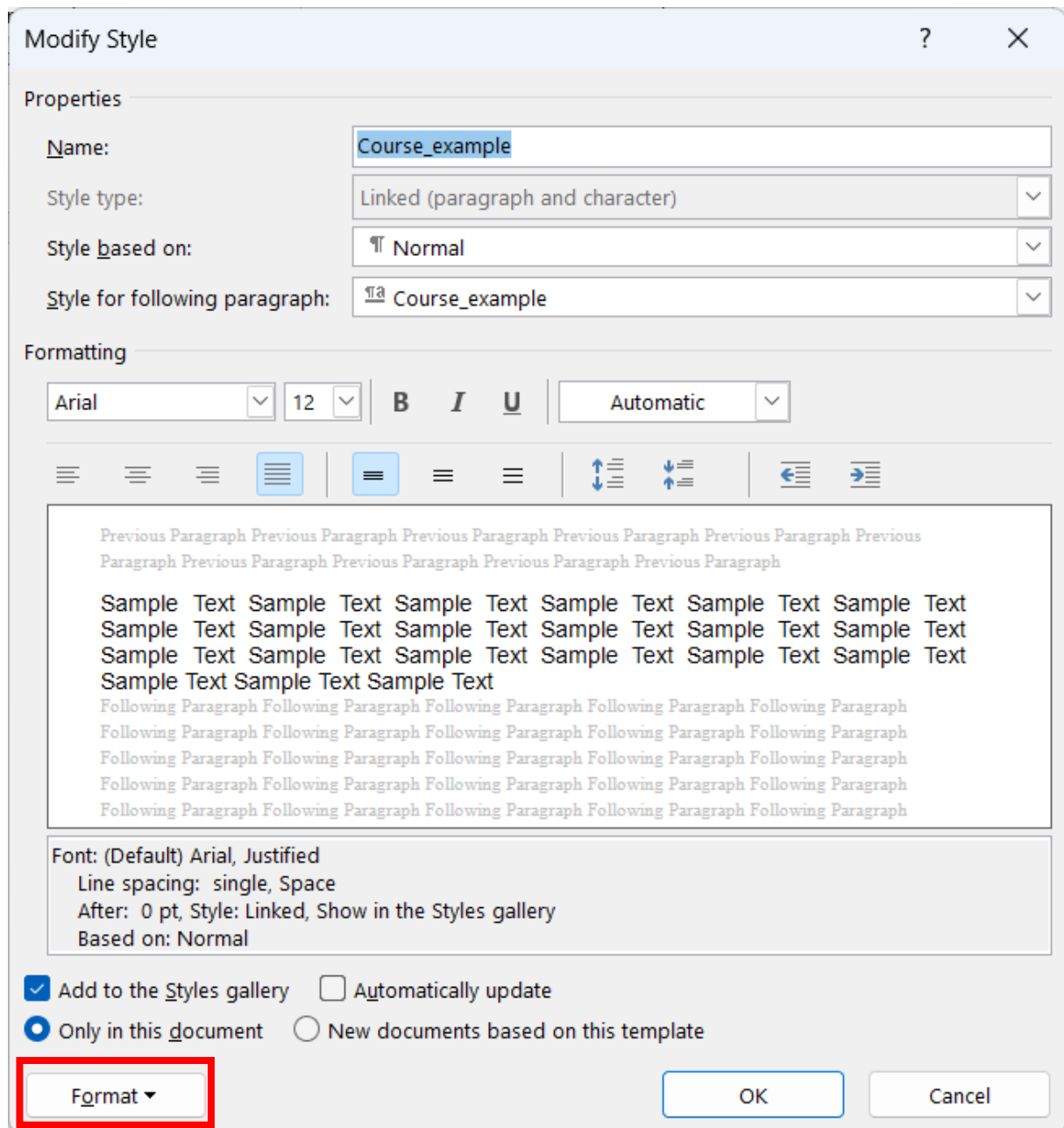


Figure 23 Create a new paragraph style “*Course_example*”.

We click on the button “Format” in the bottom left corner of the dialogue box and choose the option “*Border...*”. The dialogue box in Figure 24 opens. We change the line thickness to 1.5 points and click the left border of the schematic diagram. If you click the “*Options...*” button, you can also change the offset of the line compared to the paragraph.

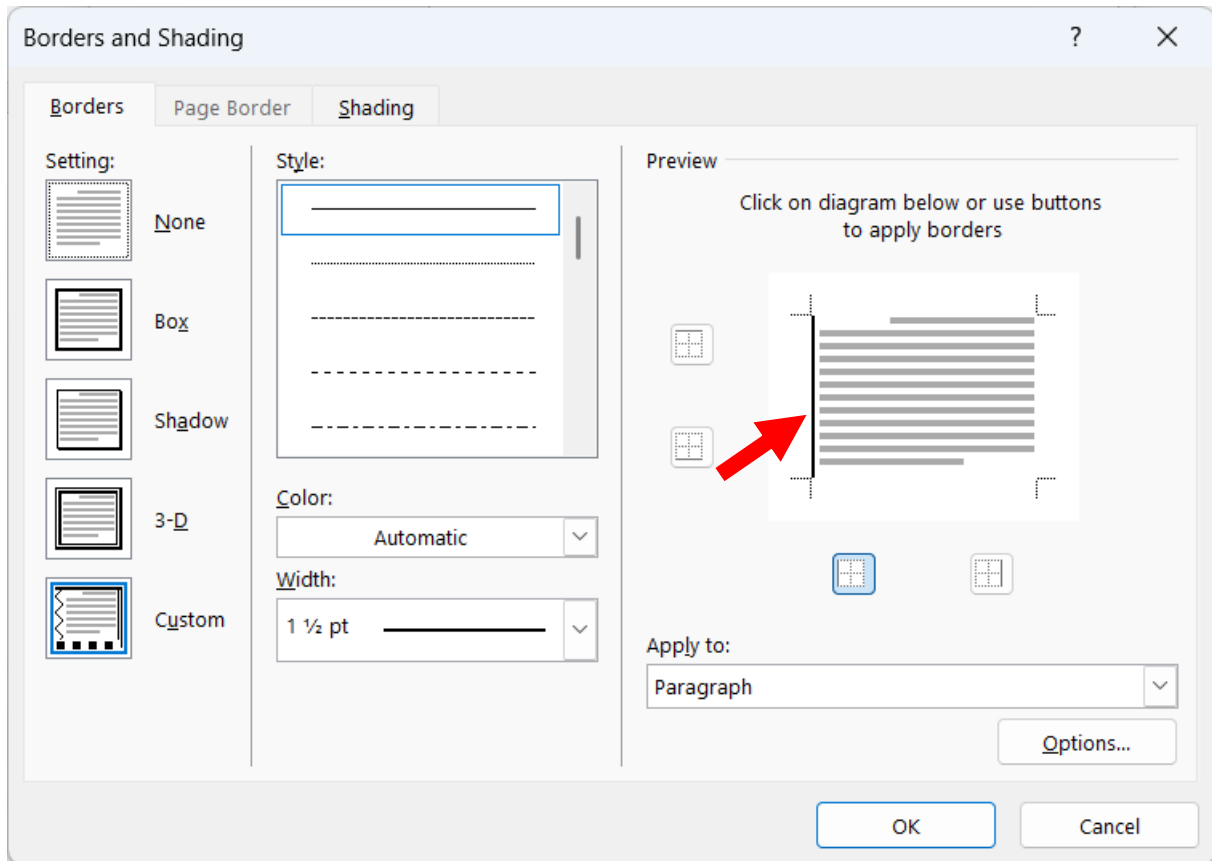


Figure 24 Defining the left border of the paragraph.

When you apply this paragraph style to any paragraph, the vertical left border appears in front of it.

With this example, I also want to illustrate something else. Suppose we already had an existing paragraph with some text with bold/italic/underlined formatting, and different font types, like the example below.

This paragraph illustrates different ***formatting marks*** and different font types.
This line is in another font type.

If I select these two paragraphs and apply the “*Course_example*” paragraph style, the result is like shown below.

This paragraph illustrates different ***formatting marks*** and different font types.
This line is in another font type.

However, suppose I have an existing paragraph like my course example above, which looks like this:

Example 1.1

Text example 1

Figure example 1

If I apply now the paragraph style “*Course_example*” to these existing paragraphs, the result looks like this:

Example 1.1

Text example 1

Figure example 1

Surprisingly, the bold/italic/underline formatting of the text “Example 1.1” has all disappeared, and turned into plain text.

At first sight, this can be a very unexpected behaviour, and it actually is. The reason for this strange behaviour is the way that Word handles styles. If formatting is applied to less than half of the paragraph, the formatting is assigned to the characters of that text, and this is called “direct formatting”. If more than half of the paragraph is formatted (in our case, the complete paragraph), the formatting is “attached” to the paragraph style.

When applying the new paragraph style “*Course_example*”, it overrides the old paragraph style and hence removes the formatting.

For this particular case, it would be better to assign the border manually, or as we will see later, write a macro to automate it.

2.4. PARAGRAPH STYLE WITH BORDERS AND SHADING

The last example combines the use of borders and shading to give some more visual emphasis to titles and headings.

We create three paragraph styles for the three heading levels of appendices. The paragraph styles are called Appendix_L1 (level 1), Appendix_L2 and Appendix_L3. They look like the examples below.

Appendix Level 1

Appendix Level 2

Appendix Level 3

Later in this tutorial, we will use those paragraph styles again for creating a table of contents, including appendices.

So we create three new paragraph styles, and just like in Figure 23, we click again the button “*Format*” in the bottom left corner of the dialogue box and choose the option “*Border...*”. The dialogue box in Figure 25 opens again. This time we define a light blue shading, combined with a border line at the left and bottom of the paragraph.

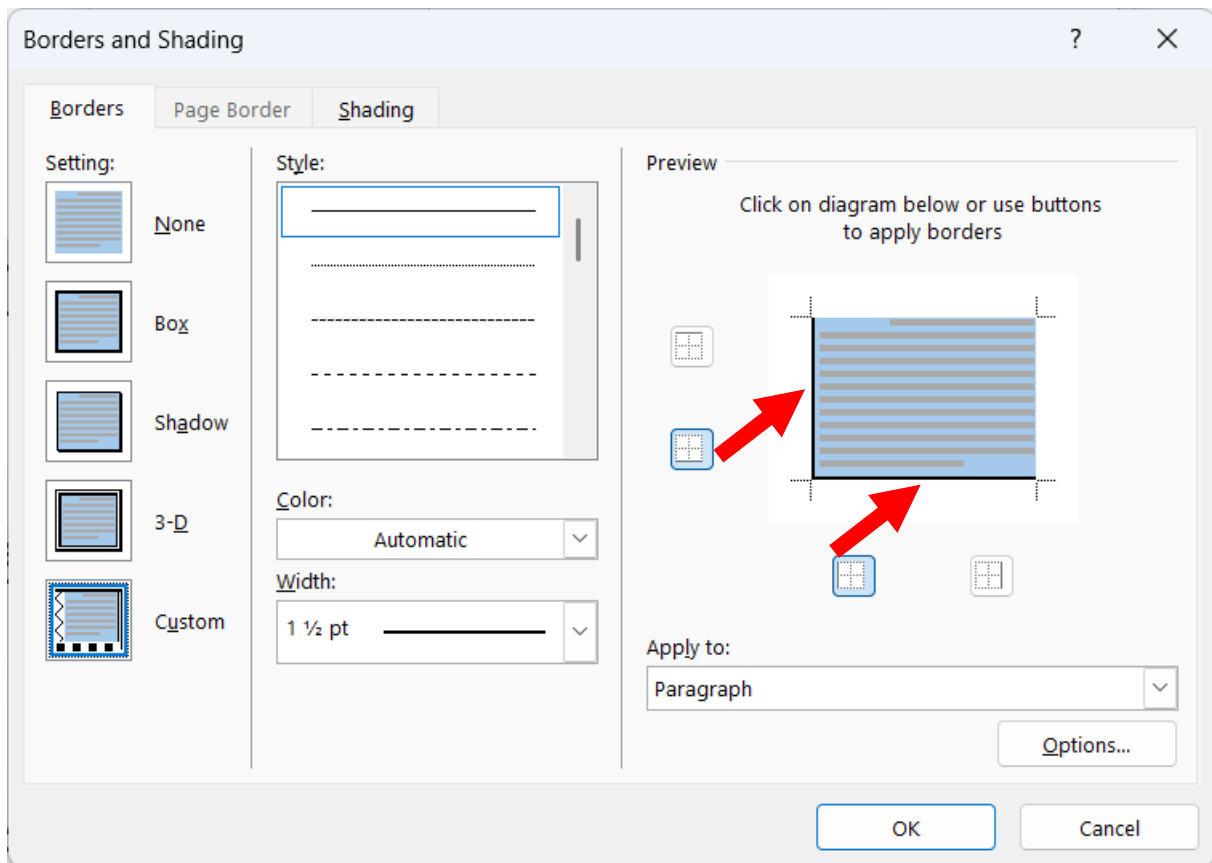


Figure 25 Assigning shading and left and bottom border to a paragraph style.

In a similar way, we create the paragraph styles Appendix_L2 and Appendix_L3. The result looks like Figure 26.

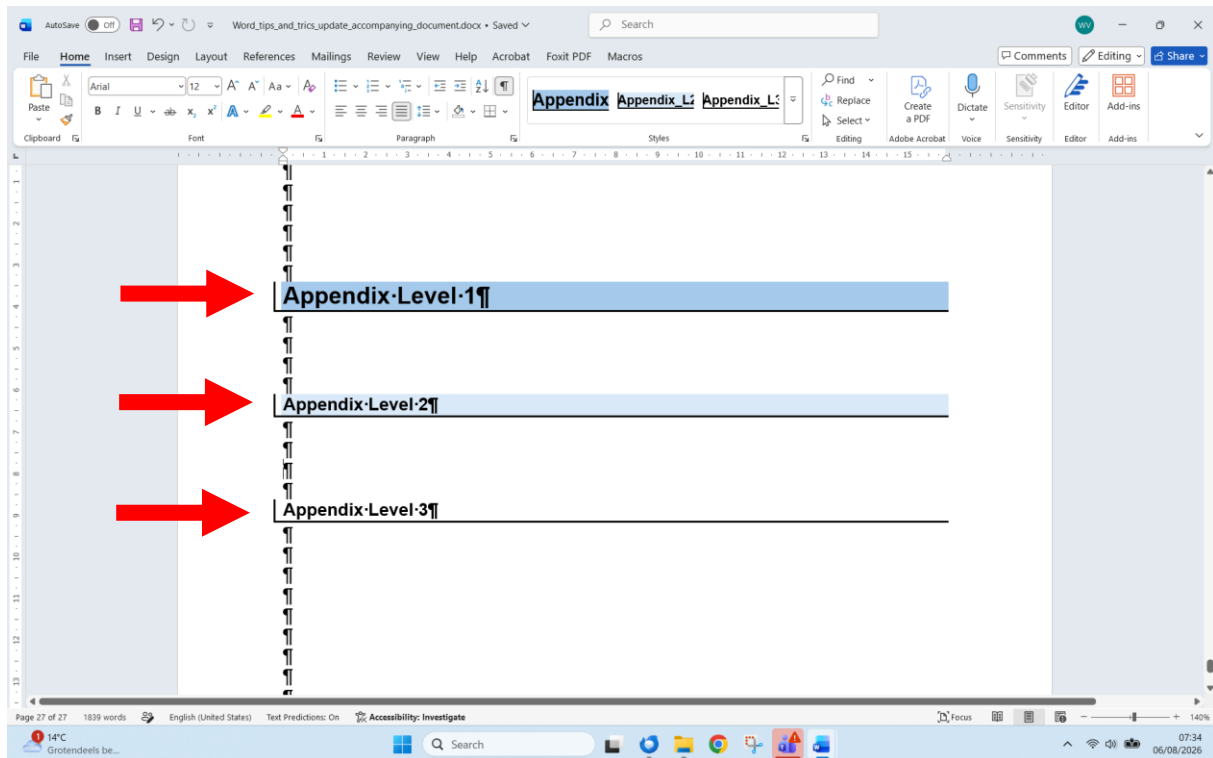


Figure 26 Result of the three paragraph styles for Appendices.

Now we move to another fundamental tool of MS Word : field codes and bookmarks.

3

Using field codes and bookmarks

3.1. FIELD CODES

Most tutorials about Microsoft Word do not even mention what field codes are. However, field codes are widely used in every Word document, but normally you will not be aware that a field code exists. Field codes are used in a document to generate information which needs to be changeable. Whenever you add page numbering to a document, a page field code is inserted into the header or footer. You might also add a date field code, either in a header/footer (to indicate when the document was last printed) or to a letter template (to save you typing in today's date each time you want to send out a letter). Field codes are also used to create an index or table of contents and in mail merge, to pick up information from a data source. You can also use field codes to insert document information (such as the author or file name), to perform calculations, to create links and references to other documents or items, and to perform many other special tasks.

As we have turned on the option *Always* for *Field shading* in the *Options* dialogue box (see Figure 2), field codes will be shown in every document with a grey background color. This background color will never be printed on paper, but is just a mark to easily recognize whether a certain part of your text is a field code or not.

By default, field codes are never shaded in Word documents, that is also why field codes go unnoticed. They appear as regular text, but are in fact a changeable content.

Let us start with an easy field code: inserting the current date and time in your document. Click *Insert > Date and Time...* to open the *Date and Time* dialogue box. The dialogue box in Figure 27 appears. You can choose the date and time format you like and then press the OK button.

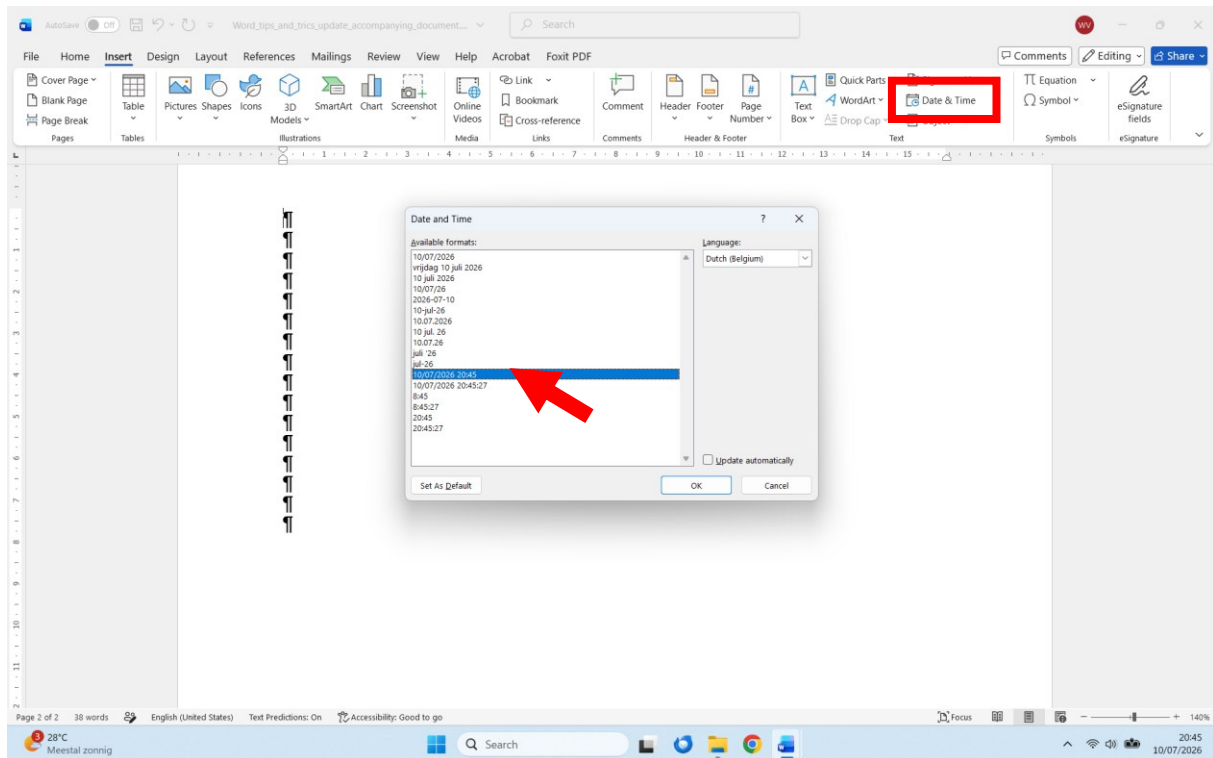


Figure 27 *Date and Time* dialogue box.

The current date is inserted in your document, as shown in Figure 28.

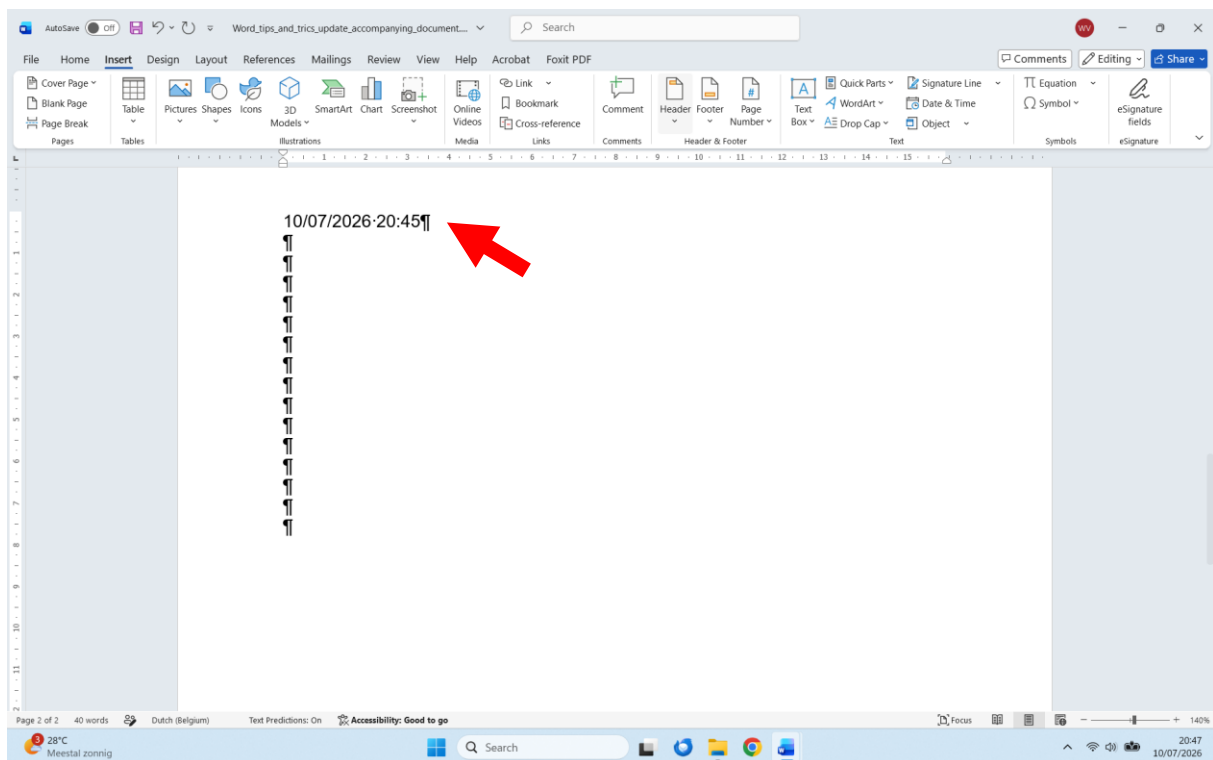


Figure 28 *Date & Time* inserted in the document (notice that the text is not grey shaded).

We indicated before that field codes should always be shaded with a grey background color, so this is not a field code, but plain text. Indeed, it was mentioned before that field codes are meant for information that can be updated. So when you return to the *Date and Time* dialogue box, you will notice that there is a checkbox “*Update automatically*” in the bottom right corner (see Figure 27). If you tick that box before inserting the date and time, the new result now looks like Figure 29.

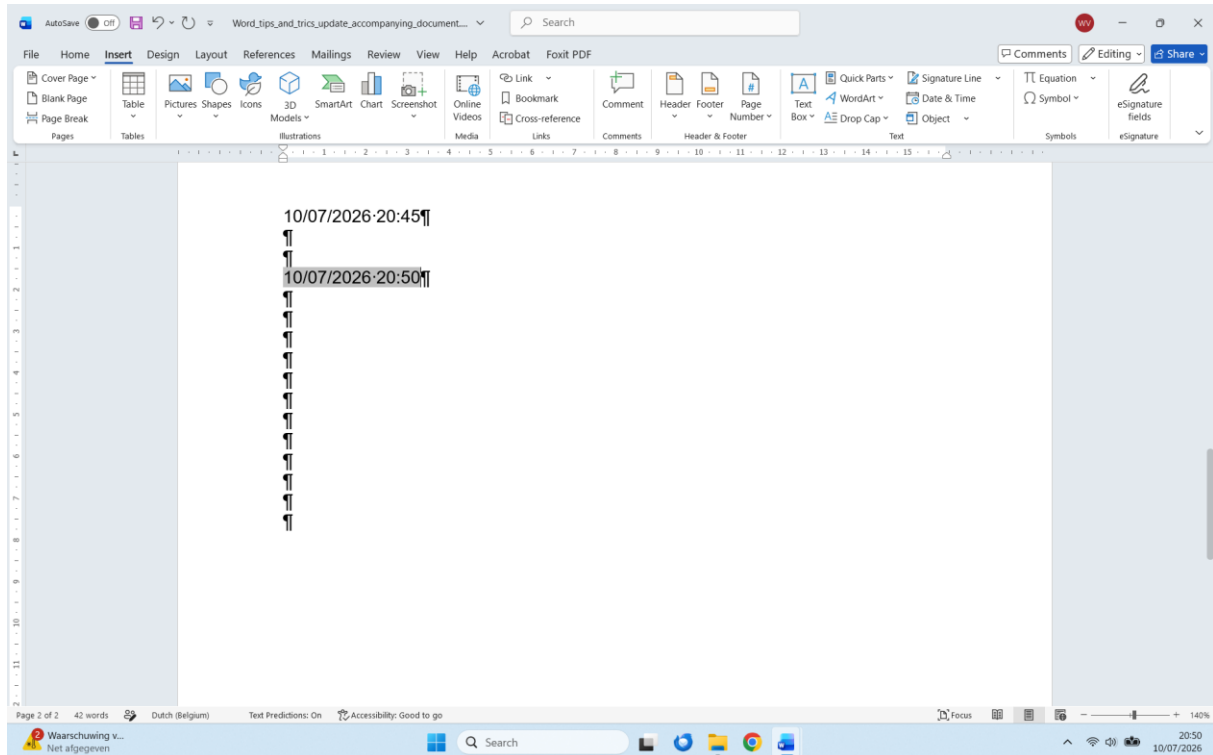


Figure 29 Current date & time inserted as a field code in the document.

The text that you see – 10/07/2026 20:50 – is only the result of the field code. The actual field code itself can be seen by pressing the keyboard shortcut <Alt>+<F9>. The field code in Figure 30 is shown.

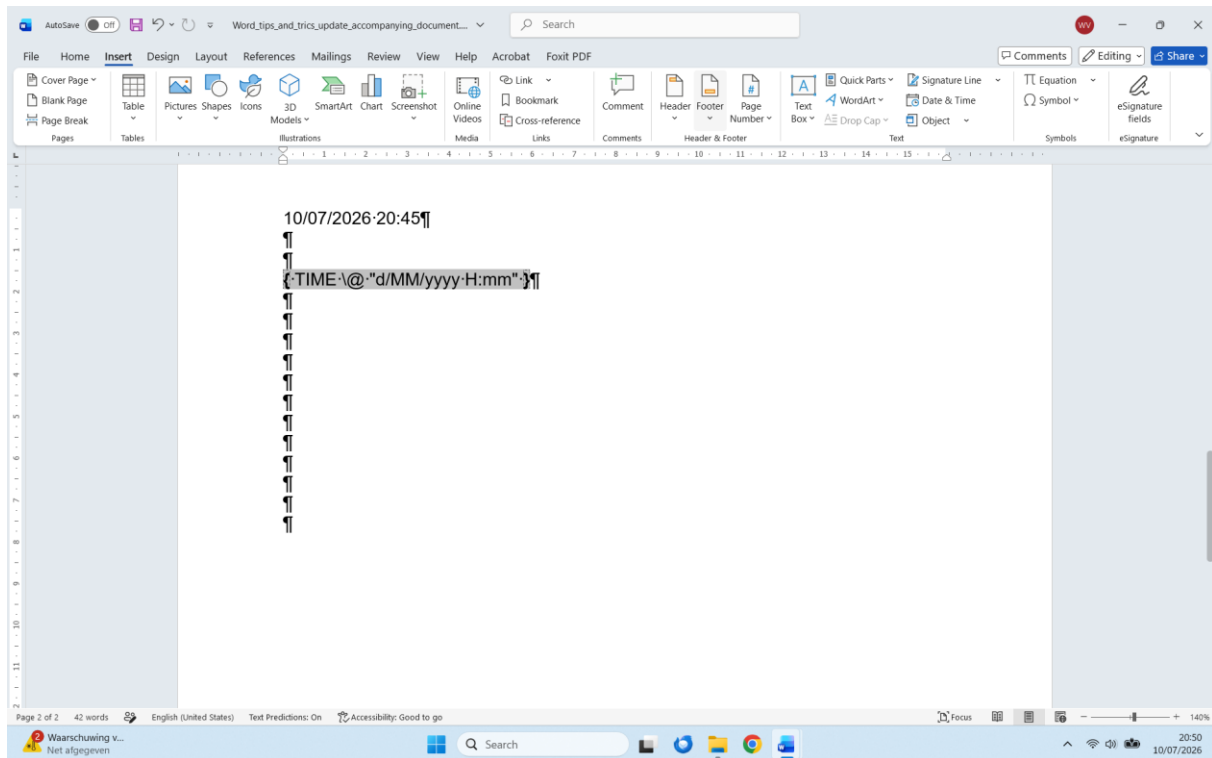


Figure 30 Visualizing the Date & Time field code by pressing <Alt>+<F9>.

The special field code TIME is displayed enclosed by curly brackets {}, known as braces. Most field codes also have switches or other options which extend their capabilities. In this case the switch \@ “d/M/yyyy” specifies the date and time format.

In the Help information, you can always find a detailed description of the field codes and the use of their switches (search for the keywords “field code TIME” in the Help screen, see Figure 31).

The result is shown in Figure 32. The switches are documented in detail in the help information.

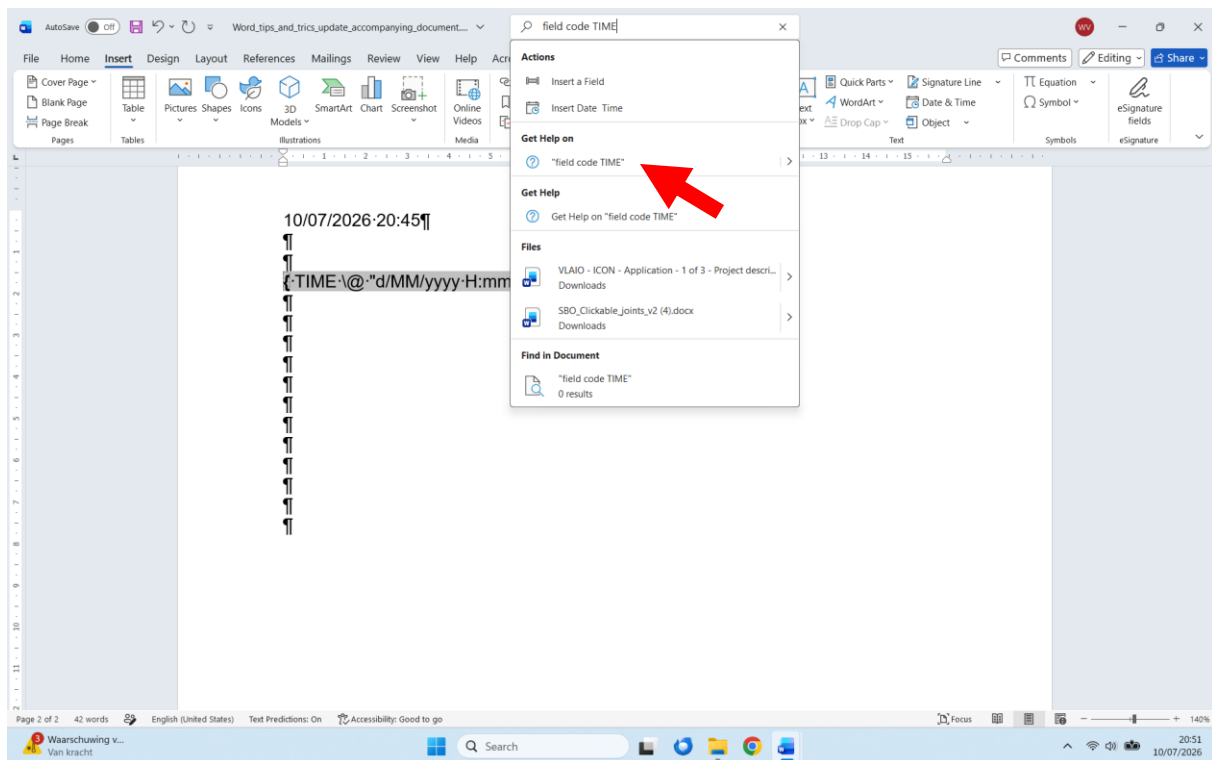


Figure 31 Searching Help information about the field code TIME.

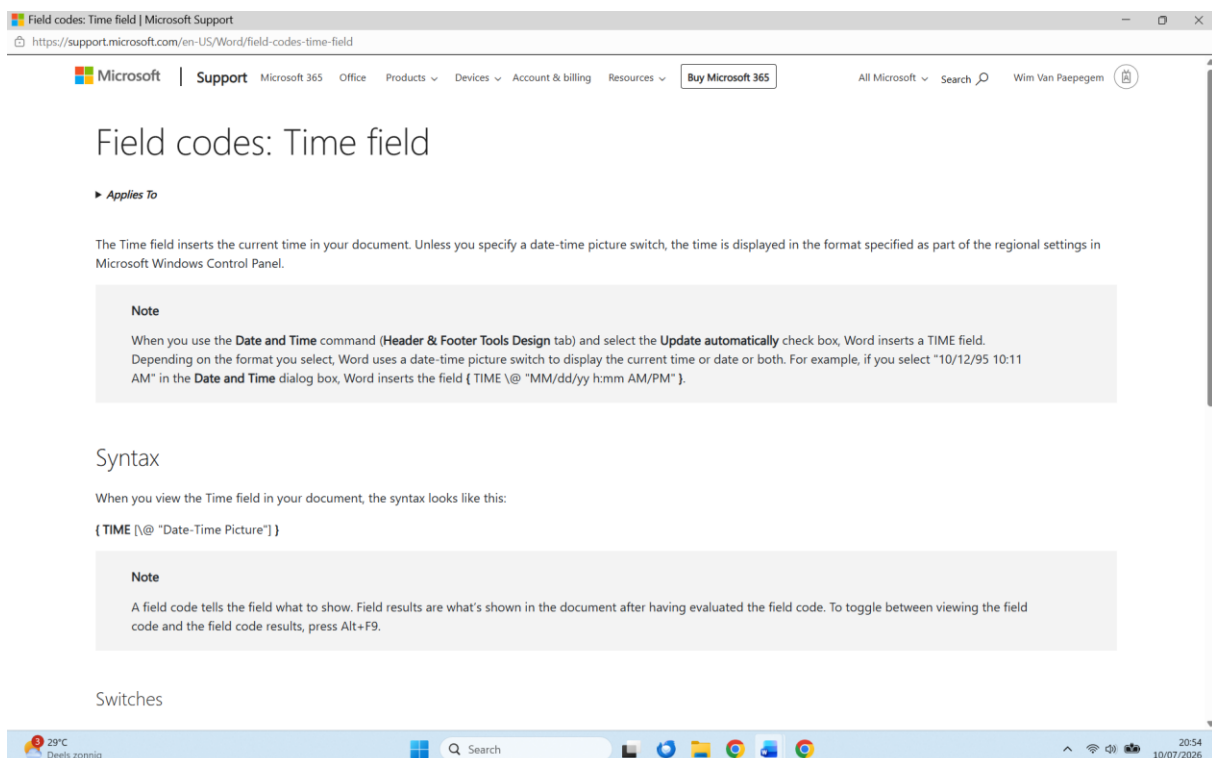


Figure 32 Help information on the field code TIME.

A lot more fields exist and most of them can be inserted by using the *Field* dialogue box. You can access it by clicking *Insert > Quick Parts > Field...*, as shown in Figure 33. The *Field* dialogue box is shown in Figure 34.

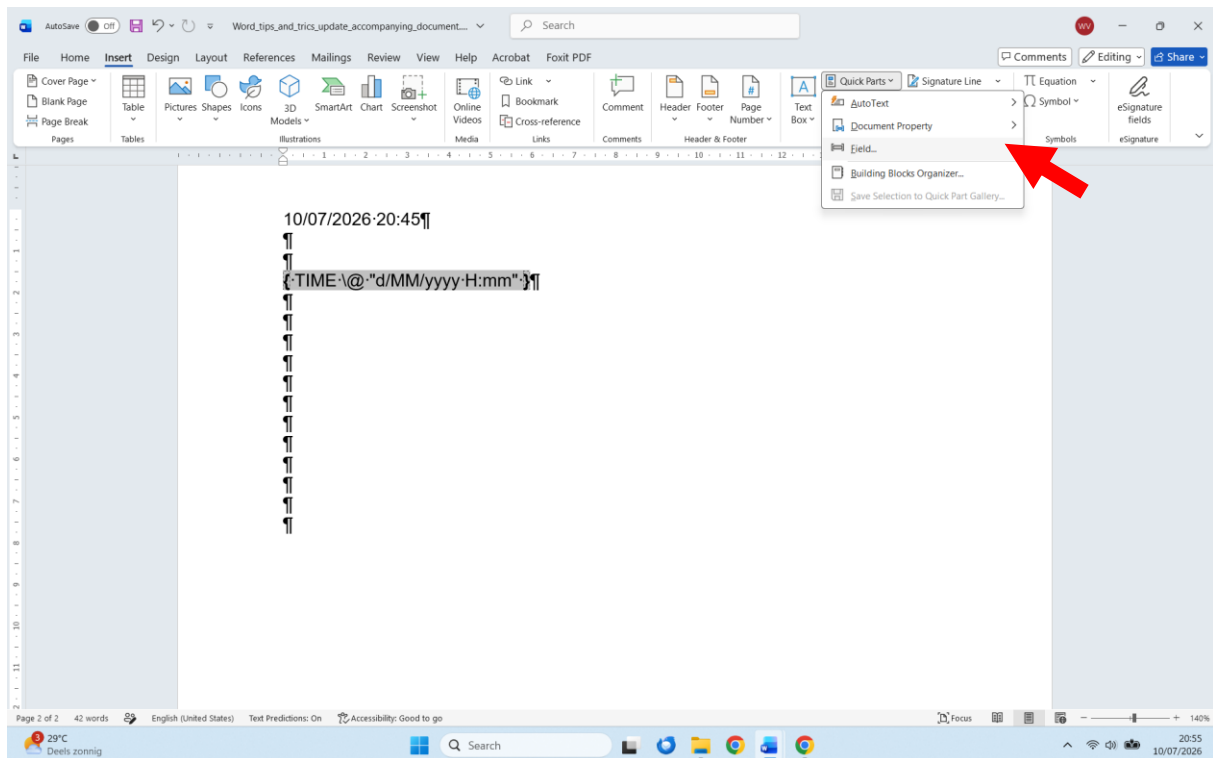


Figure 33 Inserting Field codes.

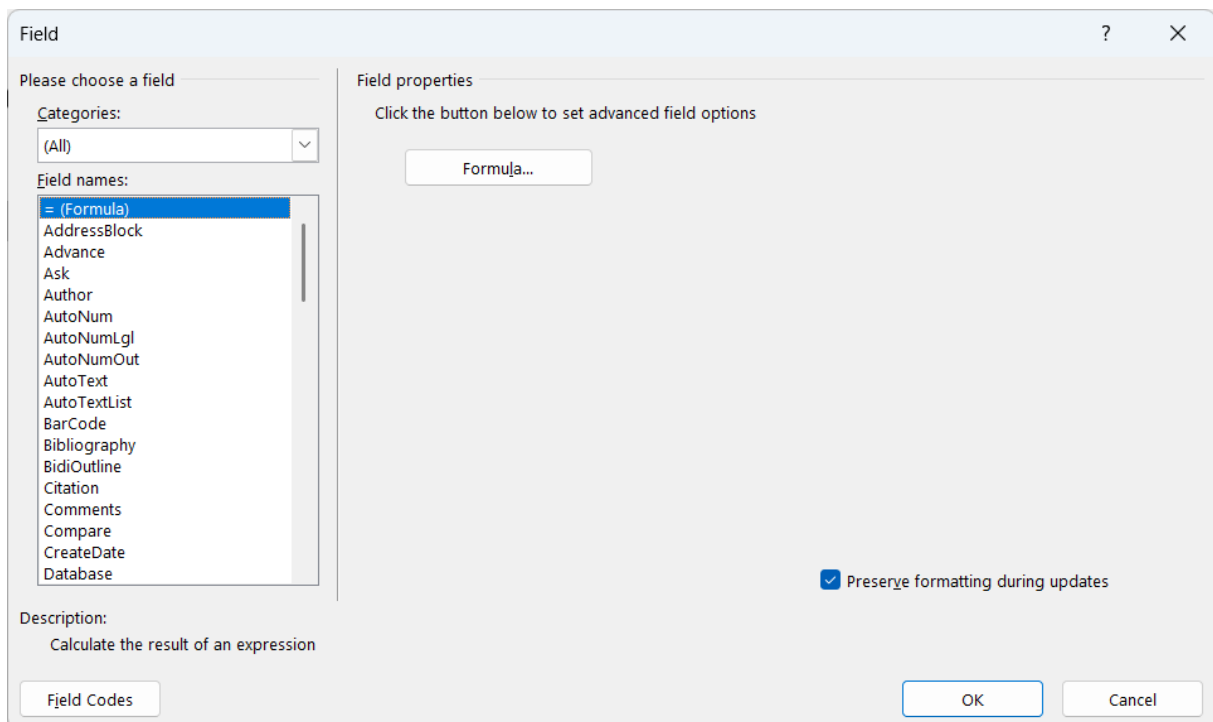


Figure 34 The *Field* dialogue box.

Some interesting field codes are listed below:

- **AUTHOR:** insert the author's name from the *Summary* tab in the *File > Properties* dialogue box,
- **DATE:** insert today's date and time,
- **FILENAME:** inserts the file name of the document, as recorded on the *General* tab in the *File > Properties* dialogue box,
- **NUMPAGES:** inserts the total number of pages in the document, using information from the *Statistics* tab in the *File > Properties* dialogue box,
- **PAGE:** inserts the number of the page that the PAGE field is located on,
- **TIME:** insert current time.

Instead of using the *Field* dialogue box, you can insert field codes directly into your text. However you cannot insert the necessary field braces {} by just typing these characters on the keyboard, though you can generate them by pressing <Ctrl>+<F9>. Next you can type any field code you want. For example, retype the TIME field code, but now with a different date and time format, as shown in Figure 35:

```
{ TIME \@ "dd/MM/yyyy HH:mm:ss" }
```

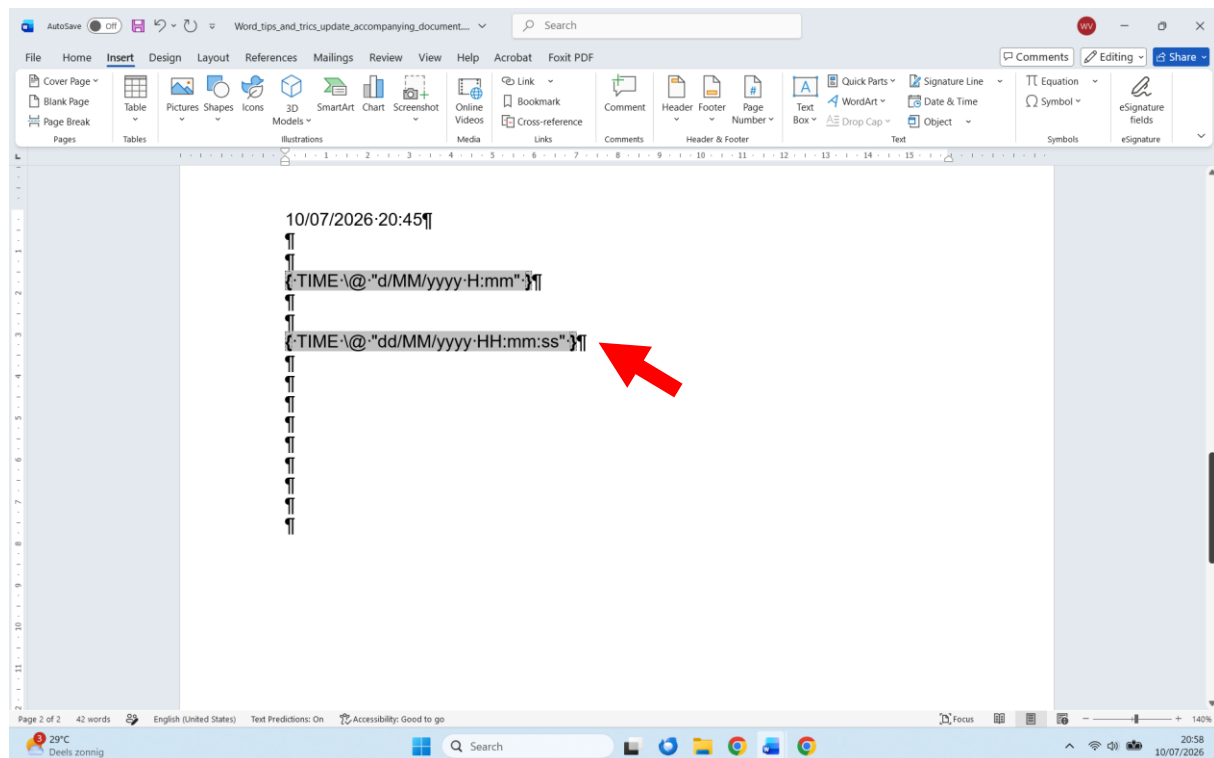


Figure 35 Inserting the TIME field code with a different date and time format.

If you press <Alt>+<F9> again, you would expect to see the result of this new field code, but only the result of the first field code is shown, not of the second one. The reason is that MS Word does not automatically update the results of new field codes. You have to explicitly force the updating of field codes by selecting the whole document (by clicking <Ctrl>+<a>) and then pressing F9 (function key for updating fields). Now you should see the result of Figure 36. You see that the Date & Time format of the last line is slightly different from the Date & Time format on the line before.

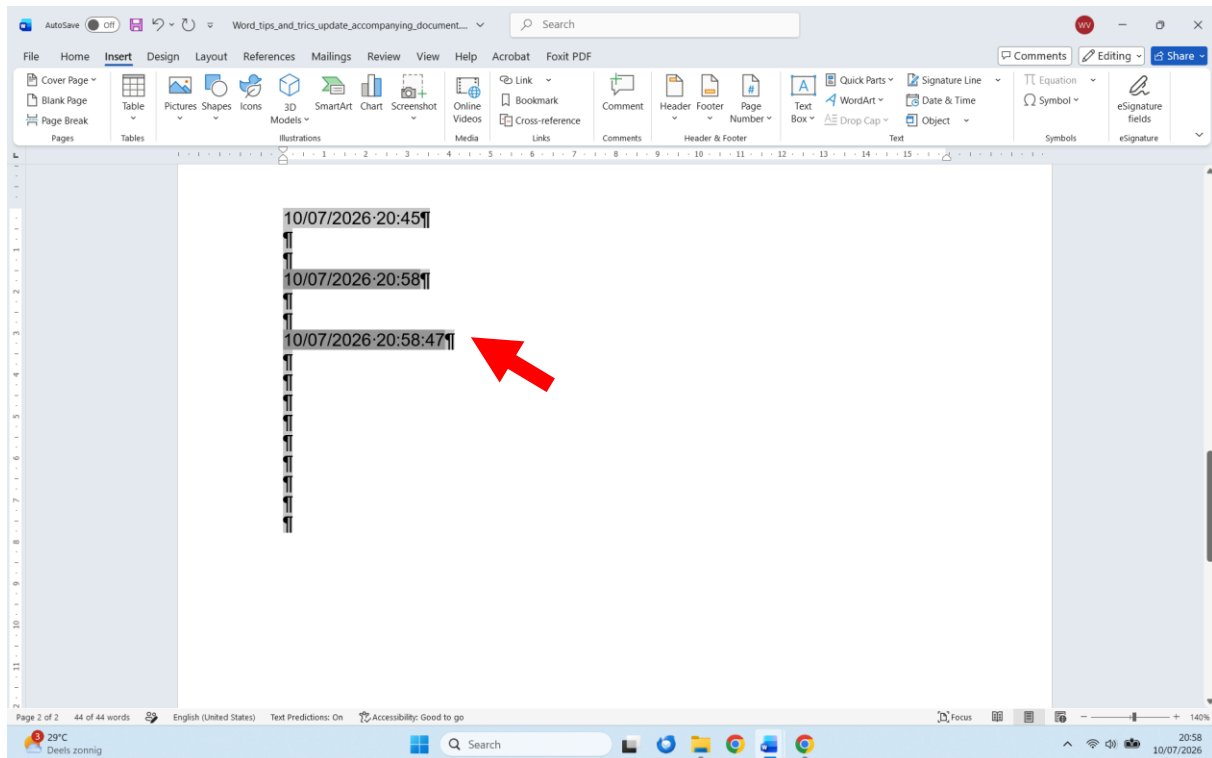


Figure 36 Updated TIME field codes with two different date and time formats.

Another solution to update the field code result, is to insert a new field with `<Ctrl>+<F9>`, and instead of pressing `<Alt>+<F9>`, you just press `<F9>` to update only the field code that is currently modified. To toggle between the field code and the result of the field code for one single field code, you do not press `<Alt>+<F9>`, but `<Shift>+<F9>`.

Finally, you may want to convert the result of the field code to regular text, so that the field code is not updated any longer. Therefore you select the field and press `<Ctrl>+<Shift>+<F9>`. You will see that the grey background disappears and that the field code has turned into regular text.

If you want that updating of the field code is only blocked temporarily, you can lock the field code by pressing `<Ctrl>+<F11>`.

Summarized, the following shortcut keys apply to field codes.

<code><Ctrl>+<a></code>	select whole document
<code><F9></code>	update one or more selected fields
<code><Alt>+<F9></code>	toggle field code display for all fields in the document
<code><Ctrl>+<F9></code>	insert new field code
<code><Ctrl>+<Shift>+<F9></code>	unlink field code and convert to regular text
<code><Shift>+<F9></code>	toggle field code display for one individual field
<code><F11></code>	go to next field
<code><Shift>+<F11></code>	go to previous field
<code><Ctrl>+<F11></code>	lock field (prevent manual or automatic update)
<code><Ctrl>+<Shift>+<F11></code>	unlock field (allow updating)

3.2. THE SEQ FIELD CODE

There is one very interesting field code that you should know and that is the field code SEQ. This is the field code that I use most.

This field code sequentially numbers chapters, tables, figures, and other items in a document. If you add, delete, or move an item and its respective SEQ field, you can update the remaining SEQ fields in the document to reflect the new sequence. The field code SEQ must be followed by an identifier. This is the name that you assign to a series of items that you want to number. The name must start with a letter and is limited to 40 characters (only letters, numbers and underscores). For example, the name for a series of equations might be “my_equations”.

If you want to number your equations, you can insert the SEQ field codes as shown in Figure 37. It is important to mention that the new paragraph style *Centered_equation* (see Figure 14 and Figure 15) has been used to insert the tab stops automatically.

You have to type the field code only once with <Ctrl>+<F9>, and then just copy the field code with the equation number. Figure 37 shows the field codes for three consecutive equations. The text of the field codes is identical and the list is called “my_equations”.

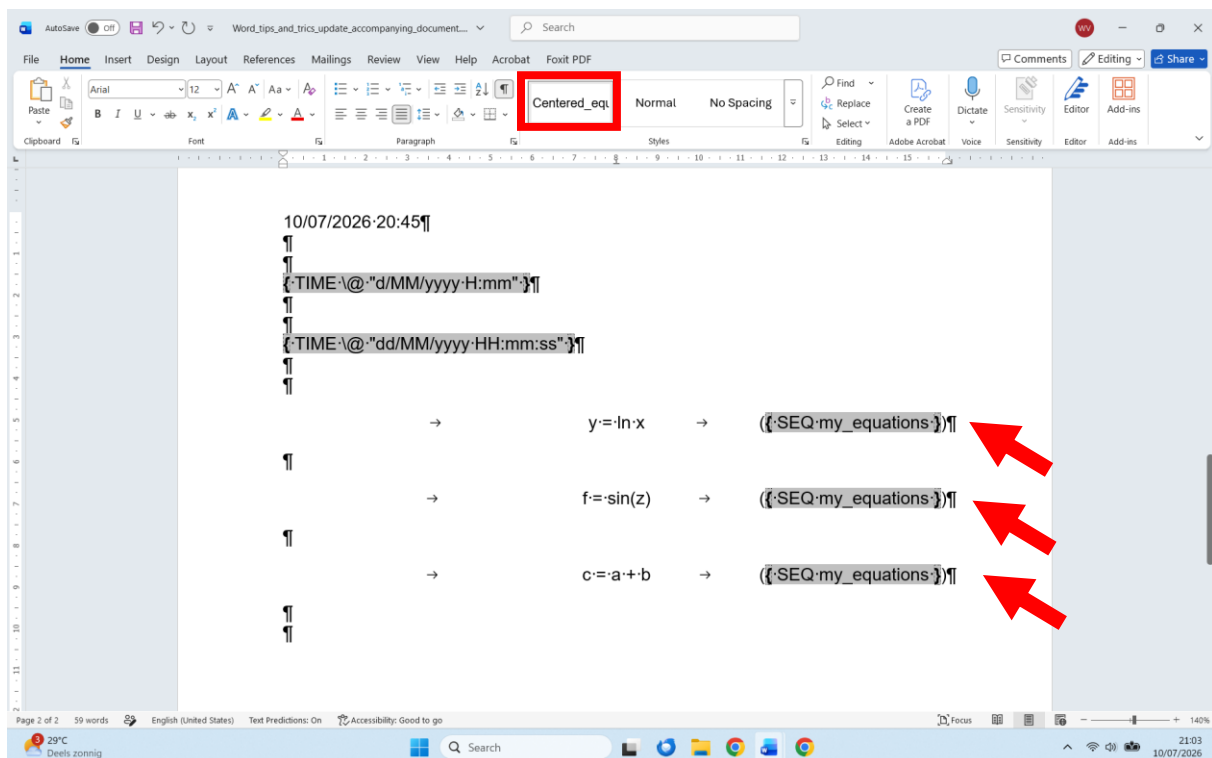


Figure 37 Automatic numbering of equations with the SEQ field code.

Next you select the whole text (<Ctrl>+<a>), update the fields with <F9> and press <Alt>+<F9> to toggle between the field codes and their results. The numbered equations are shown in Figure 38. The grey background of the numbers indicates that they are in fact field codes.

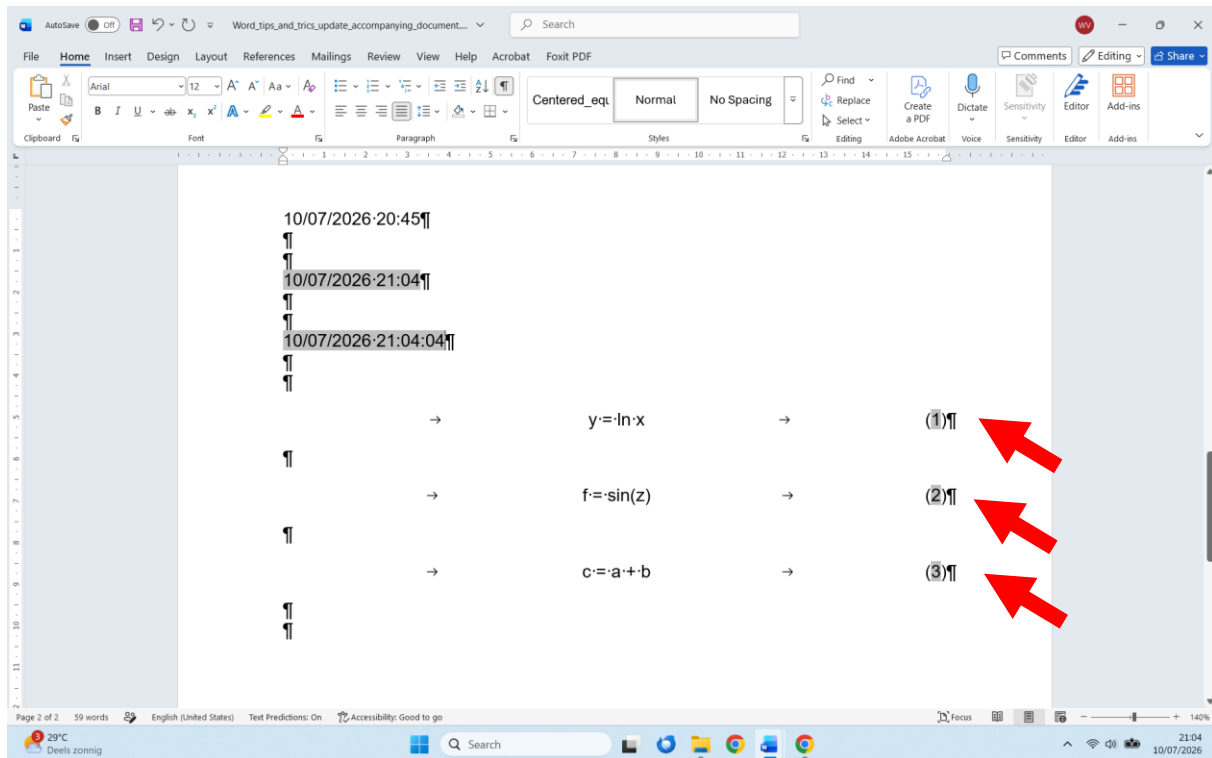


Figure 38 Result of the numbered equations after updating all fields (<Ctrl>+<a> and <F9>).

The power of field codes is only becoming clear once you start to move text. Let us say that Equation 3 ($c = a + b$) should in fact be the first equation, instead of the last one. You just cut the whole line and paste it in front of the two other equations. Apparently, the numbering is not updated and the numbering sequence of the equations is (3), (1), (2). However, if you select all text again (<Ctrl>+<a>) and update the fields (<F9>), the numbering is updated correctly, as shown in Figure 39.

Can you imagine how much time you can save if you have a large document with a lot of equations and you want to move a few equations to a different location ?

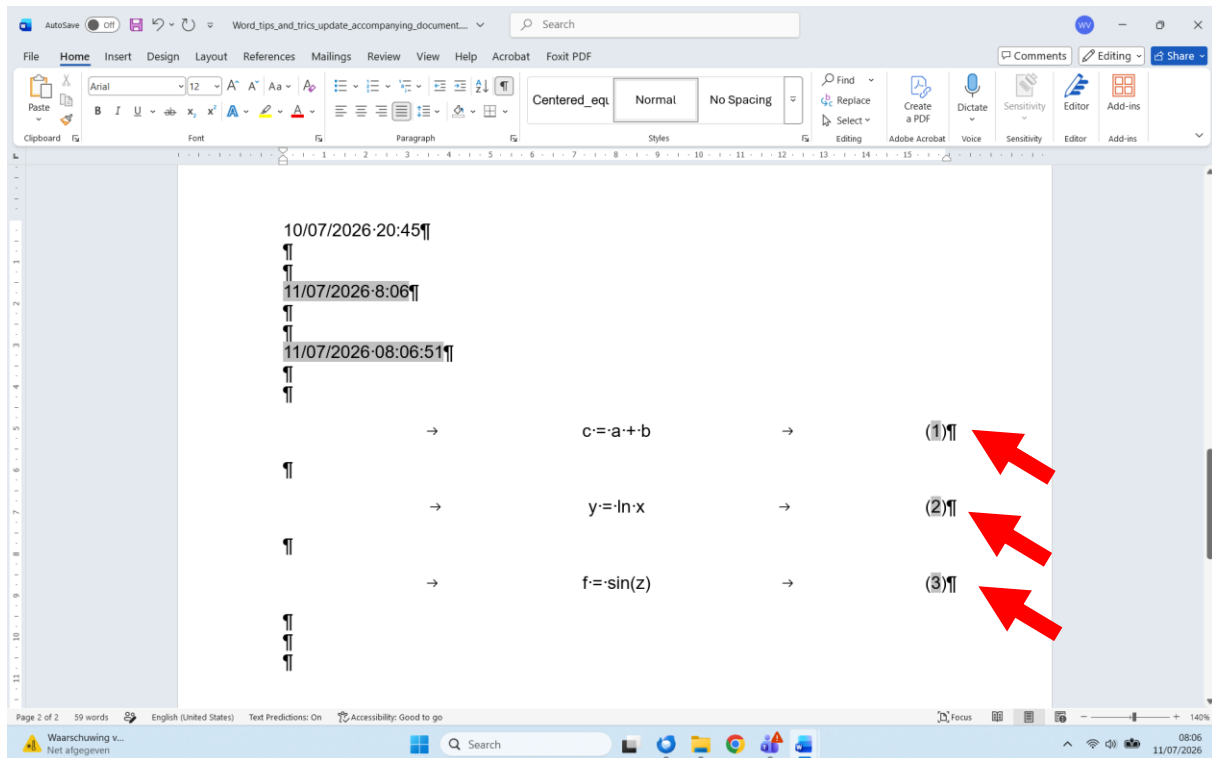


Figure 39 The new numbering sequence of the equations after moving Equation 3 to the first line.

Finally, the SEQ field code comes with several switches. We will use several of those switches in the coming chapters. Here is the summary table of the switches of the SEQ command.

Table 1 Switches of the SEQ field code.

\c	Repeats the closest preceding sequence number and does NOT increment the numbering. Mostly used to repeat the SEQ value on that page in headers/footers.
\h	SEQ increments the numbering but hides the field result from the document. Could be used to skip a number in the sequence, or refer to a SEQ field in a cross-reference without printing the number. For example, you might want to refer to a chapter but not print that chapter's number.
\n	Is not really necessary because this is the SEQ default. It increments to the next sequence number.
\r n	Changes or resets sequence numbering to the specified number n. For example, { SEQ my_equations \r 3 } restarts equation numbering at 3.
\s	Resets the sequence number at the heading level following the "s". For example, { SEQ figure \s 1 } restarts numbering at 1 after detecting the built-in paragraph style <i>Heading 1</i> .

3.3. THE STYLEREF FIELD CODE

There is another very interesting field code that you should know and that is the field code STYLEREF. The syntax is:

```
{ STYLEREF StyleIdentifier [Switches ] }
```

Again, this is a field code, so we have to insert this text with <Ctrl> + <F9>.

If we insert for example the field code { STYLEREF “Heading 1” }, we get to see the title of *Heading 1*.

If we insert the same field code with the switch \n, like { STYLEREF “Heading 1” \n }, we only get to see the chapter number, so that is what we need for example for equation numbering per chapter.

Both STYLEREF field codes are illustrated in Figure 40 below.

We will show in the next section that the combination of this field code STYLEREF with the field code SEQ gives us a powerful combination for numbering of equations and other items in your document.

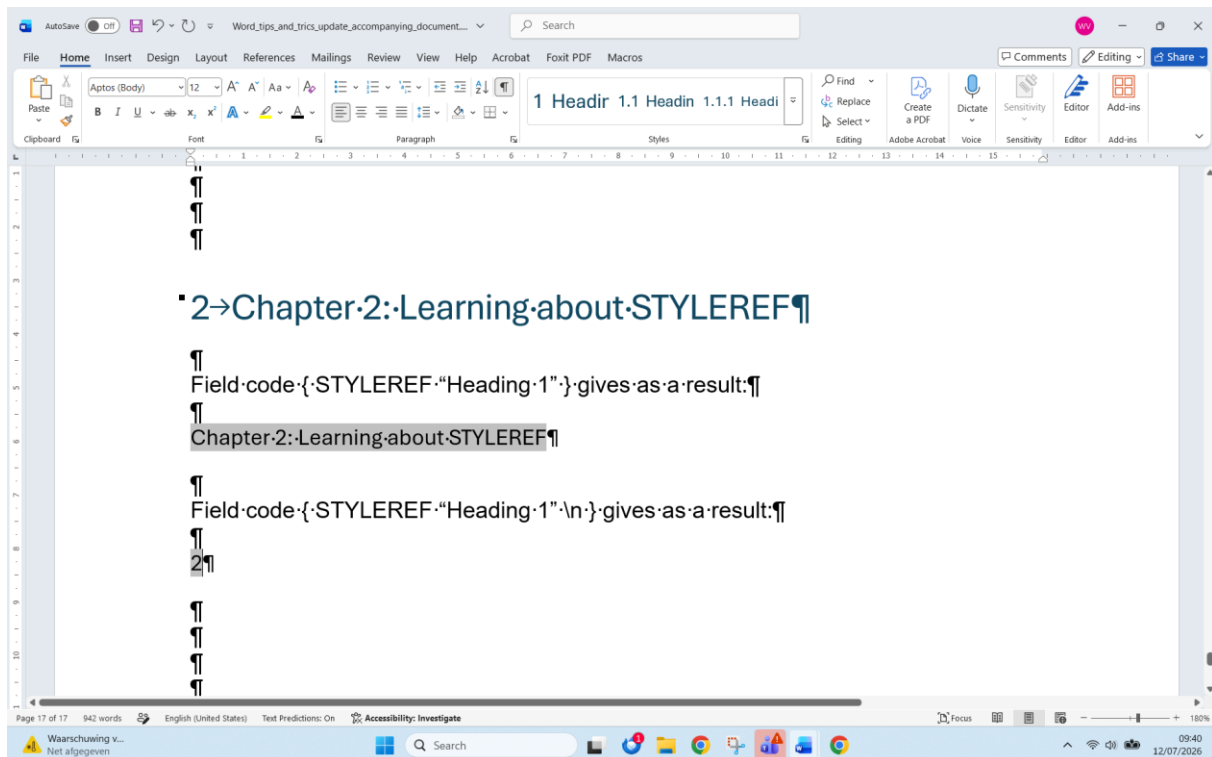


Figure 40 Use of the STYLEREF field code to refer to chapter titles and chapter numbers.

3.4. COMBINING STYLEREF AND SEQ FIELD CODES

In our previous example of equation numbering, we numbered the equations with the field code SEQ, but if we would start a new chapter, the numbering of the equations would just continue from the last number of the previous chapter. One step further is to not only have the equation centered and numbered, but also add the chapter number to it. If the equation is the first equation of chapter 6, the numbering should become:

$$f = \sin(z) \quad (6.1)$$

The first number of the equation should refer to the chapter number, and also be automatically updated if the equation is moved to another chapter.

The field code STYLEREF offers this functionality. The syntax is:

```
{ STYLEREF "Heading 1" \n }. { SEQ my_equations \s 1 }
```

Indeed, the first field code { STYLEREF "Heading 1" \n } will return only the chapter number, not the chapter title, because of the switch \n.

The second field code { SEQ my_equations \s 1 } will return the number of the equation, but restarted at number 1, every time a paragraph with the built-in paragraph style *Heading 1* is detected.

Figure 41 shows an example where there are 6 equations in chapter 5. Next a paragraph with the built-in paragraph style *Heading 1* is inserted. The next equation is again numbered from 1.

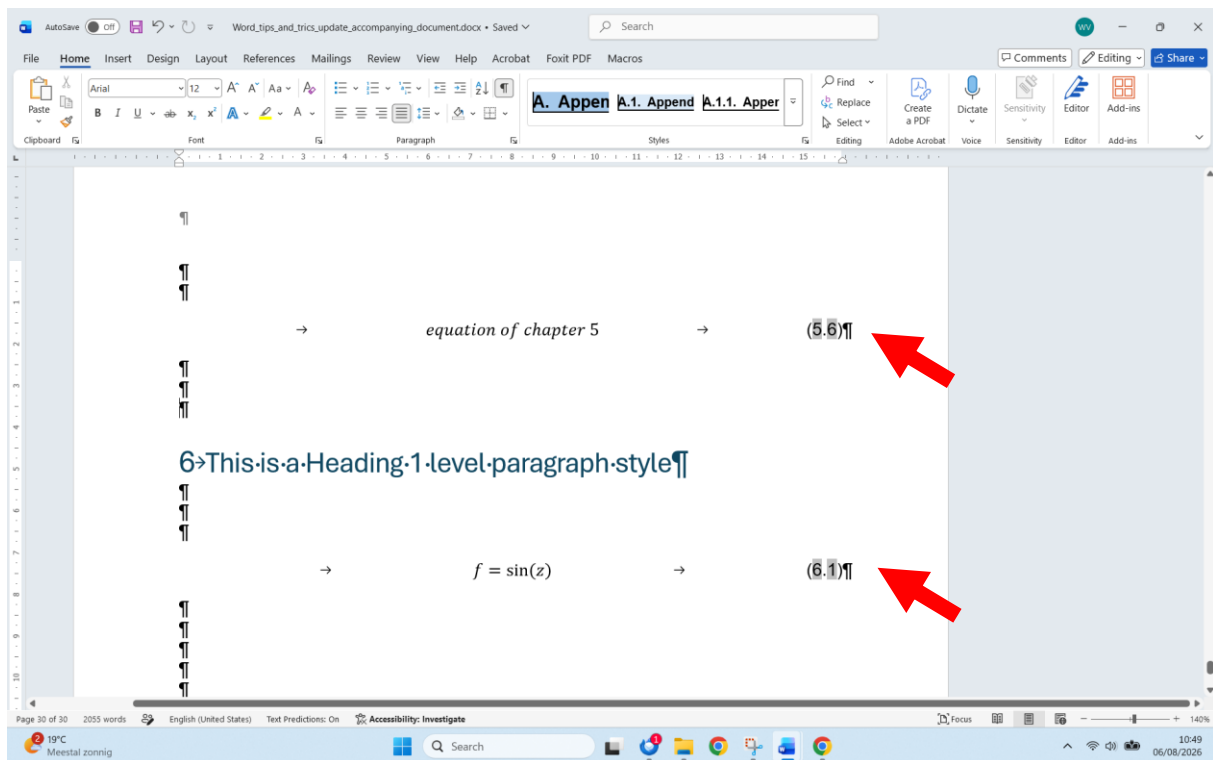


Figure 41 Example of numbering of equations with the chapter number included.

Figure 42 shows the corresponding field codes, visualized with <Alt> + <F9> to show all field codes.

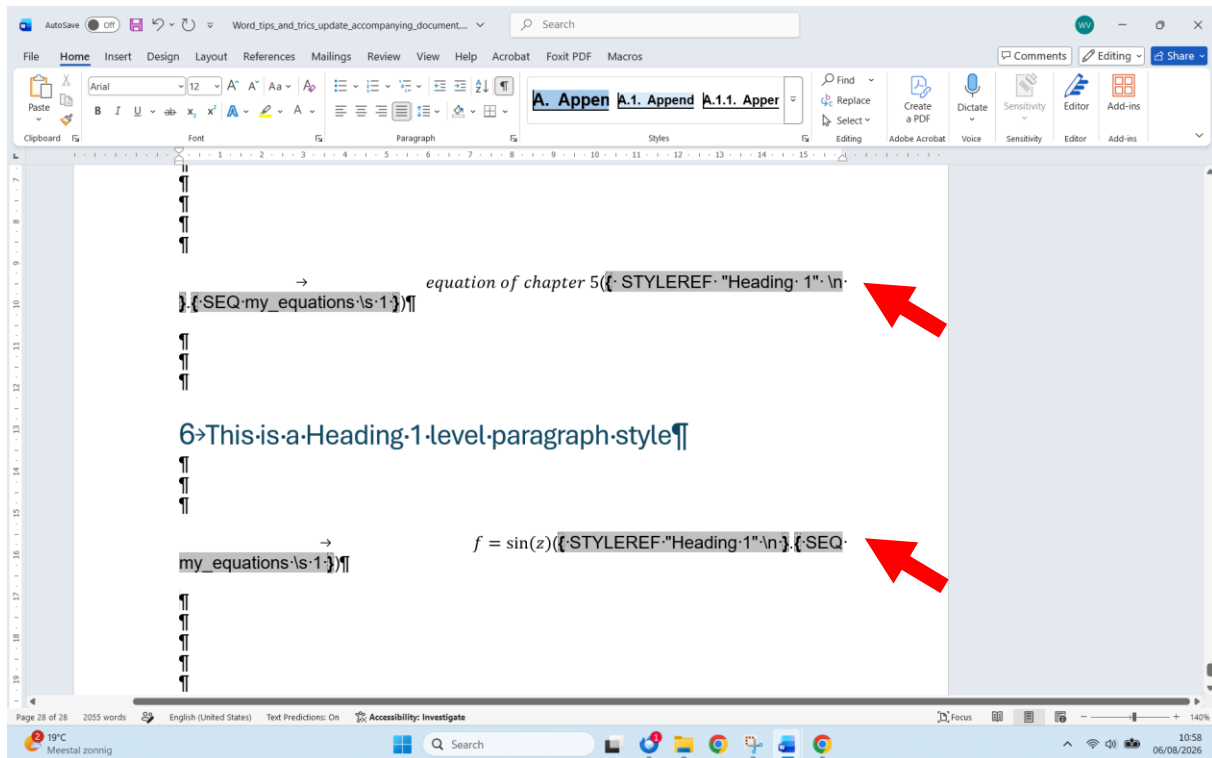


Figure 42 Corresponding field codes of numbering of equations with the chapter number included.

This example also shows one important downside of the SEQ field code. It can restart numbering after detecting a built-in paragraph style *Heading 1*, indicated by the switch \s 1. However, if you use your own user-defined paragraph style for the heading 1 level, and not the built-in paragraph style *Heading 1*, it does not work.

Indeed, the field code STYLEREF can take as an argument the name of the paragraph style it has to refer to, but the SEQ field code can only take the number of the heading level, so it only works with the built-in heading styles *Heading 1*, *Heading 2*, *Heading 3*, etc..

There is a very simple trick to solve this. When you use your own user-defined paragraph styles for the different heading levels, you just insert a paragraph with the built-in paragraph style *Heading 1* at the start of each chapter, and make the font colour white, so that the reader does not see this line of text. This is a trick that will also serve us well in later chapters.

This trick is demonstrated below. In Figure 43, the built-in paragraph style *Heading 1* is followed by a user-defined paragraph style for the chapter titles. The user-defined paragraph style is the one we want to use for the chapter titles, but the built-in paragraph style *Heading 1* is only there to reset the numbering of the equations with the SEQ field for every chapter.

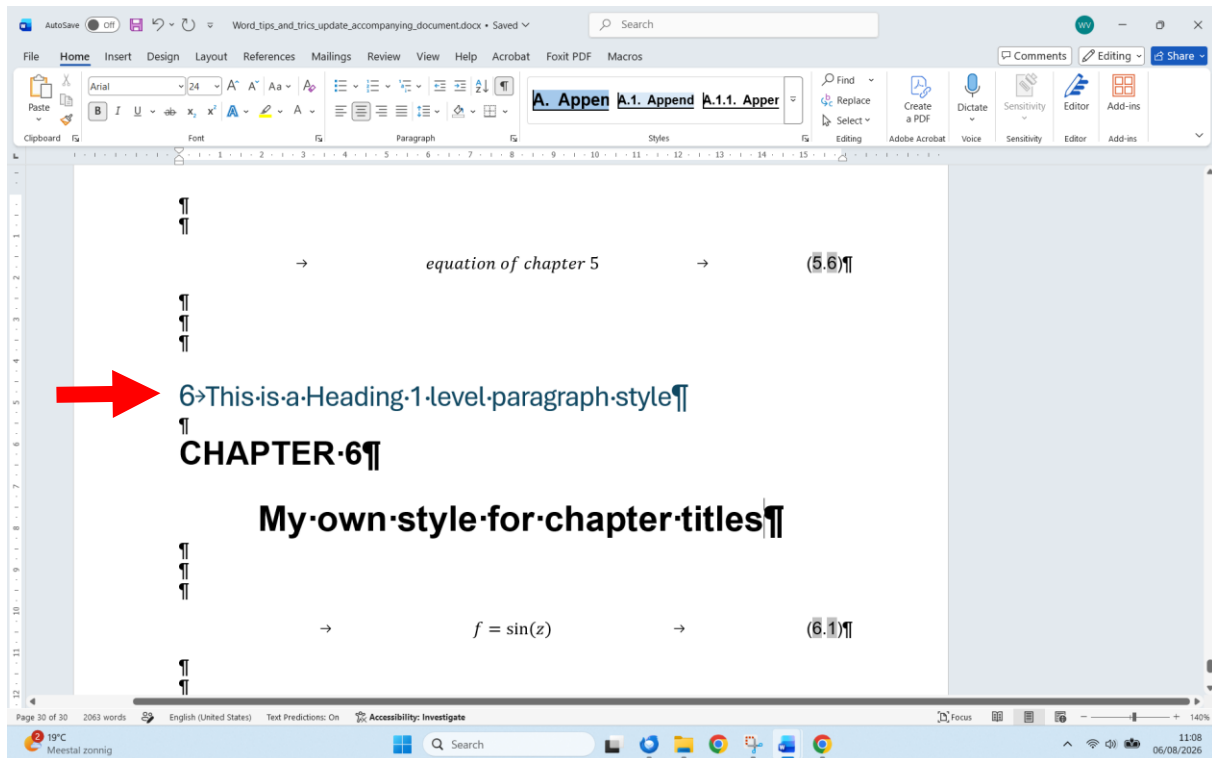


Figure 43 Built-in paragraph style *Heading 1*, followed by a user-defined paragraph style for chapter titles.

Figure 44 shows exactly the same document page as in Figure 43, but only the font colour of the line of text with the built-in paragraph style *Heading 1* is changed to white. When updating the field codes, the numbering of the equations still works correctly.

Instead of inserting a hidden paragraph with built-in paragraph style *Heading 1*, it would also be possible to insert a hidden field code at the start of each chapter as follows:

```
{ SEQ my_equations \r 0 }
```

This would reset the numbering of the equations at number 0, so the next field code { SEQ my_equations } would show the number 1 again. You do not need the switch \s 1 then for renumbering the equations for each chapter.

This field code { SEQ my_equations \r 0 } can then again be assigned the white font colour.

The reason why I prefer the method with the hidden built-in paragraph style *Heading 1*, is that it will also prove useful to number figures per chapter, as we will see in a later chapter.

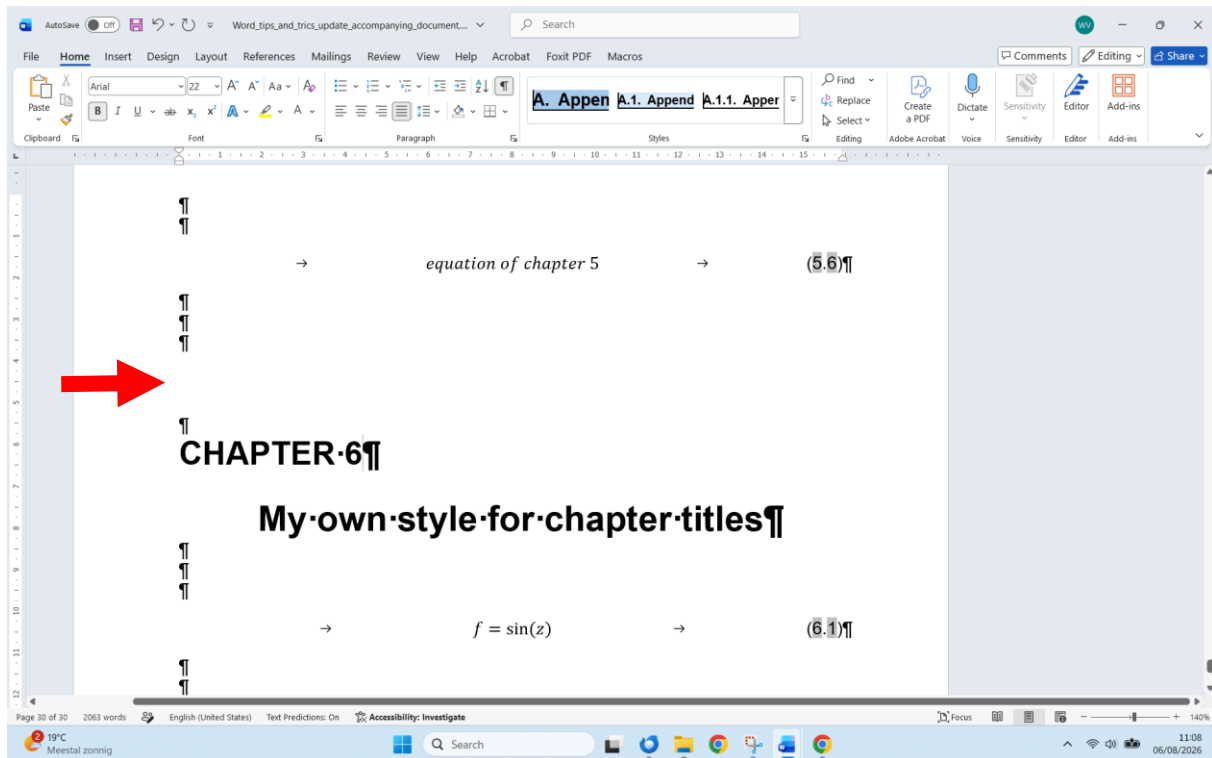


Figure 44 Result after assigning a white font colour to the built-in paragraph style *Heading 1*.

3.5. BOOKMARKS

One problem remains: if you want to refer in your text to Equation (3), and this Equation (3) becomes Equation (1), how can you update that cross-reference in your text automatically? Therefore you use bookmarks.

Bookmarks are very closely related to field codes. They identify a location within your file that you can later refer to or link to. They are a sort of “post-it” that you attach to a specific part of text in your document and this “post-it” remains fixed to that part of text, regardless where this text moves to.

To demonstrate this, we continue with the previous example. We add a bookmark to the number of the third Equation ‘ $f = \sin(z)$ ’ in Figure 39. Therefore we select the field code 3 of the equation number and then click *Insert > Bookmark...*, as shown in Figure 45.

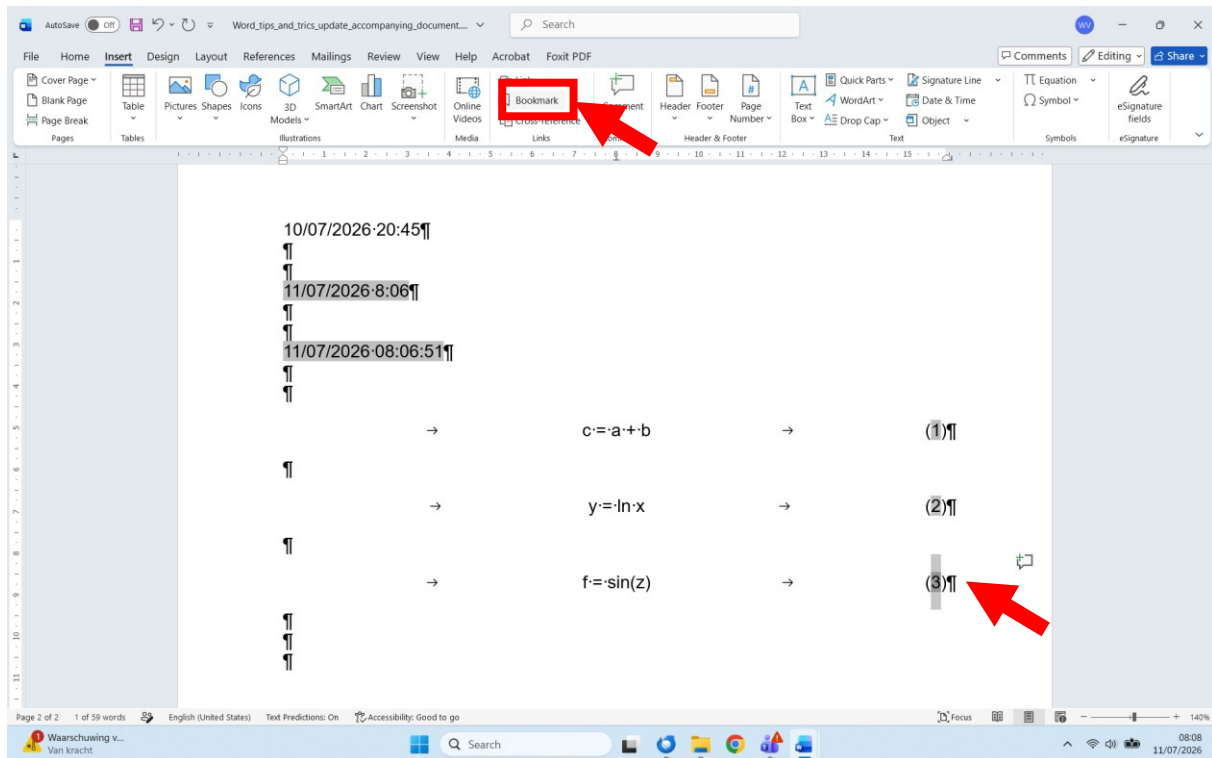


Figure 45 Inserting a bookmark for equation number 3.

The *Bookmark* dialogue box in Figure 46 appears. You can enter any name you want, but bookmark names must begin with a letter and can contain numbers. You cannot include spaces or special characters in a bookmark name. However, you can use the underscore character to separate words. In this case, you can use for example the bookmark name *Equation_f_sinz*. Do not use the equation number in the bookmark name (e.g. *My_equation_3*), because this number can change when you move the equation to a different location in the text. You click the Add button to assign this bookmark name to the field code 3.

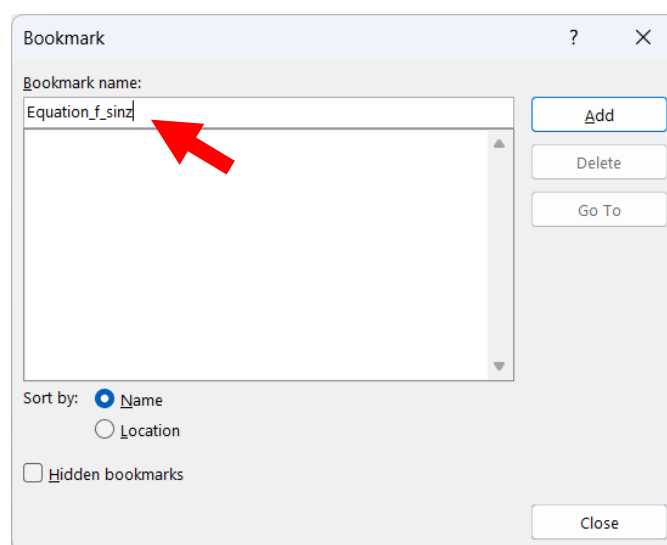


Figure 46 The *Bookmark* dialogue box.

Pay attention that after adding the bookmark to the Equation number 3, two light grey []-brackets are shown around the number 3 (see Figure 47). That is because in the first Chapter, we activated the display of bookmarks (see Figure 2). Again, this marking will not appear when printed, but is a visual aid to see that a bookmark is attached to this piece of text.

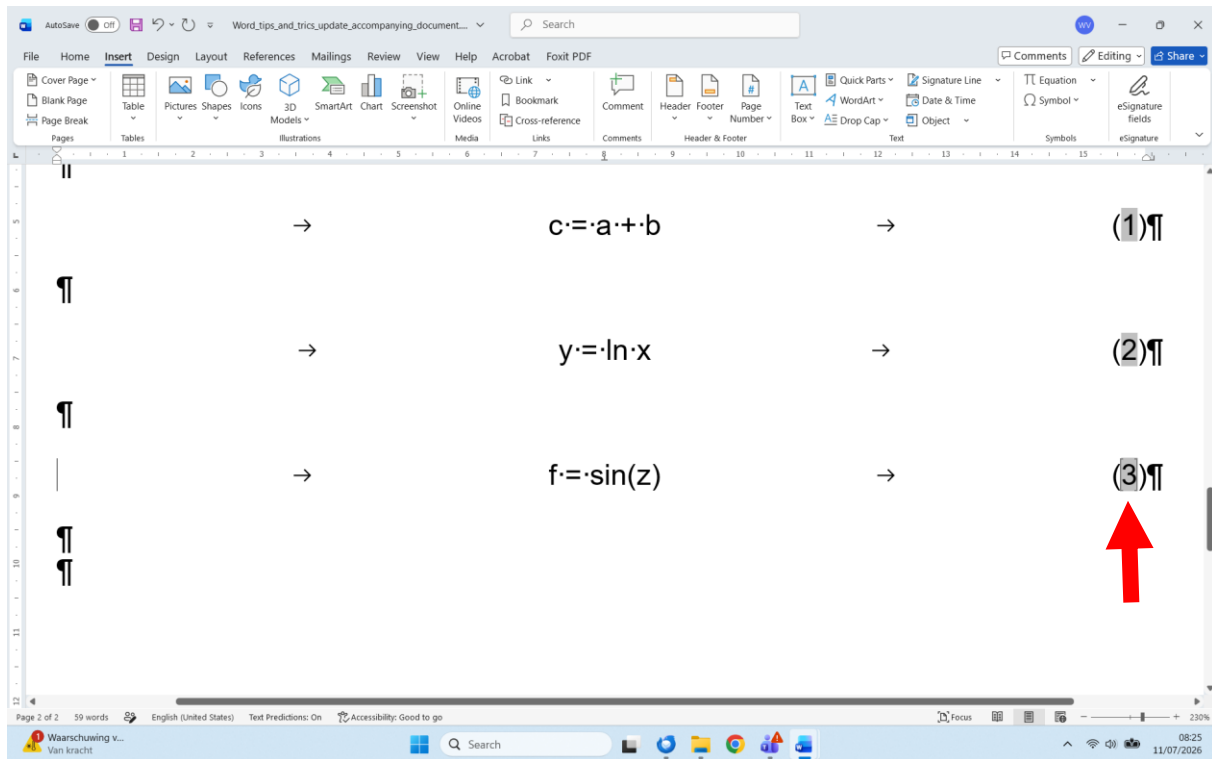


Figure 47 Highlighting the bookmark for equation number 3.

To refer to Equation (3) in your text, you first write the text “*This relation is shown in Equation (?)*”. Instead of typing now the number 3, you click *Insert > Cross-reference...* and the *Cross-reference* dialogue box in Figure 48 appears.

You select the option *Bookmark* in the drop-down list *Reference type*: and you choose the option *Bookmark text* in the drop-down list *Insert reference to*: on the right side. Now you will refer to the bookmark with name *Equation_f_sinz* and insert in the text the value of this bookmark, being the field code 3. The settings to do this, are shown in Figure 49. You confirm by clicking the *Insert* button.

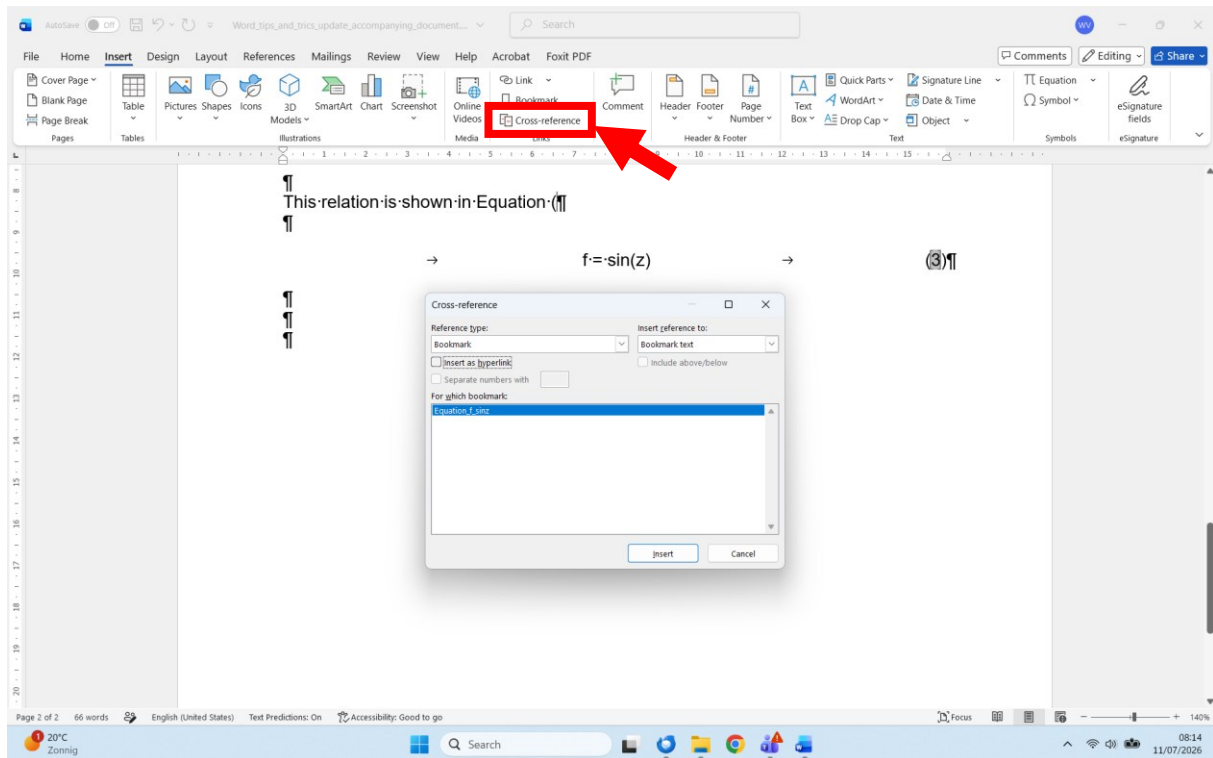


Figure 48 The *Cross-reference* dialogue box.

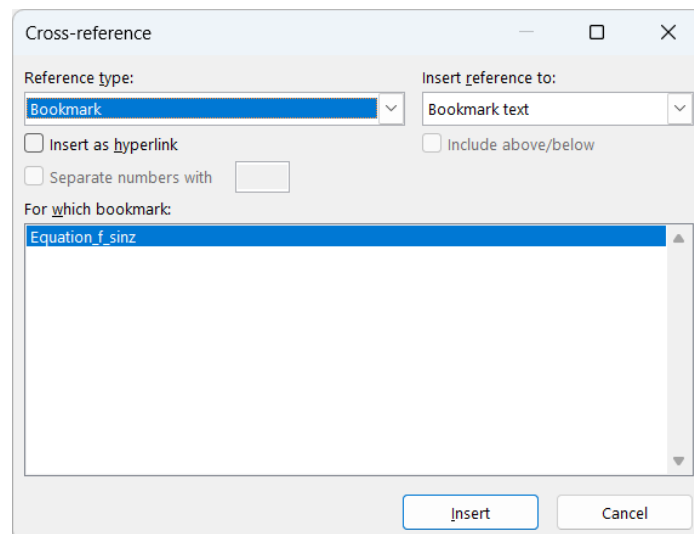


Figure 49 Settings in the *Cross-reference* dialogue box to refer to the bookmark *Equation_f_sinz*.

The result is shown in Figure 50. The grey color of the number 3 in the text indicates that this is a field code as well.

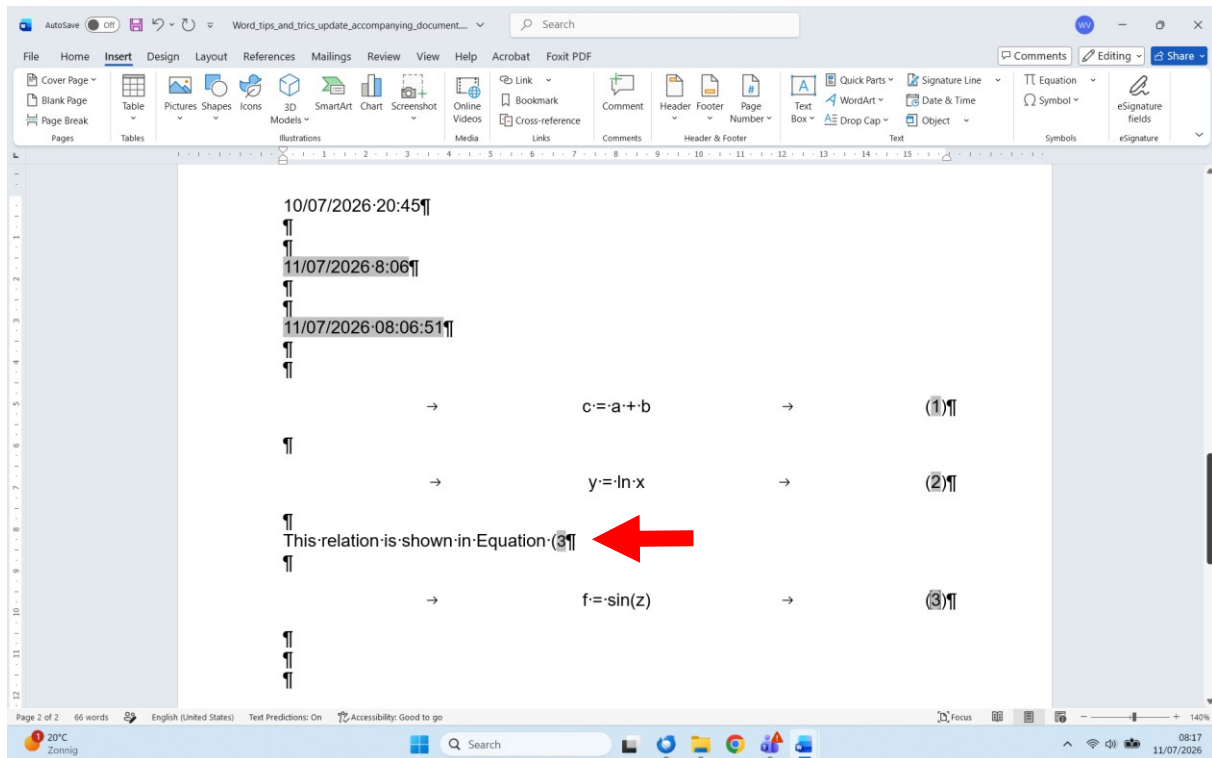


Figure 50 Result of the cross-reference to the bookmark *Equation_f_sinz*.

When you move Equation (3) and its preceding text now in front of the two other equations again, and you select all text (<Ctrl>+<a>) and update all fields (<F9>), the result of Figure 51 is shown. The Equation ' $f=\sin(z)$ ' is renumbered to (1) and the cross-reference in the preceding text is also changed to Equation (1).

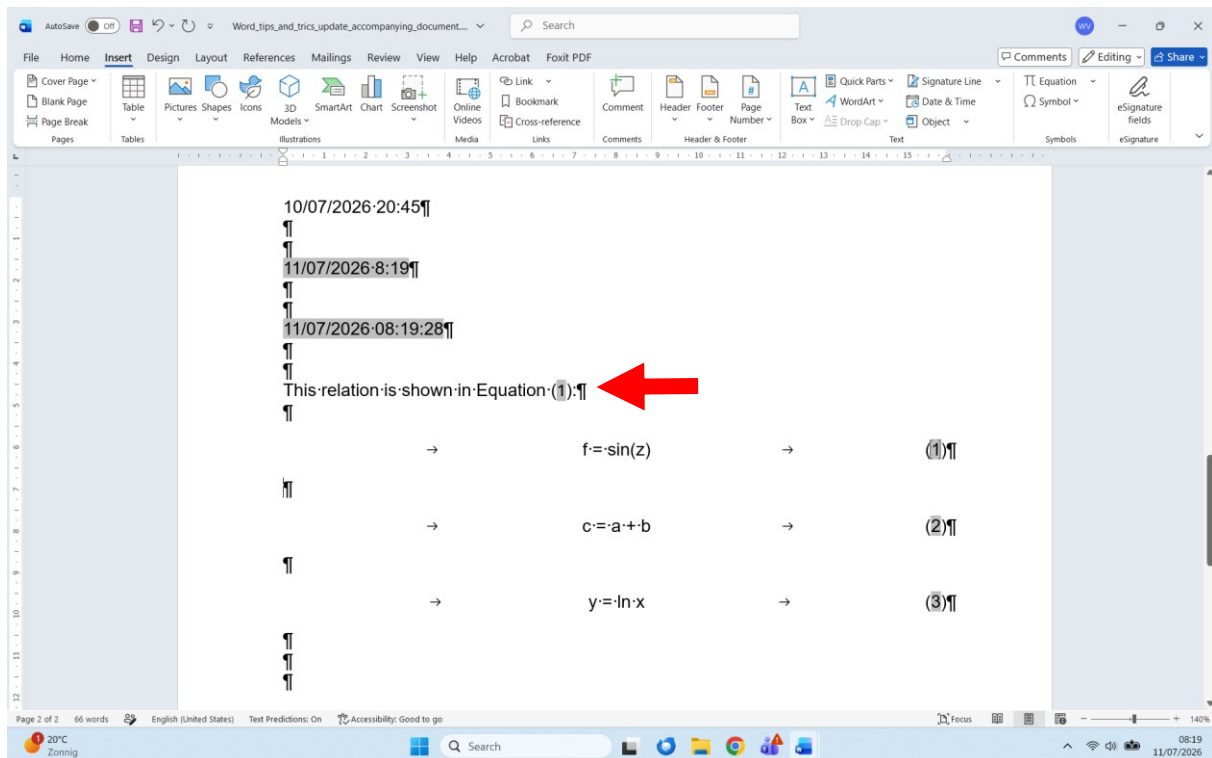


Figure 51 Updating the field codes of equation numbers and cross-reference to bookmark.

If you press <Alt>+<F9> to show the field codes (see Figure 52), you see that the cross-reference to the bookmark name *Equation_f_sinz* makes use of the field code REF, followed by the name of the bookmark *Equation_f_sinz*.

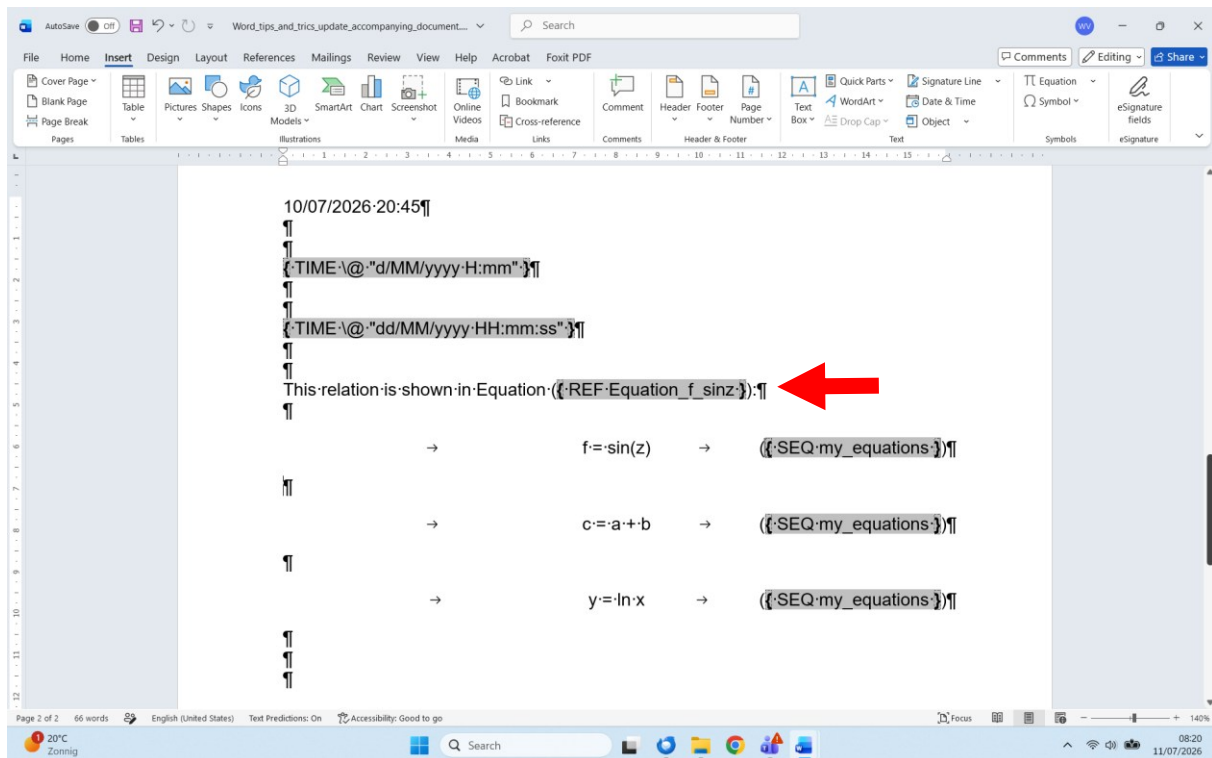


Figure 52 REF field code to insert a cross-reference.

3.6. CALCULATING WITH FIELD CODES

There are not many cases where this feature is useful, but just nice to know that you can even calculate with field codes.

If you enter the field code expression below after the set of equations of Figure 39:

```
{= { SEQ my_equations \c } - 3 }
```

Here, `{ SEQ my_equations \c }` repeats the most recent value without incrementing it. The `\c` switch is documented as repeating the closest previous sequence number, and is often used when you need to reuse the current sequence value rather than create a new one. So the calculation in the field code expression will return the number 0, because the numbering of the equations had reached number 3, and the number 3 was subtracted.

More useful applications are in combination with bookmarks. Suppose you want to mention how many equations are in a certain chapter, but that number of equations might change.

To do that, you can use a calculation with field codes.

You start with assigning a bookmark to the last equation of Chapter 1 and last equation of Chapter 2, as shown in Figure 53. Those bookmark names are respectively "Last_equation_chapter_1" and "Last_equation_chapter_2".

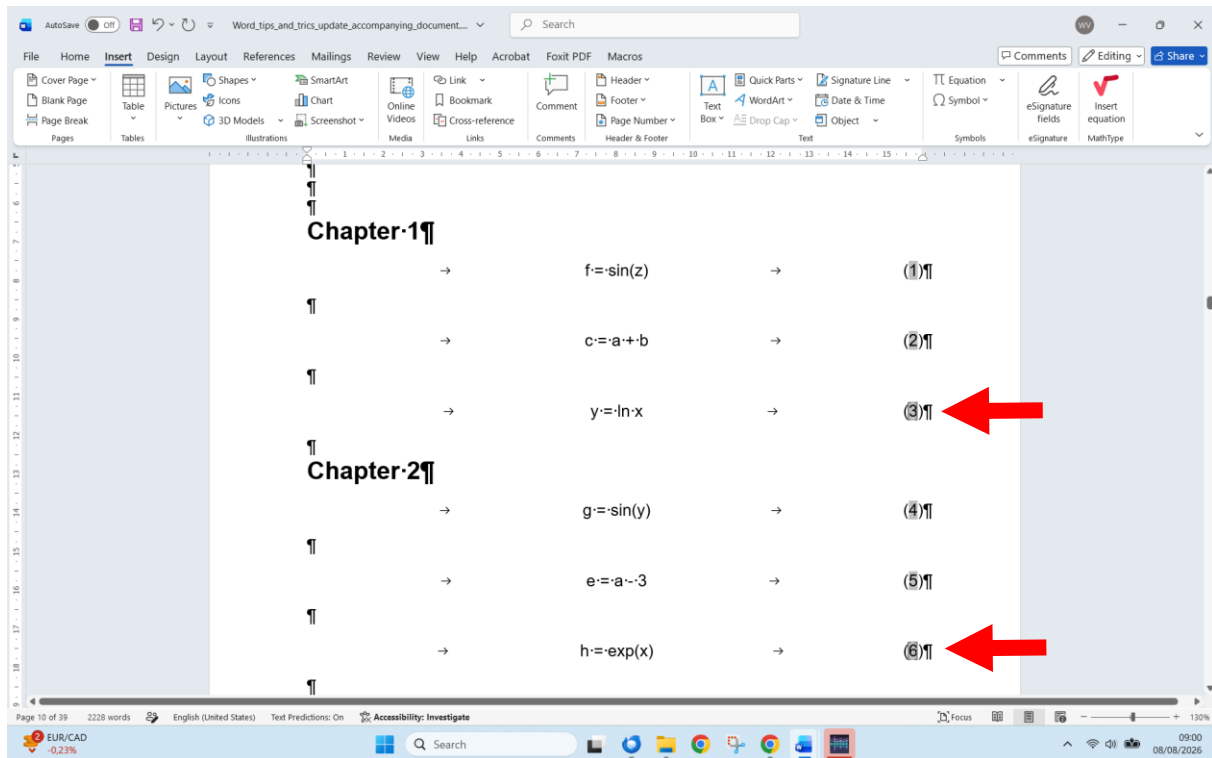


Figure 53 Adding a bookmark to the last equation of the respective chapters 1 and 2.

If you now want to write how many equations are in Chapter 2, you write the nested field code with bookmarks (do not forget to use <Ctrl>+<F9> to insert the curly brackets):

$\{ = \{ \text{REF Last_equation_chapter_2} \} - \{ \text{REF Last_equation_chapter_1} \} \}$

When you press <F9>, the field code will show the result 3, as indicated in the last line of text of Figure 54.

If you now insert an extra equation in Chapter 2, the field code calculation will automatically update to the number 4, as shown in Figure 55. Of course, if you would add a new equation after the last equation of Chapter 1 or 2, the field code expression would not give the right answer anymore, so this method has its limitations.

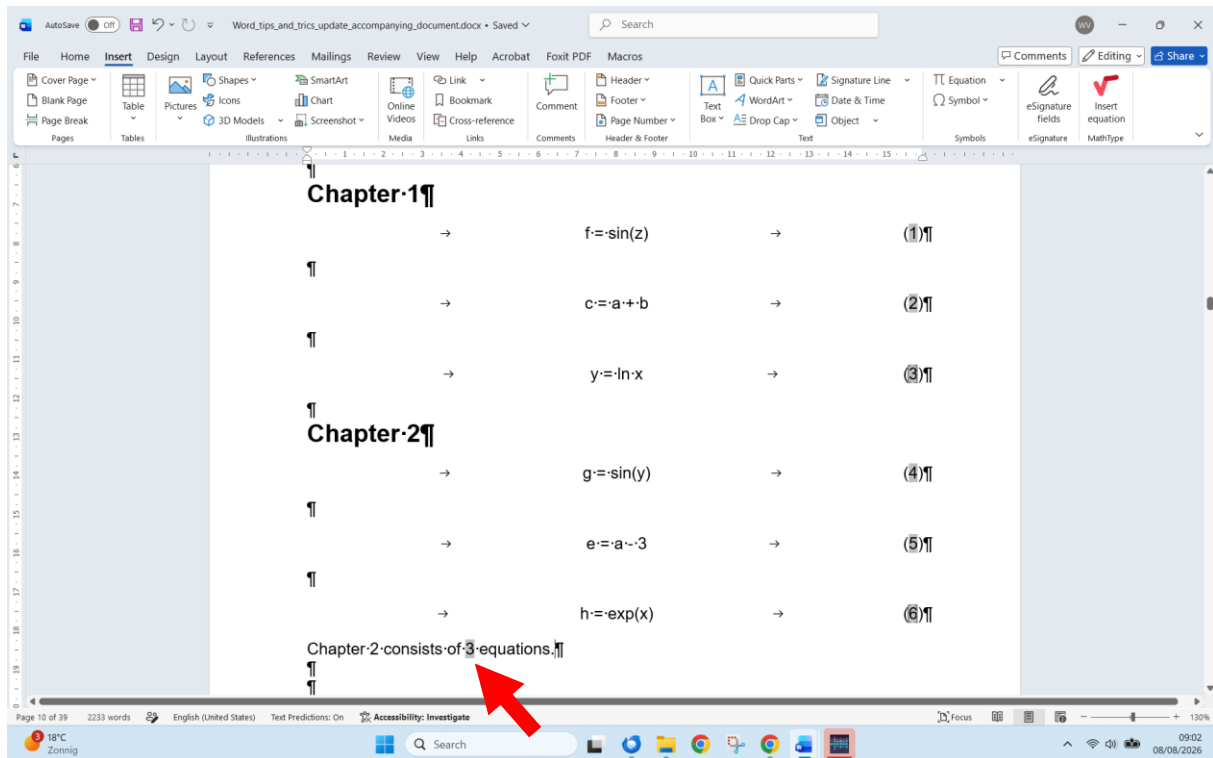


Figure 54 Result of the calculation with nested field codes that refer to bookmark names.

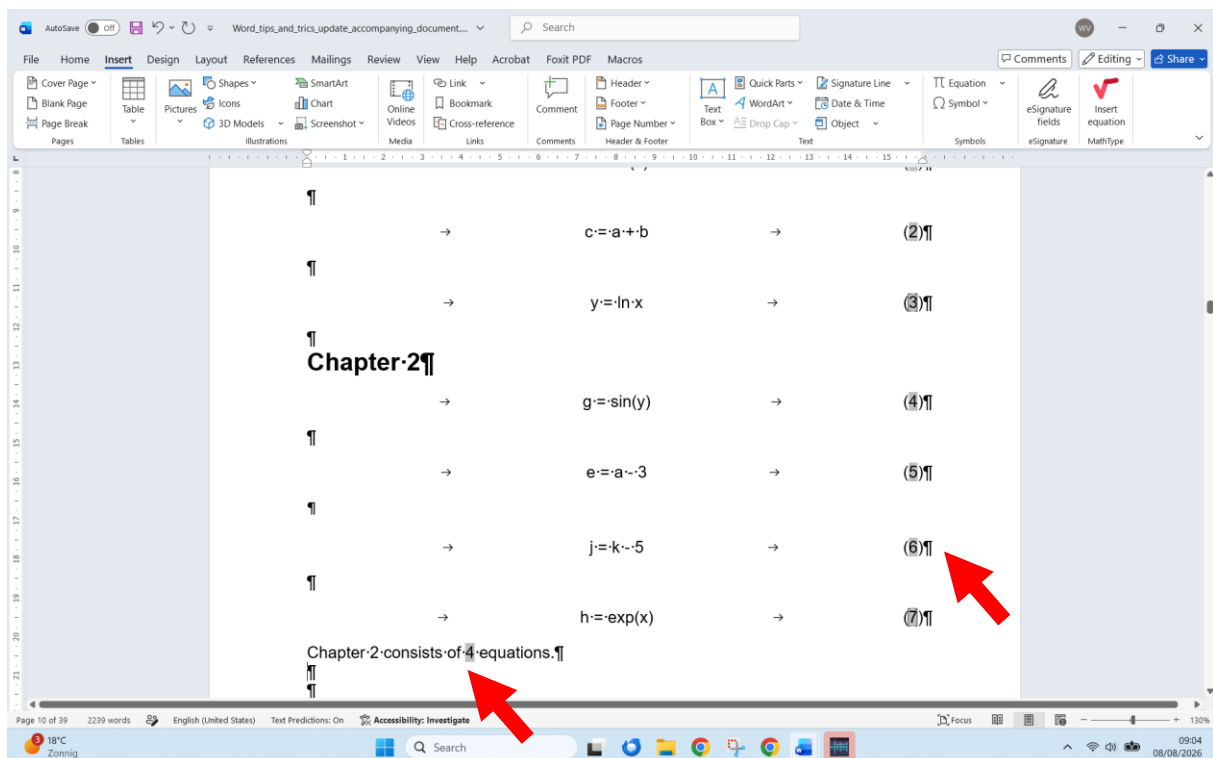


Figure 55 Automatic updating of the total number of equations in Chapter 2, after one equation has been added.

Although the use of these field codes and bookmarks is a very powerful tool, it might still seem somewhat laborious to work with them. We will see in a later chapter that these procedures on their turn can be automated by using macros to insert field codes and bookmarks with one single mouse click.

4

Using captions and cross-referencing for figures and tables

The first topic that we discussed, were styles (Chapter 2), and the second one were field codes and bookmarks (Chapter 3). This Chapter combines both previous topics to automatically number figures and tables and add cross-references in your text.

4.1. NUMBERING FIGURES AND TABLES

In a report, article, thesis, etc., every figure and table has a label, a number and a caption (descriptive text). MS Word has an automatic function to insert all three features together. Therefore you click *References* > *Caption...* The *Caption* dialogue box in Figure 56 is opened. By default, the caption has the label *Figure* and the figure number '1'.

You can change the label to any text you like (e.g. My own figure), and by clicking the *Numbering...* button, you can change the settings for the numbering. If you accept the default settings, the caption *Figure 1* is inserted in your text, as shown in Figure 57.

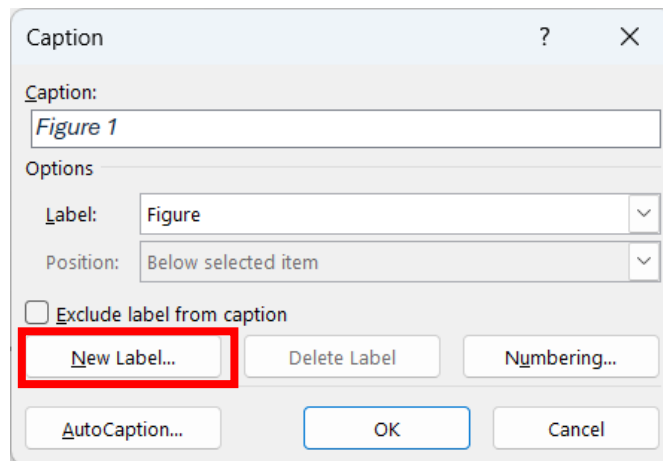


Figure 56 The *Caption* dialog box.

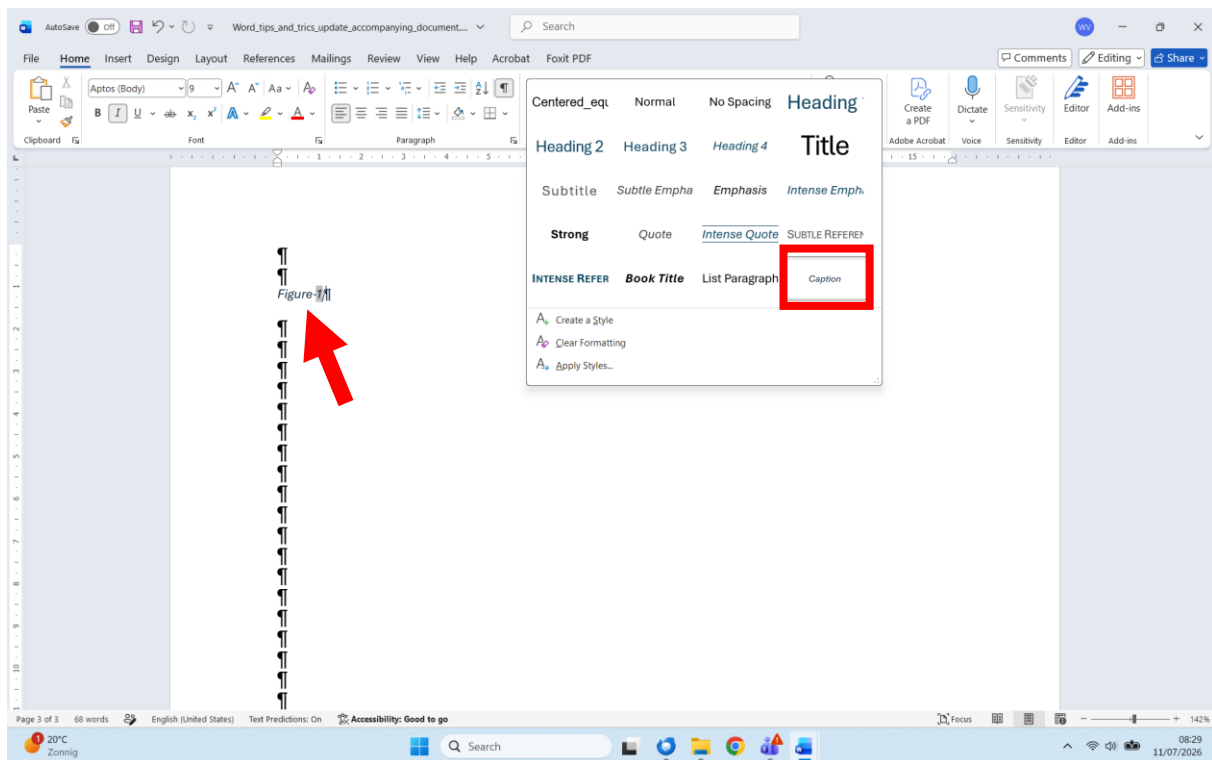


Figure 57 Inserting the default caption *Figure 1* into your text.

Two important things must be noticed:

- (i) the number '1', following the label *Figure*, has a grey background color. That means that this number is a field code. Indeed, if you press <Alt>+<F9>, the field code { SEQ Figure * ARABIC } is shown. This SEQ field code looks already familiar and differs in nothing from the SEQ field code discussed in the previous Chapter about field codes and bookmarks. The switch * details that we use arabic numbering style.
- (ii) the paragraph style has changed from *Normal* to *Caption*. You can see this by clicking on the Home button in the toolbar, and then clicking in the dropdown list of the *Styles* toolbar. Then you see that the *Caption* style is activated. This *Caption* paragraph style is a built-in

paragraph style which is assigned by default to all captions. You can modify this *Caption* paragraph style by adding additional tab stop positions or formatting marks (bold, italic, font size, etc.).

To refer to this *Figure 1*, MS Word uses again a cross-reference. You click *Insert > Cross-reference...* and the *Cross-reference* dialogue box appears. This dialogue box is exactly the same as the one we used in Figure 49 to add a cross-reference to a bookmark. In this case we want to refer to a figure, so you select the option *Figure* in the drop-down list *Reference type*: and you choose the option *Only label and number* in the drop-down list *Insert reference to*: on the right side. These settings are shown in Figure 58.

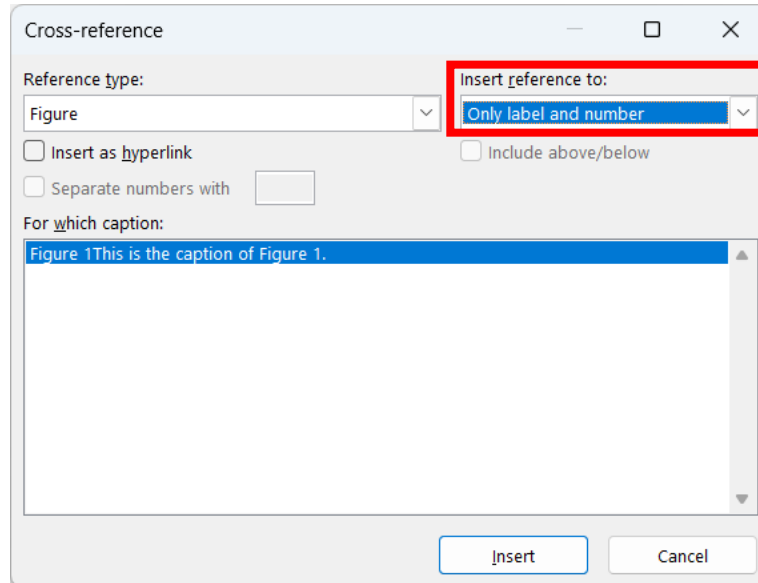


Figure 58 Settings for inserting a cross-reference to a figure caption.

If you click the *Insert* button, the result should look like Figure 59. Again the cross-reference text *Figure 1* has a grey background, which means that it is also a field code. If you press <Alt>+<F9>, you will see that this is a REF field, just as the cross-reference in Figure 52.

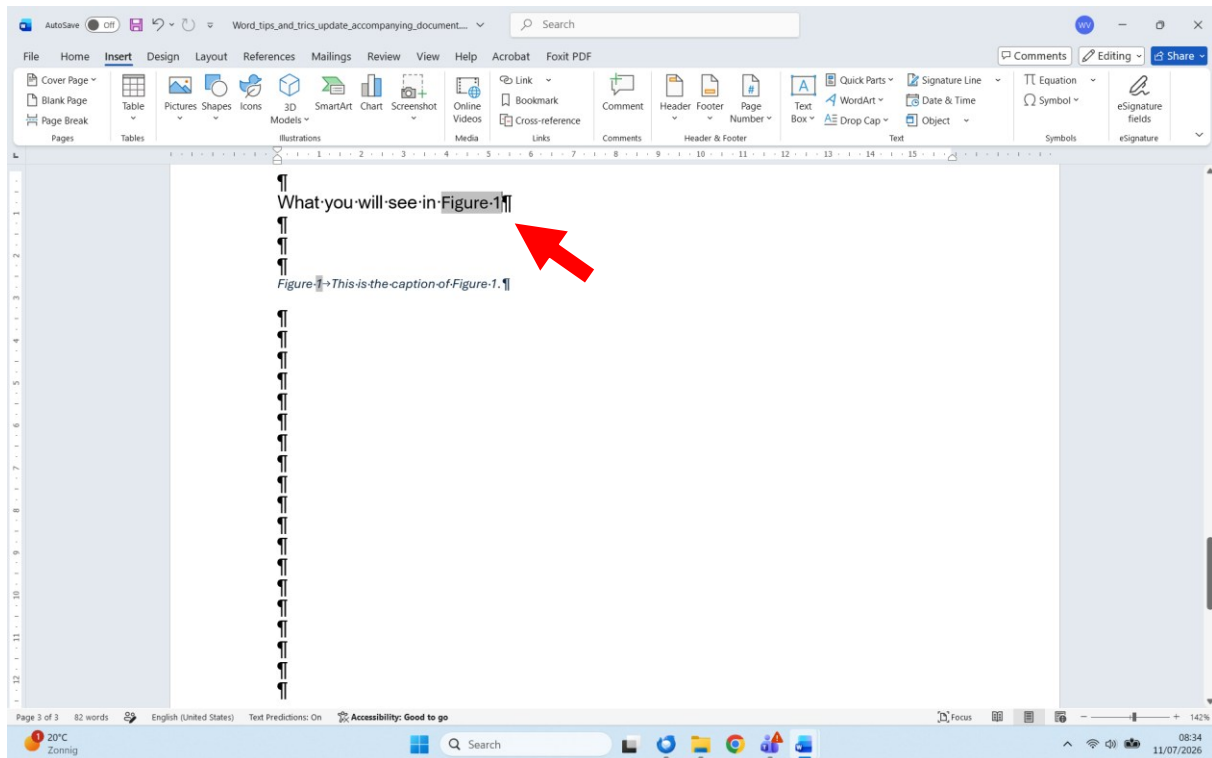


Figure 59 Result of inserting a cross-reference to *Figure 1*.

Why would you use now these captions and cross-references ? If you insert new figures in front of this *Figure 1*, the numbering of the figures and the cross-referencing will be updated automatically. For example, insert two new *Figure* captions in front of the existing caption *Figure 1*. You will see that the number of the original *Figure 1* is automatically updated to number 3. You would probably expect that also the cross-reference in the text is automatically updated, but that is again not the case. The result looks like Figure 60, and the text still refers to Figure 1, while this figure has already been renumbered to Figure 3. The reason is that not all field codes are automatically updated all the time. So to update the cross-reference as well, you have to select the whole document (<Ctrl>+<a>) and update all fields (<F9>). Now you will see that the cross-reference text changes from *Figure 1* to *Figure 3*.

If you want to remember only one tip from this chapter, choose this tip and apply at least this automatic use of captions and cross-references for your figures and tables. This feature saves you a lot of time if you start cutting and pasting figures to different locations in your text.

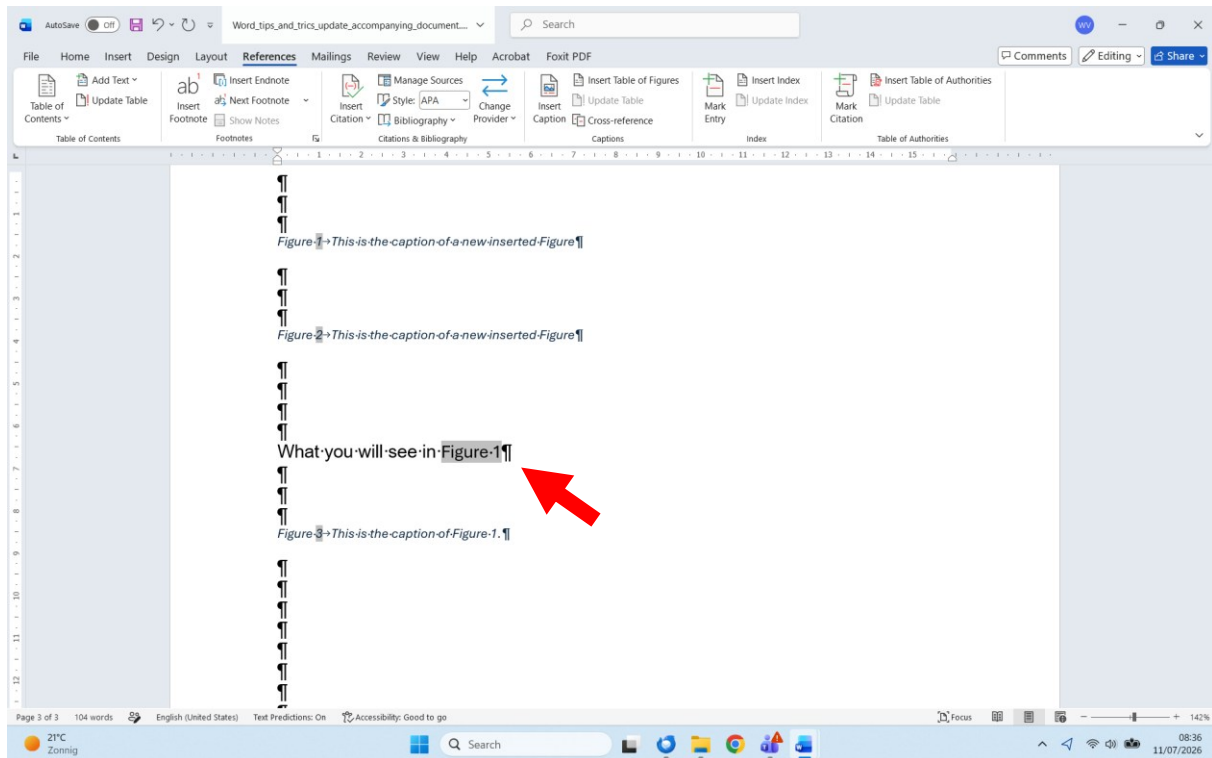


Figure 60 Result after inserting two new *Figure* captions in front of the original *Figure 1*.

4.2. NUMBERING FIGURES AND TABLES PER CHAPTER

You can also add the chapter number to the figure captions, so that figures do not number like *Figure 1*, *Figure 2*,..., but rather like *Figure 1.1*, *Figure 1.2*, etc.

You could do that manually, by defining the figure label (see Figure 56) as “*Figure 1.*” instead of “*Figure*”. However, if you would then move your figures to another chapter, or your chapter gets another number, you have to change the caption of every figure.

It is far better to include the chapter number as a field code again to your figure numbers. Word does that for you, by choosing the option “*Numbering...*” in the Caption dialogue box (see Figure 61).

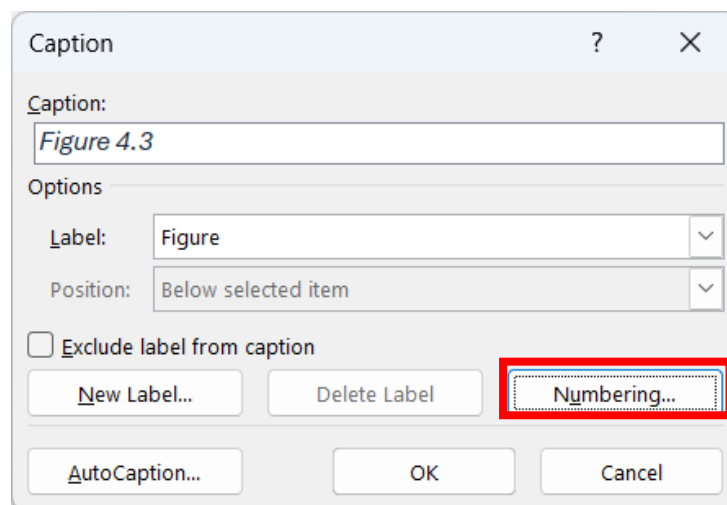


Figure 61 Choosing the option “*Numbering...*” in the Caption dialogue box.

The dialogue box in Figure 62 opens. Now you can select to include the chapter number in the figure numbering. You can also choose which paragraph style is linked to the chapter numbers (in this case the built-in paragraph style *Heading 1*).

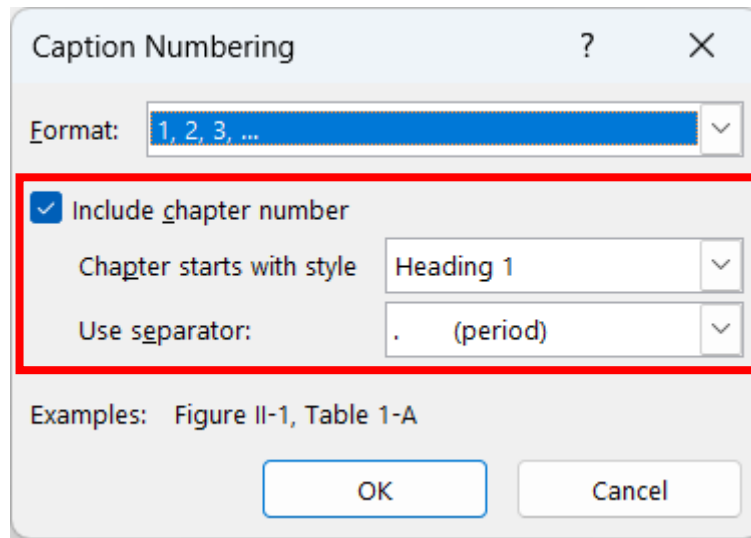


Figure 62 Including the chapter numbering in the figure caption.

Now figures are numbered with the chapter number. If I move the figures to chapter 4, the figures get renumbered as *Figure 4.1*, *Figure 4.2*,..., as shown in Figure 63.

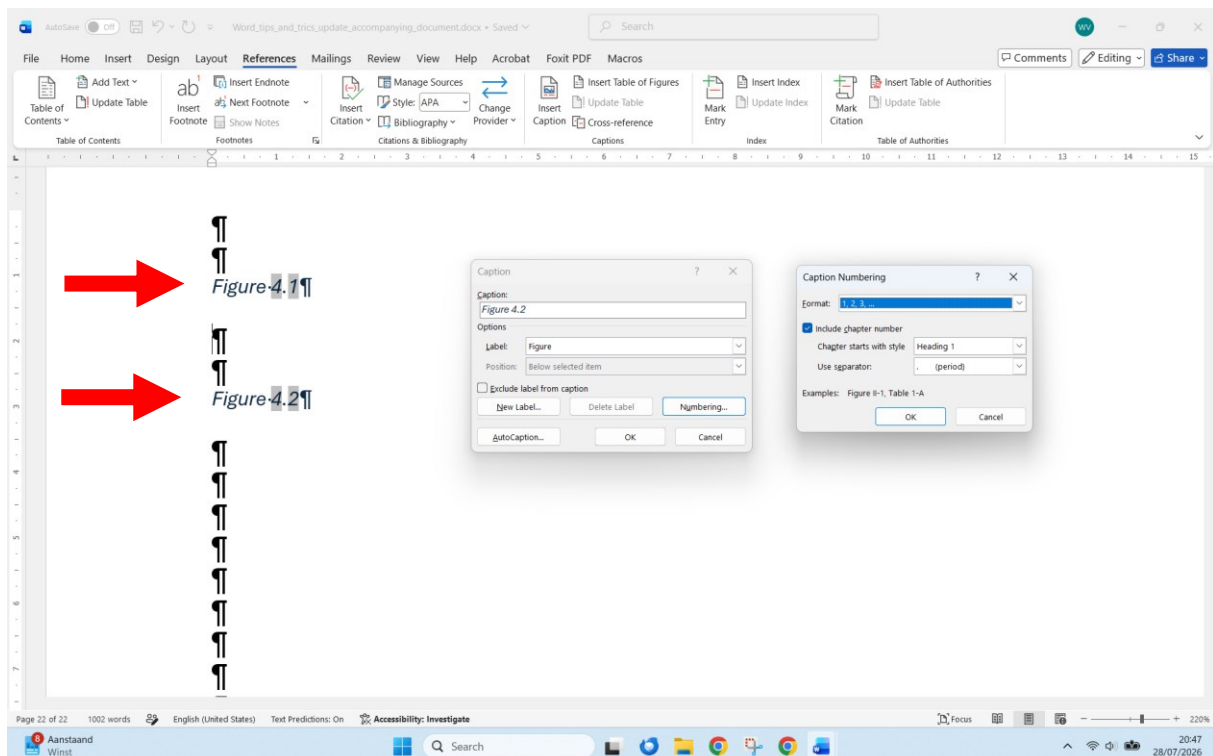


Figure 63 Automatic updating of the figure numbers in another chapter.

Once again, the strength of this versatile numbering comes from the underlying field codes. If you press <Alt> + <F9> again, you see in Figure 64 the underlying field codes STYLEREF (to refer to the built-in paragraph style *Heading 1*) and SEQ (for the numbering of the figures).

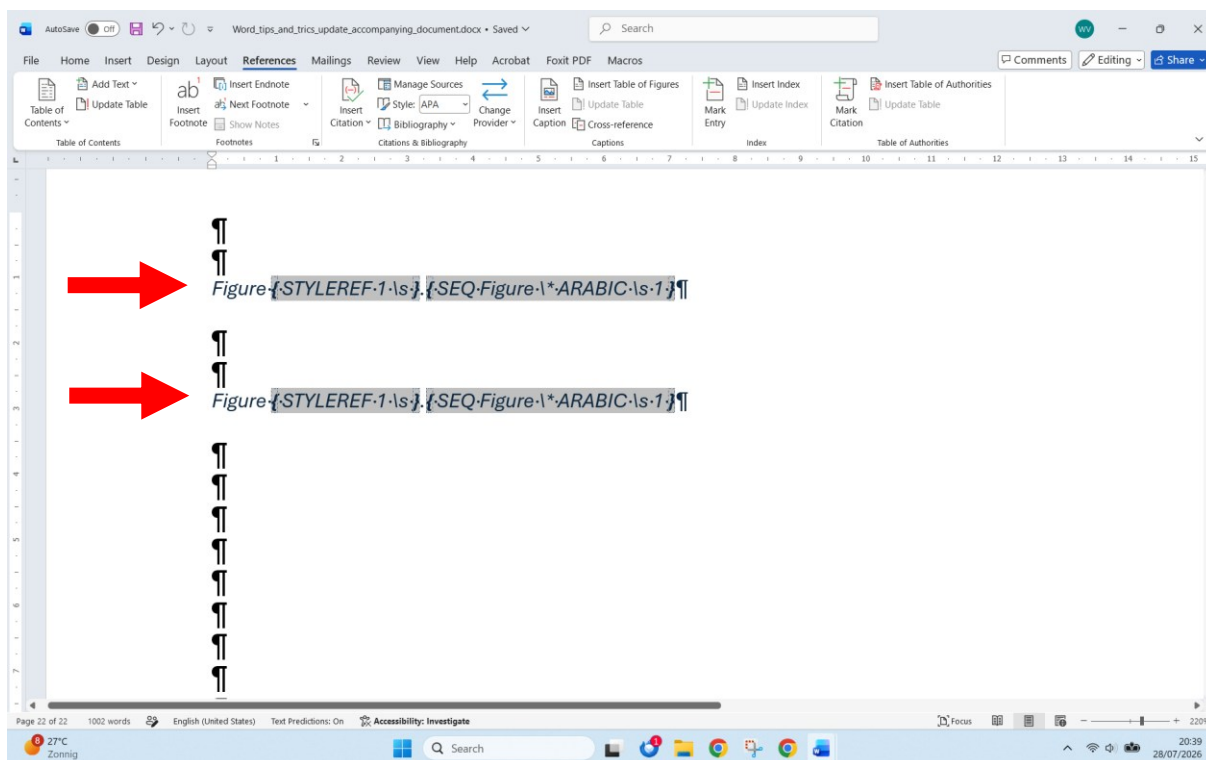


Figure 64 Underlying field codes for the figure numbering with the chapter number included.

There is an important caveat here though. In Figure 62, you can select to include the chapter number, and indicate that the chapter starts with style “*Heading 1*”. If you click on that drop-down list, you can also link the chapter number to the style *Heading 2*, *Heading 3*, etc., but you do not have the freedom to select a user-defined chapter style. This is a limitation of the SEQ field code. The switch \s 1 of the SEQ field code in Figure 64 makes it reset to 1 every time it encounters a text with built-in paragraph style “*Heading 1*”, but it cannot take as an argument a user-defined paragraph style like STYLEREF can do.

So if you use your own heading styles for chapter, subchapter, section, subsection, etc., you cannot include the chapter number into the figure numbers. Actually we already knew this from the chapter about field codes, where we already came across this limitation of the SEQ field codes for numbering equations per chapter.

There are two work-arounds in that case:

- Although less versatile, you manually add the chapter number to the figure label. So the figure label is then “Figure 1.” for Chapter 1, “Figure 2.” for Chapter 2, etc. As mentioned before, if you copy then figures from one chapter to another, you have to manually change the chapter number in the Figure label.
- A more elegant trick is to insert on the first line of a new chapter a small dummy text with built-in paragraph style “*Heading 1*”, and give it a white font, so that you do not see the text. This is a trick we already used with the SEQ field codes and that we will still use a few times in this tutorial. It is a very handy trick to initialize field codes.

Let’s retake the example of Figure 43, where we wanted to restart numbering of the equations for each chapter, but use a user-defined paragraph style for the chapter titles. Because the SEQ field code could only work with the built-in heading styles, we inserted one line of text with the built-in paragraph style *Heading 1*, and assigned a white font colour to it.

If we now want to include the chapter number in the figure numbers as well, but also use a user-defined paragraph style for the chapter titles, we apply again the same trick.

Figure 65 shows the same document page as in Figure 43, but now, a figure caption is inserted which includes the chapter number. You see that it correctly numbers the figure as Figure 6.1., because it detects a paragraph before with the built-in paragraph style *Heading 1*.

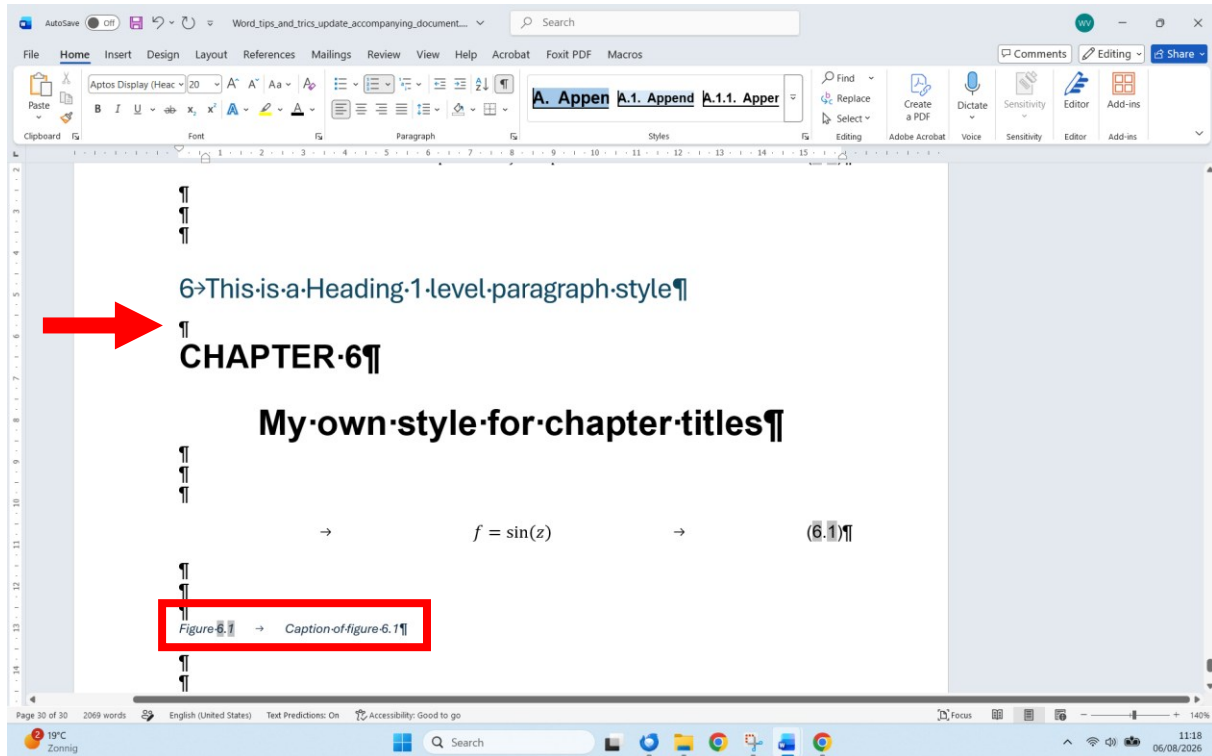


Figure 65 Insertion of a figure caption that includes the chapter number from built-in *Heading 1* paragraph style.

If we now assign a white colour to that line of text, the result looks like Figure 66.

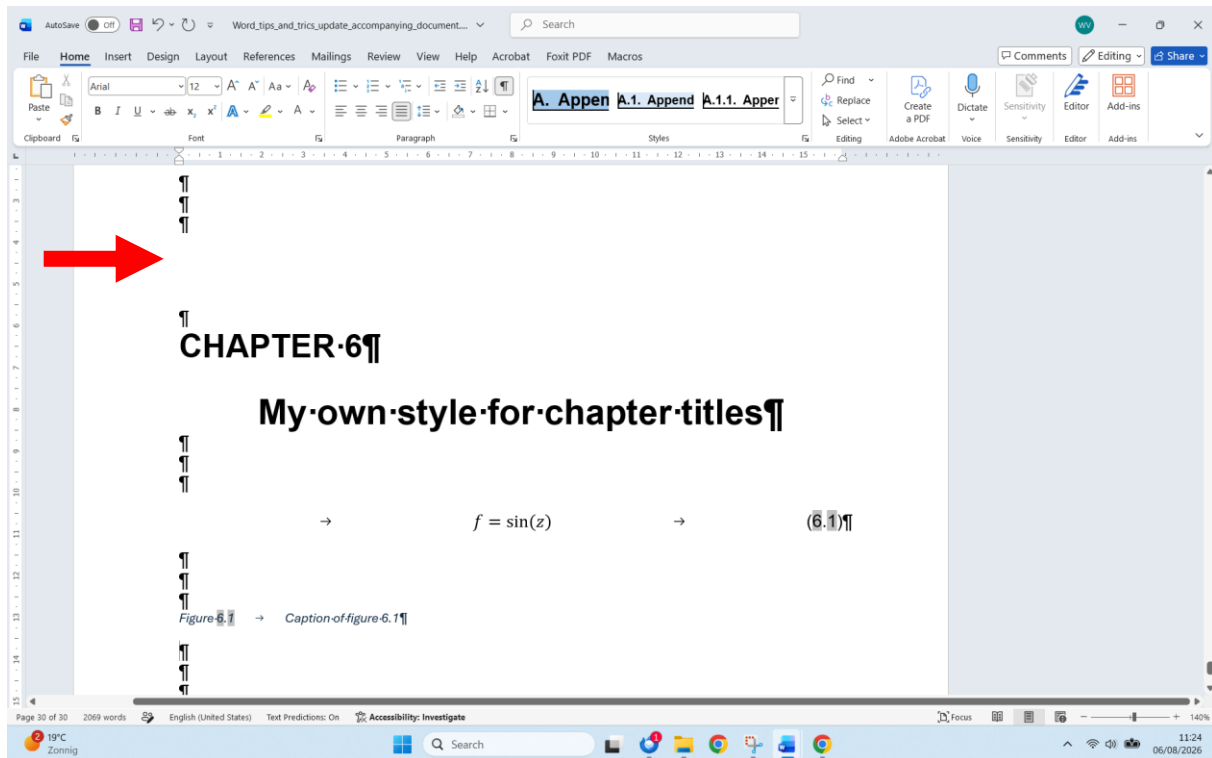


Figure 66 Figure caption that includes the chapter number, with white font colour for the built-in paragraph style *Heading 1*.

4.3. EQUATION NUMBERING WITH CAPTIONS

I have deliberately first introduced field codes in Chapter 3, and then only figure captions in this chapter, because in this way, you can understand that figure captions are in essence nothing else than field codes.

However, now that you know both field codes and figure captions, you can argue that the numbering of equations can also be done simpler by using a caption with the label “Equation”. That is true to some extent. If you select in the menu *References > Insert Caption*, you can create a new label “Equation” or “Eq.”, as shown in Figure 67. From then on, you can insert automatically numbered captions that read like “Equation 1”, “Equation 2”, etc.

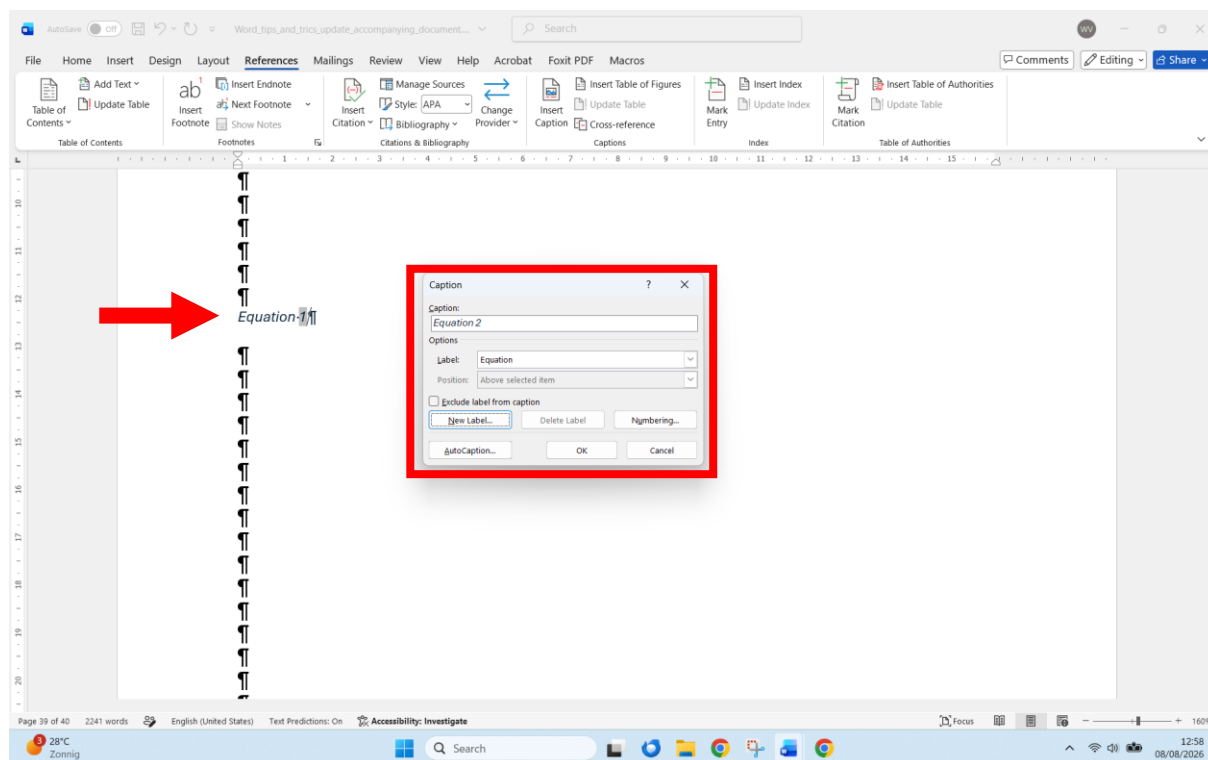


Figure 67 Defining a new caption label for Equations.

Once you have defined this new caption label, the Equations will also pop-up as a new class of cross-references in the menu *References* > *Cross-references*, as shown in Figure 68. So you can also include cross-references to these equations.

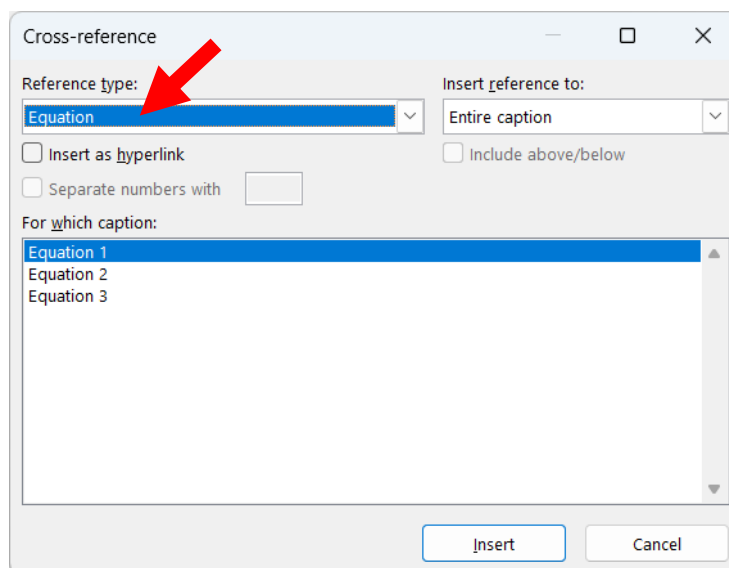


Figure 68 Cross-references to the items with caption label “Equation”.

If we retake our 3 equations from the example in Chapter 3 (see Figure 38), you could create an alternative numbering of the 3 equations in the way as shown in Figure 69.

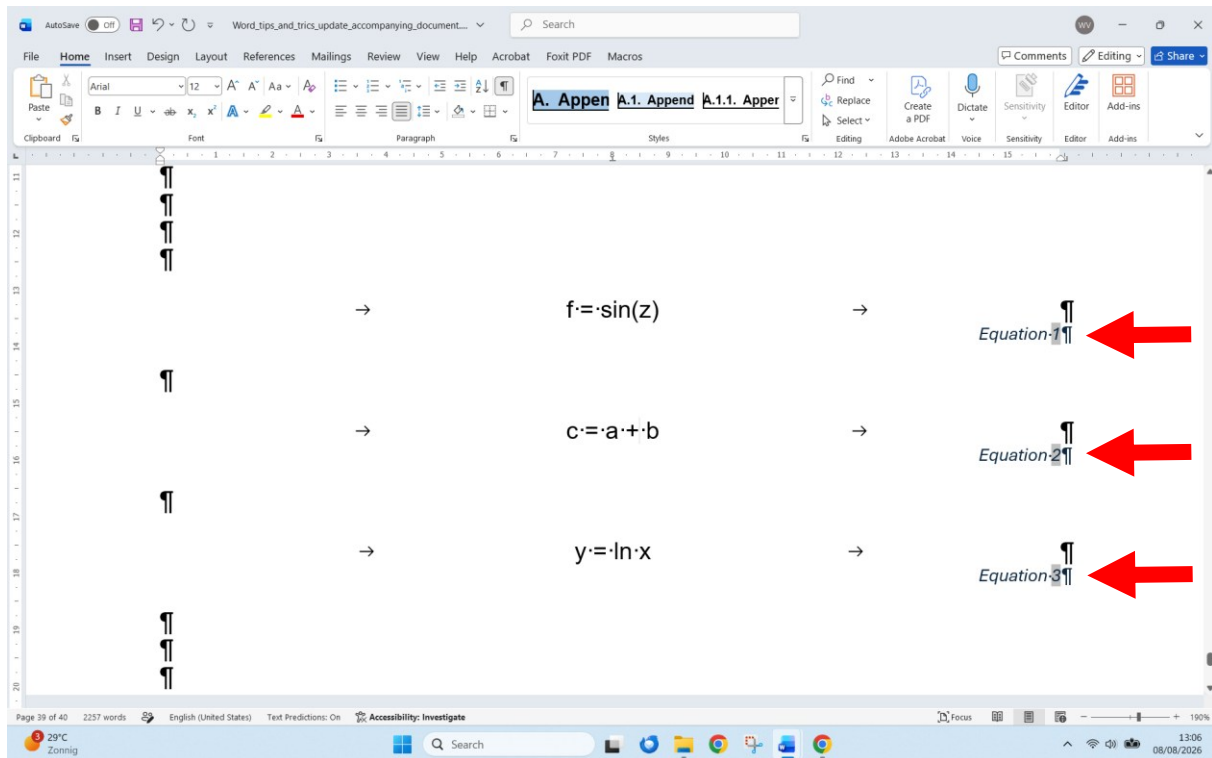


Figure 69 Numbering of equations with the Caption label “Equation”.

Although this approach of numbering equations uses standard built-in tools of Word and seems easier, it also has several drawbacks:

- You need to add a caption label “Equation” or “Eq.”, and cannot only use the equation number,
- The labels “Equation 1”, “Equation 2”, etc. get the paragraph style “Caption”, so you have to be careful to put them on the same line with the actual equation (you can do that with a table with two columns),
- In the list of cross-references for the Equations, you see Equation 1, Equation 2, Equation 3,..., but if you have many equations, these labels are not self-explaining which equation this is and inserting cross-references becomes cumbersome.

That is why I still prefer the approach with SEQ field codes and bookmark names assigned to the equation numbers, as explained in Chapter 3. And as mentioned before, once we have introduced macros in a later chapter, we can automate all these steps and have a single-click approach for numbering equations and cross-referencing them.

But nevertheless, now that we know how to properly number figures and tables and add cross-references to them, we can have a look at the inserting and formatting of figures in the next chapter.

5

Inserting and formatting figures

Figures are on the most common frustrations with Word. Figures can jump around when paragraphs are moved or cut and pasted. Also arranging several pictures in one figure can cause a headache.

In this chapter, we learn how to properly insert and format figures.

For this chapter, it is important to not only visualize all formatting marks and field codes, but also the so-called object anchors (see also Figure 1).

5.1. INSERTING FIGURES

In my modest opinion, most problems with figures only appear when you insert them as floating objects. If you paste a figure into a Word document, it can be pasted as a floating object. When a figure is a floating object, an “object anchor” is linked to it. And this anchor on its turn is “anchored” to a certain paragraph, as illustrated in Figure 70.

In the case of Figure 70, you can notice that the object anchor is anchored to the second line below the text. If you now insert extra paragraphs below the existing lines of text, the figure is drifting away from the original paragraph where it was supposed to be linked to.

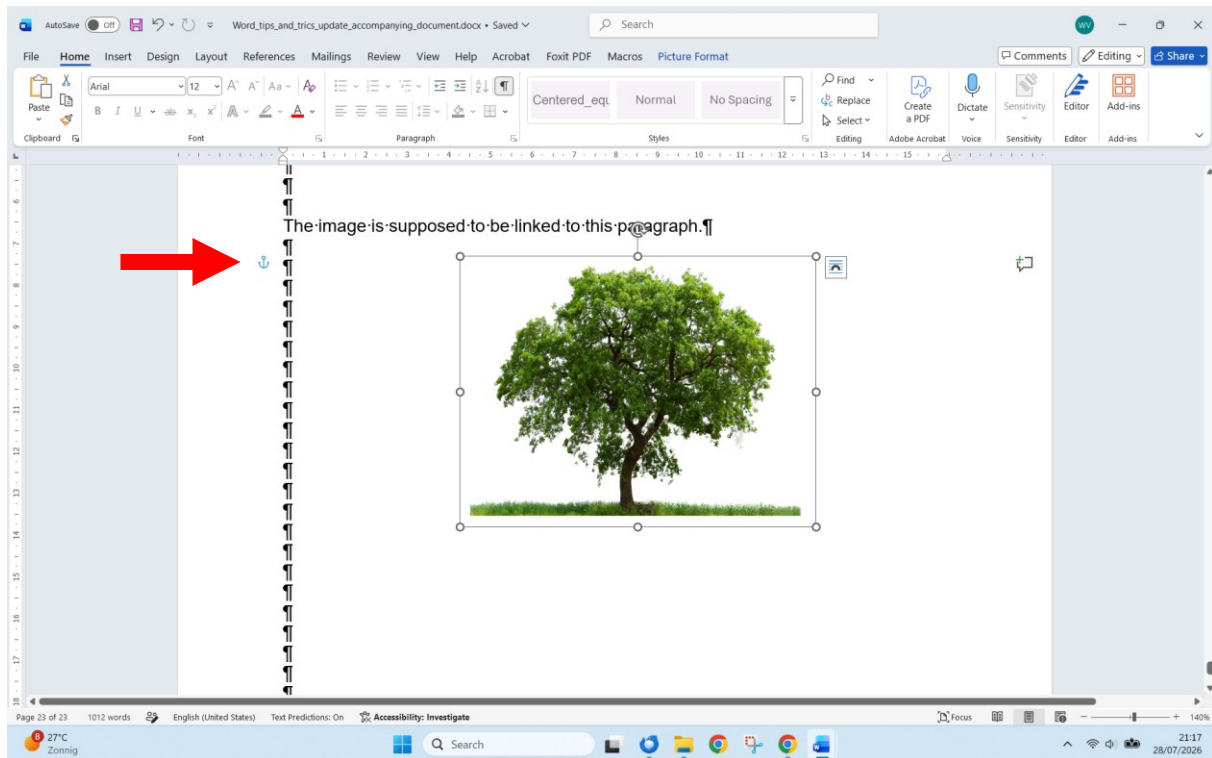


Figure 70 Floating figure with an object anchor.

Figure 71 shows what happens then, but in fact, it is still very easy to see what is going on, because you have visualized the object anchors and the paragraph formatting marks, so you can see which paragraph the figure is linked to.

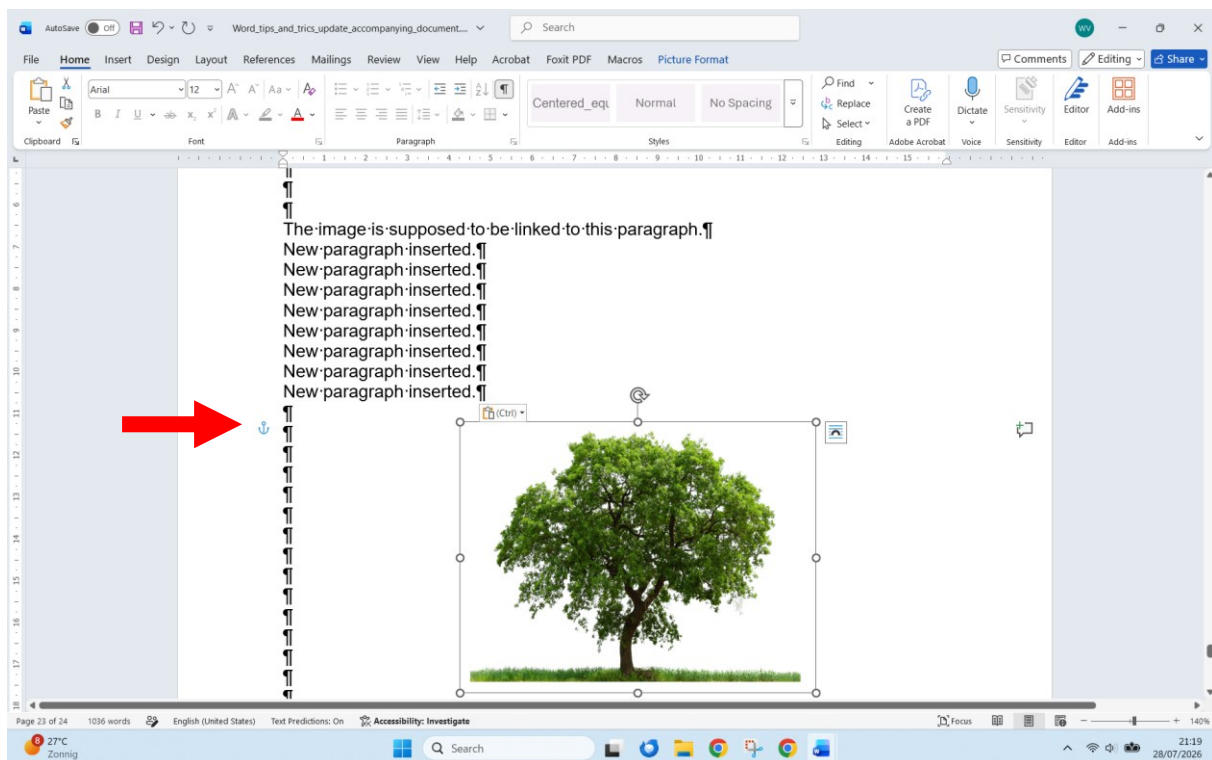


Figure 71 Separation of figure and its linking paragraph.

However, imagine that you use Word like 99% of other people do, and you hide again the formatting marks, object anchors, field codes, etc. Then you see the screen of Figure 72. Now it becomes much more difficult to see what is going on and the frustration starts.

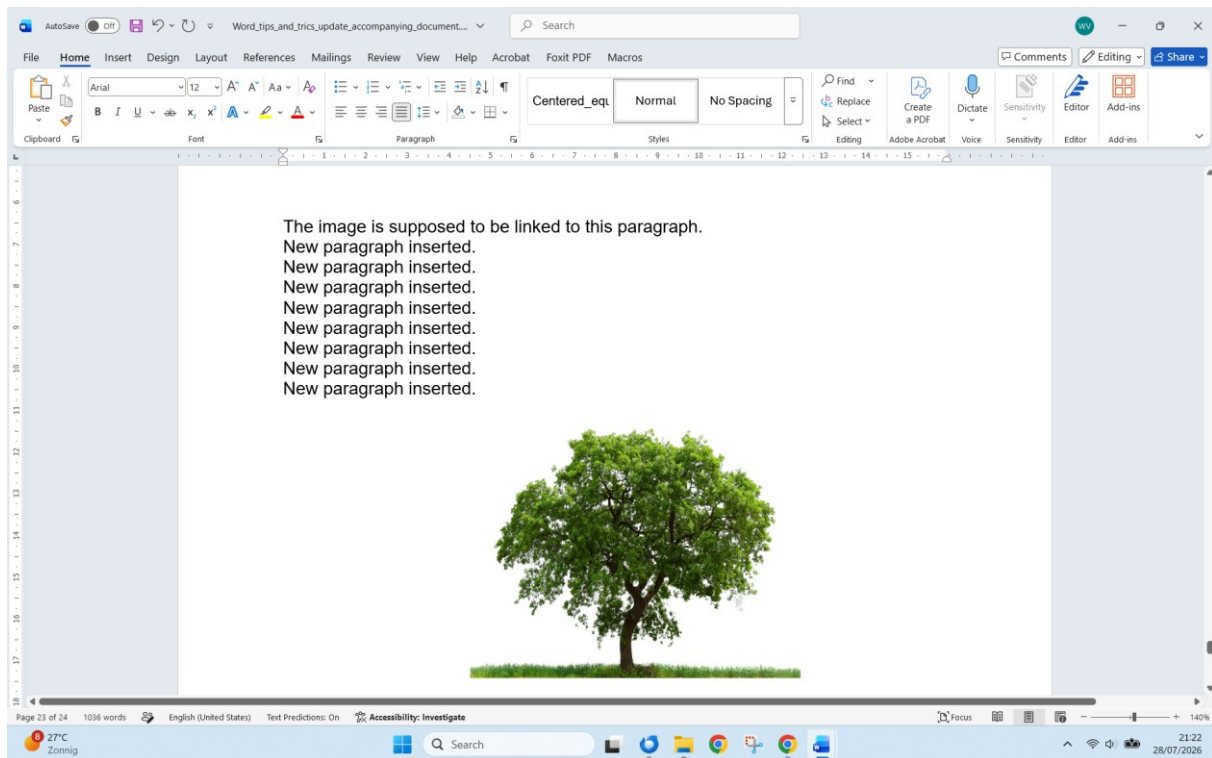


Figure 72 Same separation of figure and its linking paragraph, but without visualizing formatting marks and object anchors.

To avoid all these issues, it is far easier to insert figures “*inline with text*”. Then figures are not floating objects, but inserted as a sort of paragraph, contained between two paragraphs. To that purpose, right-click on the figure and select *Wrap Text > In Line with Text*, as shown in Figure 73.

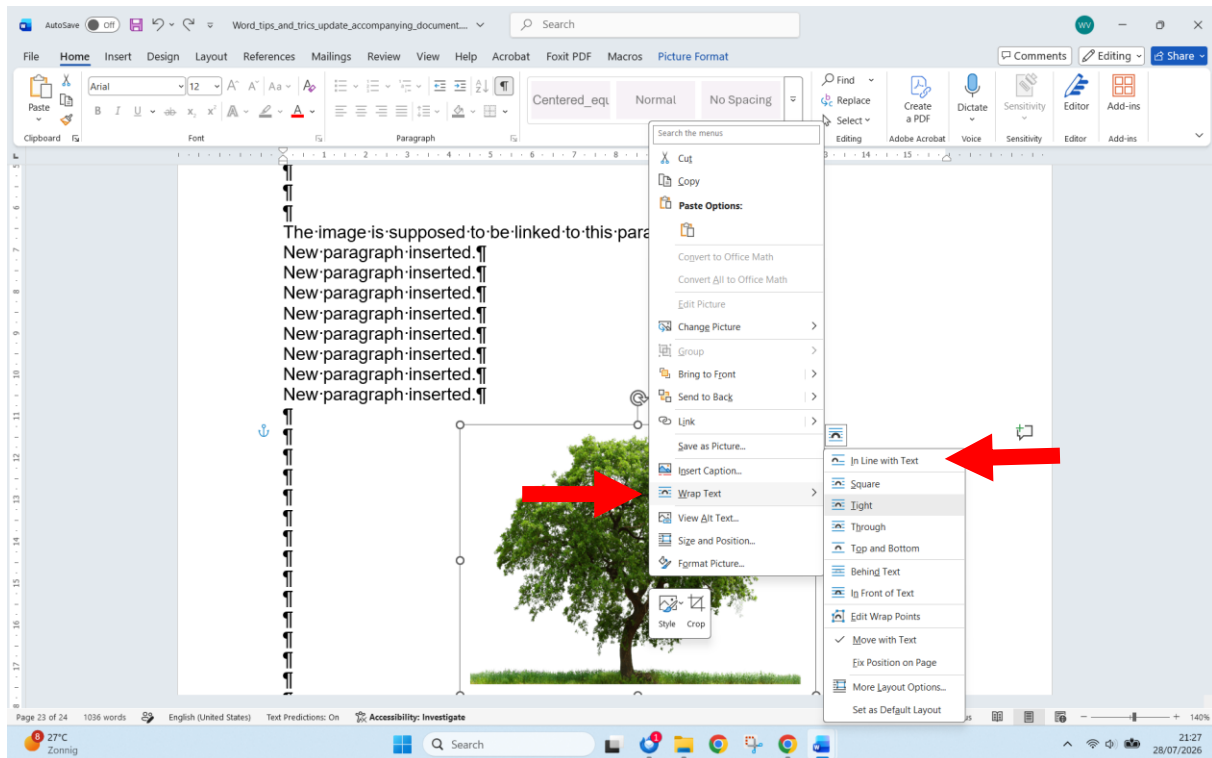


Figure 73 Change the figure insertion to “In Line with Text”.

Figure 74 shows the result of this action. You can see that all paragraph marks at the left side of the figure have disappeared, and the figure acts now as a paragraph on its own. Also the object anchor has disappeared and the figure cannot float anymore. Object anchors only appear for images with text wrapping styles other than “*In Line with Text*” (such as *Square* or *Tight*).

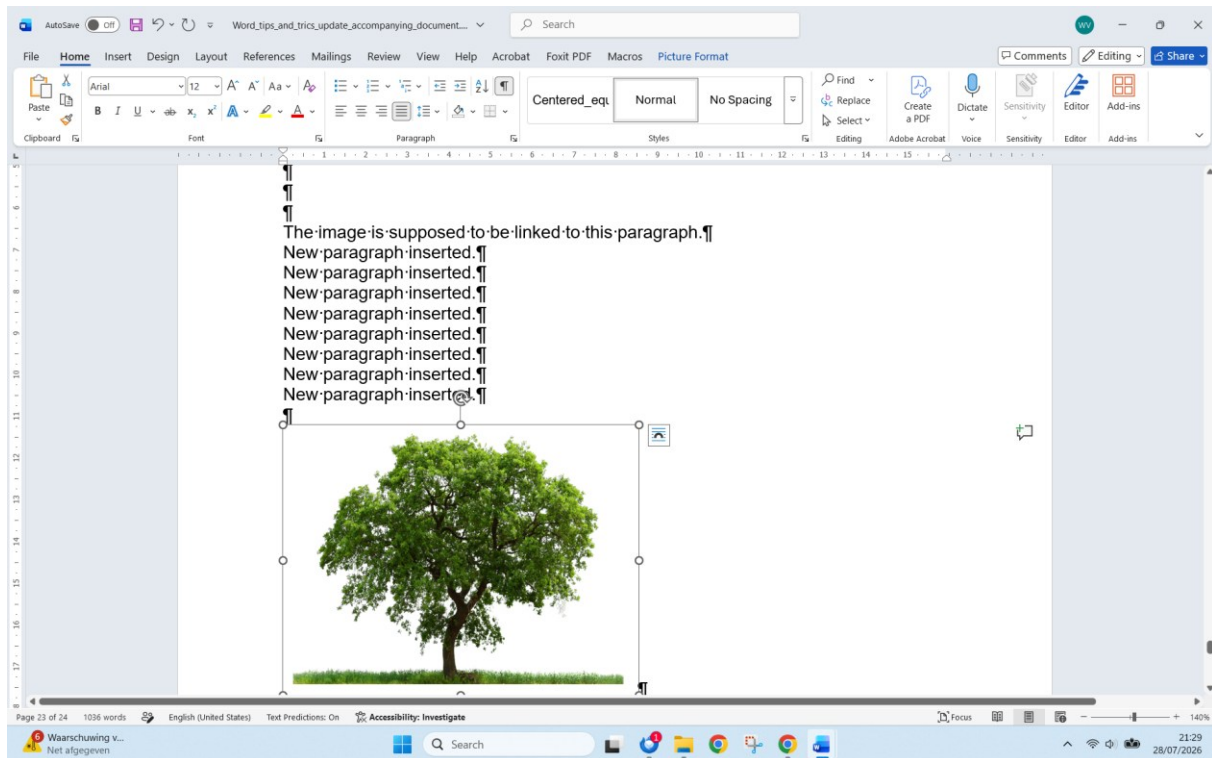


Figure 74 Result of the figure inserted “In Line with Text”.

One side-effect of the figure acting as a sort of paragraph, is that it also obeys the paragraph style in the sense that all paragraph formatting marks apply it. One very common side effect of that is that your figures can sometimes seem to disappear when you insert them as “*In Line with Text*”, as illustrated in Figure 75.

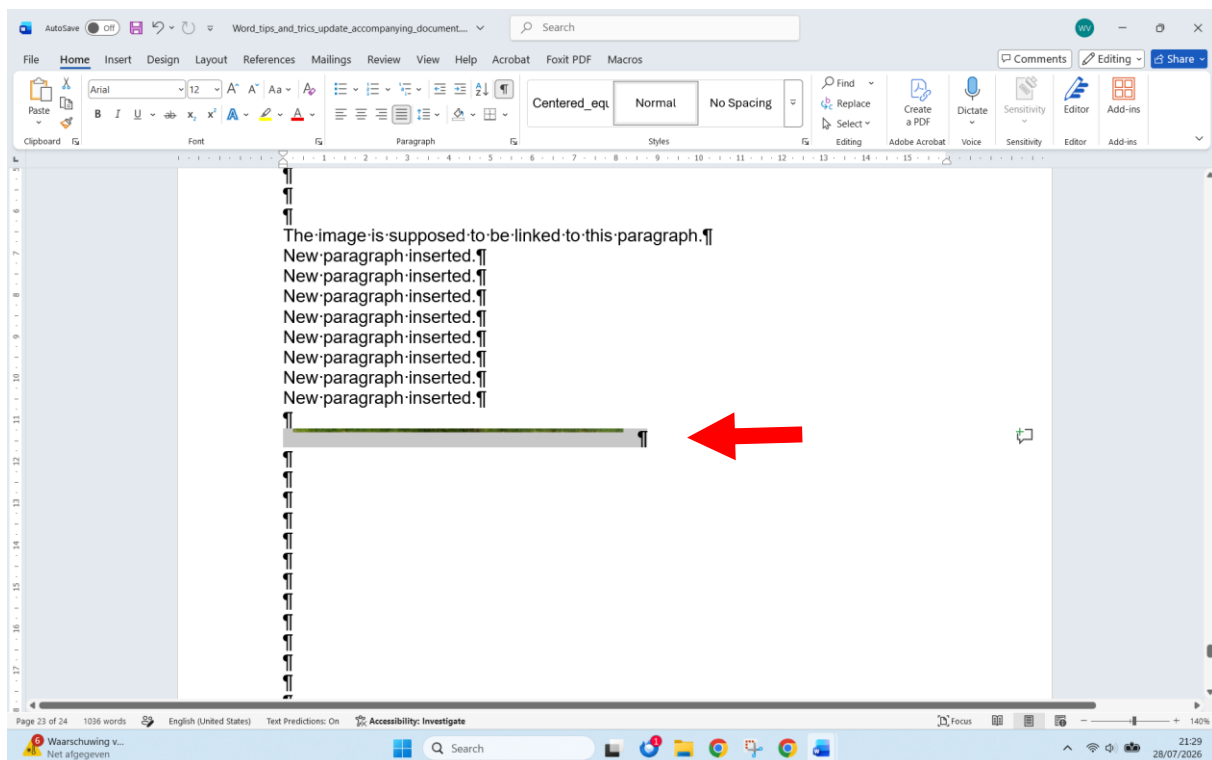


Figure 75 Result of the figure insertion when the line spacing is enforced to an exact value.

This happens when the paragraph style in your document has a line spacing that is enforced to an exact value, as shown in Figure 76.

In Figure 75, you could still see the bottom of your figure, which could somehow give you a clue what is going on. However, if your figure has some white space at the bottom and you have turned off the formatting marks and object anchors, it seems like the picture has really disappeared or has not been inserted at all.

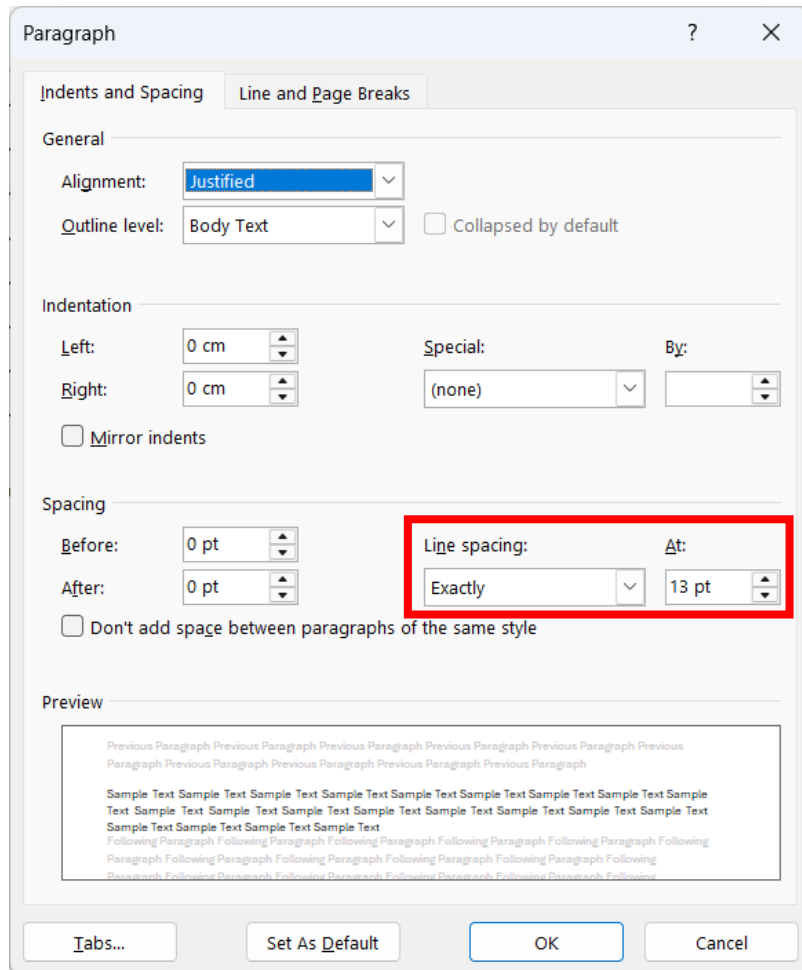


Figure 76 The figure is inserted in a paragraph with an exact value of line spacing.

Finally, you can define in the Word options what is the default manner to insert a figure. In my case, it is set to “*In line with text*”, to avoid all these issues with floating figures. You select **File** > **Options...** to open the *Options* dialogue box, as shown in Figure 77. Select the *Advanced* tab and check the option “*In line with text*” for the action “*Insert/paste pictures as:*”.

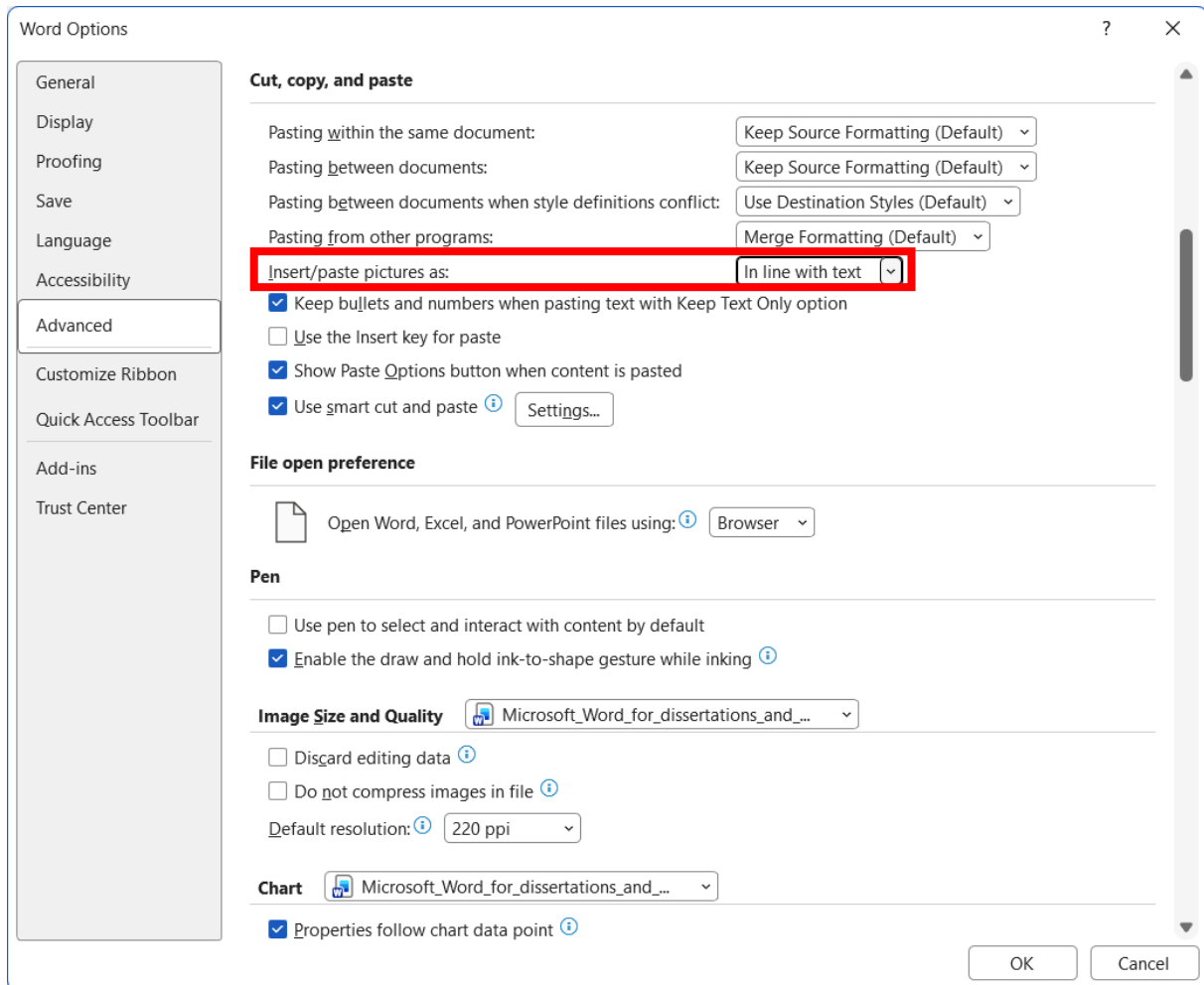


Figure 77 Setting the default insertion method of figures to “*In line with text*”.

5.2. FORMATTING FIGURES

When combining several pictures or graphs in one figure, tables are very useful to properly align them.

For example, you want to show a sequence of four images from left to right in one figure. First you define a table with two rows and four columns (*Insert > Table...*). Next you insert the four images from left to right in the first row, again not as floating objects, but “*In line with text*”. In the second row, you center the labels (a), (b), (c) and (d).

Finally, you open the *Borders and Shading* dialogue box with the command *Borders* in the *Table Design* toolbar (Figure 78).

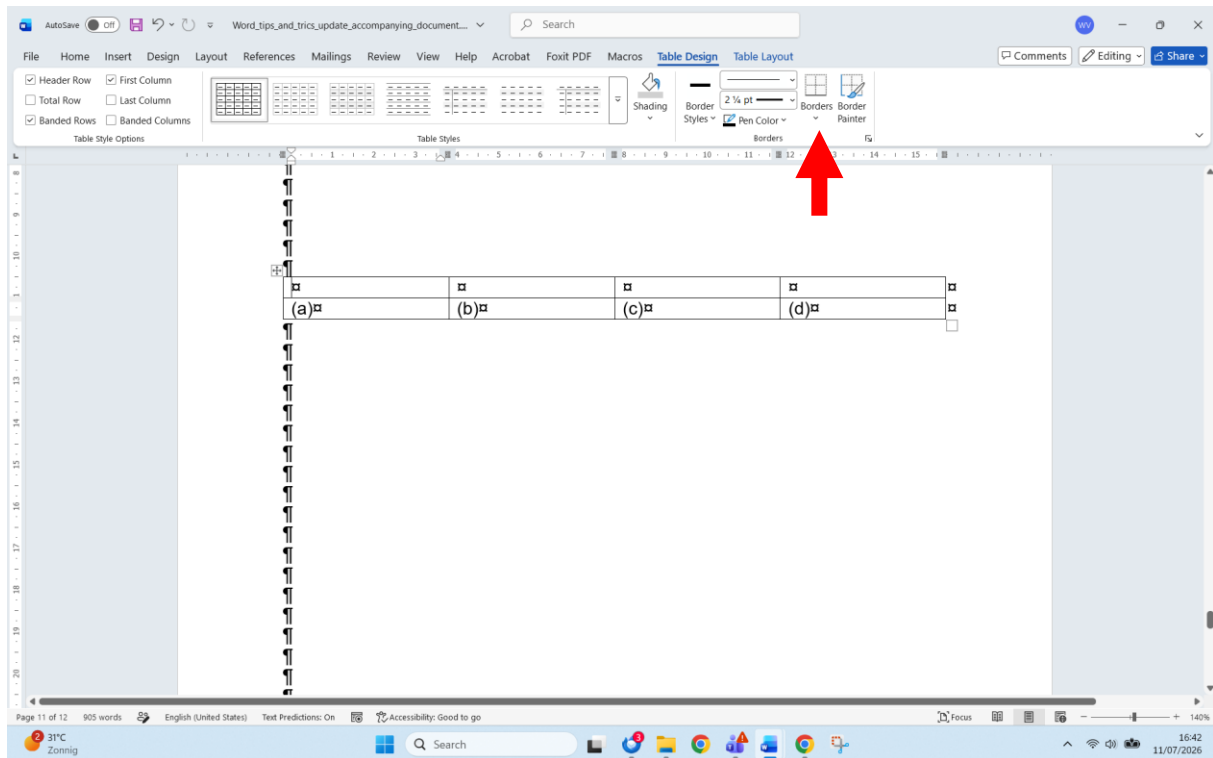


Figure 78 Selecting the command **Borders** in the **Table Design** toolbar.

Under the *Borders* tab, you select the setting “No Border”, as shown in Figure 79. You will still see light grey borders of the table, but these will not be printed. These borders are only there to indicate the borders of rows and columns while editing the table.

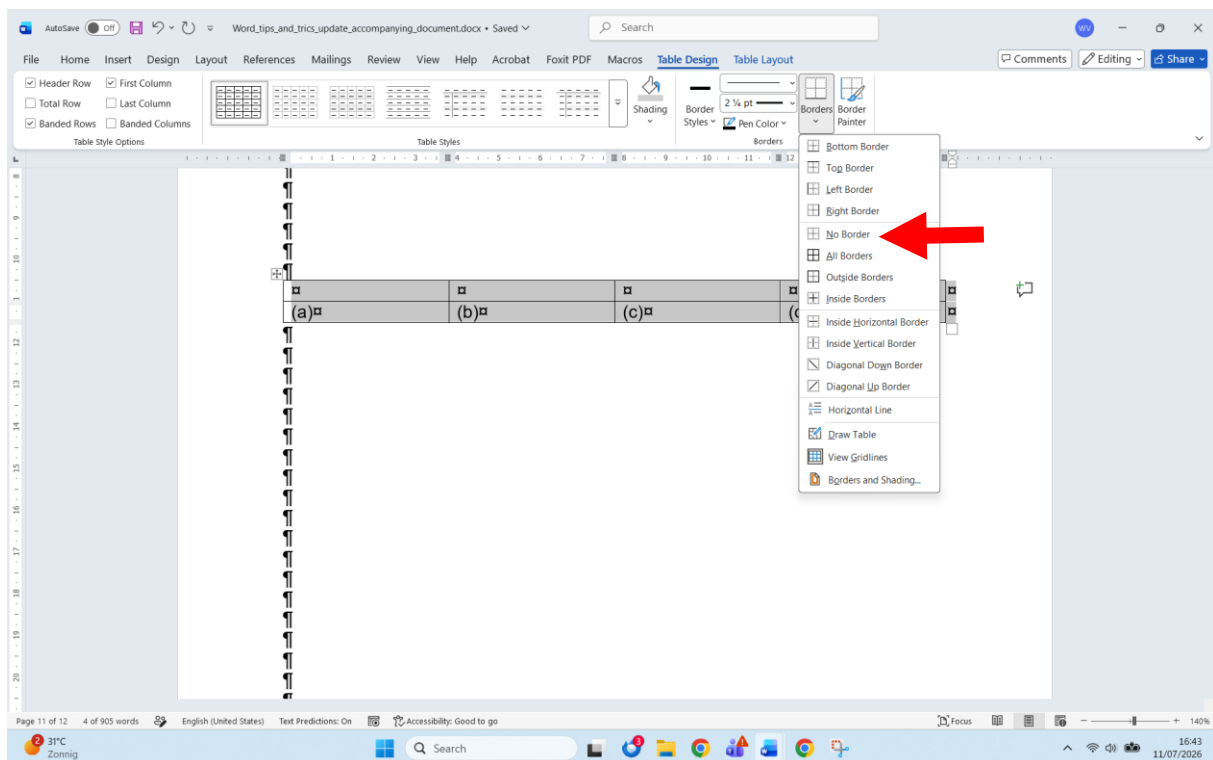


Figure 79 Making the borders of the table invisible.

In that way, the table borders will be invisible when you print the document. The result looks like Figure 80.

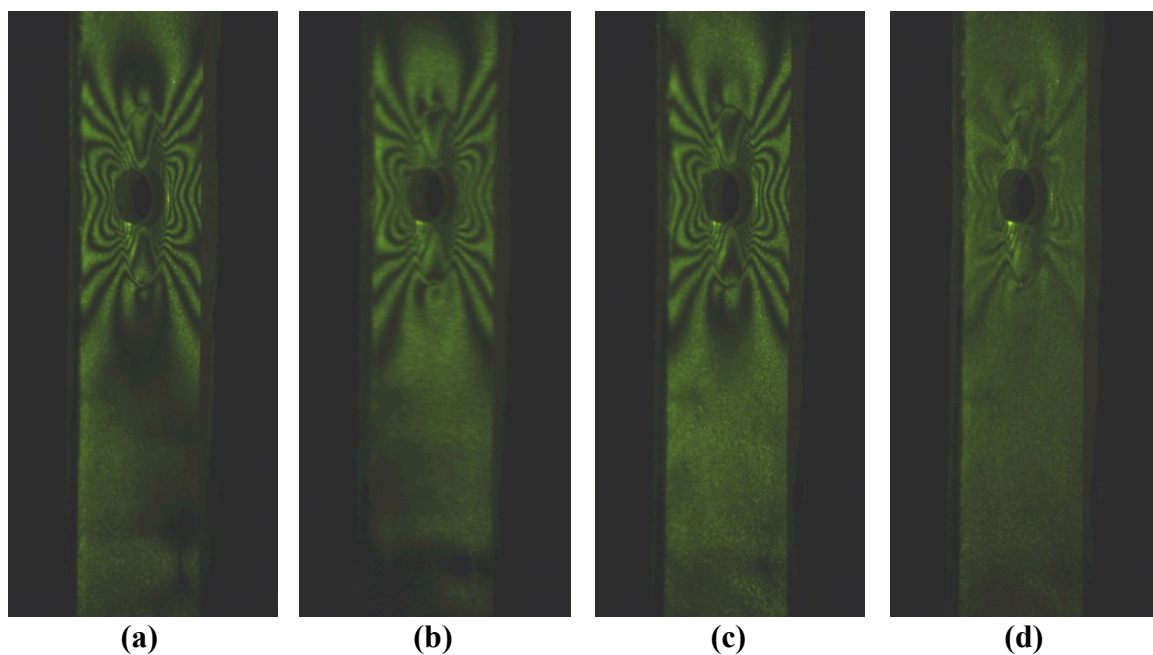


Figure 80 Example of aligning different pictures with a table.

You can also extend this formatting trick to the figure caption as well. If you want the figure and its caption to never separate, you define an extra row in the table, where the figure caption sits in. In this way, the layout of the figure is very stable.

6

Using page breaks and section breaks

Page breaks and section breaks are easy tools to enforce a certain page layout. A page break simply enforces Word to start a new page, while section breaks are much more versatile and allow to fundamentally change the page layout (page orientation, headers, footers, etc.) from one section to another.

A page break enforces the text to continue on the next page. You can easily insert a page break with the shortcut key <Ctrl>+<Enter> or by clicking *Insert* > *Page break*.

An example is shown in Figure 81. Here again, the benefit of showing all formatting marks is demonstrated (see Figure 1). The formatting mark *Page break* is clearly shown, but if you choose to hide all formatting marks, it can be very difficult to understand why these two paragraphs do not want to close up together.

A page break can be useful to force pictures to be on top of a page, or to start a new heading level on a new page.

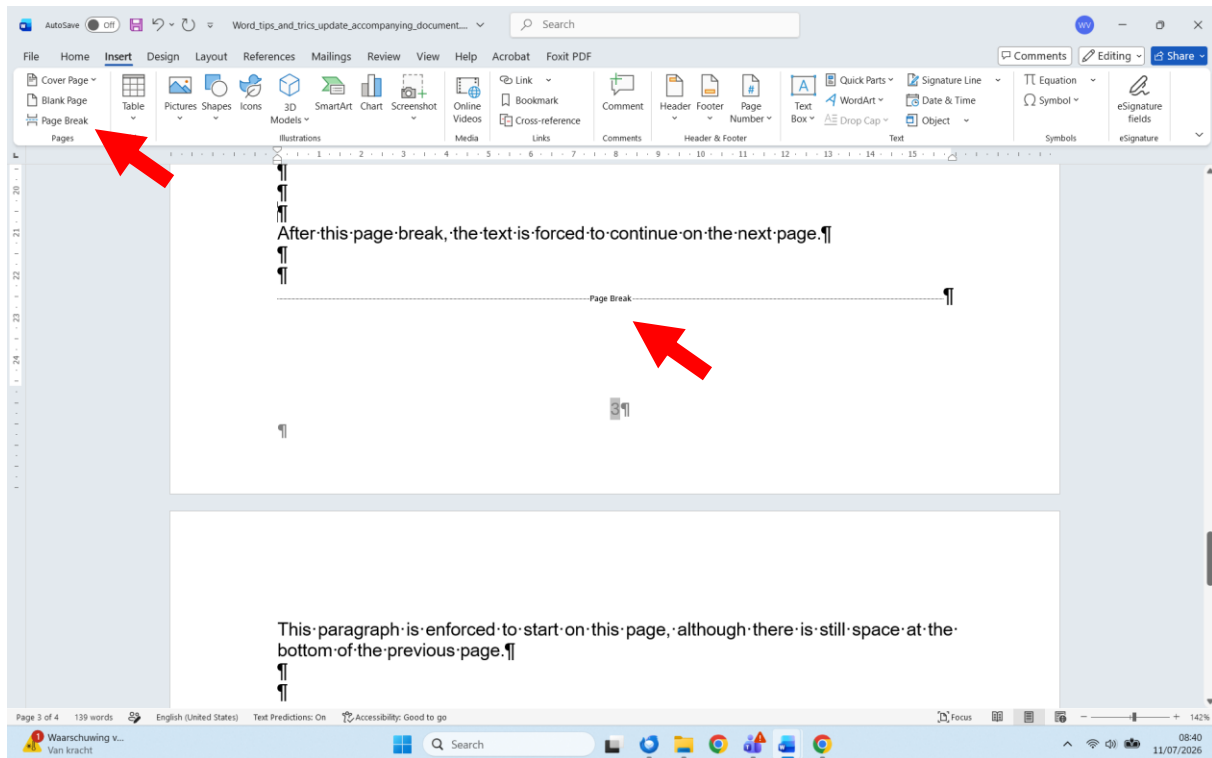


Figure 81 Inserting a page break between two pages.

A more stringent break is a section break. By inserting section breaks (*Layout* > *Breaks...* > *Section breaks* > *Continuous*), you split up your document in two or more sections. Section breaks can be inserted within a page (Continuous section break) or be forced to start on the next page (see Figure 82). You can use sections to vary the layout of a document within a page or between pages. Just insert section breaks to divide the document into sections, and then format each section the way you want.

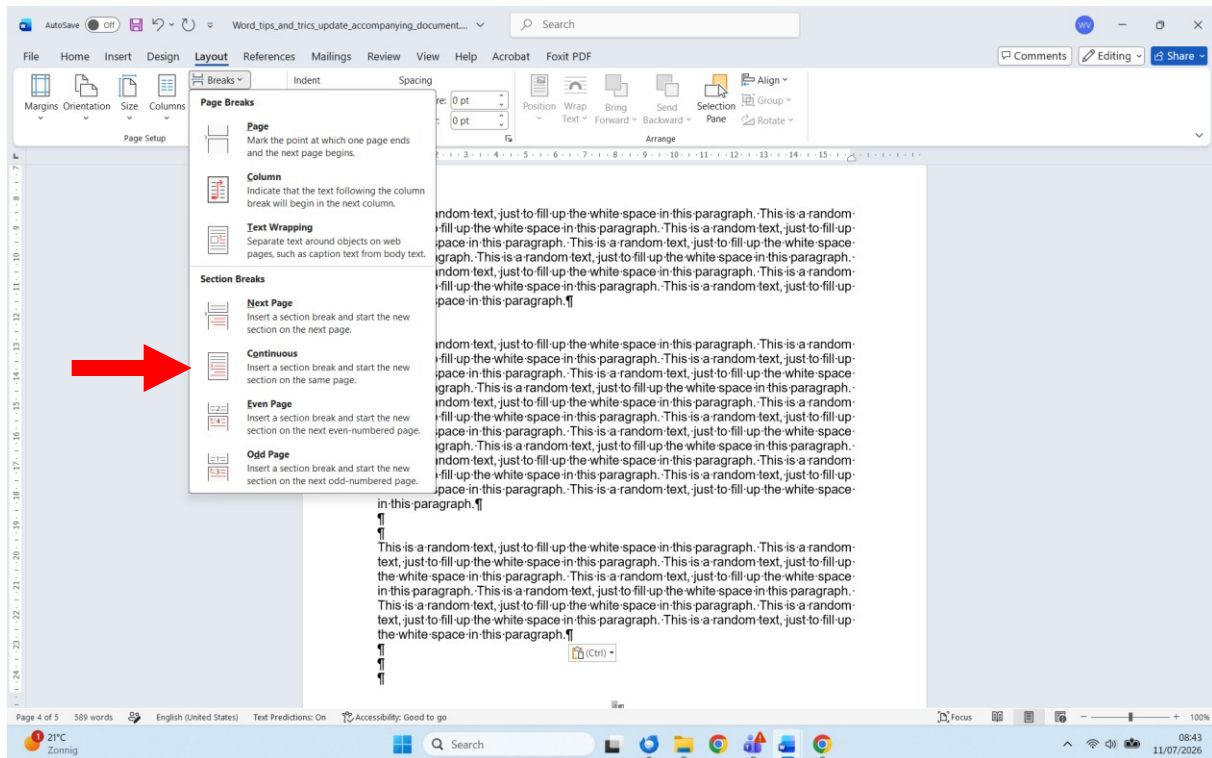


Figure 82 Inserting continuous section breaks.

For example, you want to format a section as a single column for the introduction of a report, and then format the following section as two columns for the report's body text. For that, you first add a continuous section break before and after the text that has to appear as two columns, as illustrated in Figure 83.

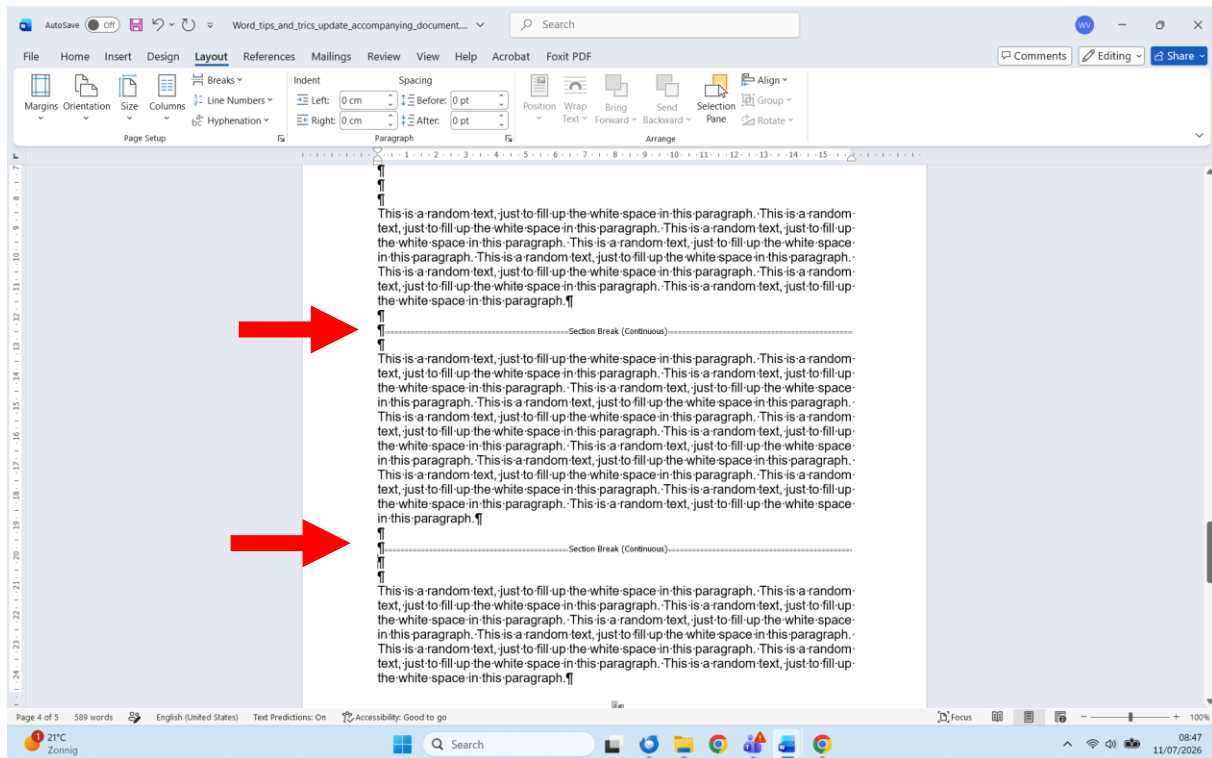


Figure 83 Inserting continuous section breaks before and after the central paragraph.

After modifying the central section into two column format (*Layout > Columns > 2 columns*), you can see the result in Figure 84. If you look carefully, you now see two continuous section breaks before and after the central paragraph in two-column format. This shows that Word is automatically inserting continuous section breaks as well to separate the text in two-column format from the text in one-column format. So in this particular case, Word automatically inserts continuous section breaks for you, and it was not necessary to manually insert them.

You can change all the following formats of a section independently from other sections:

- margins,
- paper size or orientation,
- paper source for a printer,
- page borders,
- vertical alignment,
- headers and footers,
- columns,
- page numbering,
- line numbering,
- footnotes and endnotes.

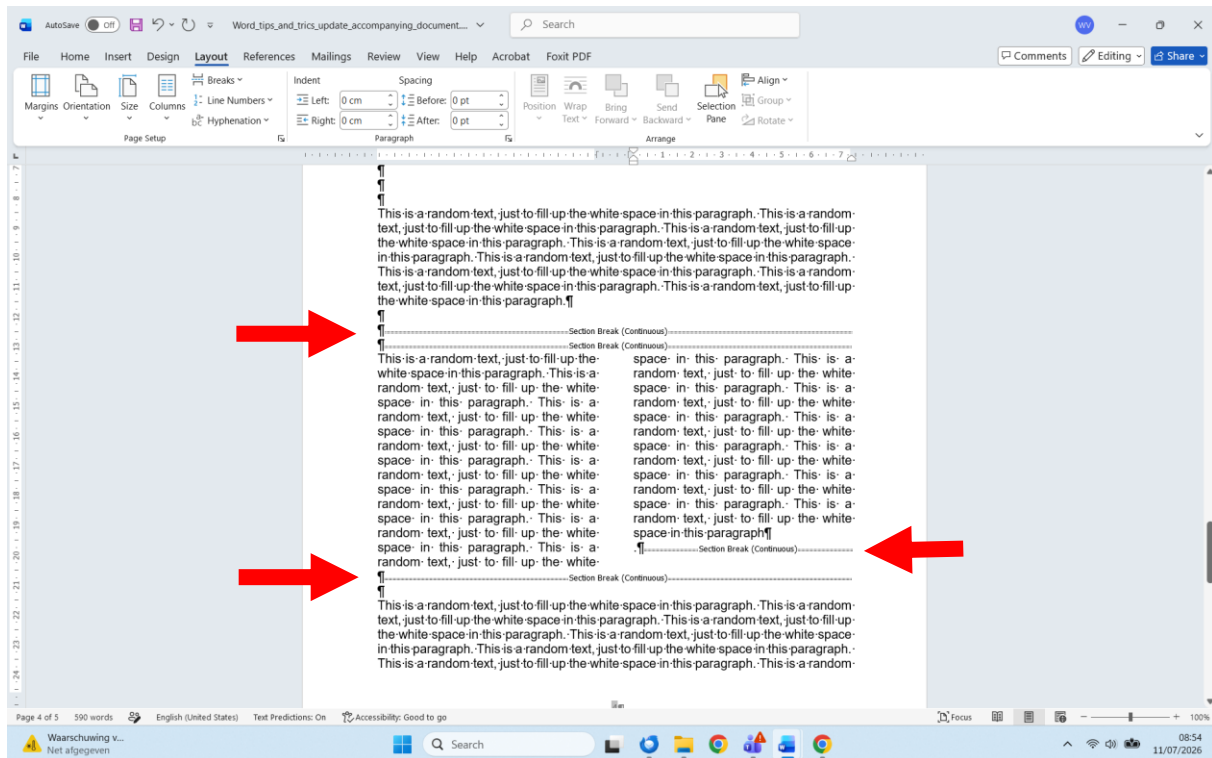


Figure 84 Inserting a continuous section break to switch from 1-column to 2-column format.

Another very useful feature of the section break is to include a page with landscape orientation between pages with portrait orientation. In a scientific manuscript or report, you can sometimes have a very wide table, that fits better on a separate page that is in landscape orientation. This can be done with a section break. To that purpose, you need to insert a section break of the type “Next page”, as illustrated in Figure 85.

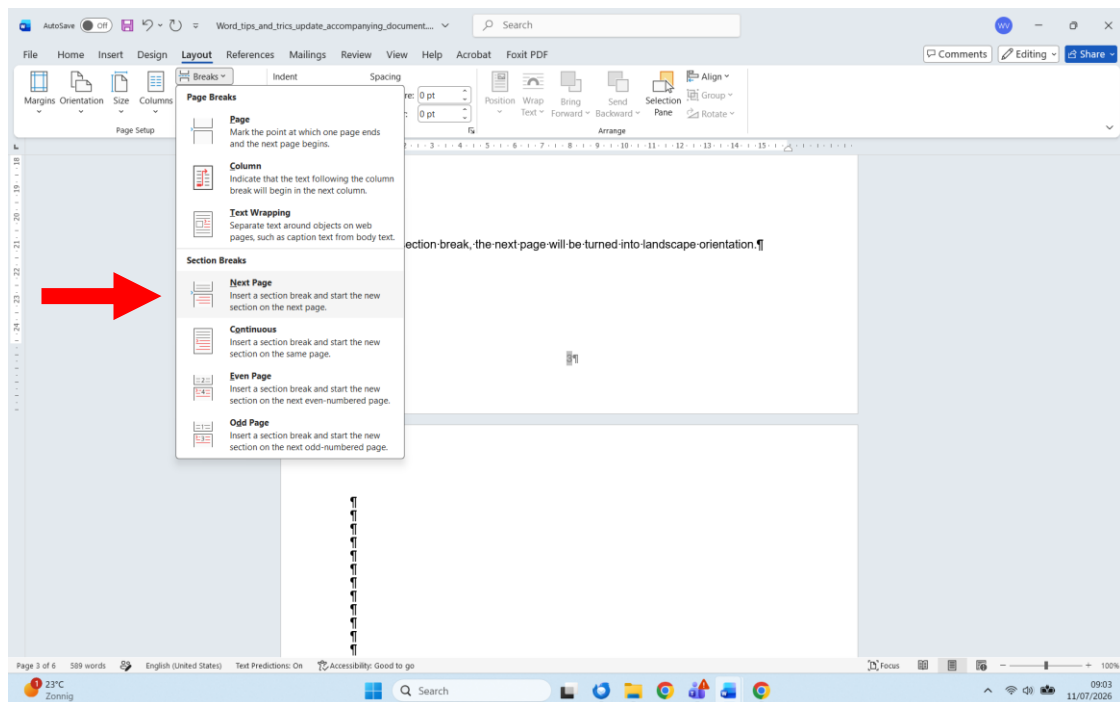


Figure 85 Inserting “Next page” section breaks.

As illustrated in Figure 86, you can now change the page orientation from *Portrait* to *Landscape* in the next section.

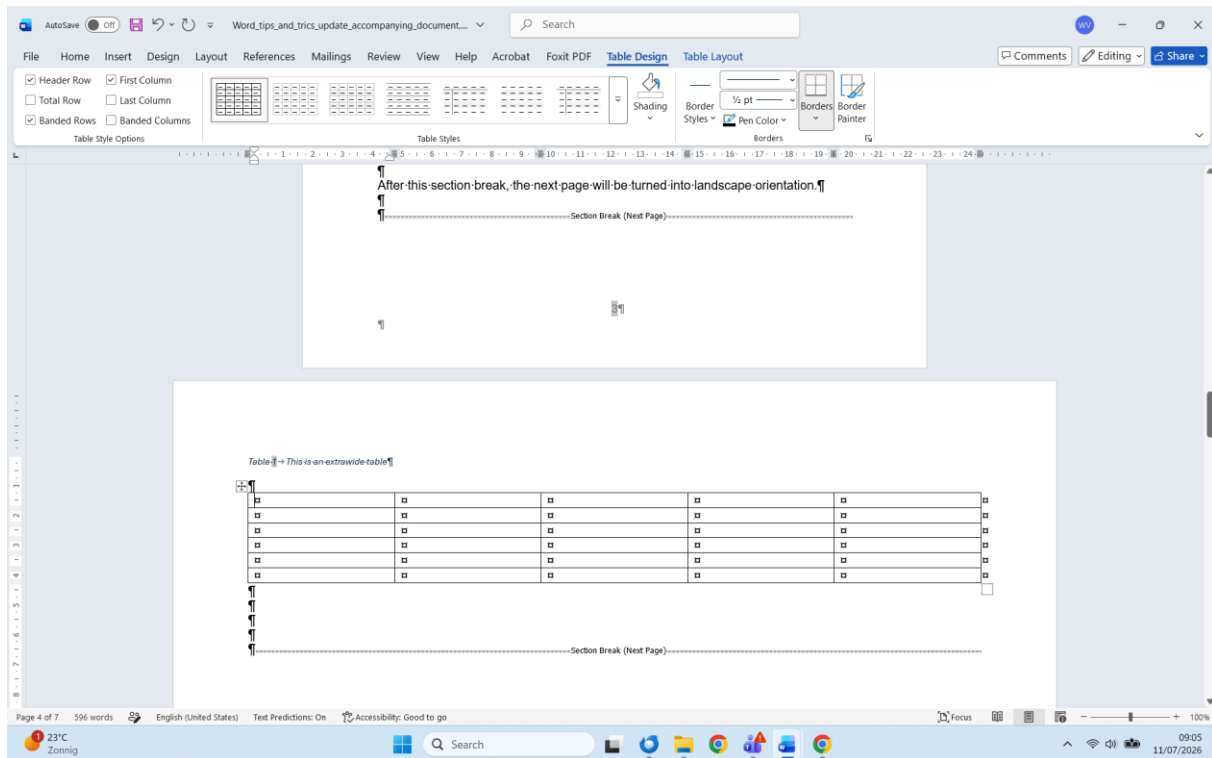
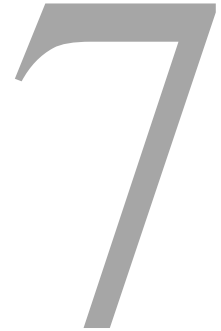


Figure 86 Section after “Next page” section break has been changed to landscape orientation.

Keep in mind that a section break controls the section formatting of the text that precedes it. For example, if you delete a section break, the preceding text becomes part of the following section and assumes its section formatting. Note that the last paragraph mark in the document controls the section formatting of the last section in the document — or of the entire document if it does not contain any sections.



Using headers, footers and page numbering

A header, which can consist of text or graphics, appears at the top of every page in a section. A footer appears at the bottom of every page. Headers and footers often contain page numbers, chapter titles, dates, or author names. I deliberately write “in a section”, because headers, footers and page numbers can indeed again be different in different sections.

The combined use of headers and footers with section breaks is very useful for larger documents. For example, in the beginning of your document, you have a table of contents, where the page numbering in the footer is typically in lower case Roman numerals (i, ii, iii, ...) and the header has the running title *Table of Contents*. After the table of contents, you have the first chapter, where the page numbering in the footer is in arabic numbers (1, 2, 3, ...) and the header has the running title *Chapter 1* or something similar.

To realize this example, you first define the header and footer for the Table of Contents. Click *Insert > Header*, so that the *Header* dropdown list appears, as shown in Figure 87. Select “*Edit Header*”.

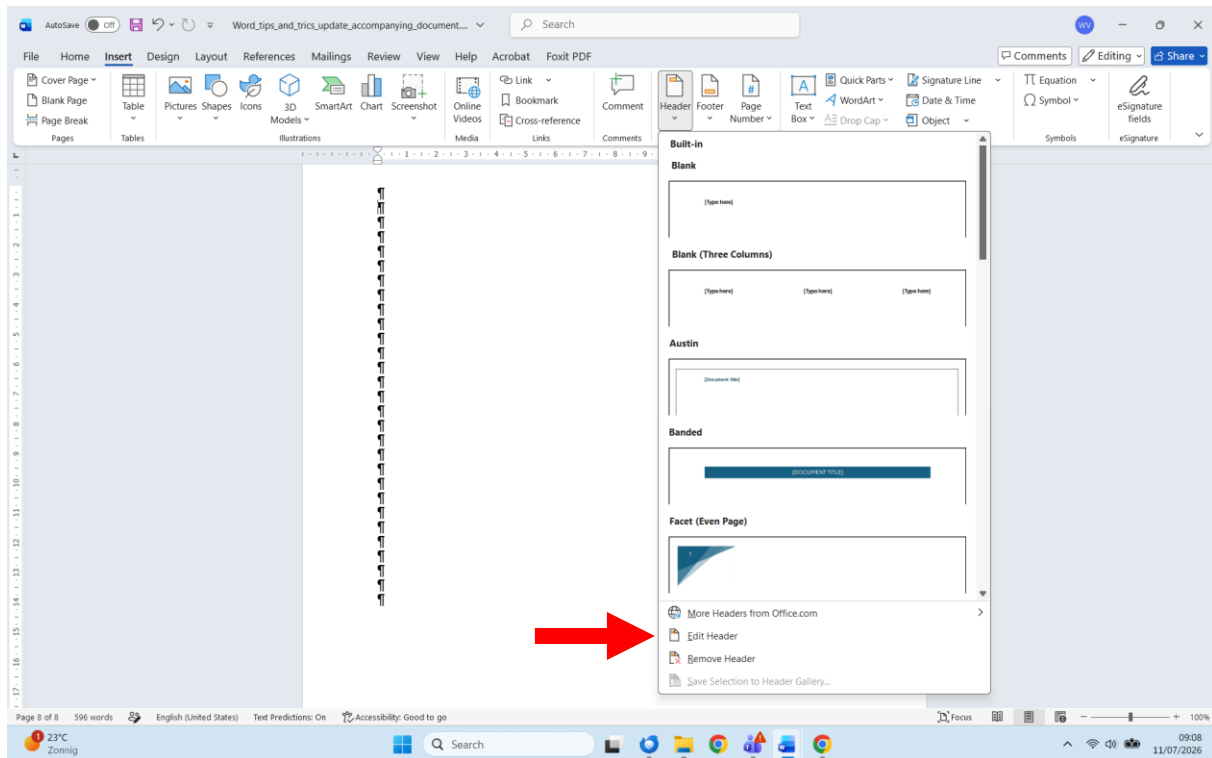


Figure 87 Header toolbar.

Now you can insert the running title *Table of Contents* in the header, for example italic, right aligned and with a line under it (with the *Home > Borders* toolbar in Figure 88). We will later see that the header titles can again be field codes of the type STYLEREF, to be automatically updated if you change the title of your sections. The result can look like Figure 88.

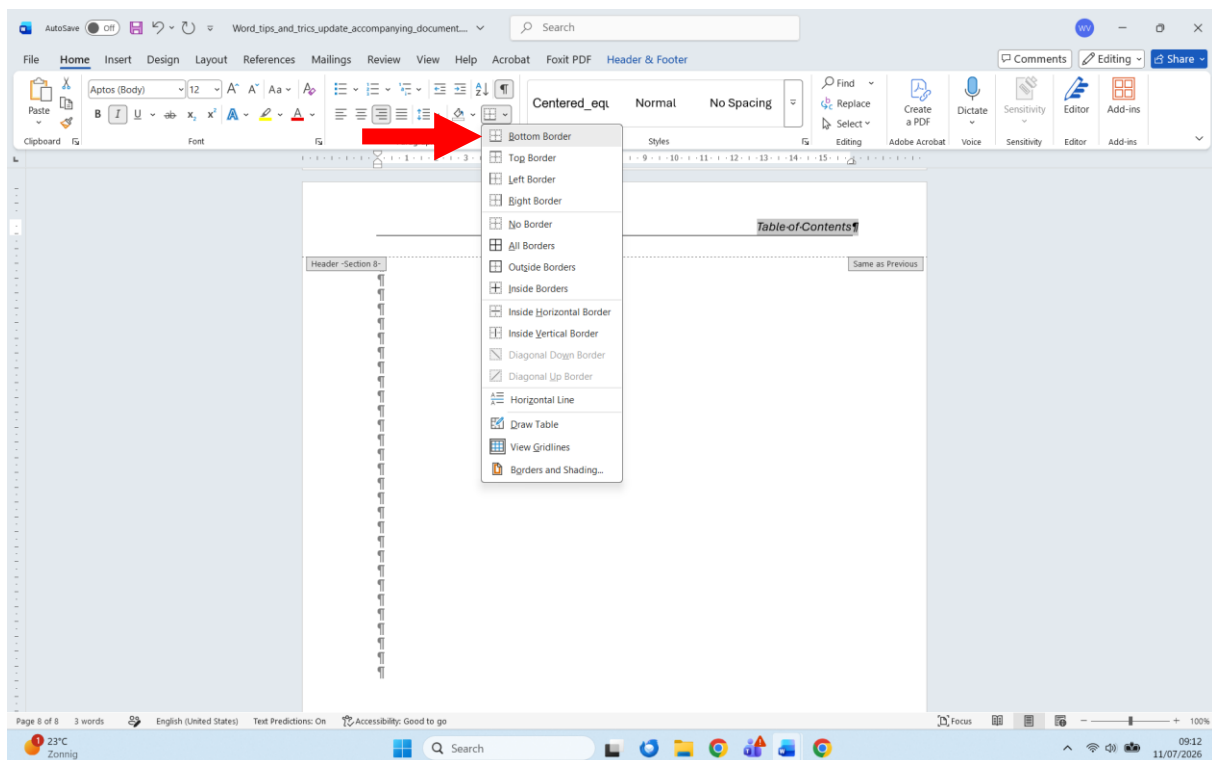


Figure 88 Layout of the header for the Table of Contents.

Next you switch to the footer to insert for example the page number for automatic page numbering. You click on the *Header and Footer* menu on the top menu bar, and then on the button “Go to Footer” in the toolbar, as shown in Figure 89.

Now, you insert a page number in the footer by clicking the button *Page Number* on the toolbar, as shown in Figure 90.

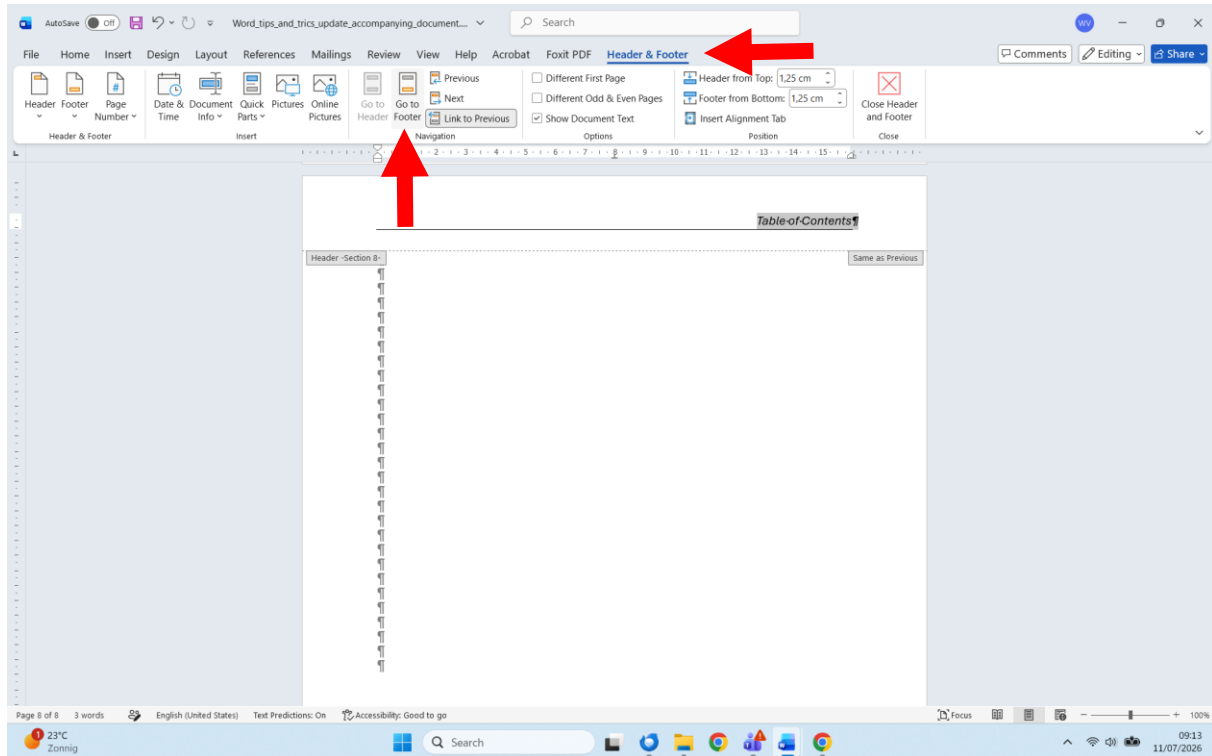


Figure 89 Switching to the footer.

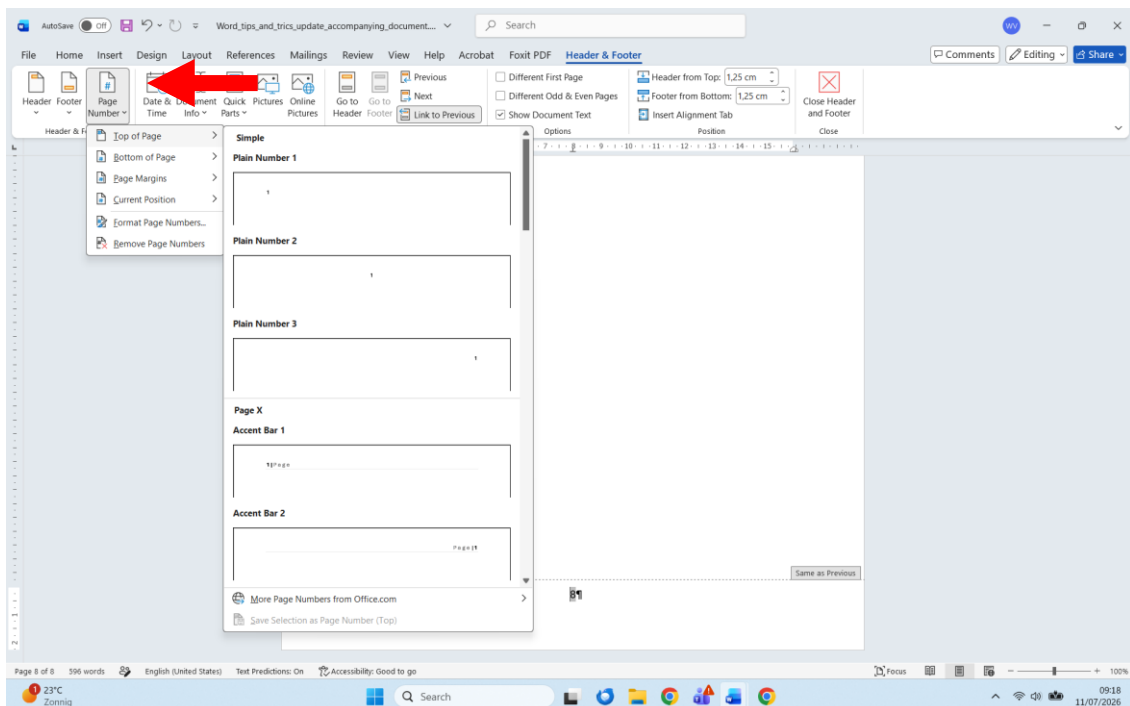


Figure 90 Inserting an automatic page number in the footer.

If you select the inserted page number and click again on the button *Page Number* from the *Header and Footer* toolbar (see Figure 91), you can format the page number and its style. The *Page Number Format* dialogue box appears, as shown in Figure 92.

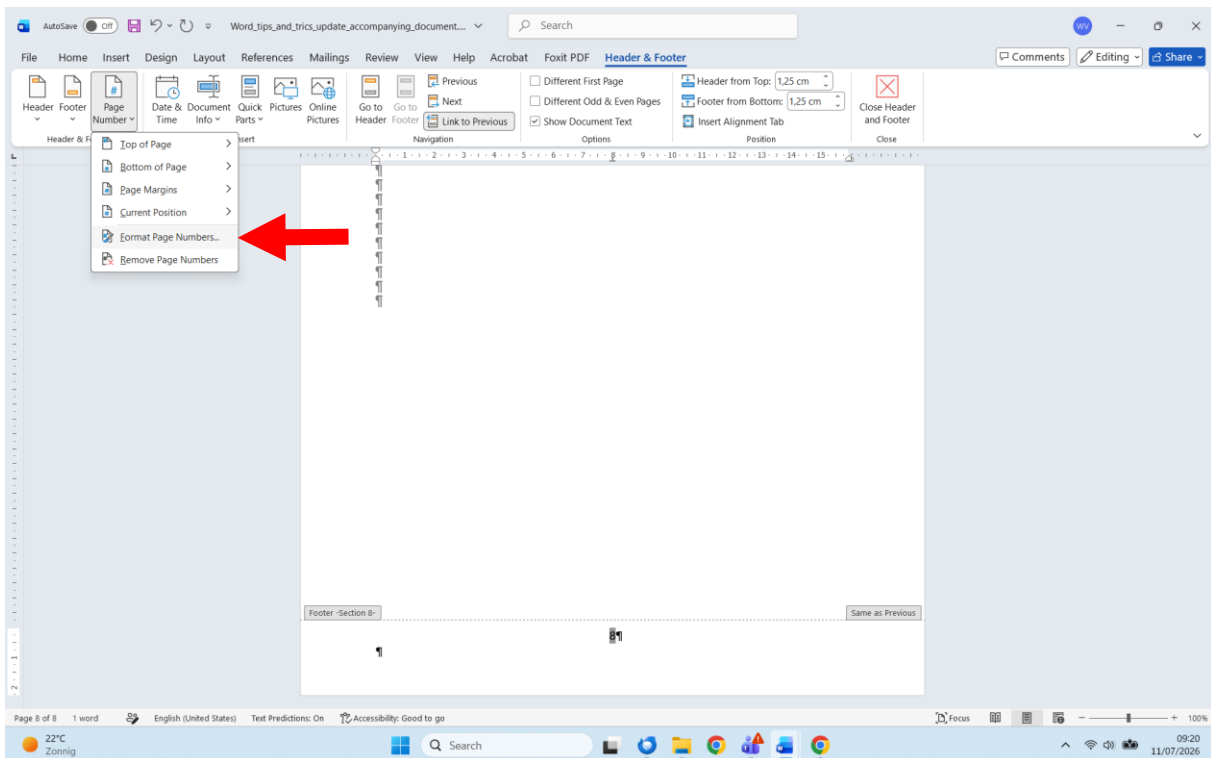


Figure 91 Formatting the automatic page number in the footer.

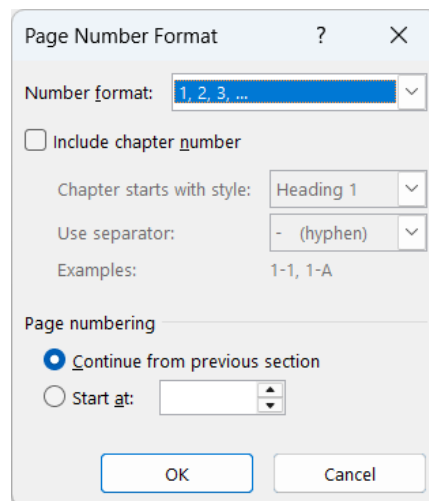


Figure 92 *Page Number Format* dialogue box for formatting the automatic page numbers.

Here the power of using styles and section breaks, is proved again. You can choose to include the chapter number automatically. If you select this option, MS Word will detect the chapter number from the style that you have defined for the chapter titles (e.g. *Heading 1*). Further you can choose to start the page numbering at a specific number or continue from the previous section.

In this case, we choose lower case Roman numerals (i, ii, iii,...) for the *Table of Contents*.

Now, you insert a section break between the last page of the *Table of Contents* and the first page of *Chapter 1*. Then you navigate to the first page of *Chapter 1* and you open the *Header and Footer* toolbar again. Now you can change the header to “Chapter 1”, as illustrated in Figure 93.

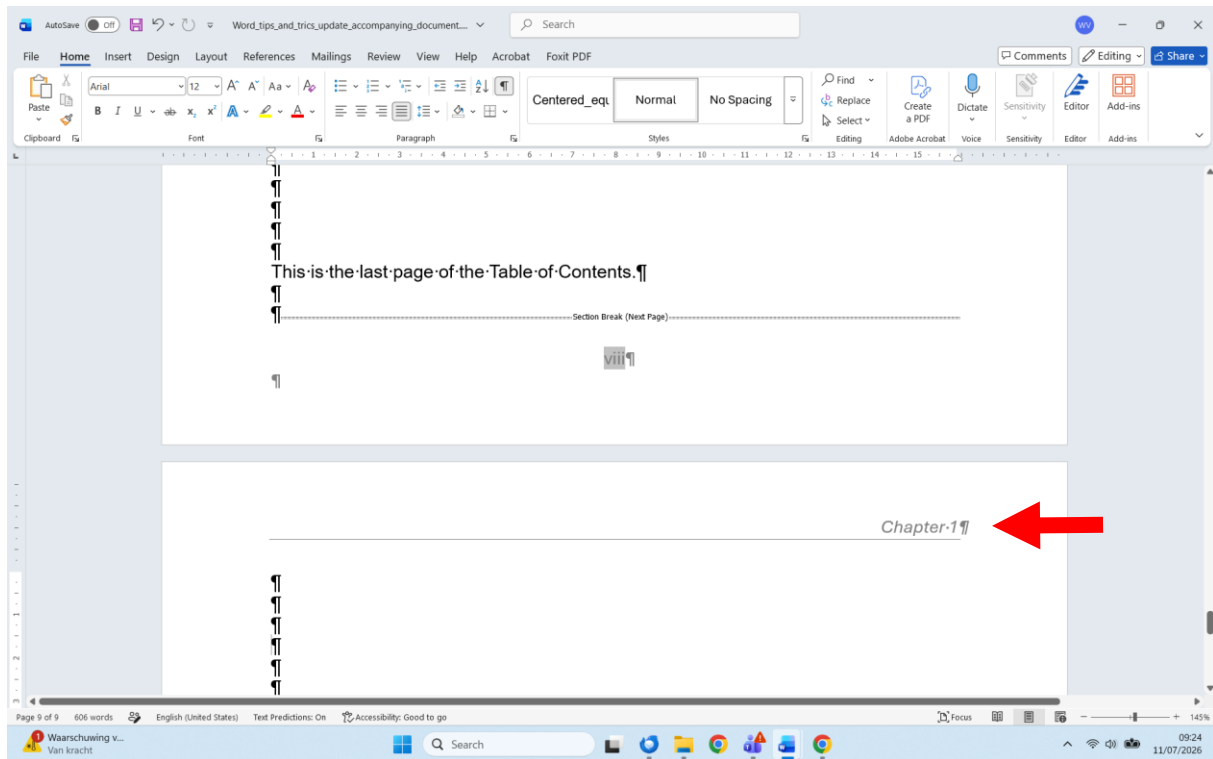


Figure 93 Changing the header and footer for the next section.

There is a pitfall though. If you now scroll up again to the *Table of Contents* section, you will see that the header of the Table of Contents is also changed to “*Chapter 1*”, which was not the intended goal.

To understand what happened, you have to pay attention to the coupling between this section and the previous section. As can be seen from Figure 94, the header of the section *Chapter 1* mentions “*Same as previous*” in the right margin.

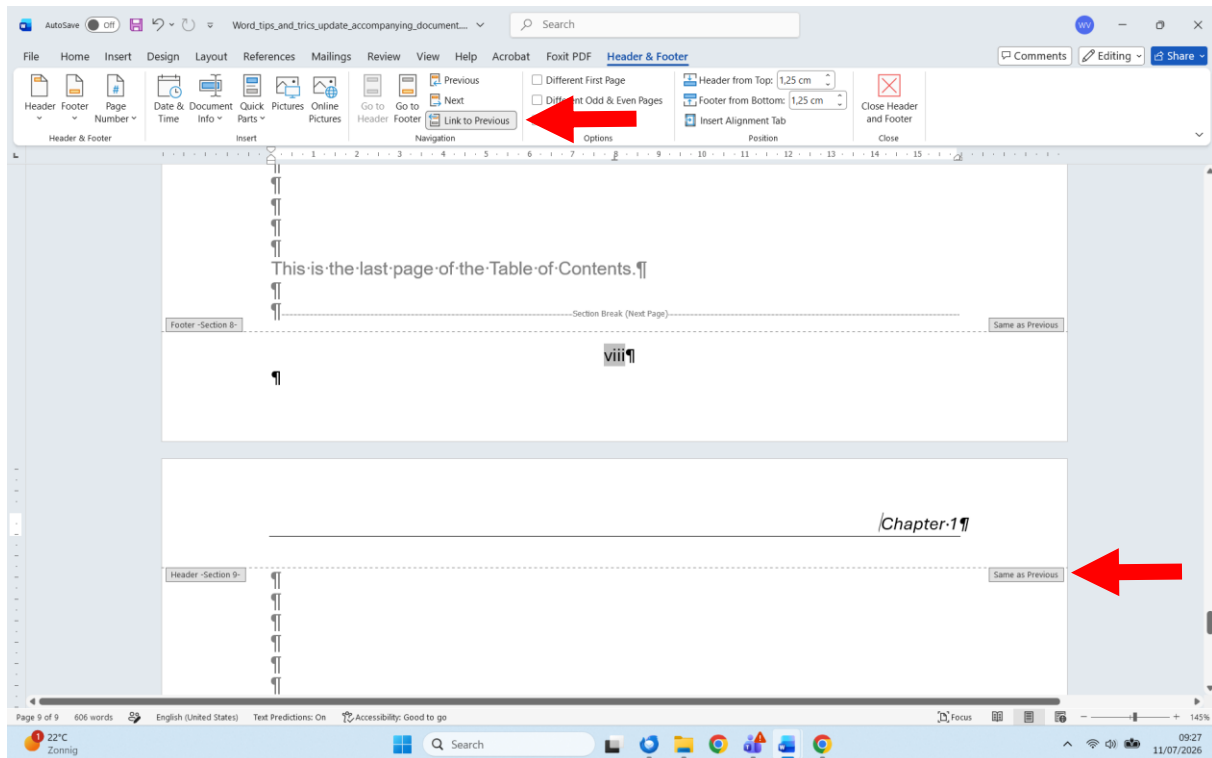


Figure 94 Header of *Chapter 1* is linked to the header of the previous section.

However, because you have created a new section, you can decouple the header and footer of this section from the header and footer of the previous section. This can also be seen in the *Header and Footer* toolbar, where the button “*Link to previous*” is activated.

Therefore you deselect the button “*Link to previous*” from the *Header and Footer* toolbar, as shown in Figure 95. This button determines if the header/footer is linked to the previous header/footer. You can remove this link only for the header, only for the footer, or for both.

Once you have removed the “*Link to previous*” for the header, you can change the running title into *Chapter 1*. Similarly, you can change the numbering style to arabic numbers (1, 2, 3,...) in the footer and restart the page numbering at ‘1’.

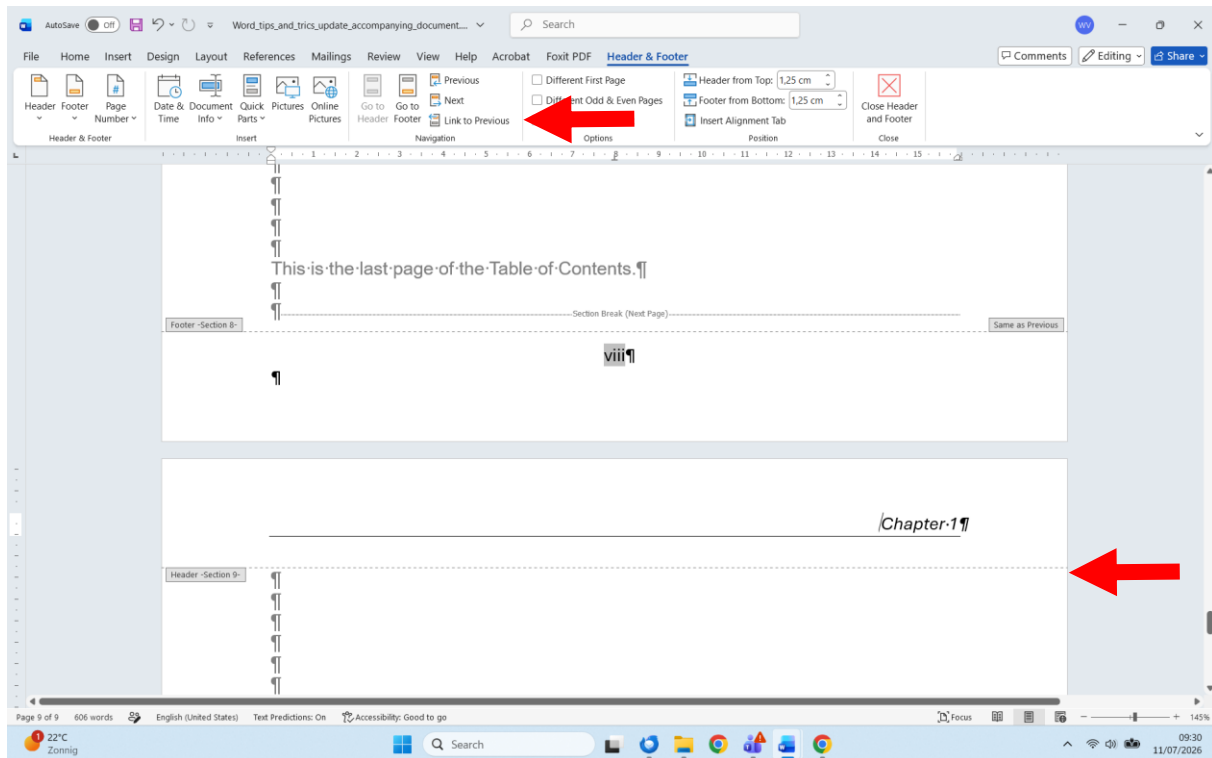


Figure 95 Removing the link to the previous header and the currently active header.

If you want the header to show on every page, except on the first page of each chapter, you need to insert again a section break before each chapter. Then you select the option *Different first page* in the *Header and Footer* toolbar, which allows you to have another header (or empty header) for the first page of a section.

Figure 96 shows that the option “*Different First Page*” is activated for the header, and also in the left margin of the header, it is explicitly mentioned “*First Page Header – Section 10*”.

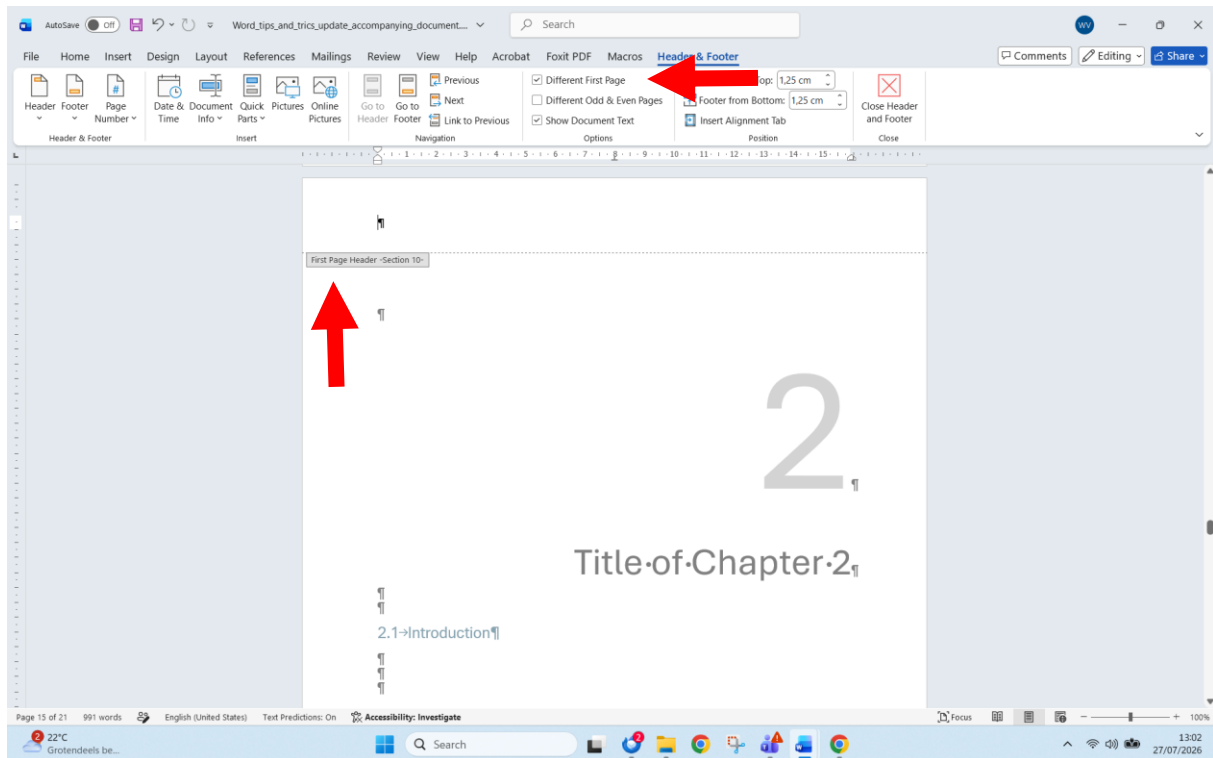


Figure 96 Different header for the first page of the section.

You can also copy the chapter title in the header with the field code `STYLEREF`. If you then change the title of the chapter, that is also automatically updated in the header. If you also want to have the chapter number in the header, you need to insert the `STYLEREF` field code twice, but with a different switch.

To have the chapter number and chapter title automatically appearing in the header, you insert the field codes below with `<Ctrl>+<F9>`:

```
{ STYLEREF "Heading 1" \n }. { STYLEREF "Heading 1" }
```

This results for this chapter in the header with text “6. Using headers, footers and page numbering”.

Finally, these section breaks have an unwanted side effect that the page numbers in the footers always start again at 1 for every chapter, even if the footer is linked to the previous footer. You must explicitly set in the page number settings the option “Continue from previous section”, as shown in Figure 97.

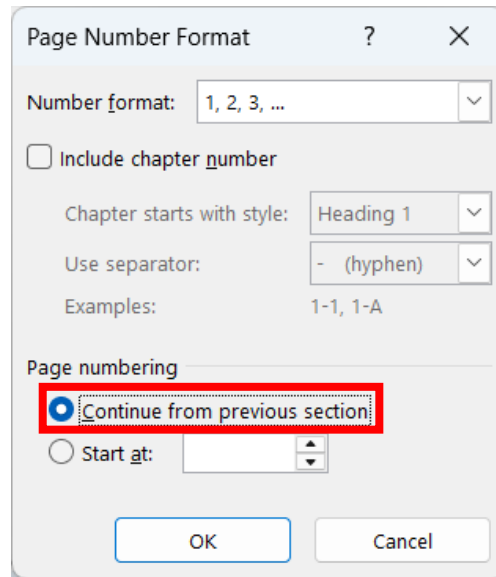


Figure 97 Continue the page numbering from one section to another.

8

Using numbered chapters and sections

Numbering chapters and sections in Word is a very tricky thing. Automatic numbering of chapters, subchapters, sections, etc. should NEVER be done by just applying the automatic list numbering. It gives you a lot of trouble when generating automatically the table of contents, or cross-referencing to chapters or sections.

Figure 98 shows the list types that you can find on the Home Ribbon of the Word toolbar. 99% of the Word users use the two first lists: (i) bulleted lists and (ii) numbered lists.

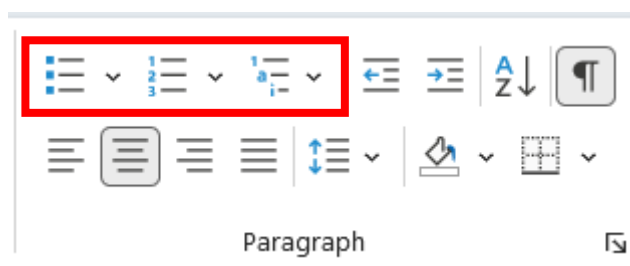


Figure 98 List types on the Paragraph toolbar.

The problematic and misleading thing is that if you apply a numbered list to a *Heading* style, it actually works. As shown in Figure 99, I have written a chapter title, assigned the built-in paragraph style “*Heading 1*” to it, and then clicked on the button “*Numbered list*” in the *Paragraph* toolbar. As you can see, the *Heading 1* is now correctly numbered.

Despite the intuitive character of this, it causes a lot of trouble if you want to automatically generate tables of contents, and number different section levels in a clean way.

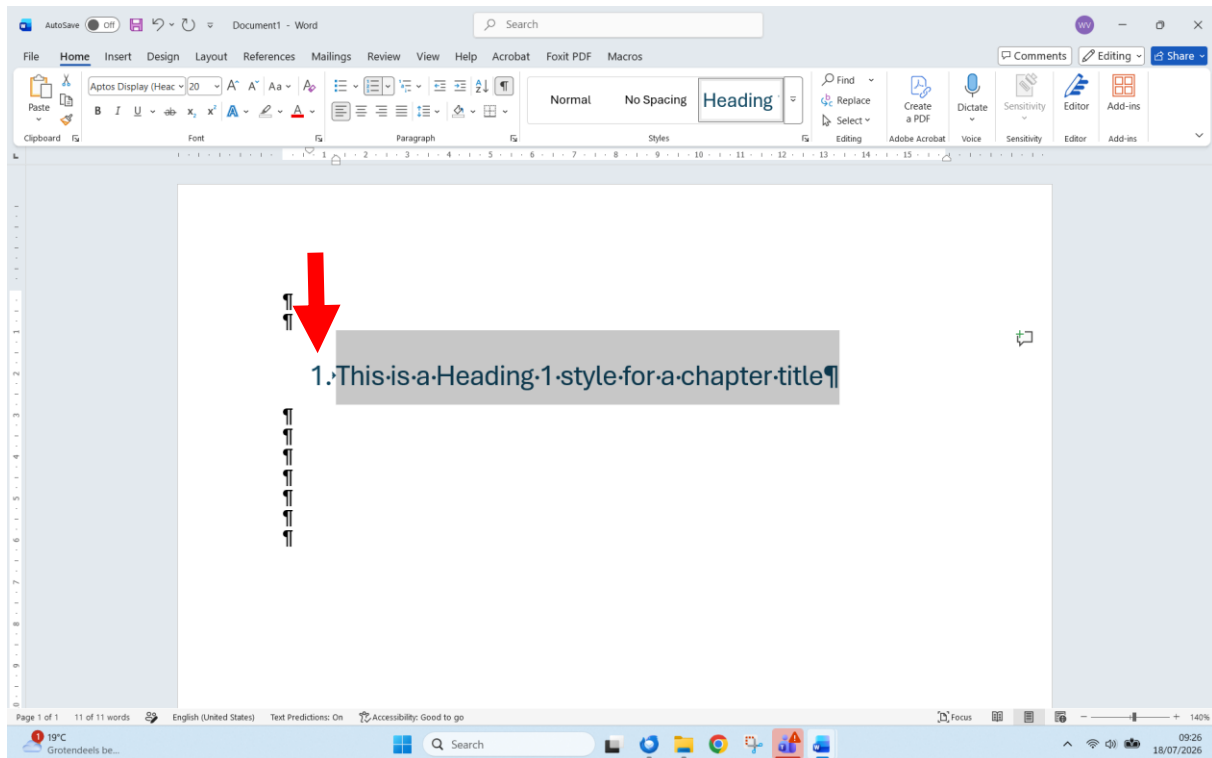


Figure 99 Assigning numbering to the Heading 1 style in the wrong way.

8.1. DEFINING MULTI-LEVEL LISTS

The only right way to do it, is to use “Multi-Level lists”. This is the third button of the list types in Figure 98. If you hover with your mouse over this button, it shows the explanation in Figure 100. This multi-level list is a list type which couples a certain paragraph style (e.g. *Heading 1*, *Heading 2*,...) to a certain level in the document (e.g. chapter, subchapter, section, subsection,...). Once you understand how it works, you can couple any paragraph style that you define yourself, to any heading level that you want.

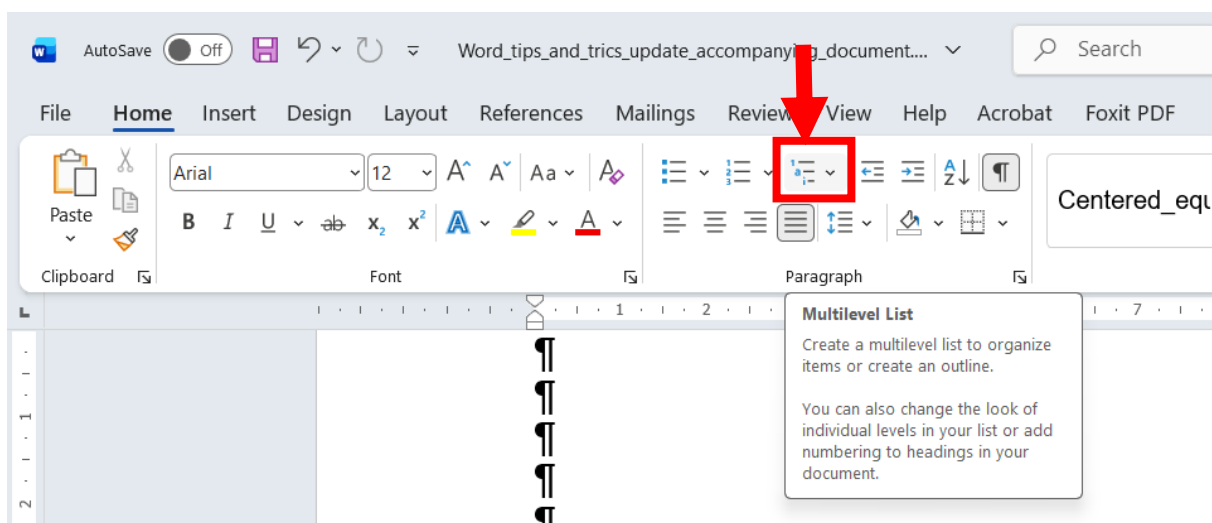


Figure 100 Selecting the Multi-level List type.

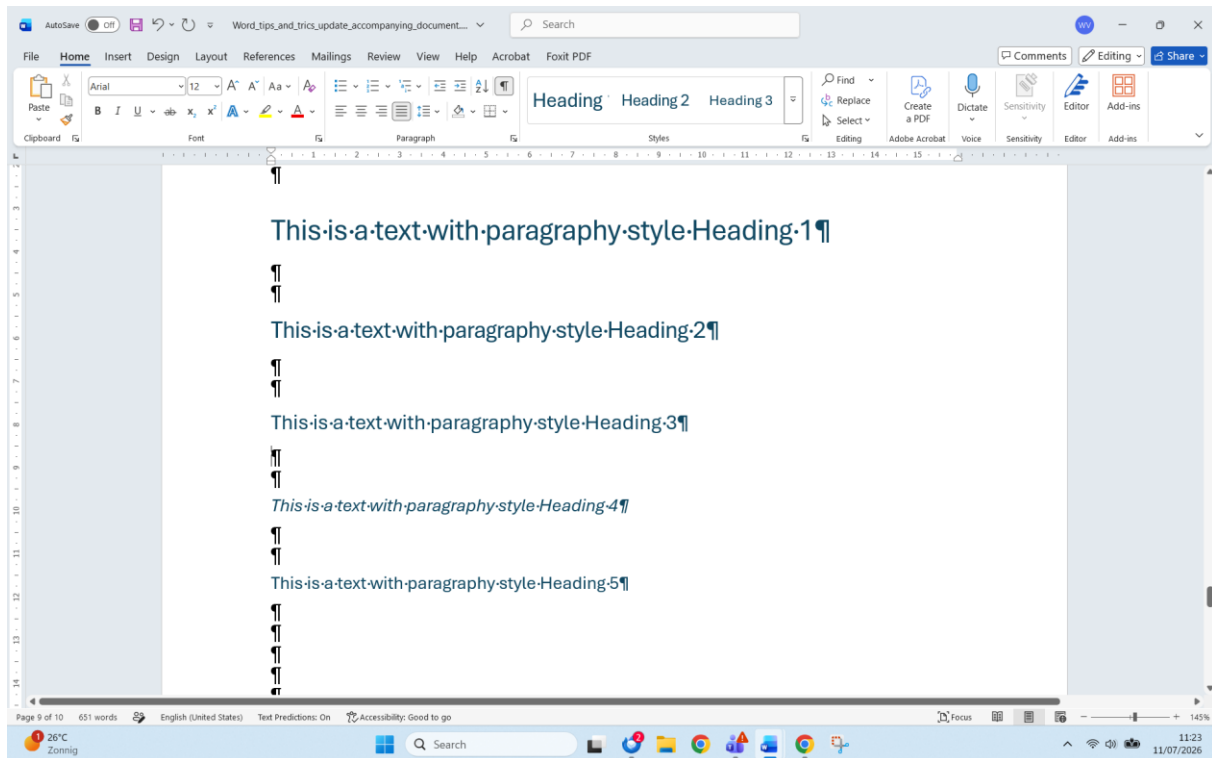


Figure 102 Document with 5 heading levels, not yet numbered.

Now, we select the Multi-level List icon and select the pre-defined list type from the List Library, which links the 5 heading levels with their respective numbers, as shown in Figure 103.

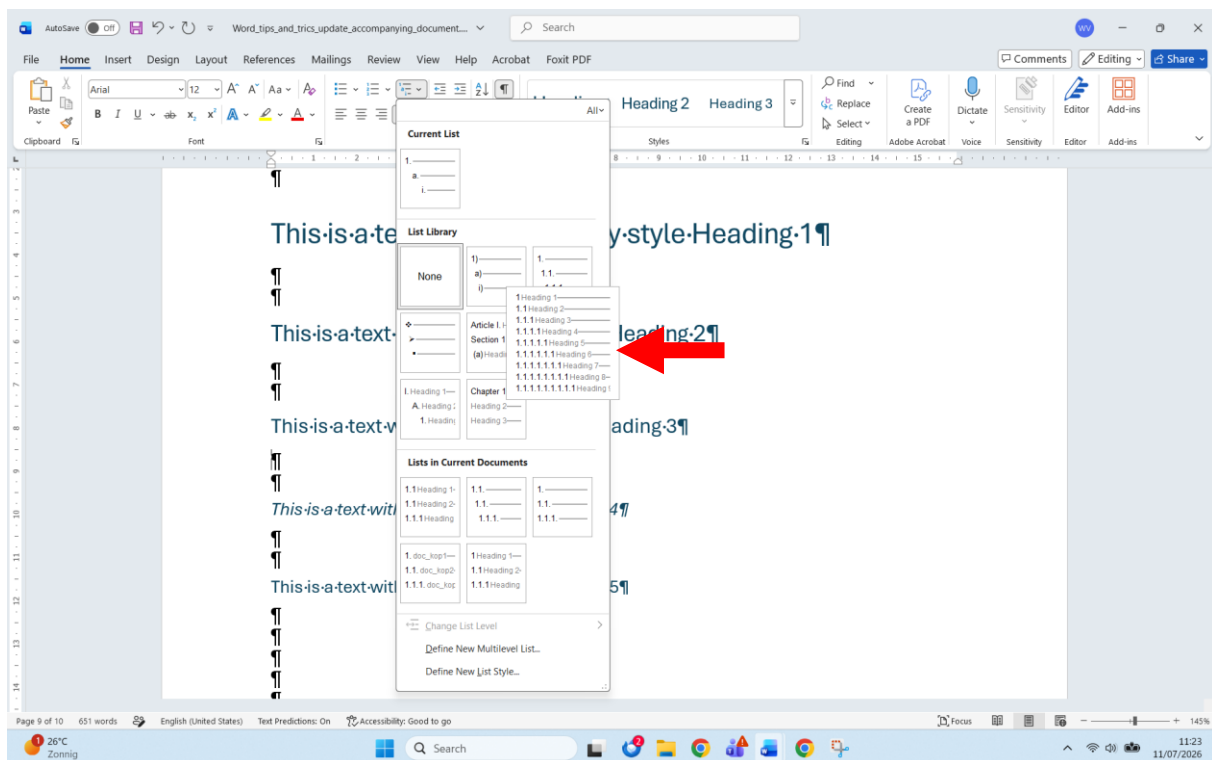


Figure 103 Selecting a Multi-level List type from the List Library.

As you can see in Figure 104, all 5 heading levels are now automatically numbered. Please notice that also in the *Paragraph* toolbar of the Home Ribbon, the heading paragraph styles are now numbered.

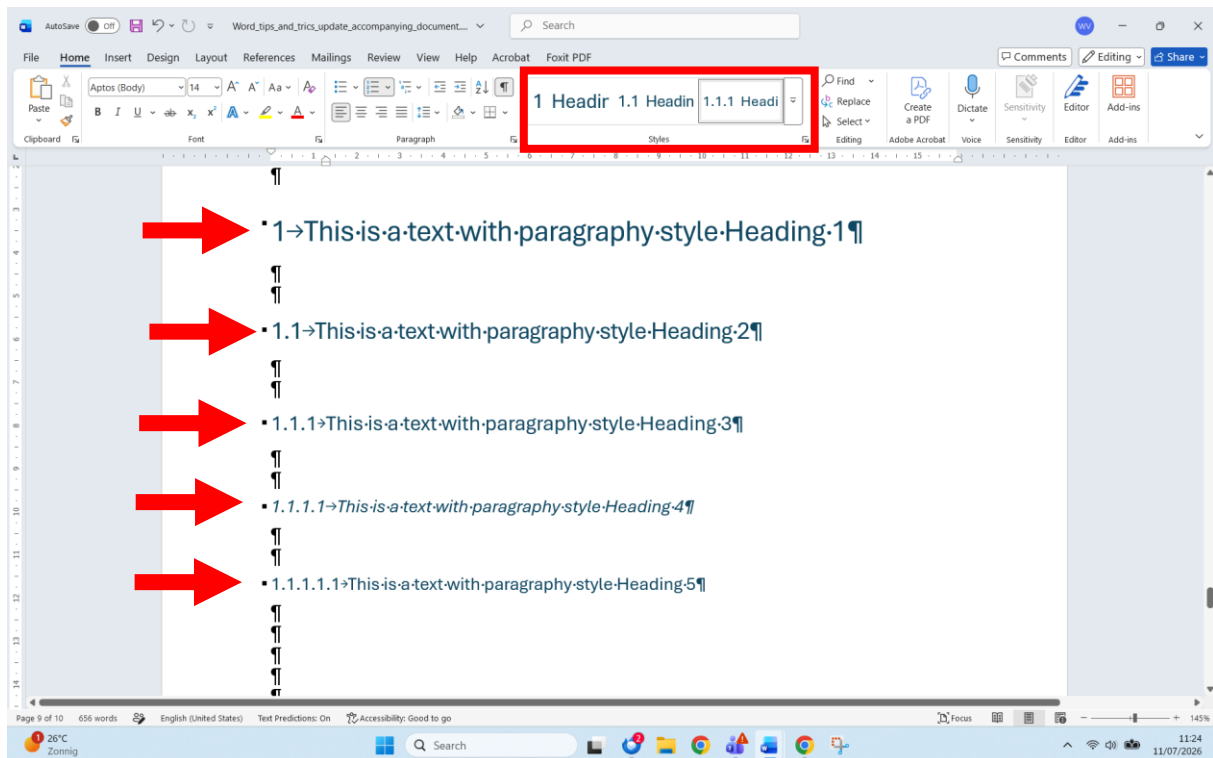
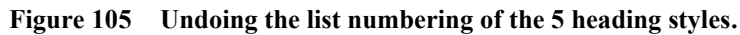


Figure 104 Numbering of the 5 heading styles with a pre-defined Multi-level List style.

Undoing the numbering, is not possible with a simple mouse click. If you go back to the Multi-level List and you select “None” in the List library, Word does not undo the numbering. You have to select all text and then undo the numbering by selecting “None” in the List library, as illustrated in Figure 105.



AutoSave On Word_tips_and_tricks.update_accompanying_document... Search

File Home Insert Design Layout References Mailings Review View Help Acrobat Foxit PDF

Clipboard Font Paragraph Styles

1 Headin 1.1 Headin 1.1.1 Headin

Find Replace Select Editing Adobe Acrobat Dictate Voice Sensitivity Editor Add-ins

This is a text with paragraph style Heading 1

This is a text with paragraph style Heading 2

This is a text with paragraph style Heading 3

This is a text with paragraph style Heading 4

This is a text with paragraph style Heading 5

Page 9 of 10 45 of 651 words English (United States) Text Predictions: On Accessibility: Good to go

26°C Zonnig 11:28 11/07/2022

92

Now, to better understand how Multi-Level Lists work, we will define a Multi-Level List ourselves. Open the Multi-Level List dropdown toolbar again and select “*Define New List Style...*”, as shown in Figure 107.

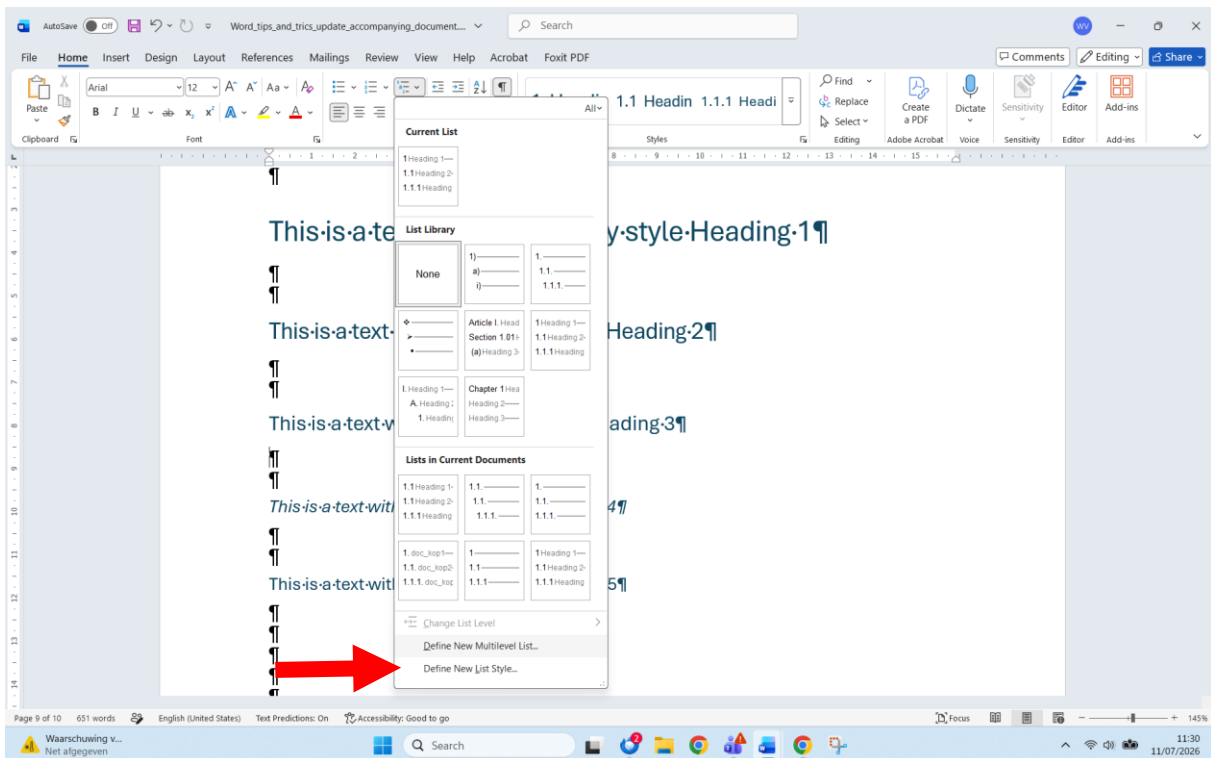


Figure 107 Definition of a new List style.

The dialogue box in Figure 108 opens. First, you can enter the name for the new Multi-level list, e.g. “Thesis_list_style”. Next, you click on the button “*Format*” in the lower left corner.

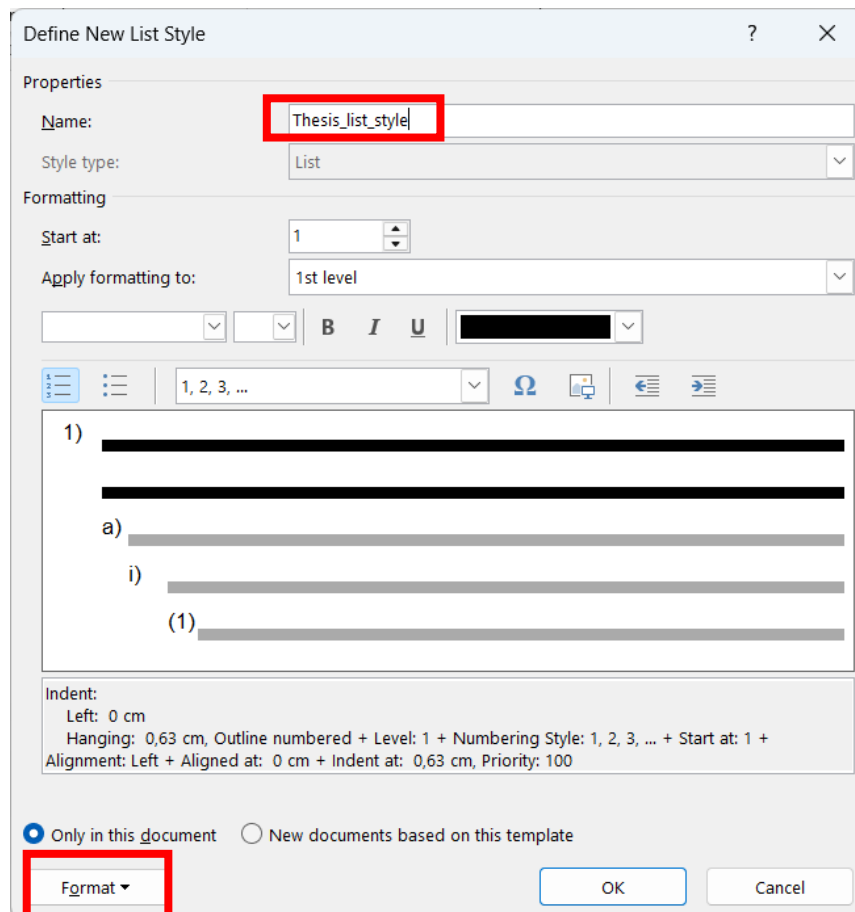


Figure 108 Definition of a new List Style.

As you can see in Figure 109, most formatting options are greyed out, and indeed, a list style is actually just used to define a hierarchy between existing paragraph styles, and is not assigned as such to a paragraph. It is situated more in a layer “above” the document.

We only need the formatting option “*Numbering*” to define the different levels of heading styles and the numbering style. It is exactly this defined hierarchy that will be used by Word to automatically generate the Table of Contents with the right heading styles and the right numbering styles.

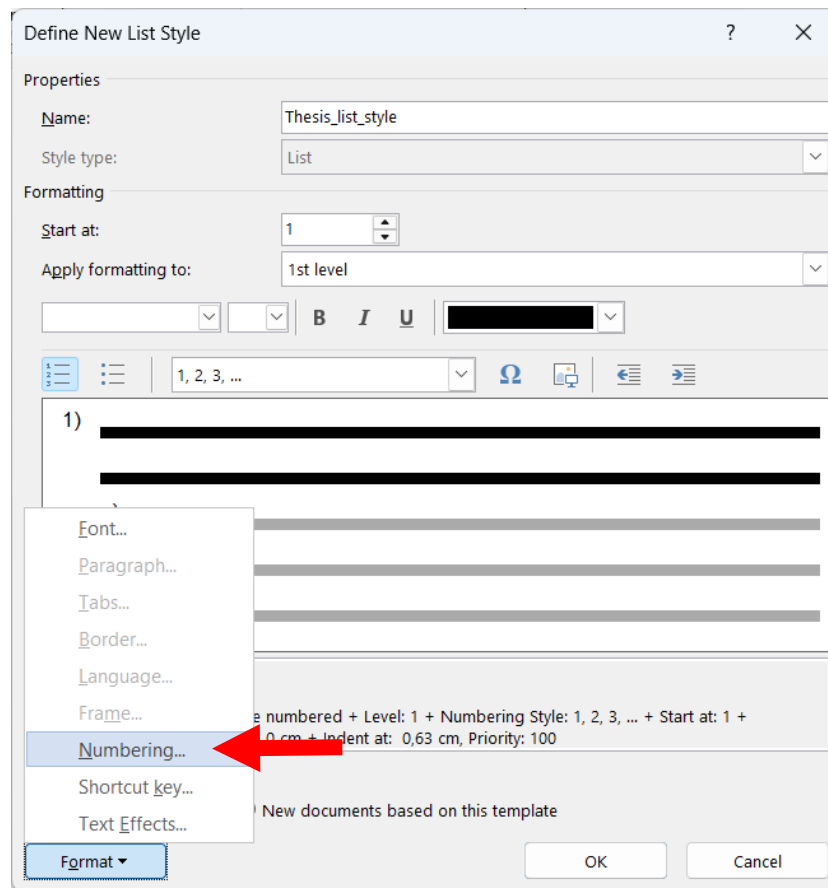


Figure 109 Select the formatting option “Numbering”.

If you select this formatting option “*Numbering*”, the dialogue box in Figure 110 appears. It is advised to click immediately in the left-bottom corner on the button “*More >>*”, which shows also the advanced options.

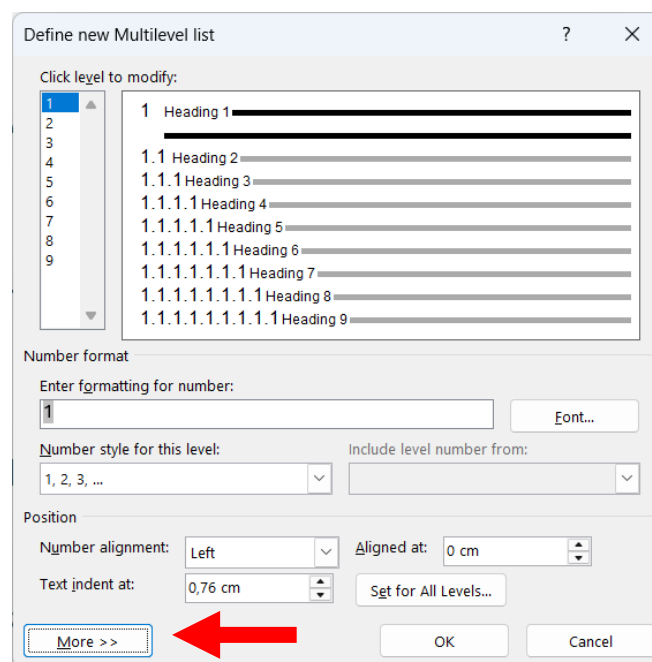


Figure 110 Dialogue box to define the hierarchy of the new Multi-level list.

Then the dialogue box expands to the view of Figure 111. You can now define many things:

- Link every heading level to a specific paragraph style. This does not have to be the built-in paragraph style “*Heading 1*” for level 1. You can link it to any paragraph style you like.
- Define if the numbering of every level also includes the numbering of the previous levels. Notice again that the numbers of the levels have a grey shaded background, because these are again field codes.
- Define the numbering style of each level (1, 2, 3, or a, b, c, or i, ii, iii, etc.)
- Define the alignment, indentation, etc. of each number.

Figure 111 Expanded view of the dialogue box for definition of a new Multi-level list.

For example, I can now link Level 1 in the hierarchy to the built-in paragraph style “*Heading 1*”, and I can also remove the right bracket) after the chapter number, as illustrated in Figure 112.

Define new Multilevel list

Click level to modify:

- 1 Heading 1
- 1.1 Heading 2
- 1.1.1 Heading 3
- 1.1.1.1 Heading 4
- 1.1.1.1.1 Heading 5
- 1.1.1.1.1.1 Heading 6
- 1.1.1.1.1.1.1 Heading 7
- 1.1.1.1.1.1.1.1 Heading 8
- 1.1.1.1.1.1.1.1.1 Heading 9

Apply changes to: Whole list

Link level to style: Heading 1

Level to show in gallery: Level 1

ListNum field list name:

Number format

Enter formatting for number: 1

Font...

Start at: 1

Restart list after:

Legal style numbering

Position

Number alignment: Left

Aligned at: 0 cm

Text indent at: 0,76 cm

Set for All Levels...

Follow number with: Tab character

Add tab stop at: 0,76 cm

<< Less

OK Cancel

Figure 112 Linking the heading levels to their respective paragraph style.

If I move to Level 2 in the hierarchy, I see the window in Figure 113. The numbering of Level 2 has the format “a, b, c” and is not yet linked to any paragraph style.

I want to link the Level 2 to the built-in paragraph style “*Heading 2*” and also show the Level 1 number, so instead of numbering the Level 2 as:

a) Heading 2

I want to number it as:

1.1 Heading 2

So I need a reference to the previous level.

Modify Multilevel list

Click level to modify:

1
2
3
4
5
6
7
8
9

1 Heading 1
a)
i)
(1)
(a)
(i)
1.
a.
i.

Apply changes to:
Whole list

Link level to style:
(no style)

Level to show in gallery:
Level 1

ListNum field list name:

Number format

Enter formatting for number:
a)

Font...

Start at: a

☒ Restart list after:
Level 1

☐ Legal style numbering

Position

Number alignment: Left

Aligned at: 0,63 cm

Text indent at: 1,27 cm

Set for All Levels...

Follow number with:
Tab character

☐ Add tab stop at:
1,27 cm

<< Less

OK

Cancel

Figure 113 Default settings for Level 2.

To that purpose, you could think to insert manually the number 1 of Level 1 into the box for “Enter formatting for number:”, as illustrated in Figure 114. The preview on top of the dialogue box seems to show the correct formatting.

However, you notice that your manually inserted number 1 has no grey background, while the second level with number 1 has a grey background. This means that regardless in which chapter you are, the Heading 2 will always be numbered as 1.1, 1.2, even if you are in Chapter 2 or 3. So this is not the correct way.

Modify Multilevel list

Click level to modify:

1 Heading 1
 2 1.1 Heading 2
 3
 4
 5
 6
 7
 8
 9

Apply changes to: Whole list

Link level to style: Heading 2

Level to show in gallery: Level 1

ListNum field list name:

Number format

Enter formatting for number: 1.1

Font...

Start at: 1

☒ Restart list after: Level 1

☐ Legal style numbering

Position

Number alignment: Left

Aligned at: 0,63 cm

Text indent at: 1,27 cm

Set for All Levels...

Follow number with: Tab character

☐ Add tab stop at: 1,27 cm

<< Less OK Cancel

Figure 114 Manual insertion of Level 1 number.

To correctly insert the number of the higher level into the numbering of the lower level, you need to insert these numbers with the drop down list “*Include level number from:*”, as illustrated in Figure 115. If I select “*Level 1*” in the drop-down list, it is inserted at the cursor position in the box “*Enter formatting for number:*”.

Modify Multilevel list

Click level to modify:

- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9

1 Heading 1

.1 Heading 2

i)

(1)

(a)

(i)

1.

a.

i.

Apply changes to: Whole list

Link level to style: Heading 2

Level to show in gallery: Level 1

ListNum field list name:

Number format

Enter formatting for number: .1

Font...

Number style for this level: 1, 2, 3, ...

Include level number from: Level 1

Position

Number alignment: Left

Text indent at: 1,27 cm

Start at: 1

☒ Restart list after: Level 1

☐ Legal style numbering

Follow number with: Tab character

☐ Add tab stop at: 1,27 cm

<< Less

OK

Cancel

Figure 115 Inserting the numbers of higher level headings into the numbering of a lower level heading.

The result is shown in Figure 116. You see that this time, the number of Level 1 has also a grey background, and is also a field code that will be updated for each new chapter.

Modify Multilevel list

Click level to modify:

1 Heading 1
 2 1.1 Heading 2
 3
 4
 5
 6 i)
 7 (1)
 8 (a)
 9 (i)
 1.
 a.
 i.

Apply changes to: Whole list

Link level to style: Heading 2

Level to show in gallery: Level 1

ListNum field list name:

Number format

Enter formatting for number: 1.1

Font...

Start at: 1

☒ Restart list after: Level 1

☐ Legal style numbering

Position

Number alignment: Left

Aligned at: 0,63 cm

Follow number with: Tab character

☐ Add tab stop at: 1,27 cm

Text indent at: 1,27 cm

Set for All Levels...

<< Less

OK

Cancel

Figure 116 Numbering of level 2 includes the numbering of level 1 (with grey shaded background).

If I proceed in the same way, I can do this for all levels in the hierarchy. Figure 117 illustrates how the heading level for level 5 is linked to the paragraph style “*Heading 5*”, and how all the preceding levels are included in the numbering with grey shaded field codes.

Define new Multilevel list

Click level to modify:

- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9

1 Heading 1

1.1 Heading 2

1.1.1 Heading 3

1.1.1.1 Heading 4

1.1.1.1.1 Heading 5

1.1.1.1.1.1 Heading 6

1.1.1.1.1.1.1 Heading 7

1.1.1.1.1.1.1.1 Heading 8

1.1.1.1.1.1.1.1.1 Heading 9

Apply changes to: Whole list

Link level to style: Heading 5

Level to show in gallery: Level 1

ListNum field list name:

Number format

Enter formatting for number: 1.1.1.1.1

Font...

Start at: 1

☒ Restart list after: Level 4

☐ Legal style numbering

Number style for this level: 1, 2, 3, ...

Include level number from:

Position

Number alignment: Left

Aligned at: 0 cm

Text indent at: 1,78 cm

Set for All Levels...

Follow number with: Tab character

☐ Add tab stop at: 1,78 cm

<< Less

OK

Cancel

Figure 117 Linking of heading level 5 with the paragraph style “Heading 5”.

Figure 118 shows how you can for example change the numbering style of this deepest level 5 from “1, 2, 3,...” to “a, b, c,...”. I have also changed the alignment to 0 cm, so that all headings are aligned with the left margin.

Define new Multilevel list

Click level to modify:

- 1 Heading 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9

1.1 Heading 2

1.1.1 Heading 3

1.1.1.1 Heading 4

1.1.1.1.a Heading 5

1.1.1.1.a.1 Heading 6

1.1.1.1.a.1.1 Heading 7

1.1.1.1.a.1.1.1 Heading 8

1.1.1.1.a.1.1.1.1 Heading 9

Apply changes to: Whole list

Link level to style: Heading 5

Level to show in gallery: Level 1

ListNum field list name:

Number format

Enter formatting for number: 1.1.1.1.a

Font...

Start at: a

Restart list after: Level 4

Legal style numbering

Number style for this level: a, b, c, ...

Include level number from:

Position

Number alignment: Left

Text indent at: 1,78 cm

Aligned at: 0 cm

Set for All Levels...

Follow number with: Tab character

Add tab stop at: 1,78 cm

<< Less

OK

Cancel

Figure 118 Change the numbering style of level 5 to “a, b, c,...”.

If you have defined all list levels as shown in Figure 118, you can close the definition of the List style. If you now click again on the Multi-Level list icon in the Home Ribbon, the drop down list shows also the new List style, and if you hover over it, it also shows the name “Thesis_list_style” that you gave to it, as illustrated in Figure 119.

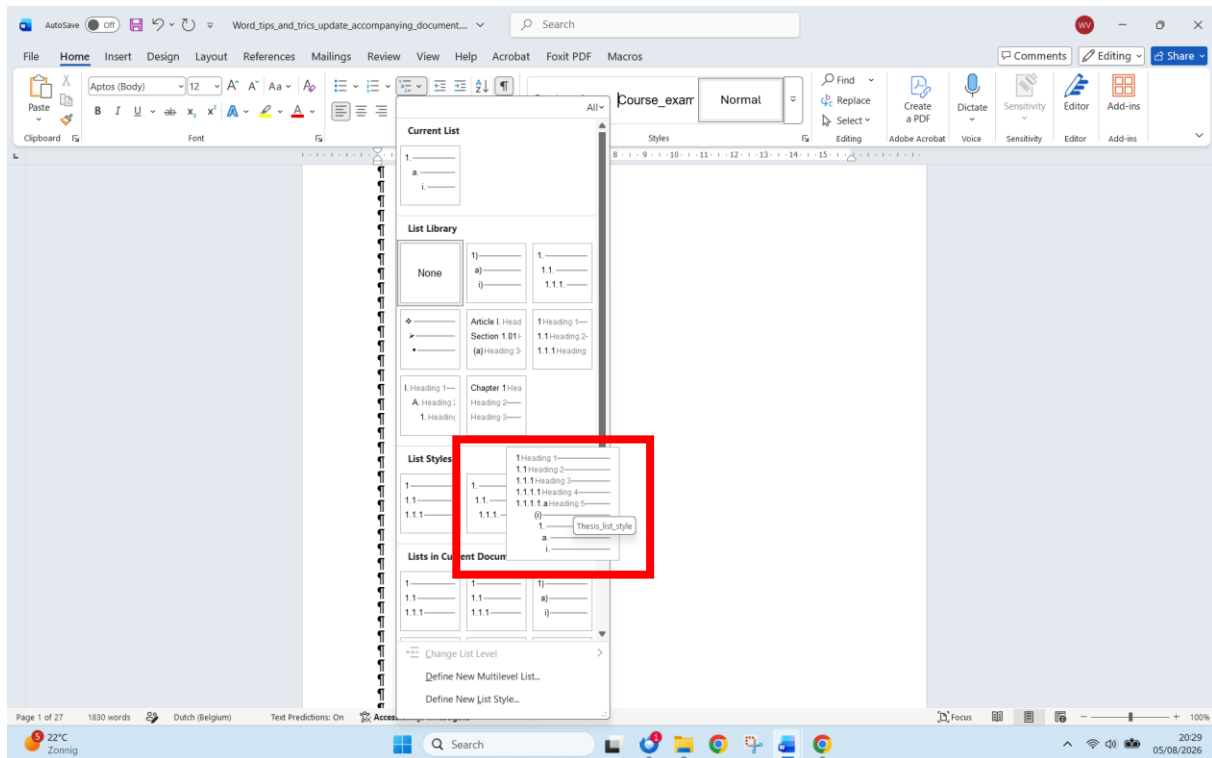


Figure 119 The new List style “Thesis_list_style” is added to the Multi-Level list styles.

You can modify this new List style by right-clicking on it in the drop-down list. So if you now want to change the numbering style of a certain level from 1, 2, 3,... to a, b, c,... or you want to change the alignment and text indentation, you modify the Multi-Level list and these changes are applied to all heading levels throughout your document.

There is one more important thing to know about these Multi-Level Lists. If you take a close look at the drop-down list in Figure 119, you see two options in the very bottom:

- Define New Multi-Level List...
- Define New List Style...

So far, we have chosen the bottom one, “Define New List Style”. However, if you click on the one above, “Define New Multi-Level List”, the dialogue box in Figure 120 opens, which is actually exactly the same as the one in Figure 110.

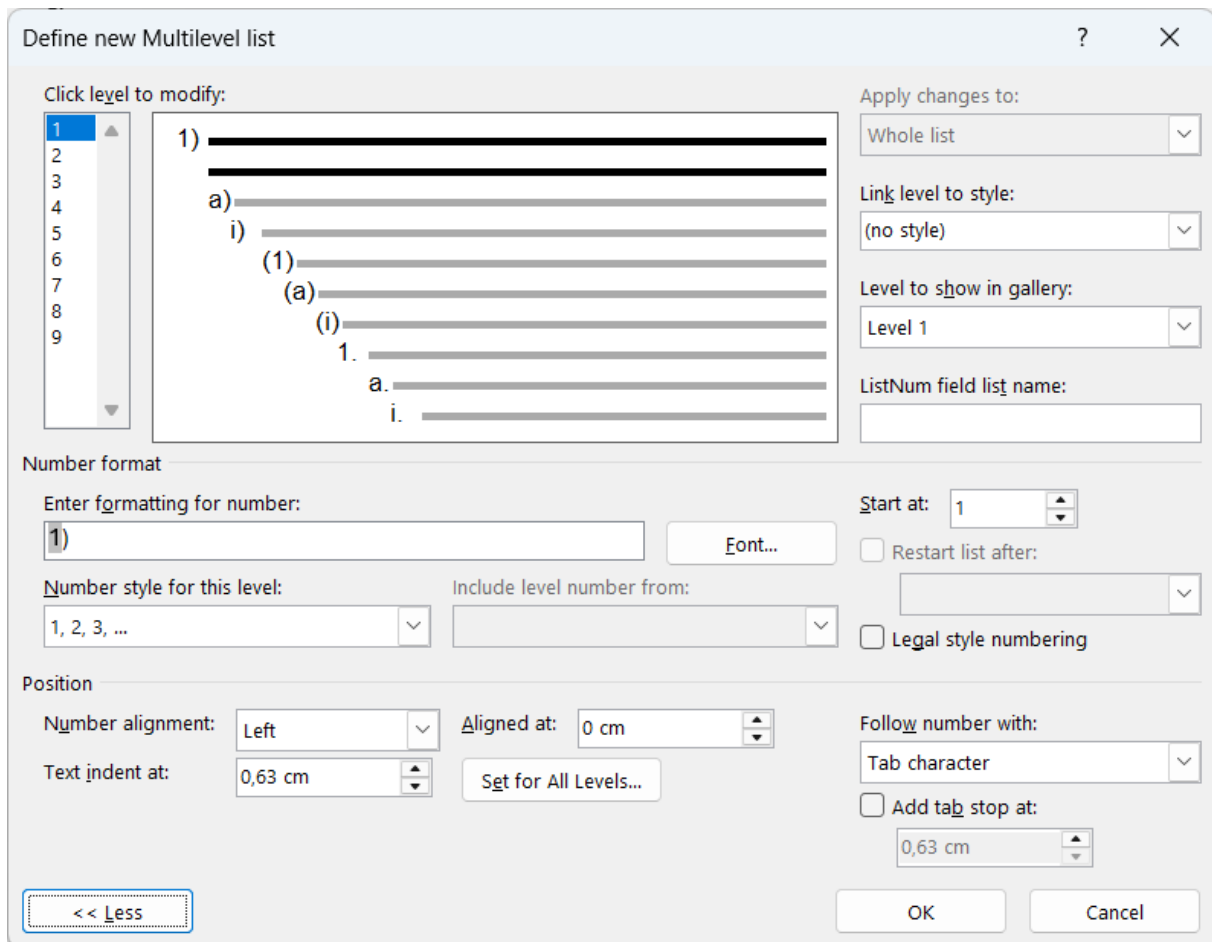


Figure 120 Dialogue box for New Multi-Level List.

So it seems like the only difference between going through the menu “*Define New Multi-Level List*” or the menu “*Define New List Style*”, is that the menu “*Define New List Style*” gives you the opportunity to give the Multi-Level List a name, as was shown in Figure 108.

I have to admit that the difference between these two ways to define a Multi-Level list is not very clear.

Microsoft advises the following:

- **Use a Multi-Level List when:** You are working on a single file, need fast custom outline numbering, or want to link specific list levels directly to built-in heading styles (Heading 1 through 9).
- **Use a List Style when:** You need to share consistent numbering rules across an entire team or organization via templates, or need to rename and centrally modify a complex multi-tier scheme.

In any case, if your Heading styles and multi-level list are not properly linked, you get classic Word numbering nightmares:

- Numbers restart unexpectedly in the middle of a document
- Copying content from another document breaks all the numbering
- The Navigation Pane shows headings correctly but numbering is wrong
- Numbers refuse to continue from a previous section

These problems happen because Word is treating the list numbering and the heading formatting as two separate, unrelated things.

I strongly prefer the bottom menu “*Define New List Style*”, because you can assign a name to the style and you can add it to the Word templates, to reuse it in other documents. It has also benefits to define multiple numbering styles, as we will show in the next section.

8.2. DEFINING MULTI-LEVEL LISTS WITH CUSTOM PARAGRAPH STYLES

A common example of two different numbering styles in one document is in case of appendices. Those might be numbered as Appendix A, Appendix B,.. and as sections A.1, A.2, etc.

If you want to define a similar hierarchy for these heading levels, you need to define a new List style. For this example, we will make use of the paragraph styles “*Appendix_L1*” (level 1), “*Appendix_L2*” and “*Appendix_L3*” that we already created in the chapter about paragraph styles (see also Figure 26).

The created paragraph styles are repeated here in Figure 121.

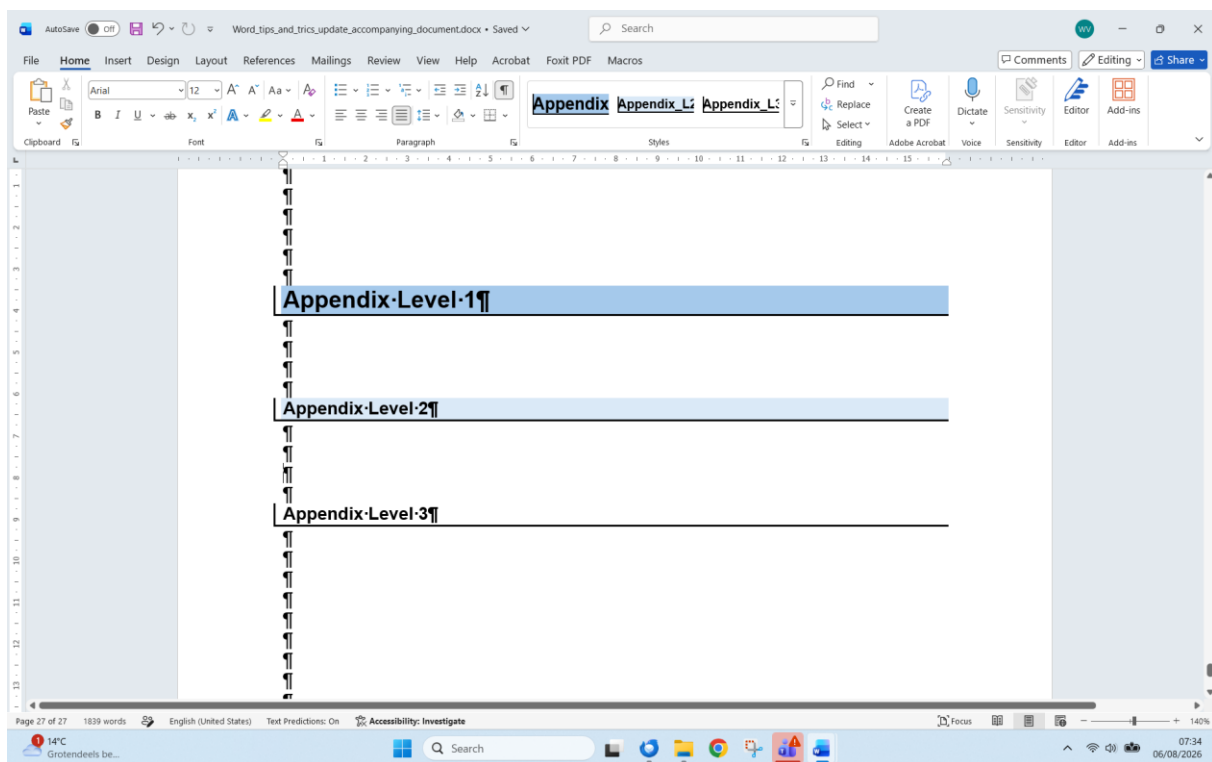


Figure 121 Creation of 3 paragraph styles for the 3 heading levels in Appendices.

We want to define a list style “*Appendix_list_style*” that relates the paragraph styles “*Appendix_L1*”, “*Appendix_L2*” and “*Appendix_L3*” to the following number styles:

- A. Appendix level 1
- A.1. Appendix level 2
- A.1.1. Appendix level 3

Following the same procedure as before, Figure 122 illustrates the definition of the new Multi-level list “*Appendix_list_style*”, with the three levels linked to the three paragraph styles for the appendices. Notice again the grey shaded numbers of the different levels in the box “*Enter formatting for number:*”.

The screenshot shows the 'Modify Multilevel list' dialog box. On the left, a list of levels from 1 to 9 is shown, with level 3 selected. The main area displays the list structure for levels 1, 2, and 3. Level 1 is 'A. Appendix_L1', Level 2 is 'A.1. Appendix_L2', and Level 3 is 'A.1.1. Appendix_L3'. The 'Enter formatting for number:' field is highlighted with a red box and contains the text 'A.1.1.'. The 'Apply changes to:' dropdown is set to 'Whole list'. The 'Link level to style:' dropdown is set to 'Appendix_L3'. The 'Level to show in gallery:' dropdown is set to 'Level 1'. The 'ListNum field list name:' field is empty. The 'Number format' section shows 'Enter formatting for number:' with 'A.1.1.' and a 'Font...' button. The 'Number style for this level:' dropdown is set to '1, 2, 3, ...'. The 'Include level number from:' dropdown is empty. The 'Start at:' spinner is set to 1. The 'Restart list after:' checkbox is checked and set to 'Level 2'. The 'Legal style numbering' checkbox is unchecked. The 'Position' section shows 'Number alignment:' set to 'Left', 'Aligned at:' set to '0 cm', and 'Text indent at:' set to '1,9 cm'. The 'Follow number with:' dropdown is set to 'Tab character'. The 'Add tab stop at:' checkbox is unchecked and the value is '1,9 cm'. The dialog has '<< Less', 'OK', and 'Cancel' buttons.

Figure 122 Definition of a new Multi-Level list “*Appendix_list_style*” with three numbering levels.

If you have finalized the creation of this second List style, you can click again on the drop-down list of the Multi-Level list icon in the Home Ribbon. The result is shown in Figure 123. If you hover over the list style, you can see its name “*Appendix_list_style*” popping up.

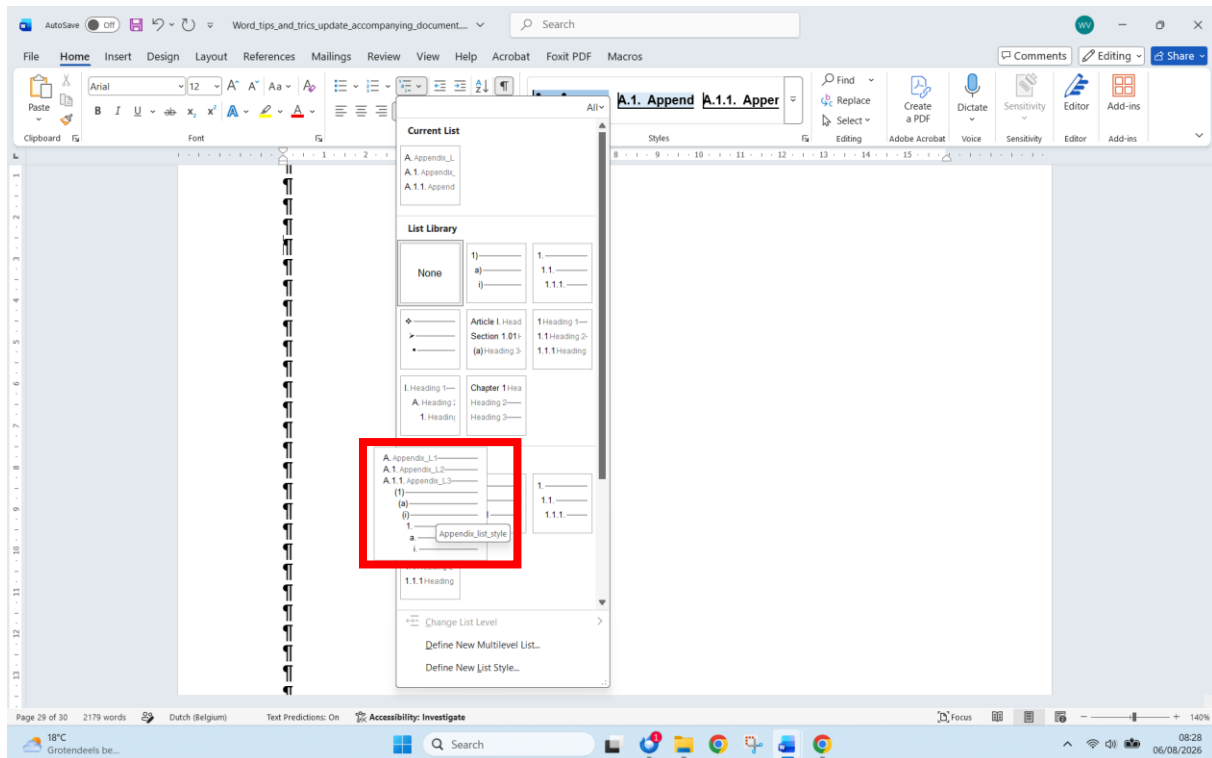


Figure 123 The new Multi-Level list “*Appendix_list_style*” appears in the dropdown menu of Multi-Level lists.

Compared to the Appendix paragraph styles in Figure 121, they are now being numbered with the numbering styles that we defined. The result is shown in Figure 124.

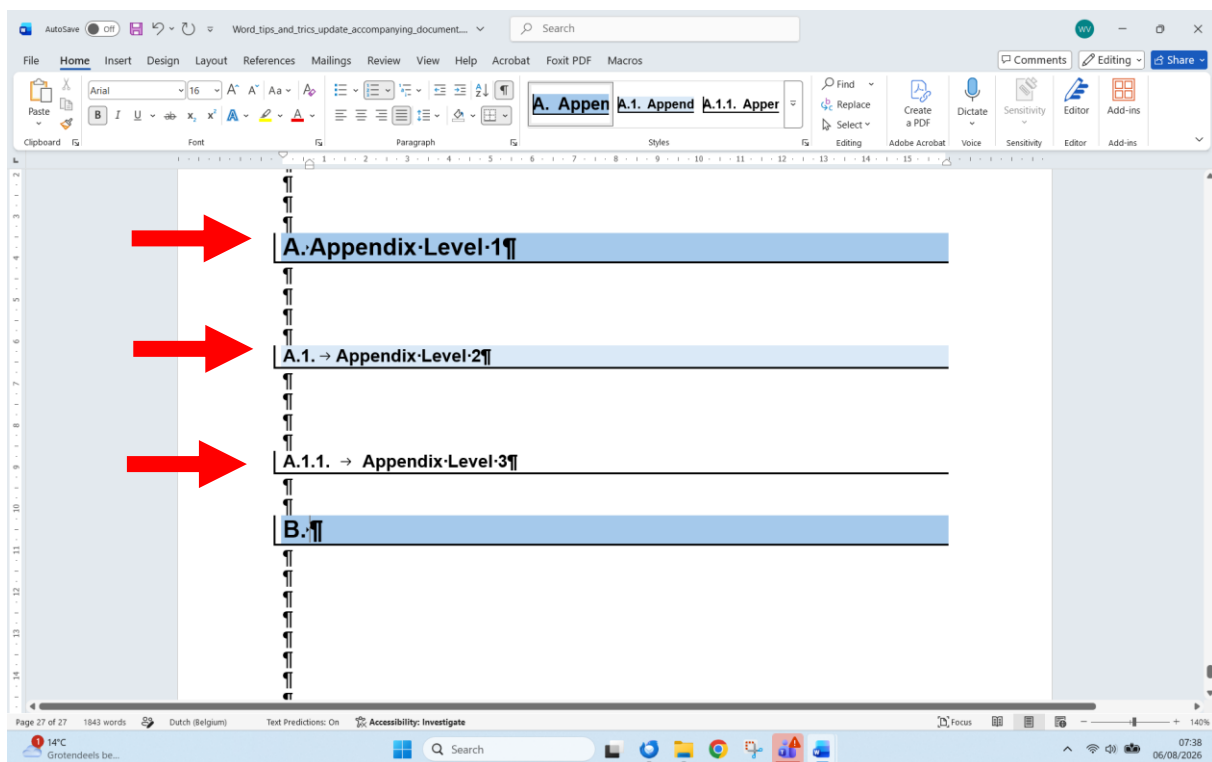


Figure 124 Numbering of the three heading levels of Appendices according to the Multi-Level list style “*Appendix_list_style*”.

It is very important to emphasize again that the creation of the Appendix paragraph styles “*Appendix_L1*”, “*Appendix_L2*” and “*Appendix_L3*” in section 2.4 was done without any numbering, and that the numbering should certainly NOT be added by the numbering icon in the Home Ribbon. Instead, it was defined now through the new Multi-Level list “*Appendix_list_style*”.

This example also shows why this List style can create so much confusion. It is not a style that you actively assign to a paragraph. As soon as the paragraph styles that you have linked in a Multi-level List, exist in a document, their hierarchy is also defined by the Multi-Level list that has been created “on top of the document”.

The big benefit of these coexisting Multi-Level lists “*Thesis_list_style*” and “*Appendix_list_style*” will become even more clear in the next chapter, where we learn how to automatically create a table of contents.

8.3. CROSS-REFERENCING TO NUMBERED SECTIONS

Now that the Multi-Level lists have defined the numbering styles of all heading levels, you can also insert cross-references to chapters, sections, subsections, etc., and those will also be automatically updated when the chapter or section changes its numbering.

You use the same menu as we used for cross-referencing to bookmarks and figures, because the cross-reference to a numbered section is again nothing else than a field code REF.

You insert a cross-reference to a chapter, subchapter, section, subsection, etc. from the menu *References > Cross-reference*. The dialogue box in Figure 125 opens. You click on the dropdown list of the “*Reference type:*” label and select the item “*Heading*”. Then all detected heading levels with their titles are shown in the window below. Next, you can choose what type of cross-reference you want to insert, under the label “*Insert reference to:*”. You can refer to the heading text, to the heading number or to the page number where the heading can be found. As indicated in Figure 125, we have chosen in this case to refer to the heading number.

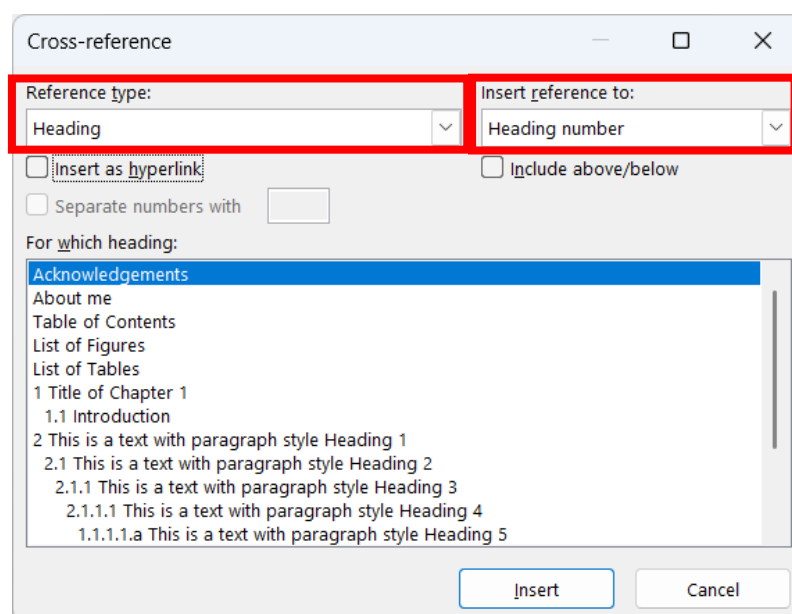


Figure 125 Insert a cross-reference to a built-in heading level in the document.

If you have defined user-defined paragraph styles for your chapters, subchapters, sections, subsections, etc., they will not appear under the reference type *Heading*, because Word only lists there the information from built-in Heading styles. However, if you have numbered all your chapters and sections with custom paragraph styles, you can still insert cross-references to them, by changing the reference type to “*Numbered item*”, as shown in Figure 126. As you can see, Word lists then all numbered items in the document. Also numbered lists that you inserted in any paragraph, will also show up there.

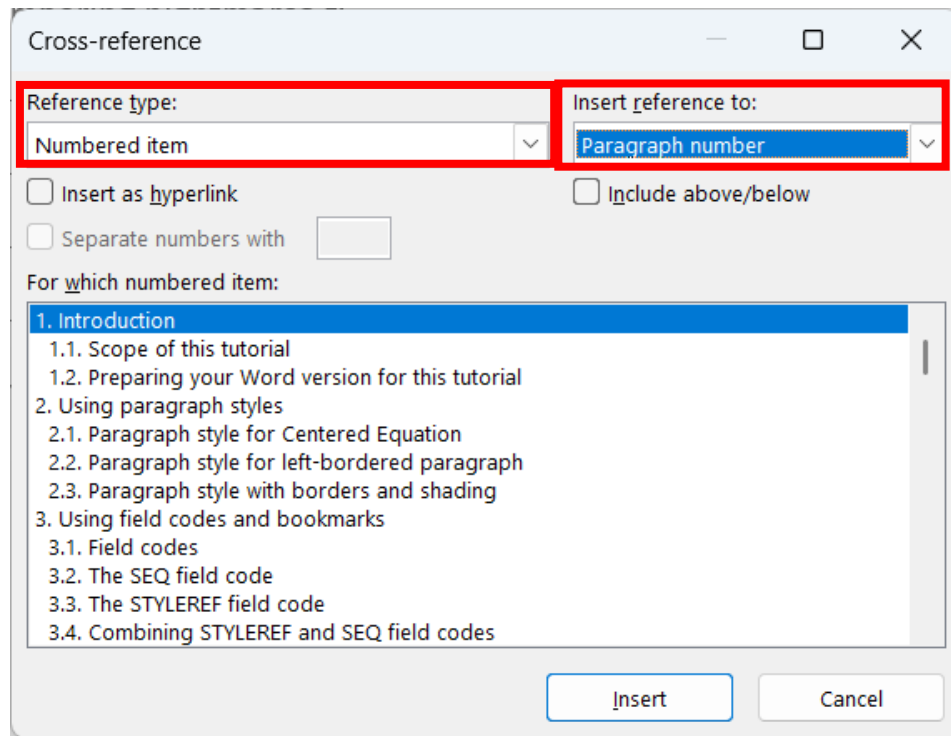


Figure 126 Insert a cross-reference to custom heading levels in the document.

As I mentioned above, the information of a cross-reference is again nothing else than a field code. If you press <Alt>+<F9>, you will see that every cross-reference is a field code of the type REF, as shown in Figure 127.

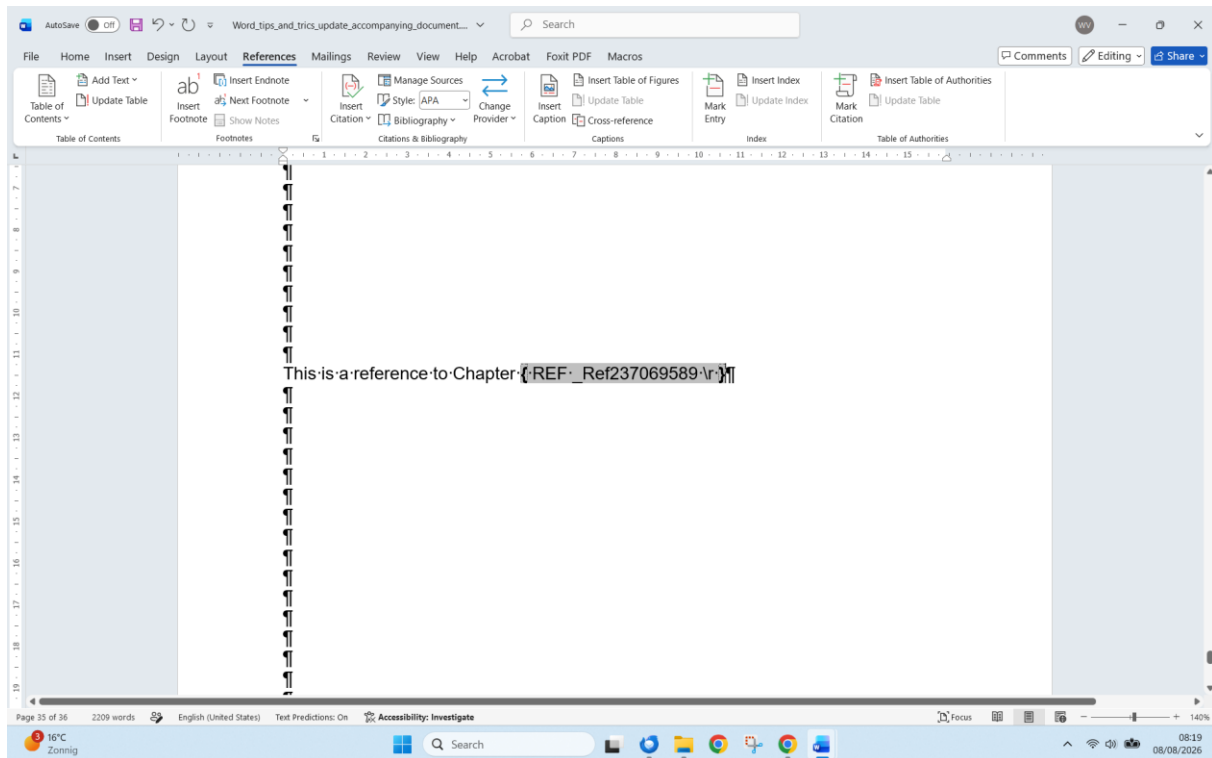


Figure 127 The underlying field code of a cross-reference to a heading or numbered item.

This chapter introduced the concept of Multi-Level lists. It is one of the most important features to know if you work with large documents with a lot of numbered chapters and sections. And it is very important to understand that level numbering should never be included while creating a new paragraph style or by using the numbered list icon, but always be defined by the Multi-Level list which lies on top of it.

If those Multi-Level lists are set-up correctly, creating an automatic Table of Contents is a matter of seconds, as will be shown in the next chapter.

9

Automatically generating a table of contents

In the previous chapter we have introduced Multi-Level Lists for numbering chapters, subchapters, sections, subsections. If you combine that with automatic page numbering for the different sections, then generating a table of contents is a piece of cake.

9.1. TABLE OF CONTENTS WITH BUILT-IN HEADING STYLES

In this first example, we will automatically generate a table of contents for a Multi-Level list that is defined with the built-in paragraph styles *Heading 1*, *Heading 2*, *Heading 3*, etc. Click *References* > *Table of Contents* tab and the dialogue box in Figure 128 appears.

When you insert an automatically generated table, MS Word uses by default the *Heading* styles 1, 2 and 3 for generating the table of contents and automatically includes the page numbers.

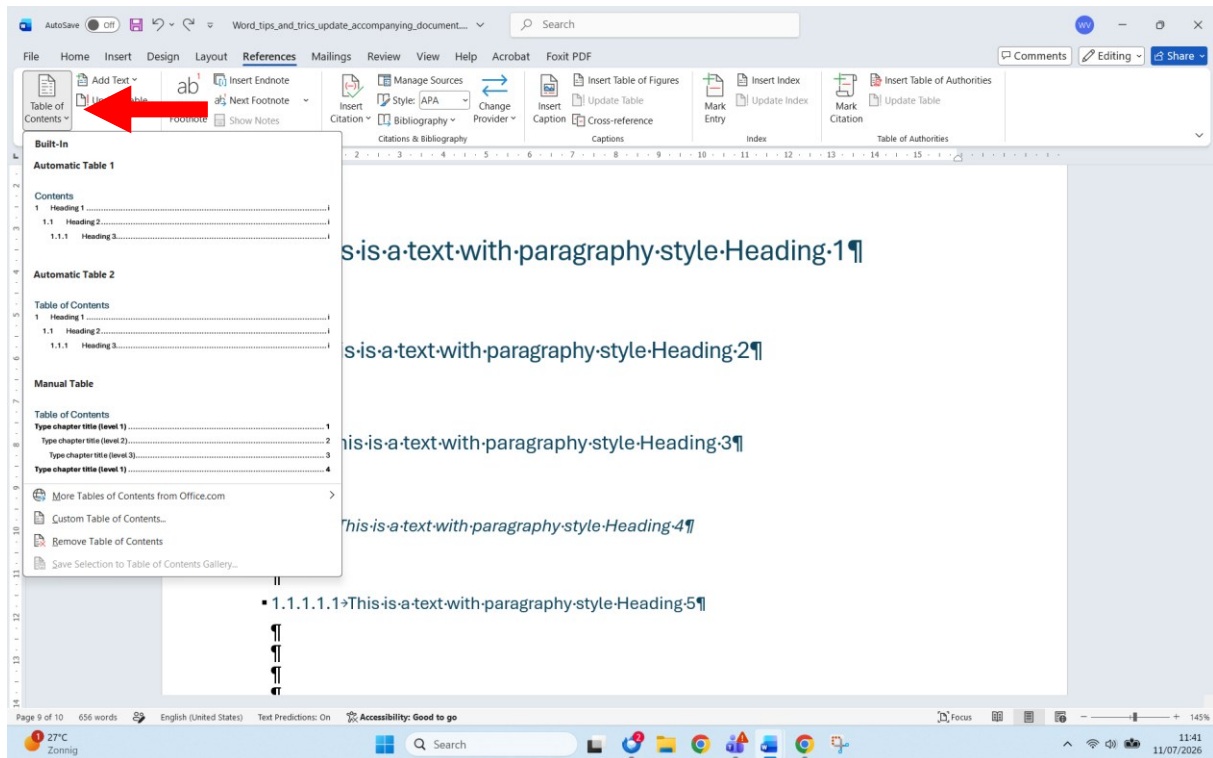


Figure 128 The Table of Contents dialogue box.

Figure 129 shows how the Table of Contents looks like, when default settings are used. You see that the lines of the table of contents are again shaded with a grey background colour, because these are again field codes that cross-reference to the titles of the different headings.

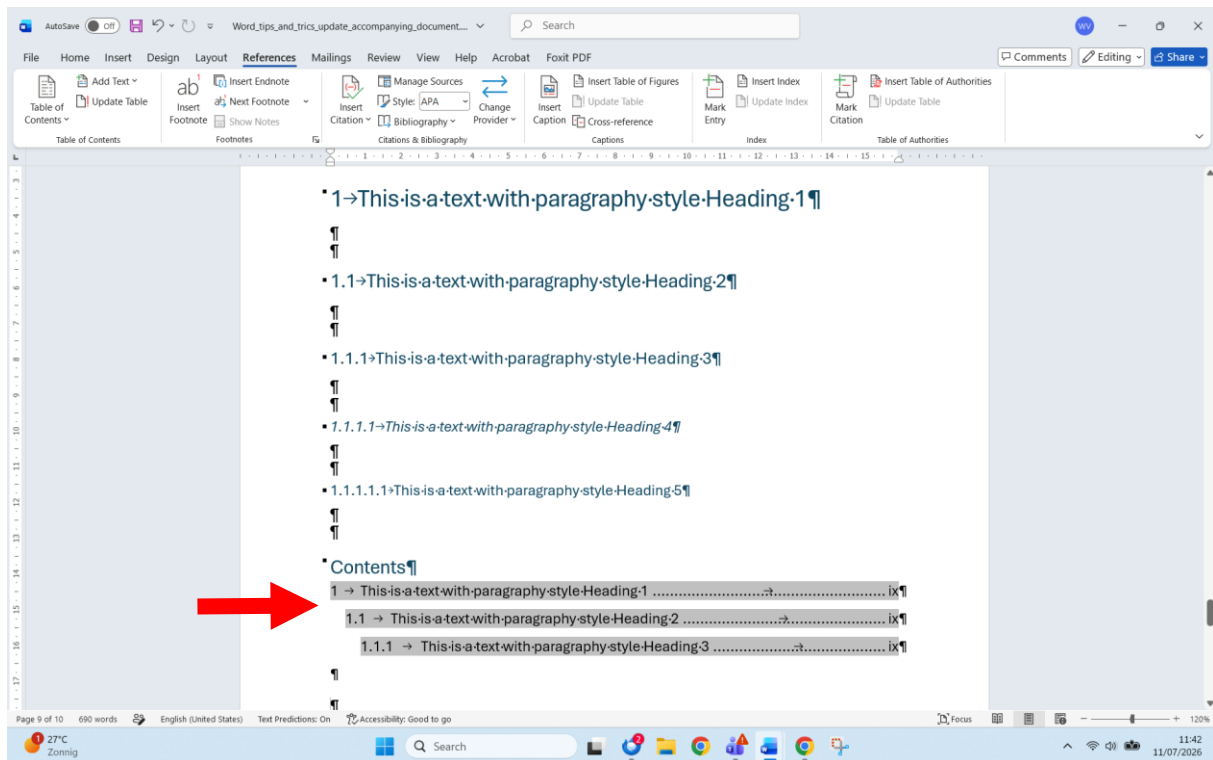


Figure 129 Automatic generation of the Table of Contents with the default settings.

If you press <Alt>+<F9> to see the field codes, you will see that the table of contents is also nothing more than a field code.

Indeed, if you toggle the field code with <Alt>+<F9>, you will see the following field code (Figure 130):

```
{ TOC \o "1-3" \h \z \u }
```

If you look up the help information for the field code TOC, you can find what all the different switches mean (search for keywords “field code TOC”).

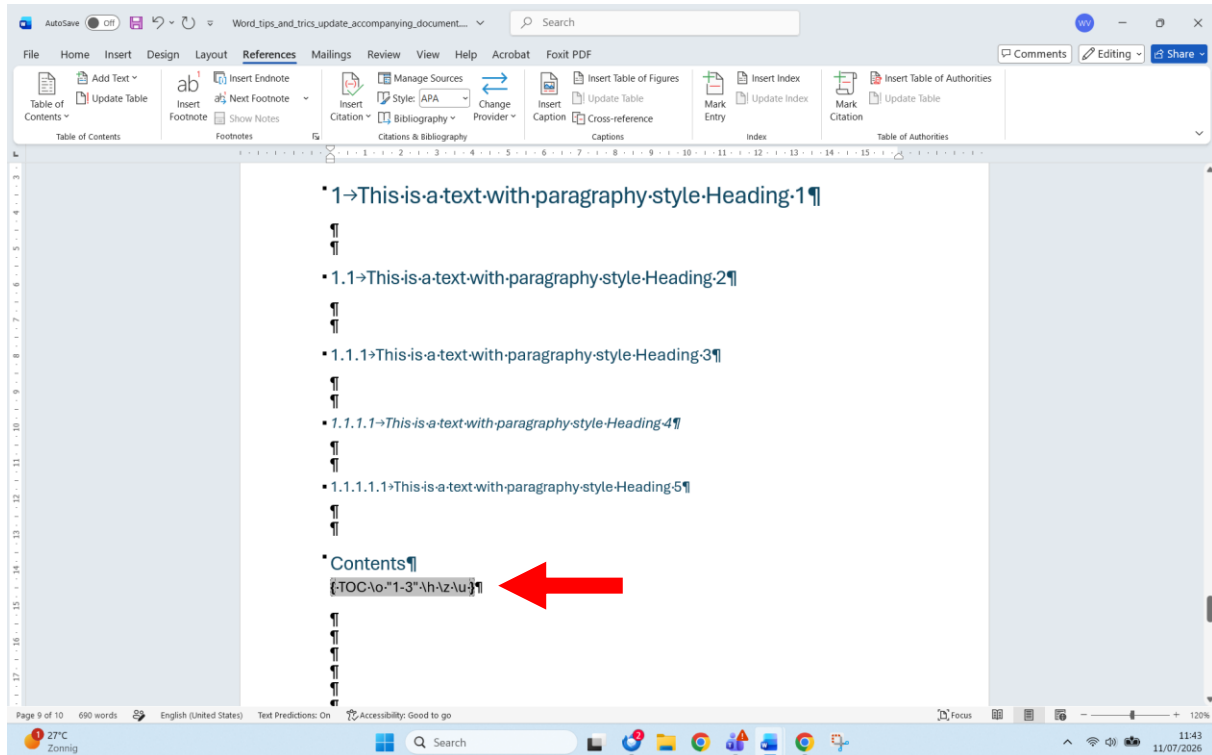


Figure 130 Pressing <Alt>+<F9> shows the field code for the Table of Contents.

Now, we will show how to define a custom Table of Contents. Click again on the *References* > *Table of Contents* button and the drop down list in Figure 131 appears. This time, pick the option “Custom Table of Contents...”.

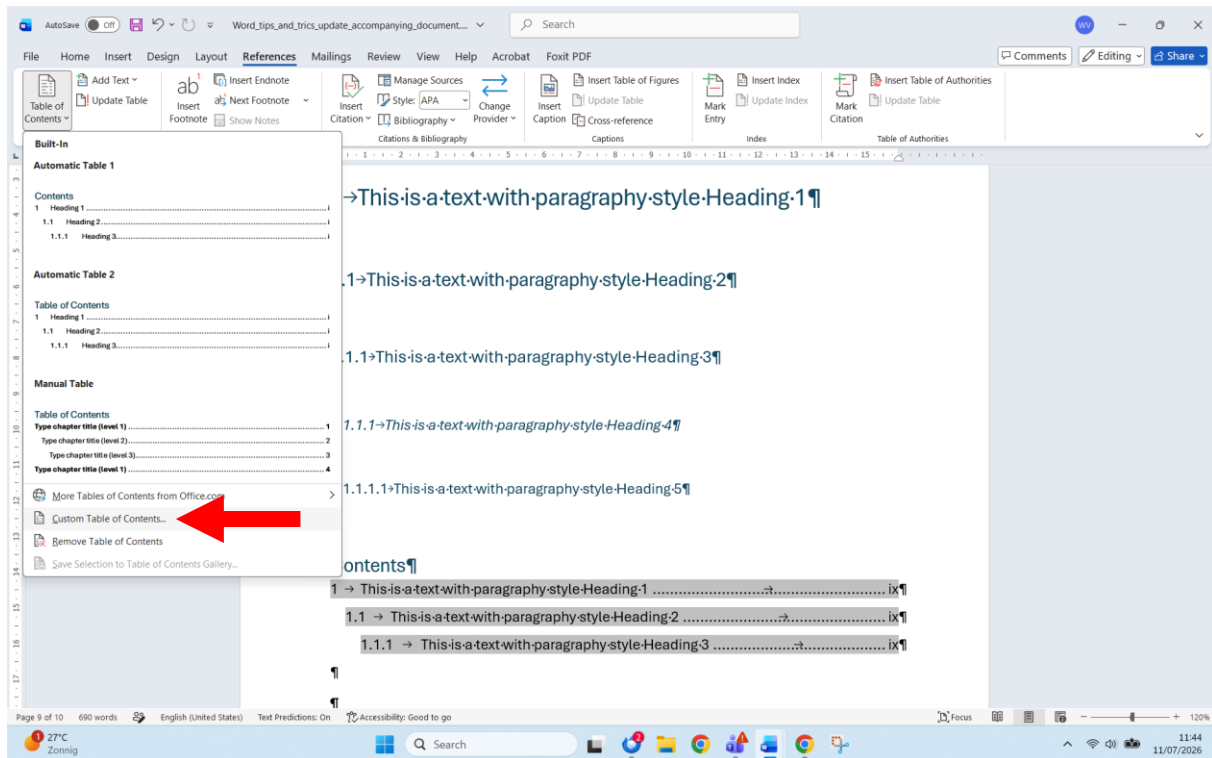


Figure 131 Selection of Custom Table of Contents.

The dialogue box in Figure 132 appears.

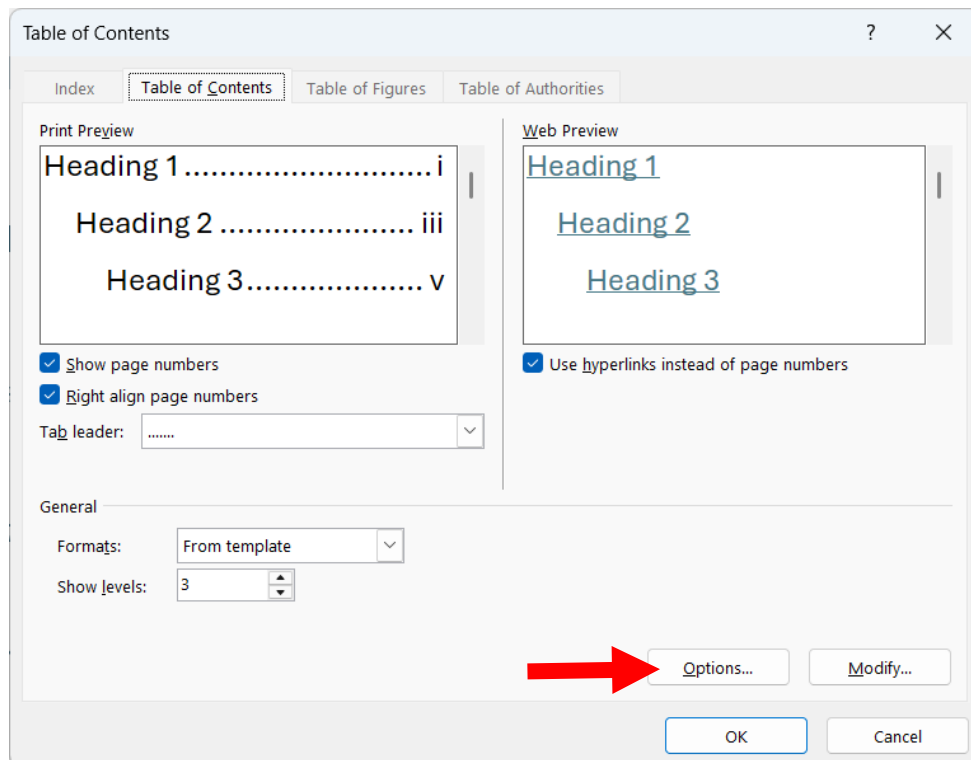


Figure 132 Changing the styles for generating the table of contents.

In this dialogue box, you can do much more:

- you can select different formats for the layout of the table of contents,
- you can increase or decrease the number of levels that are included in the table of contents (default is 3). If you choose for example 5, then subsections and subsubsections with *Heading 4* and *Heading 5* style are also indexed in the table of contents,
- you can build a table of contents based on your own styles. If you click on the *Options...* button in the right bottom corner of Figure 132, the *Table of Contents Options* dialogue box is opened, as shown in Figure 133. You can also here increase the number of indexed levels to 5, as shown in the next Figure 134.
- if you remove the number 1 to 3 from *Heading 1*, *Heading 2* and *Heading 3*, and you assign these numbers to different styles, then these styles will be used to build up the hierarchy of the table of contents.

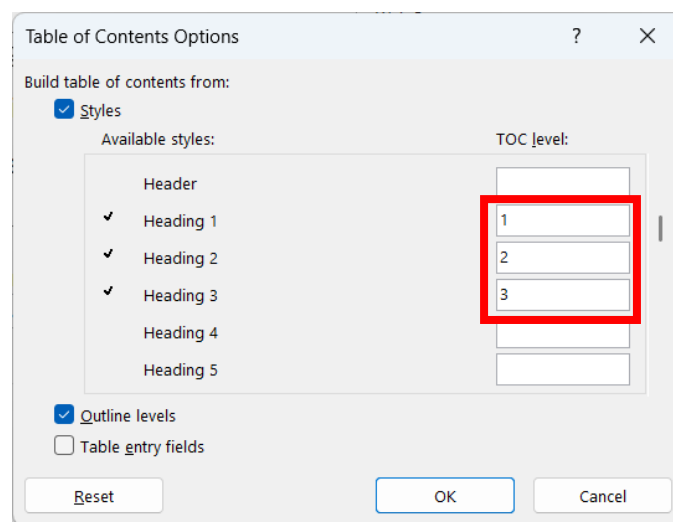


Figure 133 Table of Contents Options dialogue box.

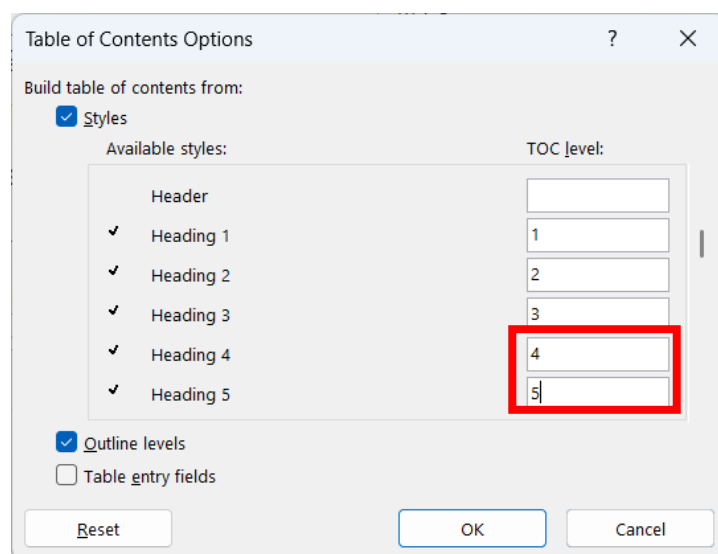


Figure 134 Selecting 5 levels for indexing the table of contents.

When you now return to the document, you see that the table of contents shows 5 heading levels, as illustrated in Figure 135.

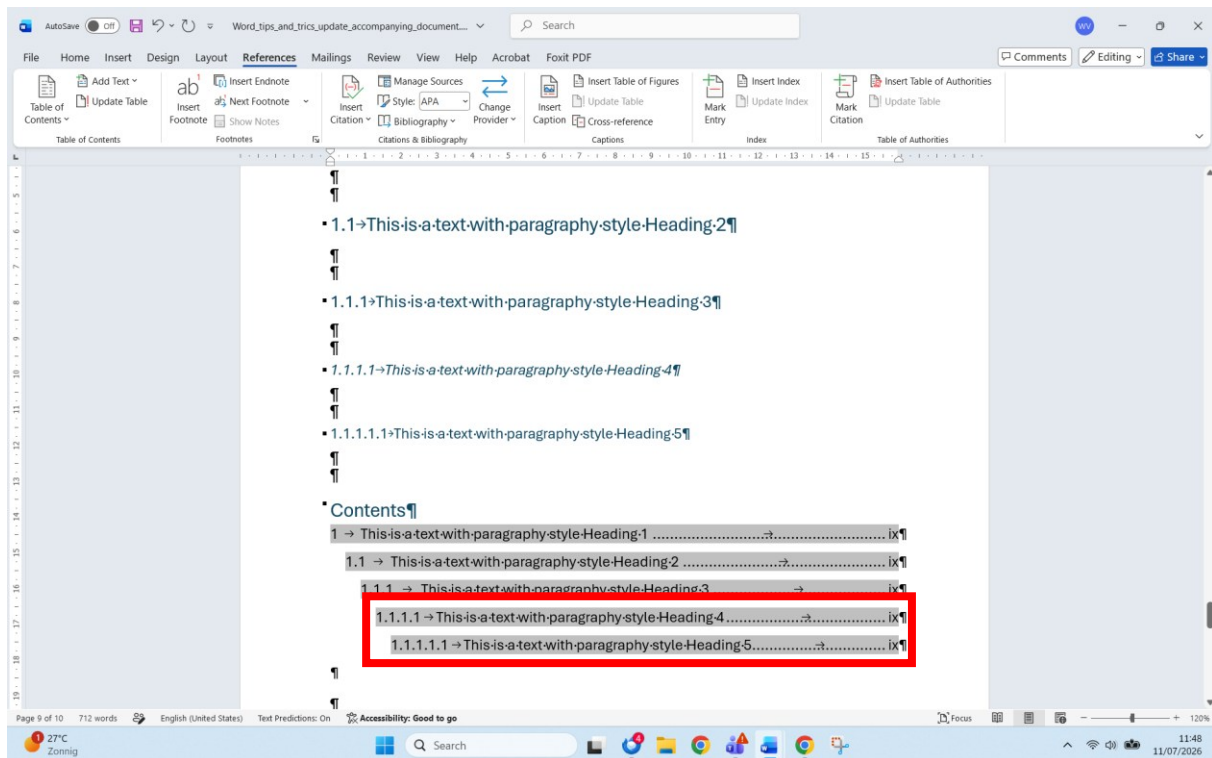


Figure 135 Table of Contents with 5 heading levels instead of 3.

Finally, it can happen that you want to transfer the table of contents to a separate file. If you do that, all the linked information that this field code needs to build up the table of contents, is lost. That is why you need first to transform the field code into regular text. Therefore you select the table of contents text and then press <Ctrl>+<Shift>+<F9>. Then you will see the text as shown in Figure 136.

You might not have expected that the text would turn blue and underlined, but this is a consequence of the fact that you used the default option *Use hyperlinks instead of page numbers* when creating the table of contents (see Figure 132). If you do not select this check box, the field code will convert to regular text.

Since this is regular text, you can also change again its font type, font size, alignment, etc.

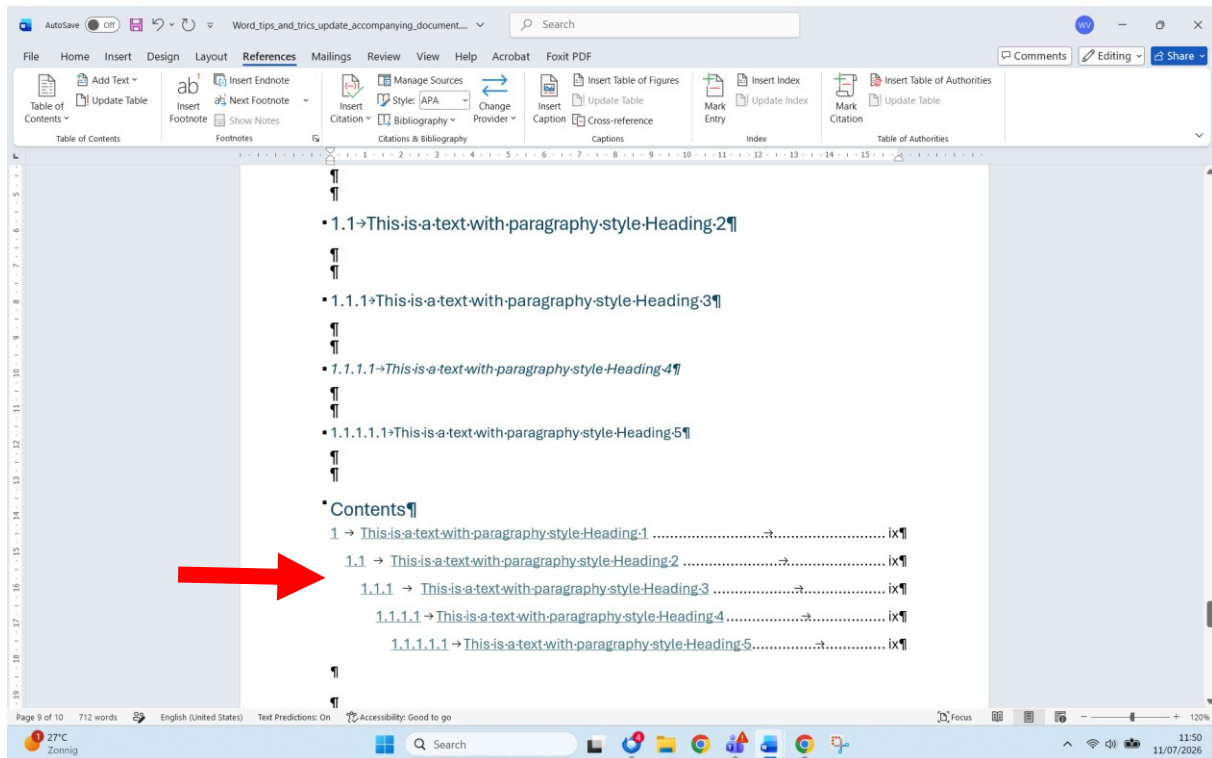


Figure 136 Converting the TOC field code to regular text.

9.2. TABLE OF CONTENTS WITH CUSTOM HEADING STYLES

In this section, we will show why the proper definition of Multi-Level lists in the previous chapter is so important to create an automatic Table of Contents.

In the previous chapter, we have created the coexisting Multi-Level lists “*Thesis_list_style*” and “*Appendix_list_style*”. Both List styles start with heading level 1, then heading level 2, heading level 3, etc., but their numbering style is different and the linked paragraph styles are different.

To automatically create one table of contents that includes both the chapters and the appendices, we insert again a Table of Contents, and we click again the *Options...* button, like in Figure 132. This time we link the three levels of the paragraph styles *Appendix_L1*, *Appendix_L2* and *Appendix_L3* respectively to Level 1, Level 2 and Level 3, as shown in Figure 137. Then we scroll further in the list of paragraph styles and we also link the built-in paragraph styles *Heading 1*, *Heading 2* and *Heading 3* respectively to Level 1, Level 2 and Level 3, as shown in Figure 138.

So Level 1 of the Table of Contents is linked to two paragraph styles at the same time: *Heading 1* and *Appendix_L1*. The same applies to Level 2 and Level 3.

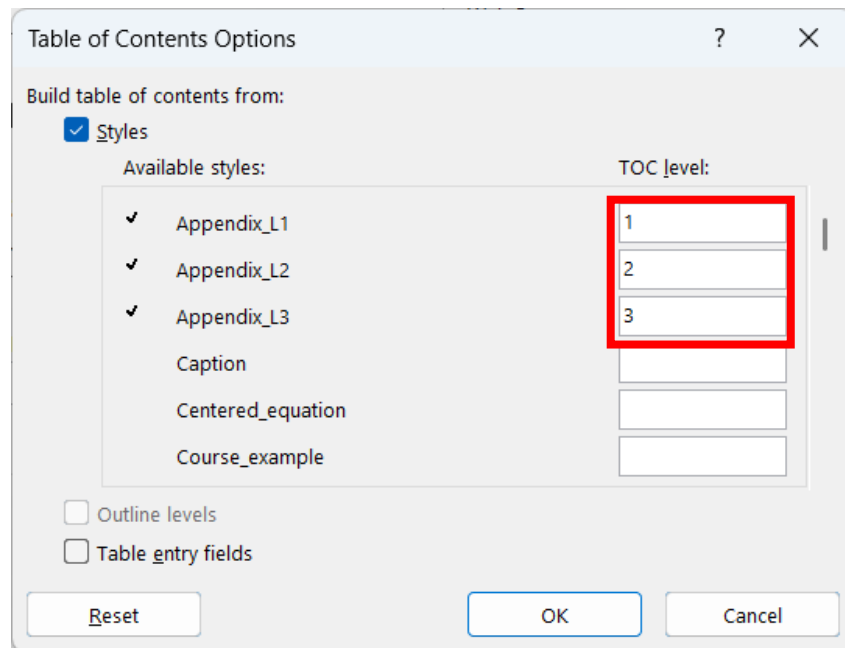


Figure 137 The three paragraph styles for the heading levels of Appendices are linked to the three first levels of the Table of Contents.

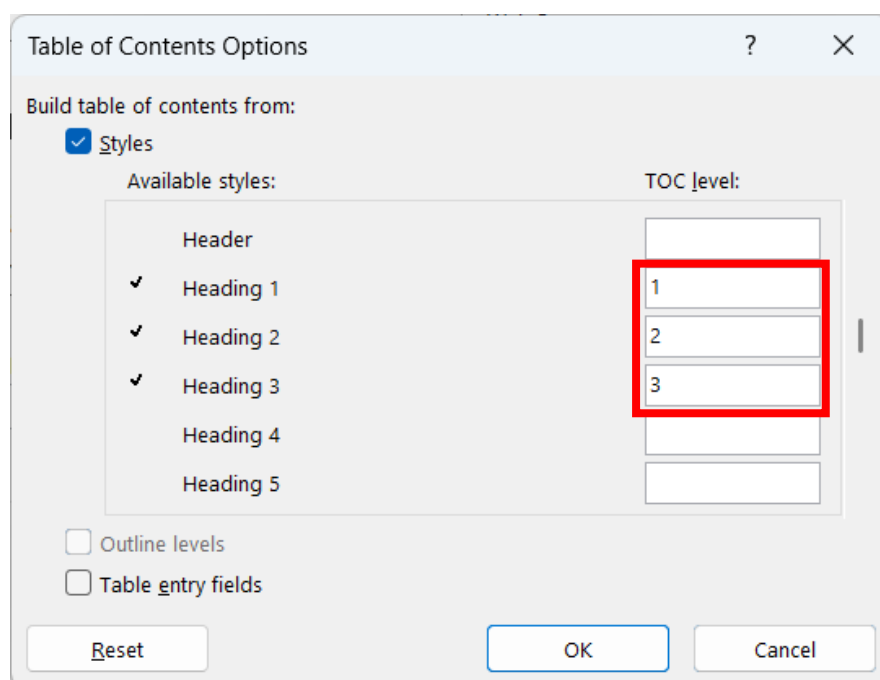


Figure 138 The three paragraph styles for the built-in *Heading* styles of the document are also linked to the three first levels of the Table of Contents.

If we now insert the Table of Contents, the result looks like Figure 139. So both the chapters and the appendices are included in the Table of Contents, and more importantly, the numbering style is taken from the Multi-Level lists “*Thesis_list_style*” and “*Appendix_list_style*”.

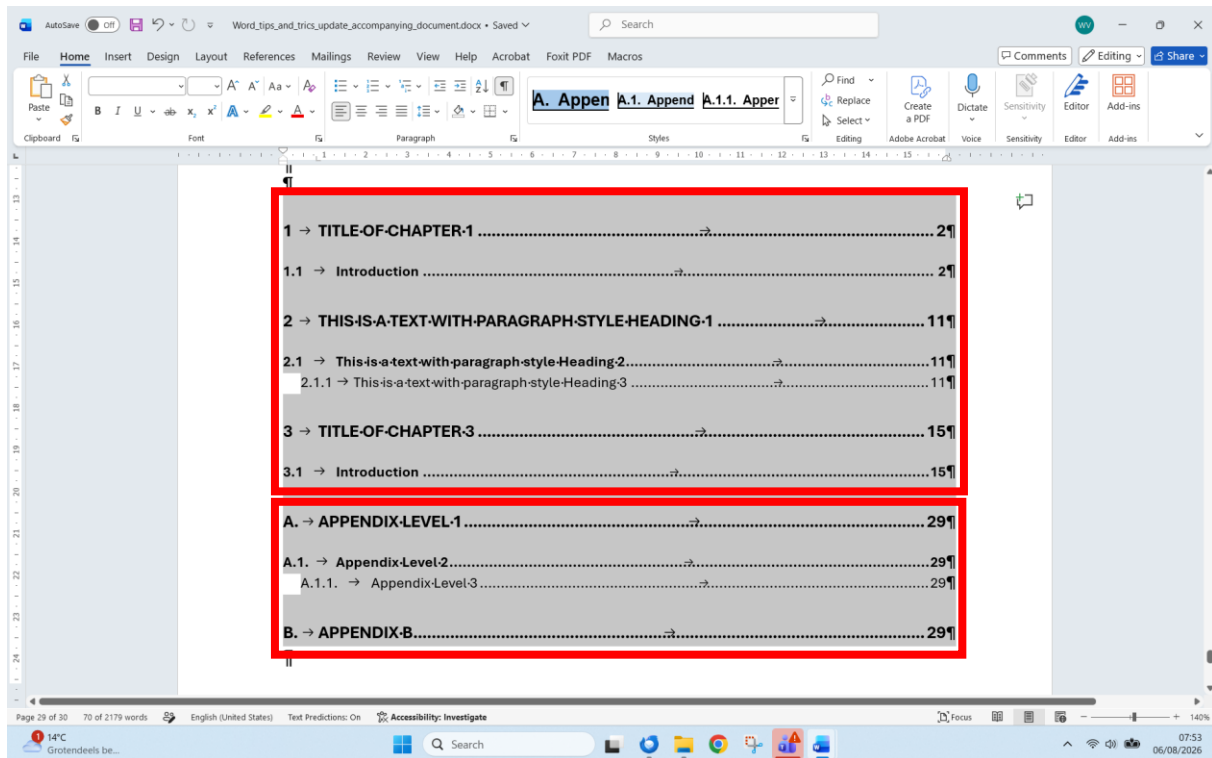


Figure 139 The Table of Contents shows both the first three levels of the built-in *Heading* styles and of the *Appendix* paragraph styles.

9.3. ADDING FRONT MATTER TO THE TABLE OF CONTENTS

The sections that come before Chapter 1 in a thesis or manuscript are collectively called the front matter (or preliminary pages).

Typical front matter includes:

- Title page
- Abstract
- Preface
- Acknowledgements
- Table of Contents
- List of Figures
- List of Tables
- List of Abbreviations
- Glossary (if applicable)

Typically, the Table of Contents is preceded by a number of unnumbered sections, but you might wish to include headings like Acknowledgements, Table of Contents, List of Figures, etc. in the Table of Contents. So at first sight, this seems to conflict with our defined Multi-Level list, where the built-in heading styles *Heading 1*, *Heading 2*, *Heading 3*,... are numbered.

You can easily achieve this by respecting a few general guidelines:

- First insert a section break (see Chapter 6) before Chapter 1
- Deactivate “*Link with previous*” for the header and footer of Chapter 1, so that the front matter can have separate numbering
- Insert page numbers for the front matter, but change the numbering style to i, ii, iii,...

- You might insert another section break for the title page, to have no numbering at all for the title page
- You include different pages for the Acknowledgements, Table of Contents, List of Figures, etc.

For the titles of these front matter pages, you just assign the paragraph style *Heading 1* as you have done for all chapter titles. Obviously this title will automatically be numbered, because you have defined the numbering in the Multi-Level list “*Thesis_style_list*” above.

If we take the example of the section Acknowledgements, which will typically come first, it will be numbered as “1. Acknowledgements”. If you just put your cursor inside the title “*Acknowledgements*”, you will see that the button “*Numbered list*” on the Home Ribbon is activated, because this style is numbered.

This is illustrated in Figure 140, where you can see that the second button “*Numbering*” in the *Paragraph* toolbar of the Home Ribbon is highlighted. You can see it in detail in Figure 141.

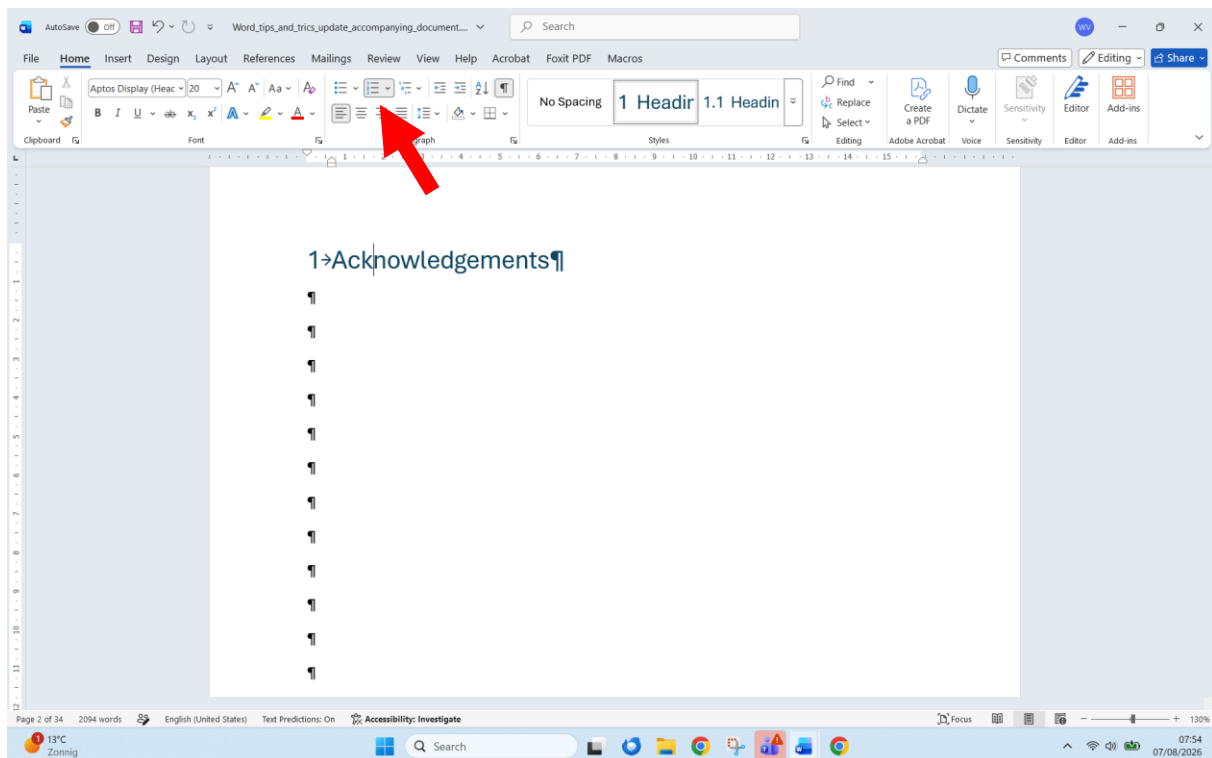


Figure 140 Front matter titles also get assigned the built-in paragraph style *Heading 1*.



Figure 141 The icon for “*Numbered list*” is highlighted.

Remember from Chapter 8 that we said to NEVER number chapters and sections with this button, but always use the third button “*Multi-Level lists*”. That is still correct, but to UNNUMBER one single section, you can use this second button “*Numbered list*”. Indeed, for

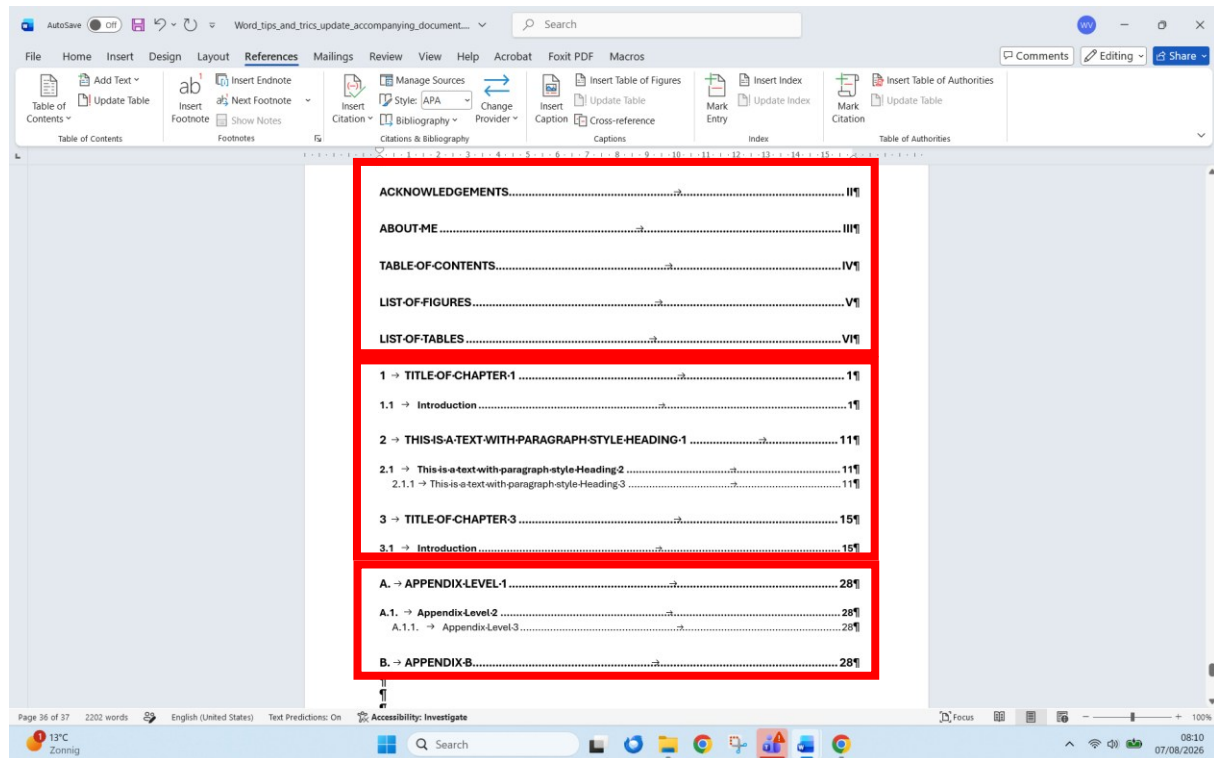


Figure 143 The Table of Contents shows both the front matter, the first three levels of the built-in *Heading* styles and of the Appendix paragraph styles.

Now that we understand how heading styles are used to build a table of contents, we can make use of this knowledge to generate more creative chapter headers and chapter opening pages.

10

LaTeX like chapter titles

If you like the typical LaTeX chapter layout, or any other fancy layout for chapter titles, the problem is that your *Heading 1* cannot be used. However, if *Heading 1* is missing, the numbering of *Heading 2*, *Heading 3*, *Heading 4*, etc. is not correct, and also the Table of Contents will not find the chapter titles. Furthermore, if you want to use the chapter title in the header of each page, that will also not work.

There is an easy trick for this. We will still insert the chapter title with *Heading 1*, but hide only the heading number with a white overlay rectangle (white font colour does not work here, because then, also the chapter title itself is turned invisible).

For the chapter title layout used in this document, the chapter number is inserted on a separate line with a very large font and grey colour. Below that large chapter number is the title of the chapter, inserted with the *Heading 1* style and numbered with the chapter number. The result is shown in Figure 144 (the grey symbol after the number 1 is the paragraph mark, that is also as large as the font size).

The only extra thing you need to do, is to put a white rectangular shape over the chapter number in front of the chapter title (in *Heading 1*), so that that chapter number is hidden for the reader. Since the chapter number is of course there, Word can refer to it in STYLEREF fields in header or footer, and also index it in the Table of Contents.

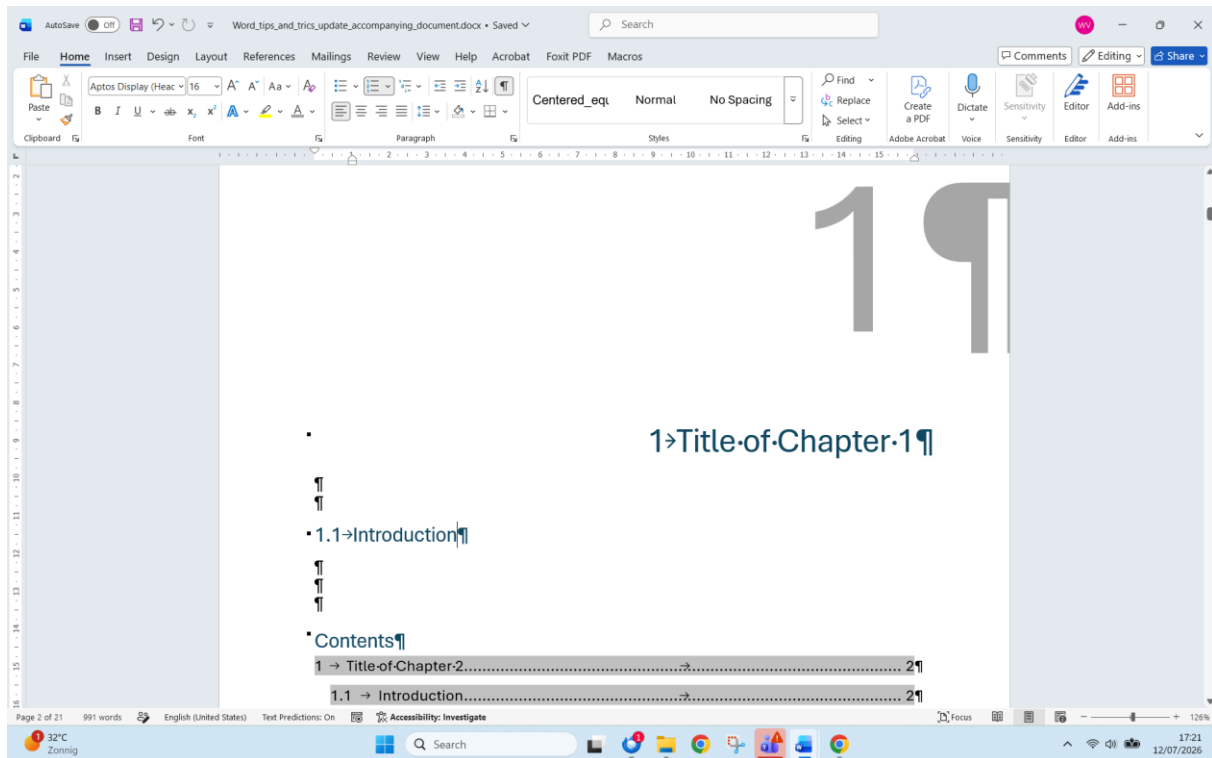


Figure 144 LaTeX like layout of the chapter header and opening page.

If you also want full freedom for the layout of the text of the chapter title itself, you have another variant of this trick.

In Figure 145 below, you see again the LaTeX like style for the chapter number and title, and you see on top still the built-in paragraph style *Heading 1* with the title of the Chapter 1. The *Heading 2* level is correctly numbered 1.1., since it finds back the *Heading 1* style above.

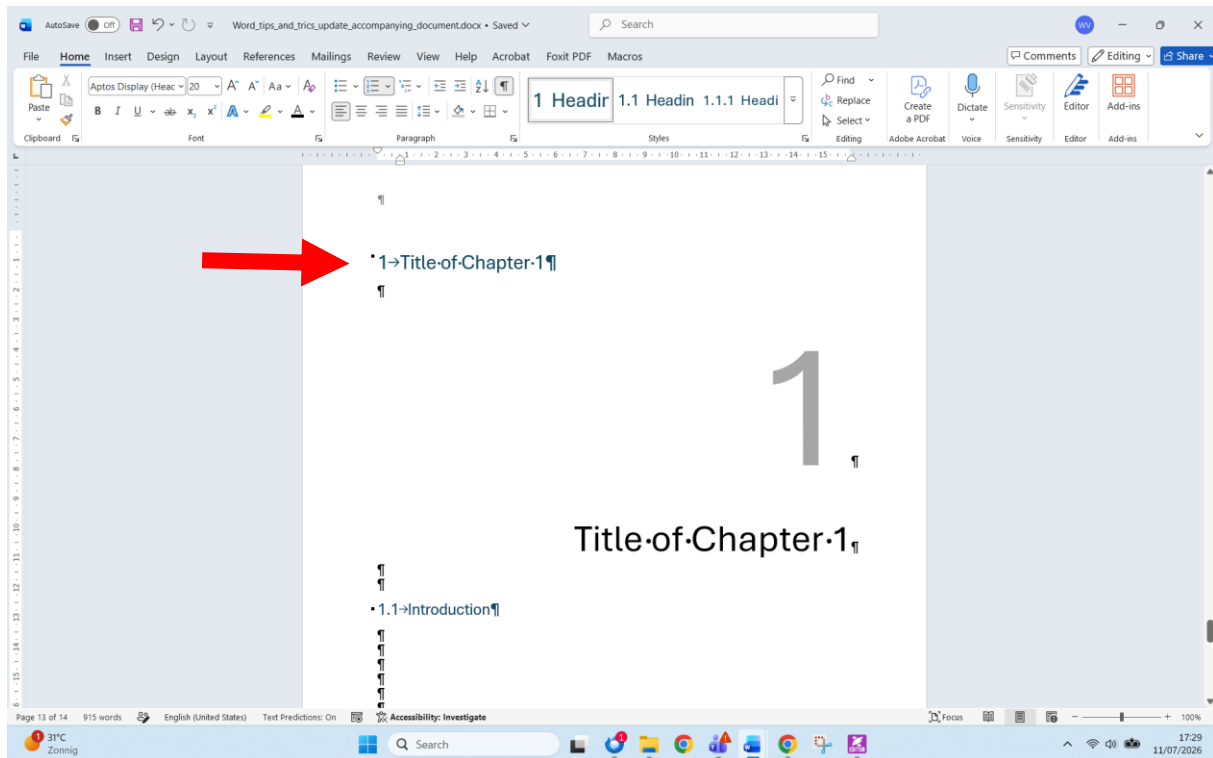


Figure 145 Variant of the LaTeX like layout of the chapter header and opening page.

The only thing you need to do again, is select the first line of the page with the built-in paragraph style *Heading 1* and chapter title, and change the font colour to white. The text becomes invisible, but Word can perfectly work with it.

The visible chapter title, which is below the large, grey chapter number, can now have any layout you want. It is again advised to create a specific paragraph style for these chapter titles, so that you can change them all at once, if necessary.

When you now create an automatically generated table of contents, the chapter titles are correctly there, as illustrated in Figure 146.

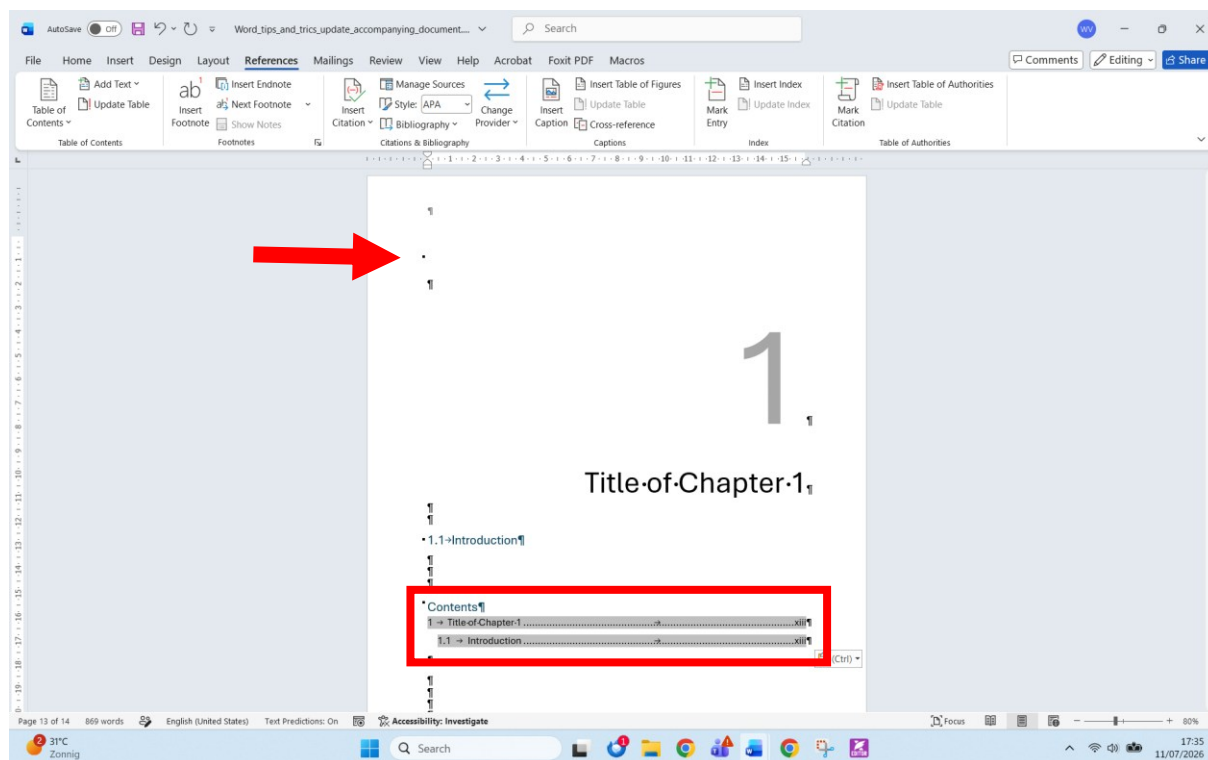


Figure 146 Generation of Table of Contents, with white font colour of the chapter title with built-in paragraph style *Heading 1*.

We have discussed already a large number of features for layout, paragraph and equation numbering, formatting styles and field codes.

Many of these tasks can still be automated to a large extent, so that a lot of consecutive steps and clicks in Word are automated into one mouse click. This can be done with macros.

11

Using macros

If you are not yet convinced of using all the features discussed above, you can make it even more efficient and faster by using macros.

A macro is an action or a set of actions that you can use to automate tasks. When you record a macro, you perform a series of actions in MS Word and MS Word converts this sequence into an executable program written in the *Visual Basic for Applications (VBA)* programming language. Although you can edit this program code with the *Visual Basic Editor*, you do not need any programming skills to record and execute macros. You can execute the macro by clicking a newly defined button on the toolbar or by pressing a newly defined shortcut key.

Below we discuss a few examples to show the power of using macros. We will record a macro to insert a centered and numbered equation, one of the missing features where Word is often criticized for. After that, we will even add the chapter number to the equation number, all just with a single click.

11.1. MACRO FOR INSERTING A CENTERED AND NUMBERED EQUATION

In this section, we will create a macro which combines several topics we learnt in the previous chapters. The macro *Insert_centered_equation* will execute by clicking one single button on the toolbars and will insert a centered equation with a right aligned automatically generated equation number. The following actions must be performed:

- assign the paragraph style *Centered_equation* (see Figure 14 and Figure 15)
- jump to the right aligned tab at the right margin
- insert the SEQ field code to number the equation automatically
- start the Microsoft Equation Editor at the centered tab position.

In this way, we can insert centered and automatically numbered equations in our articles, reports, theses, etc. with a single mouse click.

Let us now move to the macro recording. You can find the macro toolbox on the toolbar *View* > *Macros* as shown in Figure 147. Select the option *Record Macro...*

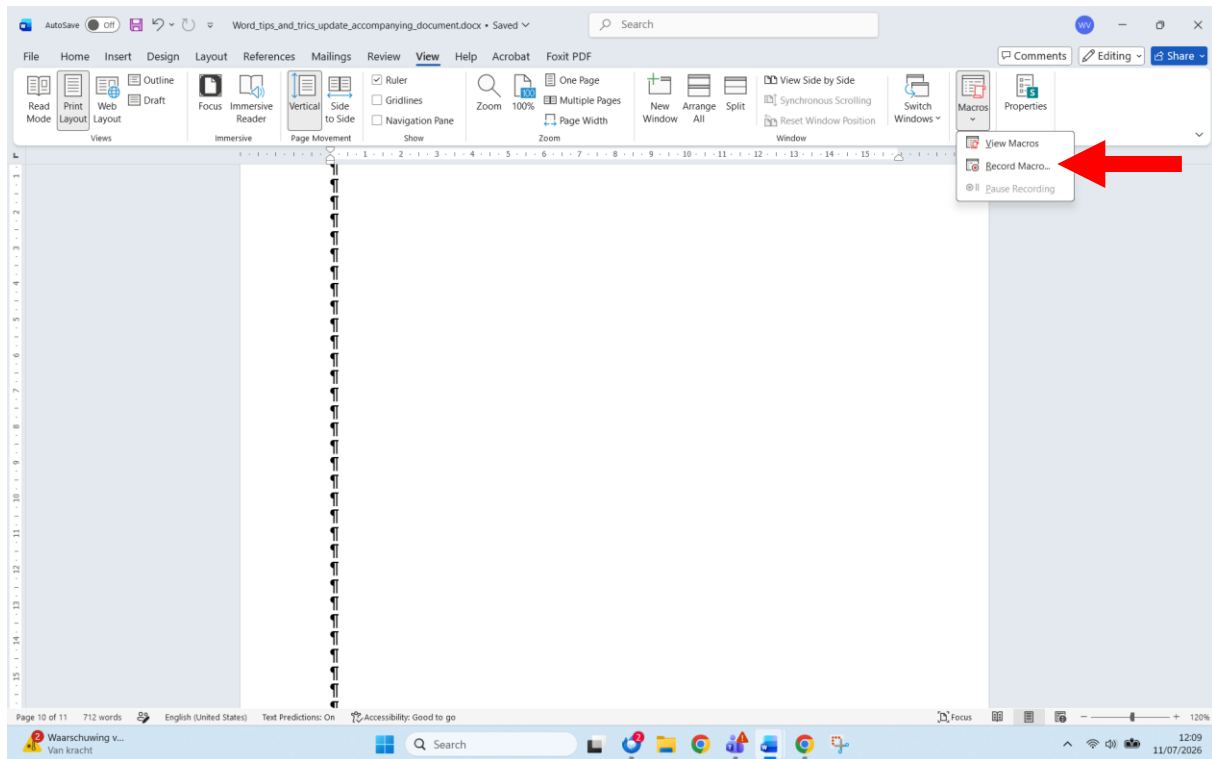


Figure 147 The Macros toolbar.

The *Record Macro* dialogue box in Figure 148 appears.

First enter the name *Insert_centered_equation* in the *Macro name:* text box.

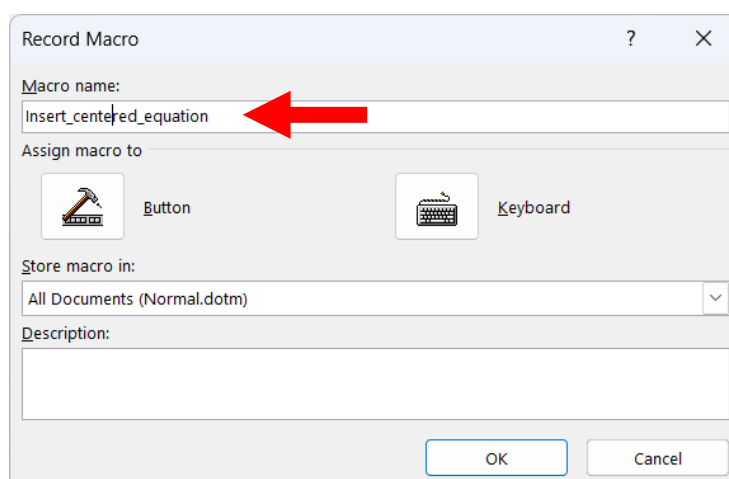


Figure 148 The *Record Macro* dialogue box.

Now you can choose to assign the command for executing the macro to a newly defined button on one of the toolbars or to a newly defined shortcut key on the keyboard.

If you choose for the toolbars, the *Customize* dialogue box in Figure 149 appears. By default, Word offers you to add the Macro execution button to the very top of your screen, where the Quick Access Toolbar is.

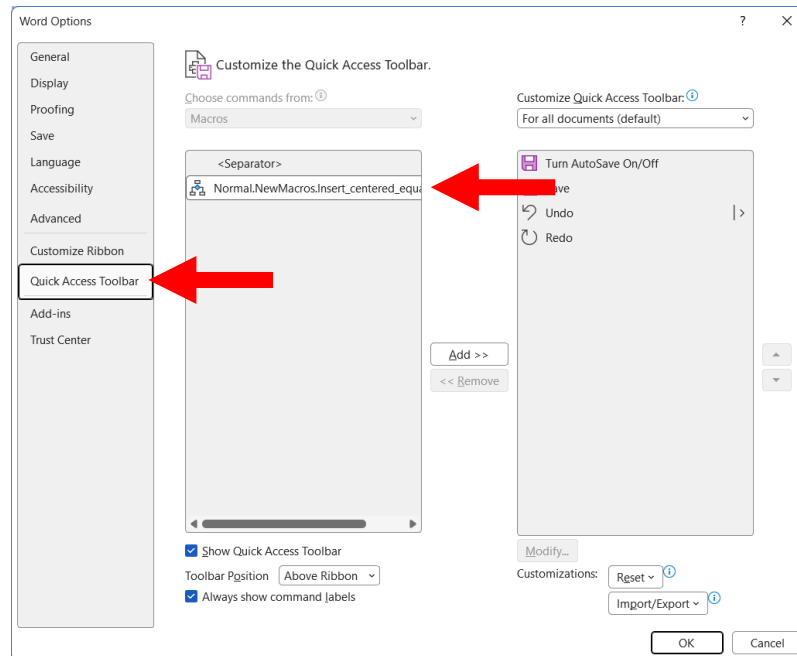


Figure 149 The *Customize* dialogue box.

If you prefer to add your macro to another ribbon, for example the Home ribbon, you have to switch to the option “*Customize Ribbon*” in the left menu. Then from the left dropdown menu, you select Macros, as illustrated in Figure 150.

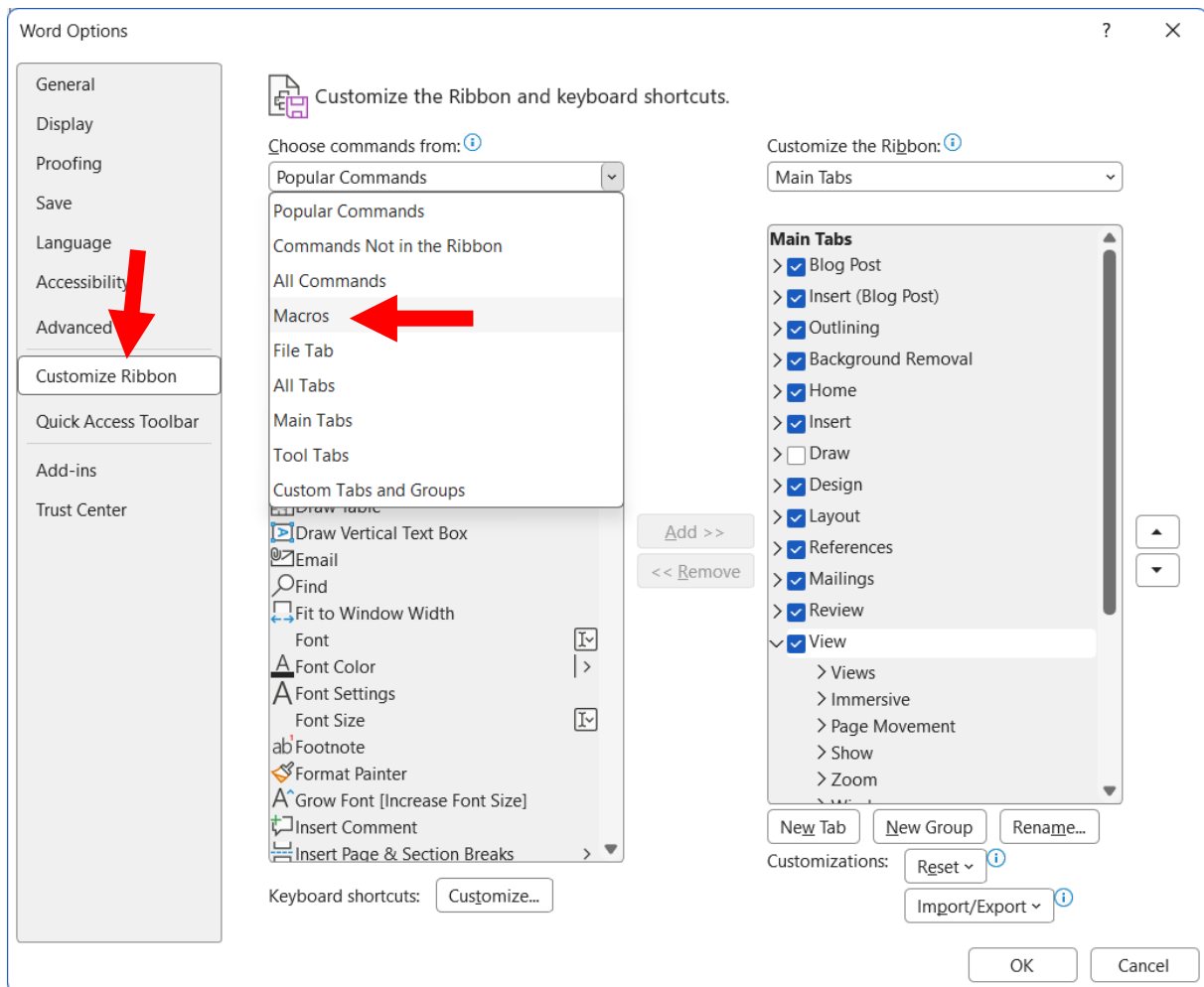


Figure 150 Select the available Macros to assign to a customized Ribbon.

Figure 151 shows the next step. Since we have only defined one macro so far, the list of macros is limited to the macro “*Insert_centered_equation*”. In this example, we show how to add the Macro execution button to an entirely new Ribbon, because the other Ribbons (Home, Insert, Design, Layout, etc.) are typically already quite full. Also, since I use quite a lot of macros, I prefer to make a new ribbon in the top menu, similar to the Home Ribbon.

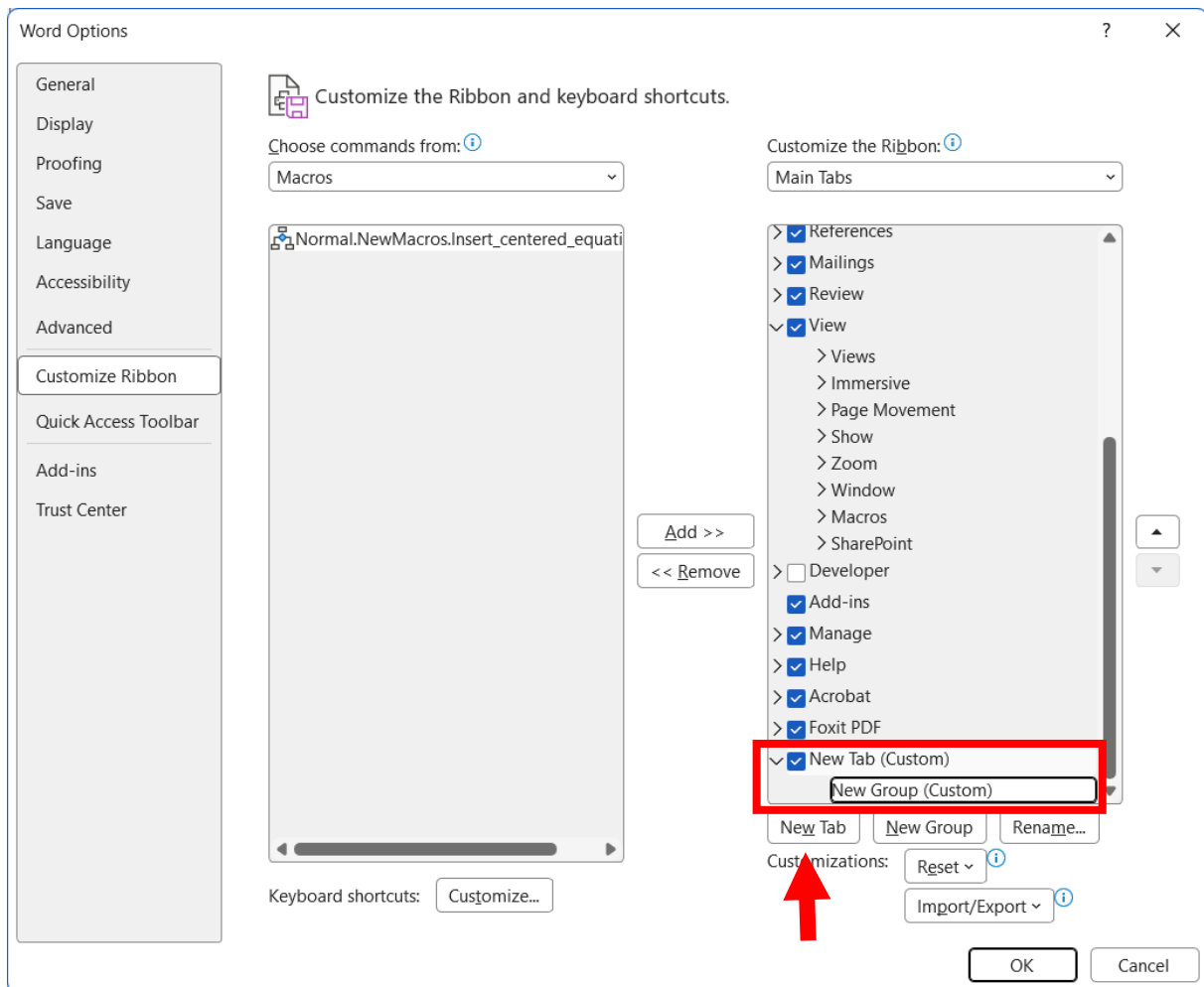


Figure 151 Adding the Macro “Insert_centered_equation” to a new Ribbon.

In Figure 151, we have chosen to create a new Tab (new Ribbon). This automatically also creates a new group, because the Ribbon can group different items in one toolbar. Figure 152 shows that we rename this custom Ribbon to “Macros”.

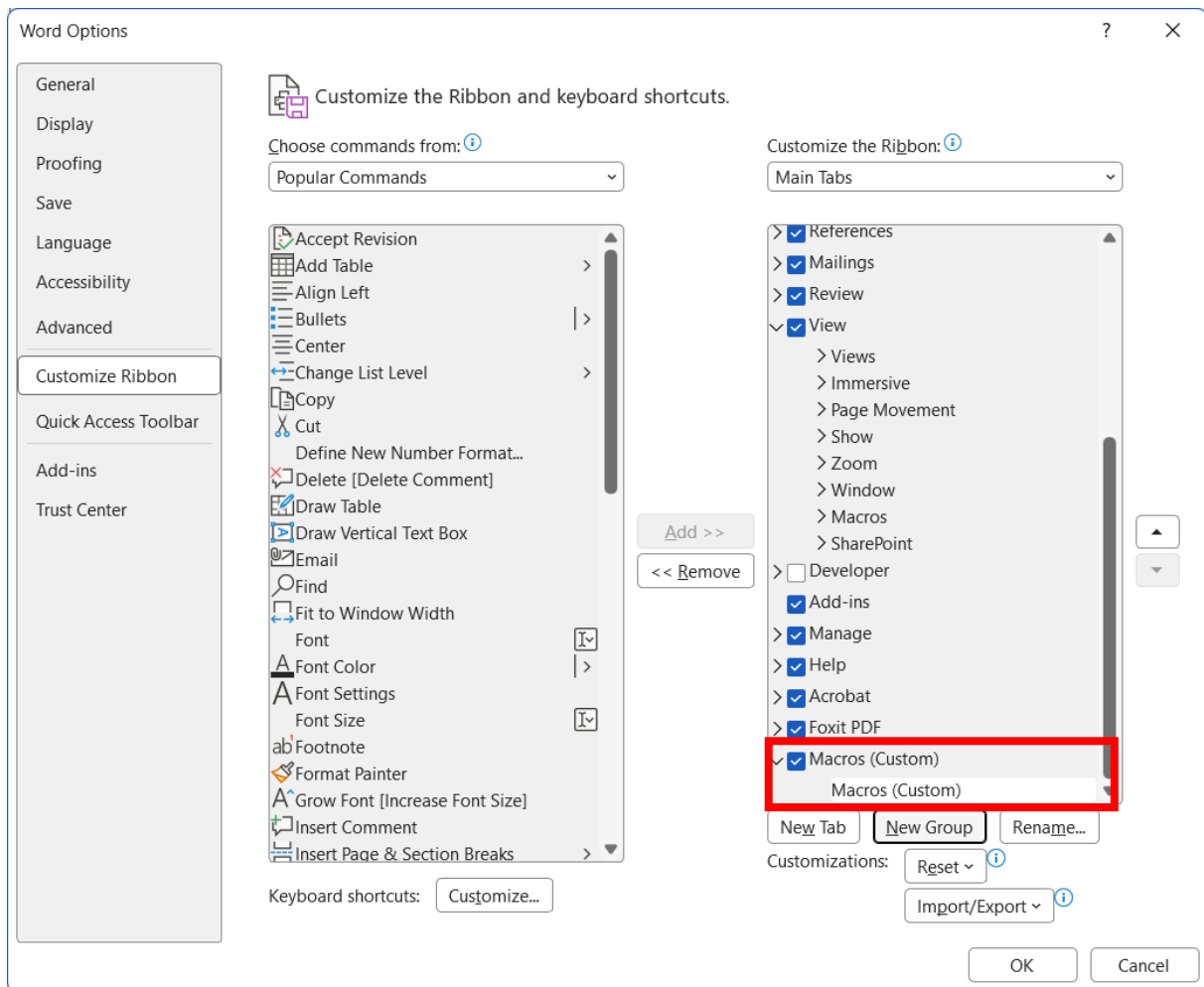


Figure 152 The custom Ribbon/Tab is renamed to Macros.

When we will have finished recording the macro, the macro will be added to a new Ribbon called Macros. Figure 153 shows how this new Ribbon will look like in the top menu, as the rightmost Ribbon of all, when we have finished recording the macro.

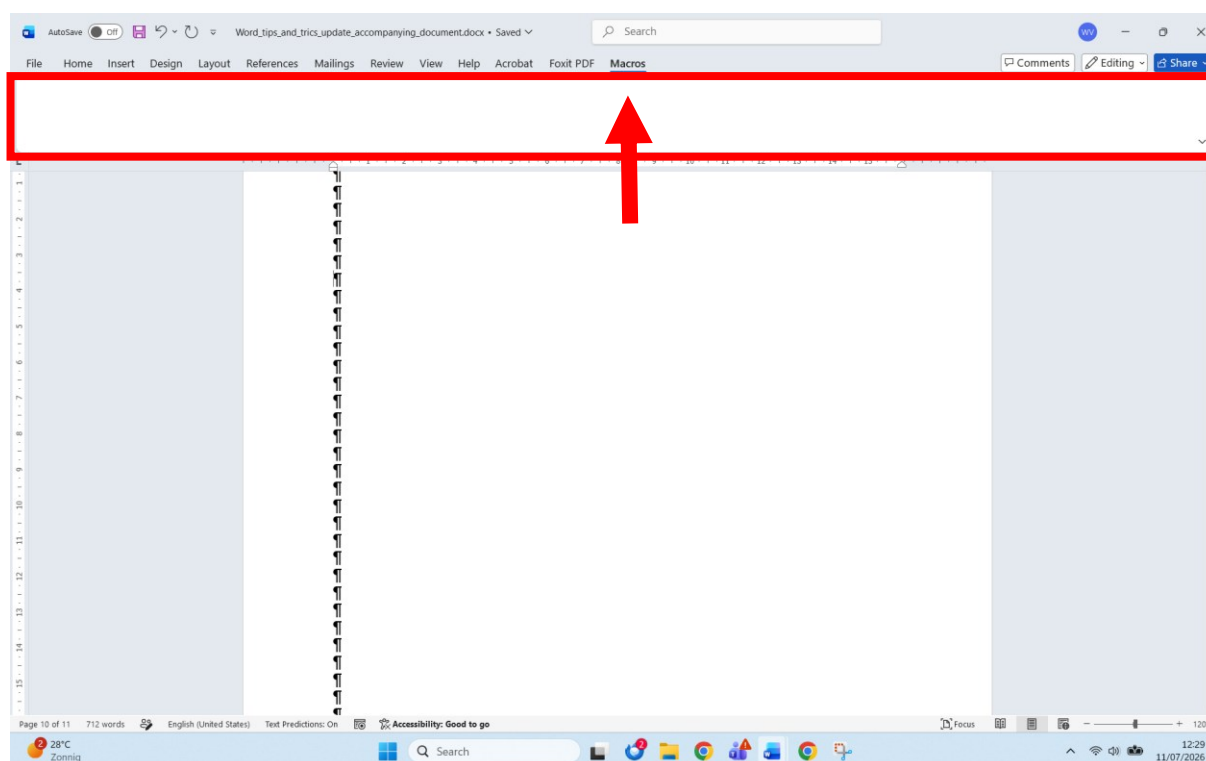


Figure 153 Creation of the new Ribbon Macro.

Be careful that if you choose the *View > Macros > Record macro...*, and you have finished the step of assigning the macro to a button on a Ribbon or a keyboard shortcut, the recording starts immediately after that. The mouse pointer changes to the symbol of a tape recorder. So we move on to the actual recording of the macro.

When recording a new macro, first do a “dry run” and practice the sequence of codes that you need to execute to achieve the desired result. This may take some trial-and-error.

For quick troubleshooting, you can choose *View > Macros > Record macro...*, let Word choose the default name “*macro1*”, press the OK button and start recording. Afterwards, you can run the macro from *View > Macros > View Macros* and debug the macro actions. If you reach the final goal, you just delete all the temporary macros from the list.

When recording a macro, it is handy to know some shortcuts for moving around with the cursor, because Macros do not record selections made with a mouse. This is a very important thing to remember: when you record a macro in Microsoft Word, mouse selections and text navigation inside the document window are NOT recorded. You can use the mouse to click menu commands and ribbon buttons, but clicking or dragging to select text or move the cursor will be ignored.

So you need to use the keyboard to select text or navigate through the document while you are recording your macro. The table below shows the most useful keyboard shortcuts for navigating in the document and selecting text.

Table 2 Keyboard shortcuts for selecting text and navigating in the document.

Keyboard shortcut	Action
Ctrl + right arrow	Move to next word
Ctrl + left arrow	Move to previous word
Ctrl + down arrow	Move to next paragraph
Ctrl + up arrow	Move to previous paragraph/start of paragraph
Right arrow	Move one character to the right
Left arrow	Move one character to the left
Home	Move to start of the line
End	Move to end of the line
Ctrl + Page up	Move to top of previous page
Ctrl + Page down	Move to top of next page
Ctrl + Home	Move to beginning of document
Ctrl + End	Move to end of document
Shift + F5	Move to the last change/edit
Shift + arrow keys	Select text
Ctrl + Shift + left arrow	Select the word to the left
Ctrl + Shift + right arrow	Select the word to the right
Shift + Home	Select from the current position to the beginning of the current line
Shift + End	Select from the current position to the end of the current line
Ctrl+Shift+Up arrow	Select from the current position to the beginning of the current paragraph
Ctrl+Shift+Down arrow key	Select from the current position to the end of the current paragraph
Ctrl+Shift+Home	Select from the current position to the beginning of the document
Ctrl+Shift+End	Select from the current position to the end of the document
Ctrl + A	Select all document content
F8	F8 repeatedly to expand the selection to the entire word, sentence, paragraph, section, and document
Ctrl+Shift+F8, then press the arrow keys	Select a vertical block of text
Esc	Stop extending the selection
Ctrl+Backspace	Delete one word to the left
Ctrl+Delete	Delete one word to the right
Shift + Enter	Insert a line break
Ctrl + Enter	Insert a page break
Ctrl + Shift + Enter	Insert a column break
Ctrl + F9	Insert an empty field
Shift + F9	Switch between a selected field code and its result
Alt + F9	Switch between all field codes and their results
Ctrl + Shift + F9	Unlink a field and convert to plain text
F11	Go to the next field
Shift + F11	Go to the previous field

For the macro *Insert_centered_equation*, this is the exact list of steps you have to record:

1. your cursor is at the start of an empty paragraph
2. assign the paragraph style *Centered equation* (see Figure 14 and Figure 15). This can be done with your mouse by clicking the style in the Styles toolbar
3. press the <Tab> key twice (you are now at the end of the line),
4. insert an opening bracket “(“,
5. press <Ctrl>+<F9> to insert a field code,
6. write “SEQ my_equations” (or any name you choose for the SEQ list),
7. click <F9> to update the active field code,
8. press the <End> key to move to the end of the line,
9. insert a closing bracket “)”,
10. press the <Home> key to move to the beginning of the line,
11. press the right arrow on your keyboard once to move to the position of the centering tab,
12. stop recording the macro by going to the *View > Macro* toolbar and stop recording. There is no keyboard shortcut for stopping the recording.

After the recording, the result of the recorded actions is as shown in Figure 154:

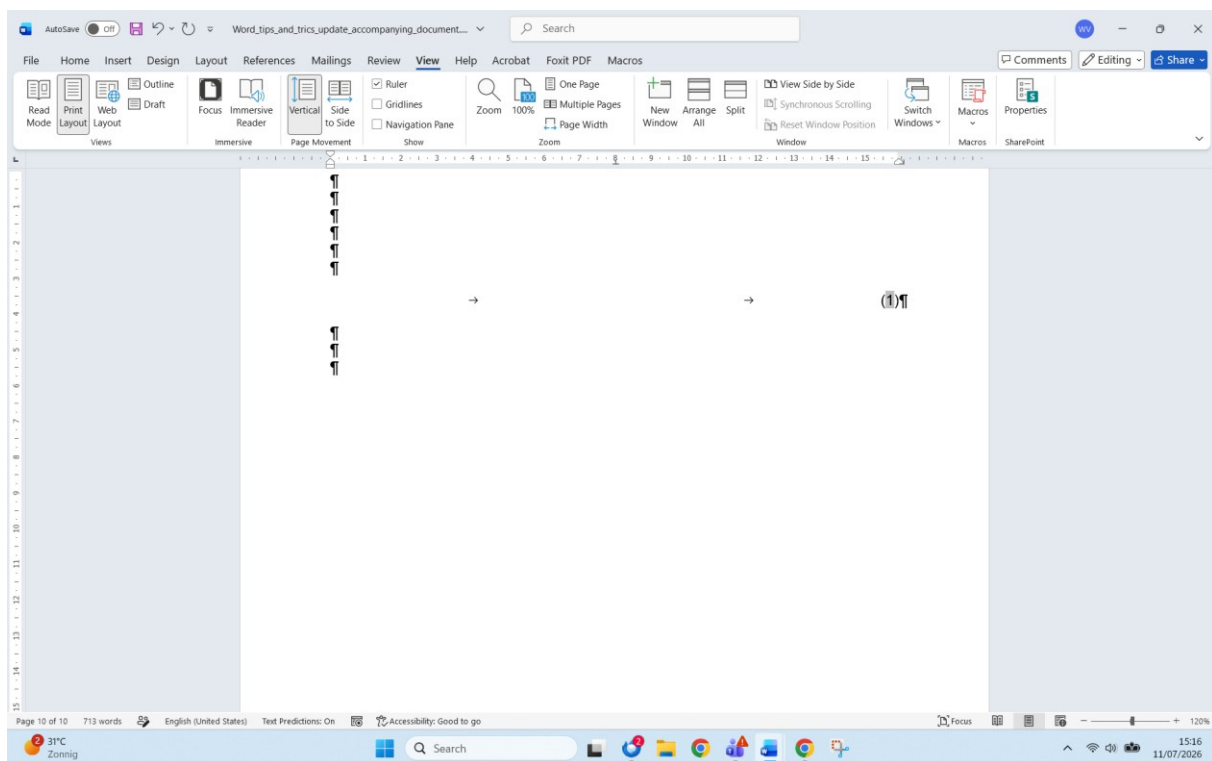


Figure 154 Result of the recording action of the macro “*Insert_centered_equation*”.

The cursor is blinking at the centered tab, and you are ready to insert an equation.

You could think that opening the Equation Editor can also be part of the macro recording, but just opening the Equation Editor, is not saved in the macro code. Even if you add some simple character in the Equation Editor, like a space or a single character, the recorded macro code just prints that character, without opening the Equation Editor.

To force the macro to open the Equation Editor as the last action of the macro, we need to edit the Visual Basic code of the macro and add a small section that calls the Equation Editor. To edit the Visual Basic code, you go to *View > Macros > View macros*, and then choose to edit the macro that you recorded. You now see the screen in Figure 155, which shows you the Visual Basic code of the recorded macro. You can easily understand the actions that you just recorded, translated into Visual Basic code.

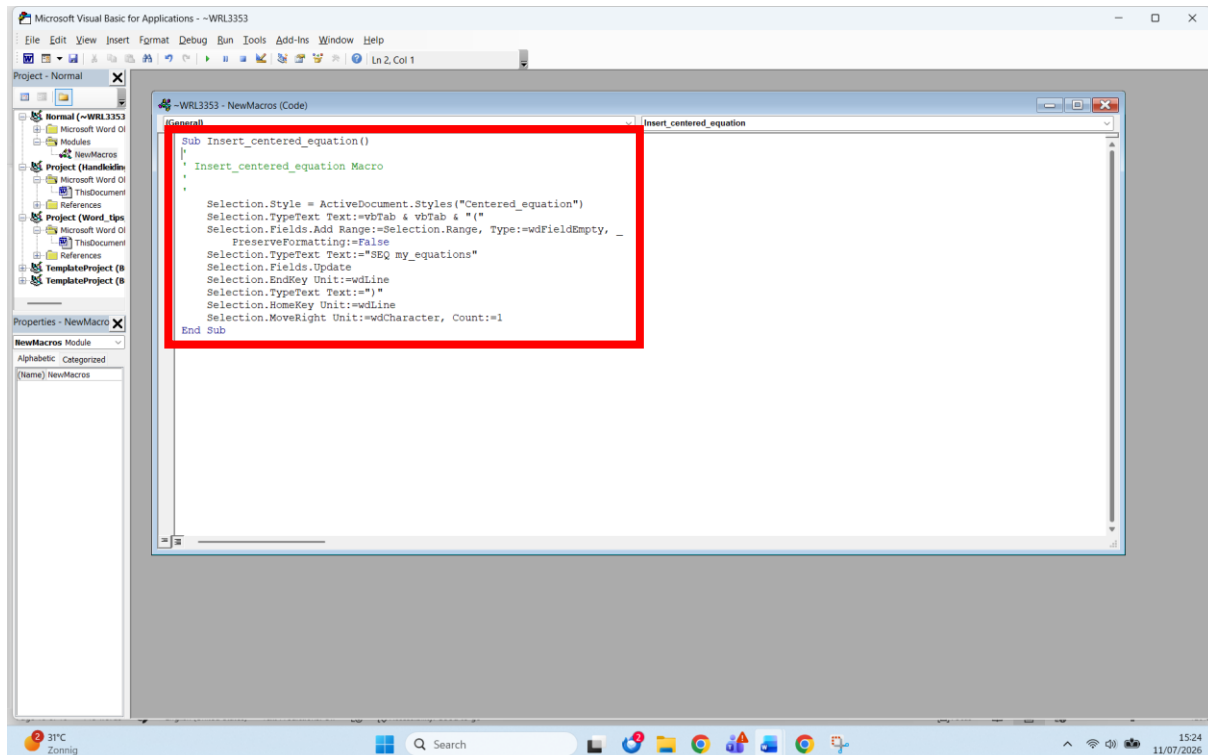


Figure 155 Corresponding Visual Basic code of the macro “Insert_centered_equation”.

Just add the following code snippet before the last line of the code “End Sub”:

```
Dim objRange As Range
Dim objOMath As OMath

' Get the current cursor location
Set objRange = Selection.Range

' Add an Office Math equation structure to the text range
Set objOMath = objRange.OMaths.Add(objRange).OMaths(1)

' Optional: Convert it to a professionally built-up visual display format
objOMath.BuildUp

' Position the cursor right inside the new equation placeholder
objRange.Select
```

The combined code is shown in Figure 156.

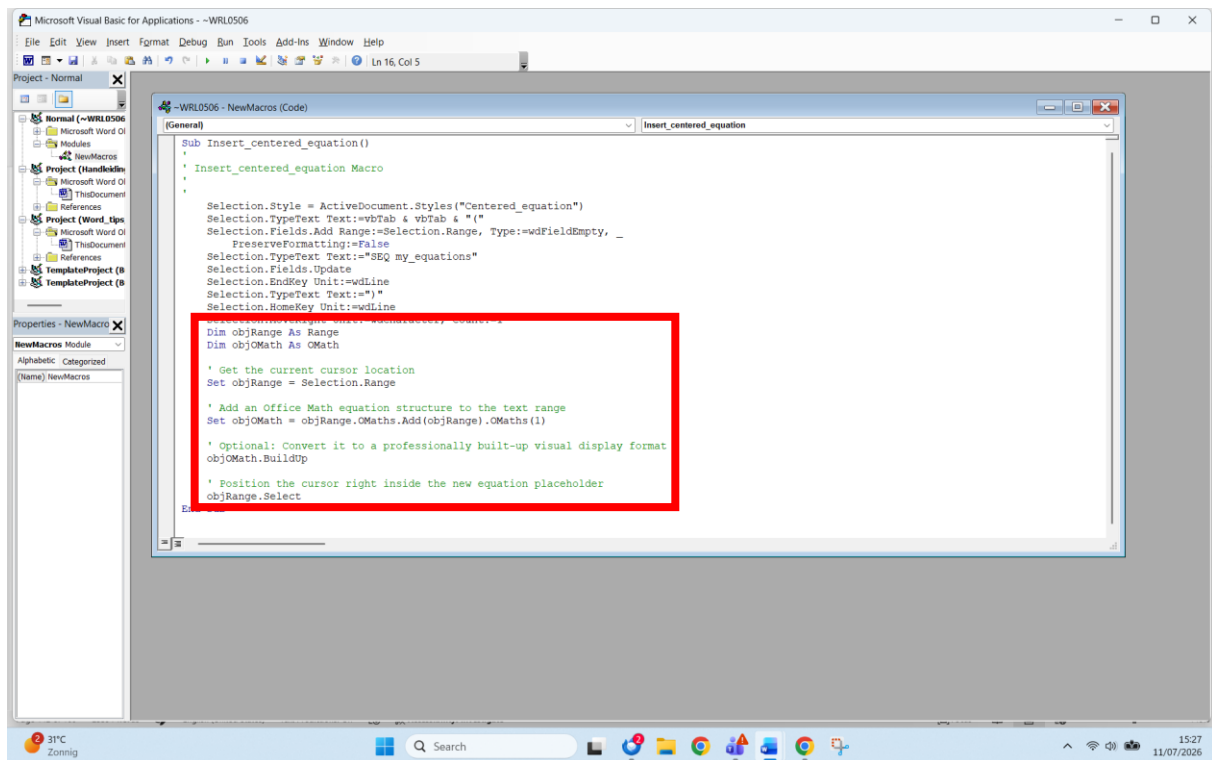


Figure 156 Visual Basic code with extra action to open the Microsoft Equation Editor.

Save the code by clicking the *Save* button on top of the Visual Basic Editor. The macro has now appeared in your new Macro ribbon on top, and you can just click it to execute it. The result is shown in Figure 157.

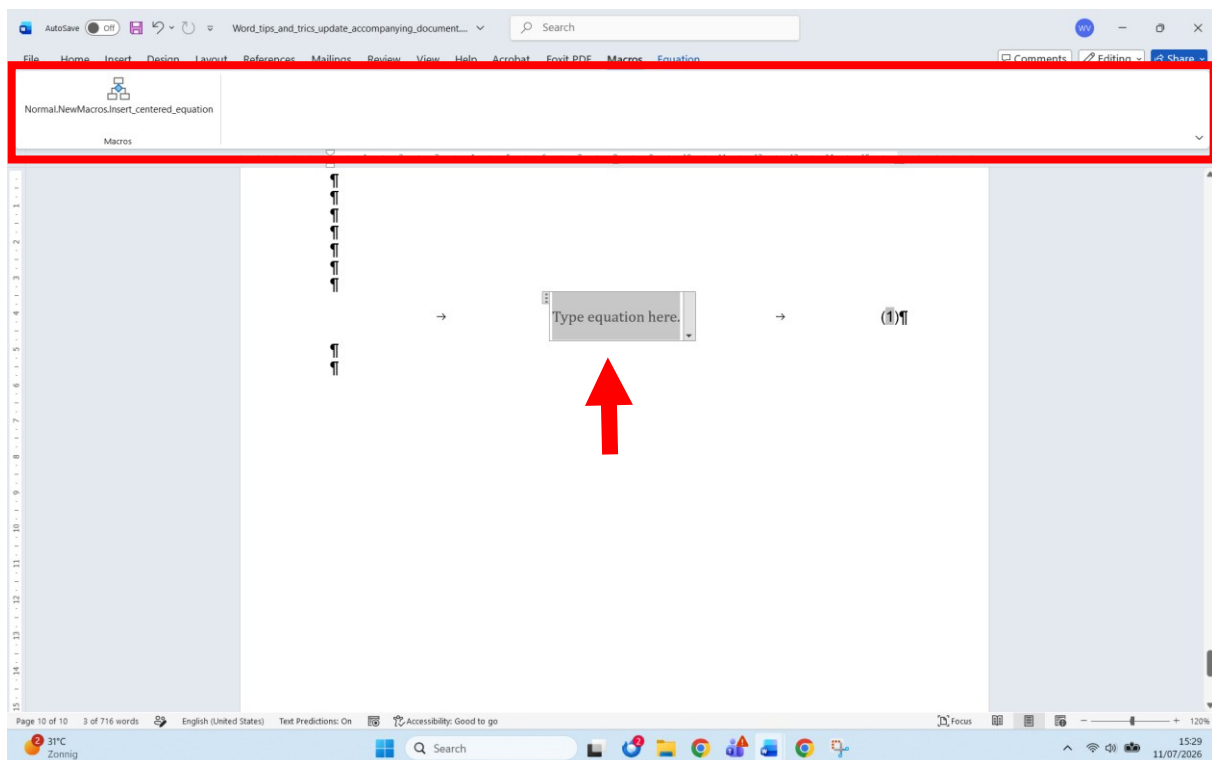


Figure 157 Result of execution of the macro "Insert_centered_equation".

Finally, to make the macro look more attractive on the Ribbon toolbar, it is possible to edit the label and icon that is shown on the Ribbon toolbar.

To that purpose, you right click with the mouse on the macro in the Ribbon toolbar, and select “*Customize the Ribbon*”, as shown in Figure 158 below.

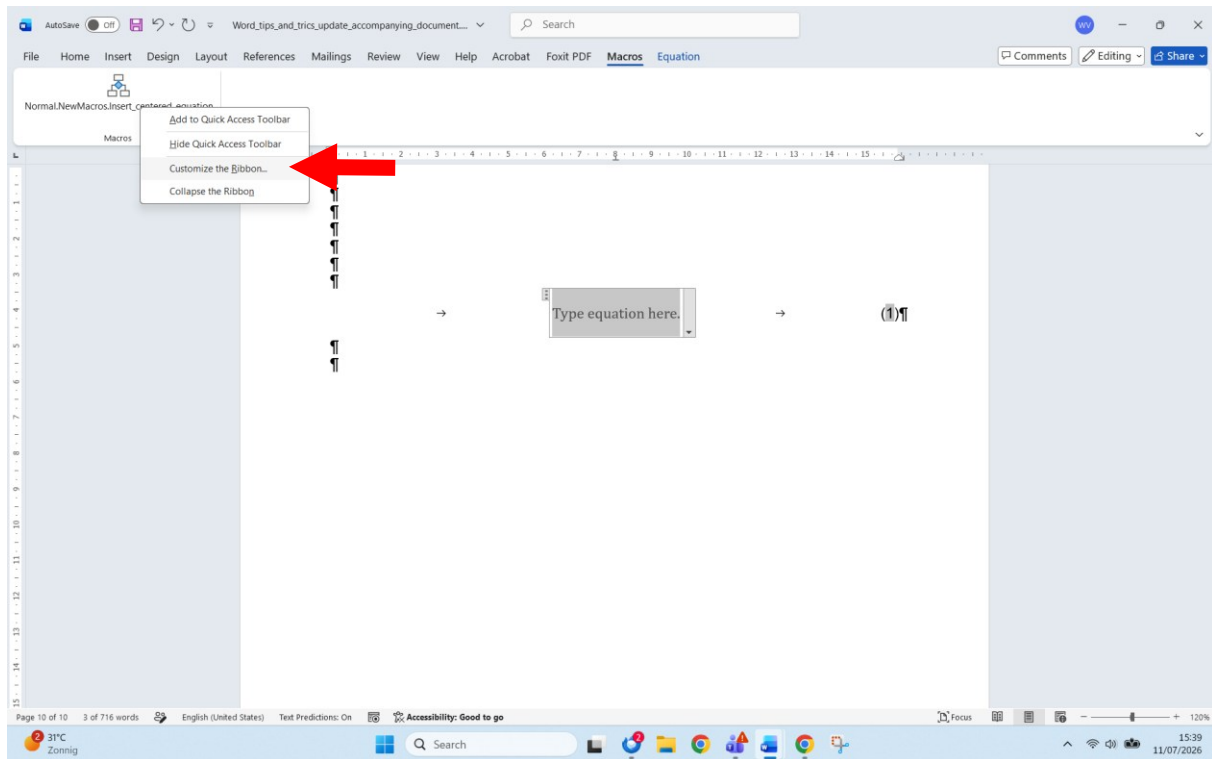


Figure 158 Customize the label and icon of the macro on the Ribbon.

The dialogue box for the customization of the Ribbons opens again, as shown in Figure 159. You select the Ribbon “*Macros*” in the right slider menu, that you had created before. In the subgroup *Macros*, you find your recorded macro “*Insert_centered_equation*”. You can rename the macro there and select an icon.

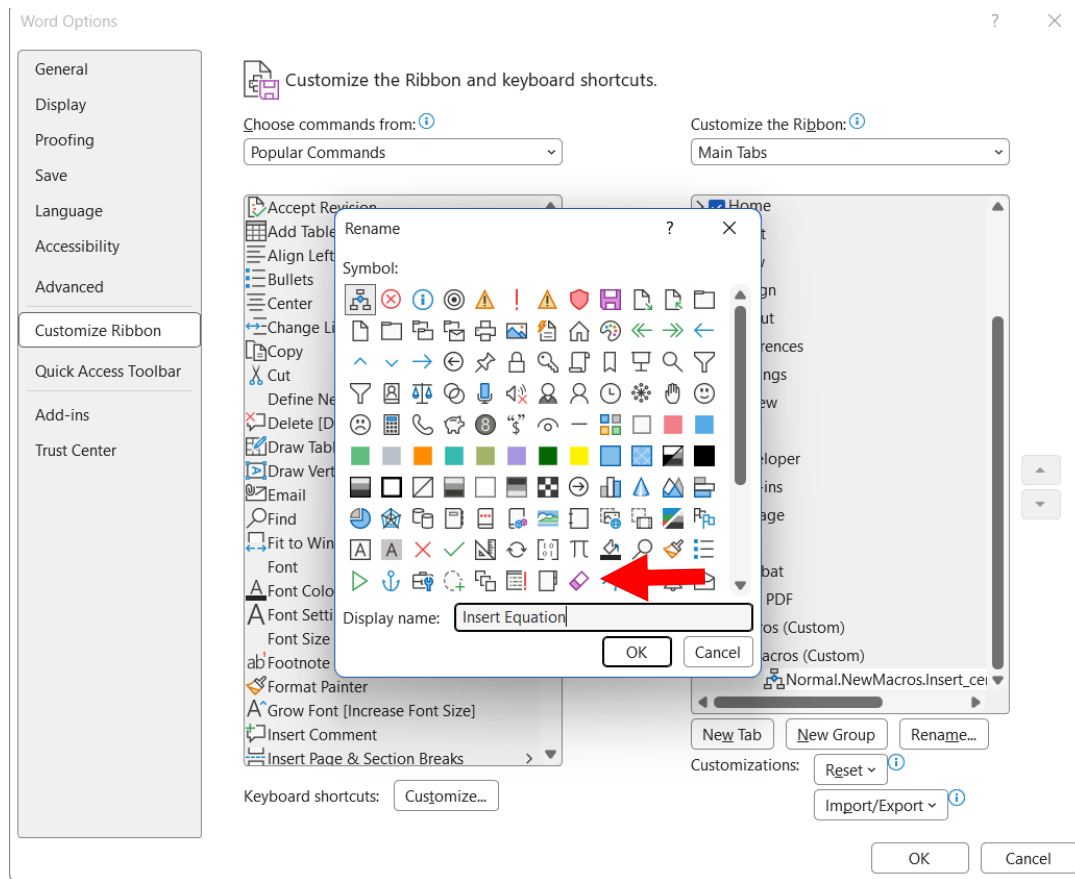


Figure 159 Editing the label and icon for the macro “*Insert_centered_equation*”.

After that, the macro appears in the Macro Ribbon as shown in Figure 160 below.

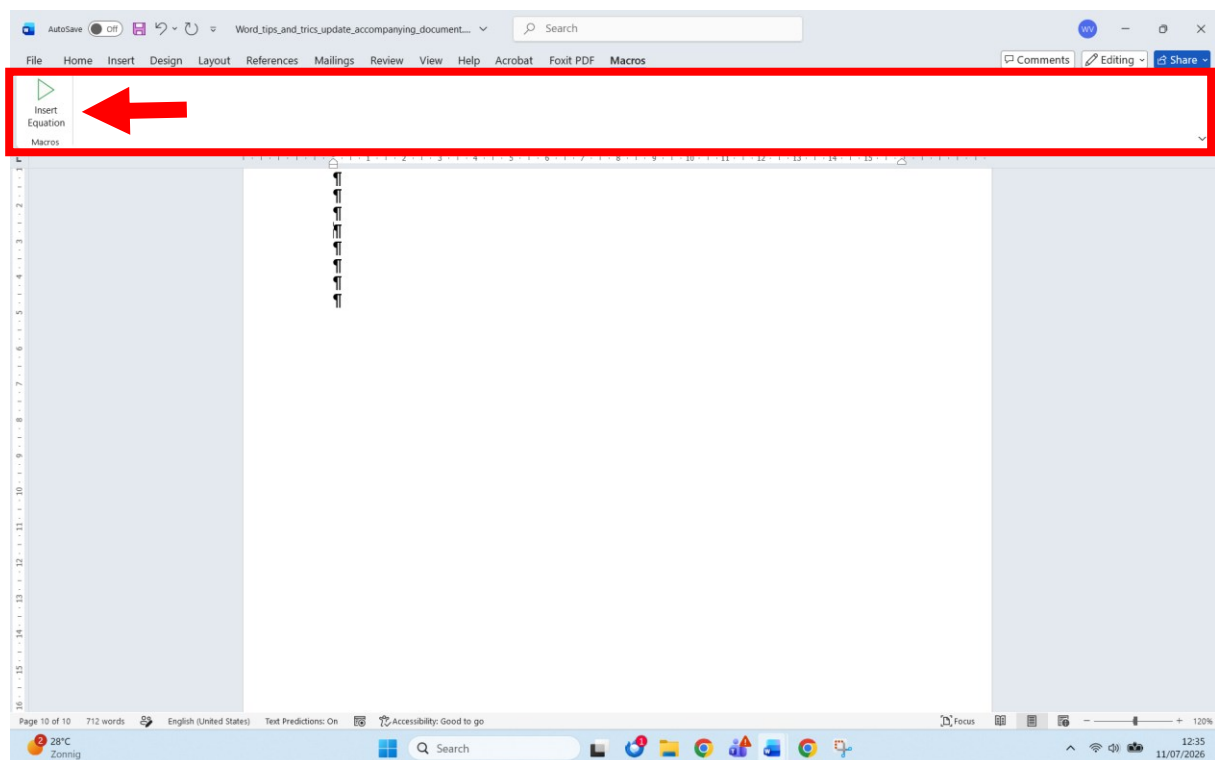


Figure 160 Customizing the newly defined button on the toolbar to execute the macro.

If you want to assign your new macro to a keyboard shortcut instead, you have to choose a shortcut key. Most combinations of <Alt> and a letter are not used by MS Word itself.

Finally, here is the full VBA code, if you want to paste the entire code instead and skip the recording of the macro.

Visual Basic code for automatically inserting centered and numbered equations

```
Sub Insert_centered_equation()
'
' Insert_centered_equation Macro
'
'
    Selection.Style = ActiveDocument.Styles("Centered_equation")
    Selection.TypeText Text:=vbTab & vbTab & "("
    Selection.Fields.Add Range:=Selection.Range, Type:=wdFieldEmpty, _
        PreserveFormatting:=False
    Selection.TypeText Text:="SEQ my_equations"
    Selection.Fields.Update
    Selection.EndKey Unit:=wdLine
    Selection.TypeText Text:=")"
    Selection.HomeKey Unit:=wdLine
    Selection.MoveRight Unit:=wdCharacter, Count:=1
    Dim objRange As Range
    Dim objOMath As OMath

    ' Get the current cursor location
    Set objRange = Selection.Range

    ' Add an Office Math equation structure to the text range
    Set objOMath = objRange.OMaths.Add(objRange).OMaths(1)

    ' Optional: Convert it to a professionally built-up visual display format
    objOMath.BuildUp

    ' Position the cursor right inside the new equation placeholder
    objRange.Select
End Sub
```

11.2. MACRO FOR INSERTING A CENTERED AND NUMBERED EQUATION PER CHAPTER

In section 3.6, we already discussed the combined use of the field codes STYLEREF and SEQ to have equation numbers that include the chapter number.

The field codes were:

```
{ STYLEREF "Heading 1" \n }. { SEQ my_equations \s 1 }
```

Remember that the SEQ field code can only restart numbering when detecting the built-in paragraph style *Heading 1*. If you use user-defined paragraph styles for the chapter titles, you need to either:

- Insert a line of text with the built-in paragraph style *Heading 1* at the start of each chapter, and make it invisible by assigning a white font colour,
- Insert a reset of the equation numbering at the start of each chapter, by inserting the field code `{ SEQ my_equations \r 0 }` and making that field code invisible. The next appearances of the SEQ field code do not need the switch `\s 1` then, because the built-in paragraph style *Heading 1* is not used.

So now we can record the macro of the centered equation again, but adding a few extra steps in the macro recording:

1. your cursor is at the start of an empty paragraph
2. assign the paragraph style *Centered_equation* (see Figure 14 and Figure 15). This can be done with your mouse by clicking the style in the Styles toolbar
3. press the <Tab> key twice,
4. insert an opening bracket “(“,
5. press <Ctrl>+<F9> to insert a field code,
6. write “STYLEREF “Heading 1” \n”,
7. click <F9> to update the active field code,
8. press the <End> key to move to the end of the line,
9. write a period as separator between the chapter number and the equation number,
10. press <Ctrl>+<F9> to insert a field code,
11. write “SEQ my_equations \s 1” (or any name you choose for the SEQ list),
12. click <F9> to update the active field code,
13. press the <End> key to move to the end of the line,
14. insert a closing bracket “)”,
15. press the <Home> key to move to the beginning of the line,
16. press the right arrow on your keyboard once to move to the position of the centering tab,
17. stop recording the macro by going to the *View > Macro* toolbar and stop recording.
There is no keyboard shortcut for stopping the recording.

Add again the extra code to call the Microsoft Equation Editor.
The final result looks like Figure 161.

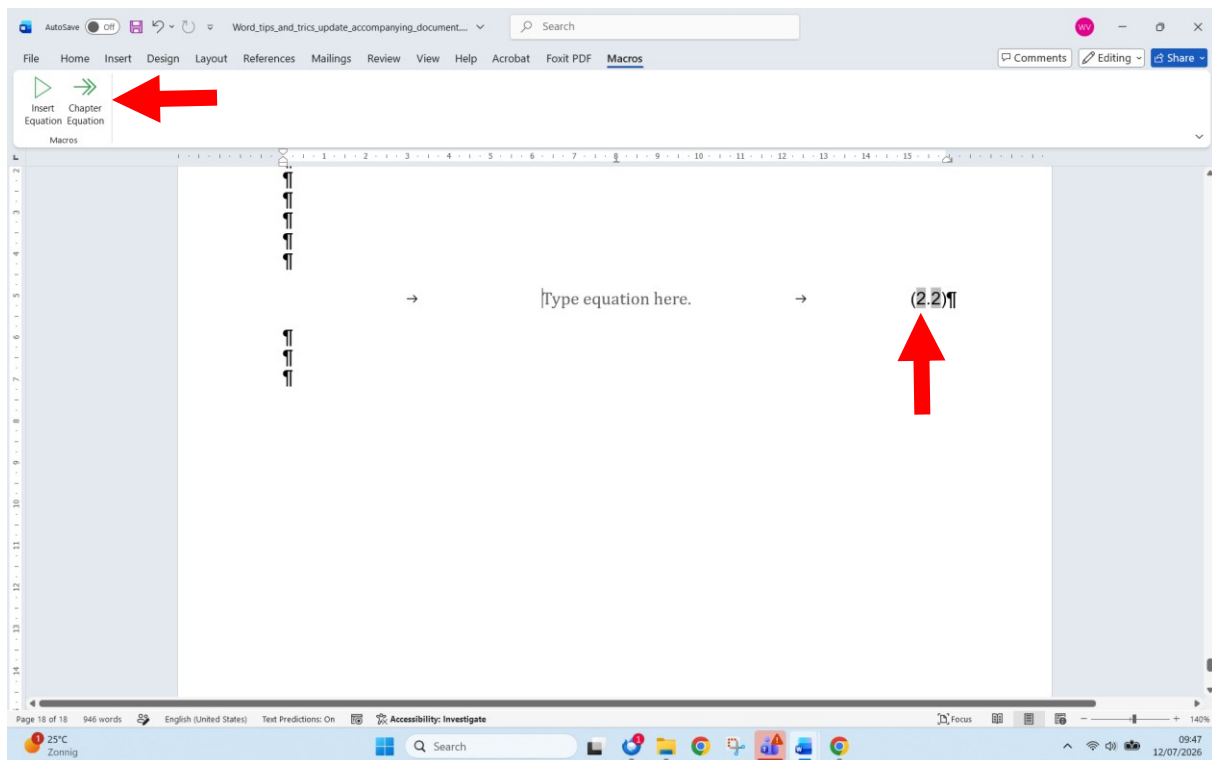


Figure 161 Updated result of the macro “Insert_centered_equation” by adding the chapter number.

This STYLEREF field code can also be used in headers/footers/etc. to refer to the title or number of chapters, subchapters, sections, etc. You can also define your own paragraph styles and refer with STYLEREF to those styles.

Finally, here is again the full VBA code for this macro, if you want to skip the recording steps.

Visual Basic code for automatically inserting centered and numbered equations per chapter

```
Sub Insert_chapter_equation()
'
' Insert_chapter_equation Macro
'
    Selection.Style = ActiveDocument.Styles("Centered_equation")
    Selection.TypeText Text:=vbTab & vbTab & "("
    Selection.Fields.Add Range:=Selection.Range, Type:=wdFieldEmpty, _
        PreserveFormatting:=False
    Selection.TypeText Text:="STYLEREF ""Heading 1"" \n"
    Selection.Fields.Update
    Selection.EndKey Unit:=wdLine
    Selection.TypeText Text:= "."
    Selection.Fields.Add Range:=Selection.Range, Type:=wdFieldEmpty, _
        PreserveFormatting:=False
    Selection.TypeText Text:="SEQ my_equations \s 1"
    Selection.Fields.Update

```

```

Selection.EndKey Unit:=wdLine
Selection.TypeText Text:=")"
Selection.HomeKey Unit:=wdLine
Selection.MoveRight Unit:=wdCharacter, count:=1
Dim objRange As Range
Dim objOMath As OMath

' Get the current cursor location
Set objRange = Selection.Range

' Add an Office Math equation structure to the text range
Set objOMath = objRange.OMaths.Add(objRange).OMaths(1)

' Optional: Convert it to a professionally built-up visual display format
objOMath.BuildUp

' Position the cursor right inside the new equation placeholder
objRange.Select
End Sub

```

11.3. MACRO FOR INSERTING NUMBERED EXAMPLES PER CHAPTER

A variant of the previous macro is a macro to insert numbered examples or illustrations, that have a left-border and are again numbered per chapter.

We would like to have the following layout:

Example 1.1

Text example 1

Figure example 1

We make use of the paragraph style “*Course_example*” that we already created in section 2.3. The numbering itself has exactly the same field codes as the previous example:

```
{ STYLEREF “Heading 1” \n }. { SEQ my_course_examples \s 1 }
```

Now the steps to record this macro, are as follows:

1. your cursor is at the start of an empty paragraph
2. assign the paragraph style *Course_example* (see section 2.3). This can be done with your mouse by clicking the style in the Styles toolbar
3. type the text “Example “
4. press <Ctrl>+<F9> to insert a field code,
5. write “STYLEREF “Heading 1” \n”,
6. click <F9> to update the active field code,
7. press the <End> key to move to the end of the line,
8. write a period as separator between the chapter number and the example number,
9. press <Ctrl>+<F9> to insert a field code,

10. write "SEQ my_course_examples \s 1" (or any name you choose for the SEQ list),
11. click <F9> to update the active field code,
12. press the <End> key to move to the end of the line,
13. press the <Shift> + <Home> keys to select the text till the beginning of the line,
14. click on the Home Ribbon the icons for Bold, Italic and Underline,
15. press the <End> key to move to the end of the line and undo the selection,
16. press the <Enter> key to move to the next line,
17. click on the Home Ribbon the icons for Bold, Italic and Underline again, so that this new line has no bold/italic/underline formatting anymore,
18. stop recording the macro by going to the *View > Macro* toolbar and stop recording. There is no keyboard shortcut for stopping the recording.

Finally, here is again the full VBA code for this macro, if you want to skip the recording steps.

Visual Basic code for automatically numbered examples

```
Sub Insert_course_example()
'
' Insert_course_example Macro
'
    Selection.Style = ActiveDocument.Styles("Course_example")
    Selection.TypeText Text:="Example "
    Selection.Fields.Add Range:=Selection.Range, Type:=wdFieldEmpty, _
        PreserveFormatting:=False
    Selection.TypeText Text:="STYLEREF ""Heading 1"" \n"
    Selection.Fields.Update
    Selection.EndKey Unit:=wdLine
    Selection.TypeText Text:="."
    Selection.Fields.Add Range:=Selection.Range, Type:=wdFieldEmpty, _
        PreserveFormatting:=False
    Selection.TypeText Text:="SEQ my_course_examples \s 1"
    Selection.Fields.Update
    Selection.EndKey Unit:=wdLine
    Selection.HomeKey Unit:=wdLine, Extend:=wdExtend
    Selection.Font.Bold = wdToggle
    Selection.Font.Italic = wdToggle
    Selection.Font.UnderlineColor = wdColorAutomatic
    Selection.Font.Underline = wdUnderlineSingle
    Selection.EndKey Unit:=wdLine
    Selection.TypeParagraph
    Selection.Font.Bold = wdToggle
    Selection.Font.Italic = wdToggle
    Selection.Font.UnderlineColor = wdColorAutomatic
    Selection.Font.Underline = wdUnderlineNone
End Sub
```

12

. Integrating user input in macros

If you are familiar with programming in the *Visual Basic for Applications* language, you can add a lot more features to macros. But nowadays even that is not necessary anymore.

With the assistance of Gemini, ChatGPT, Claude or Copilot, you can even feed the Visual Basic code of your macro to the AI assistant, and add anything you like.

In the example below, we update the macro “*Insert_centered_equation*” to ask for a bookmark name, and then add that bookmark name to the number of the equation, so that we can easily cross-reference that equation.

12.1. EXTENSION OF MACRO FOR EQUATION NUMBERING WITH BOOKMARKING

In the previous chapter, we have recorded the macro “*Insert_centered_equation*”. Now we want to implement a pop-up dialogue box to ask the reader for a bookmark name. That bookmark name is then assigned to the equation number with the field code SEQ. Then you can insert cross-references to the equation number very easily, and if the equation moves to another place in the document, the field code SEQ is updated, and so is the cross-reference as well.

Here is the VBA code for this extended functionality.

Visual Basic code for automatically inserting equations and asking for a bookmark

```
Sub Insert_centered_equation()  
,  
' Insert_centered_equation Macro
```

```

Dim BookmarkName As String
Dim IsValid As Boolean
Dim RegEx As Object
Dim EqNumberField As Field
Dim EqNumberRange As Range

Set RegEx = CreateObject("VBScript.RegExp")
RegEx.Pattern = "[A-Za-z][A-Za-z0-9_]{0,39}$"
RegEx.IgnoreCase = False
RegEx.Global = False

Do
    BookmarkName = InputBox( _
        "Enter a bookmark name for the equation. Bookmark names in Microsoft Word must begin with a letter and can be up to 40 characters long. They can contain letters, numbers, and underscores, but no spaces or special punctuation marks are allowed.", _
        "Equation Bookmark Name")

    If BookmarkName = "" Then
        Exit Sub
    End If

    IsValid = RegEx.Test(BookmarkName)

    If IsValid Then
        If ActiveDocument.Bookmarks.Exists(BookmarkName) Then
            IsValid = False
            MsgBox "A bookmark with this name already exists. Please enter another bookmark name.", _
                vbExclamation, "Bookmark Already Exists"
        End If
    End If

    If Not IsValid Then
        MsgBox "Invalid bookmark name." & vbCrLf & vbCrLf & _
            "The bookmark name must:" & vbCrLf & _
            "- Begin with a letter" & vbCrLf & _
            "- Be no longer than 40 characters" & vbCrLf & _
            "- Contain only letters, numbers, and underscores" & vbCrLf & _
            "- Not already exist in the document", _
            vbExclamation, "Invalid Bookmark Name"
    End If

Loop Until IsValid

Selection.Style = ActiveDocument.Styles("Centered_equation")
Selection.TypeText Text:=vbTab & vbTab & "("

```

```

Set      EqNumberField      =      Selection.Fields.Add(Range:=Selection.Range,
Type:=wdFieldEmpty, _
      PreserveFormatting:=False)

Selection.TypeText Text:="SEQ my_equations"
Selection.Fields.Update

Set EqNumberRange = EqNumberField.Result.Duplicate
EqNumberRange.Select
ActiveDocument.Bookmarks.Add Name:=BookmarkName, Range:=Selection.Range

Selection.EndKey Unit:=wdLine
Selection.TypeText Text:=")"
Selection.HomeKey Unit:=wdLine
Selection.MoveRight Unit:=wdCharacter, Count:=1

Dim objRange As Range
Dim objOMath As OMath

' Get the current cursor location
Set objRange = Selection.Range

' Add an Office Math equation structure to the text range
Set objOMath = objRange.OMaths.Add(objRange).OMaths(1)

' Optional: Convert it to a professionally built-up visual display format
objOMath.BuildUp

' Position the cursor right inside the new equation placeholder
objRange.Select
End Sub

```

Just copy and paste this code in the Visual Basic Editor and preserve the name of the macro. If you now run the macro, you first get the following dialogue box in Figure 162. This dialogue box also mentions the rules for the naming of the bookmark name, so that the user is instructed which bookmark names are valid.

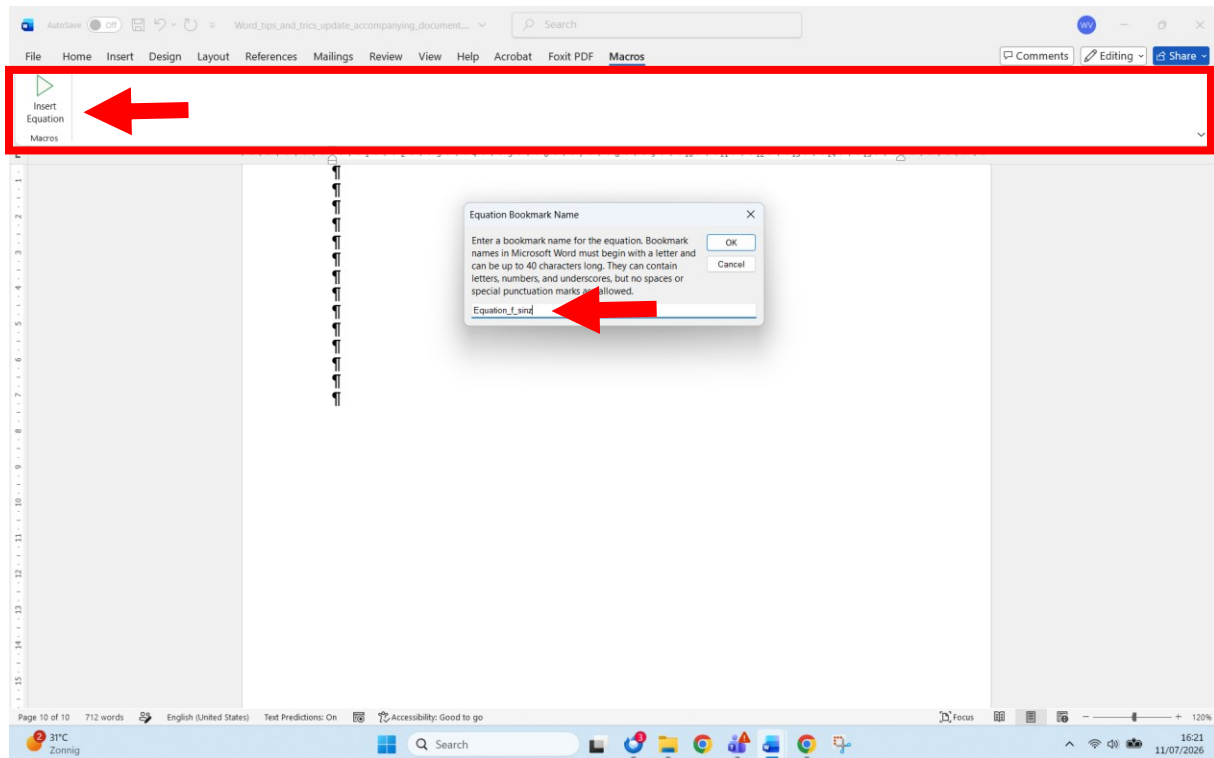


Figure 162 The macro “*Insert_centered_equation*” asks the user to input a bookmark name.

The result after execution of the macro looks almost the same, but now, there are two grey [] brackets around the number of the equation, as shown in Figure 163, and a new bookmark name is created,

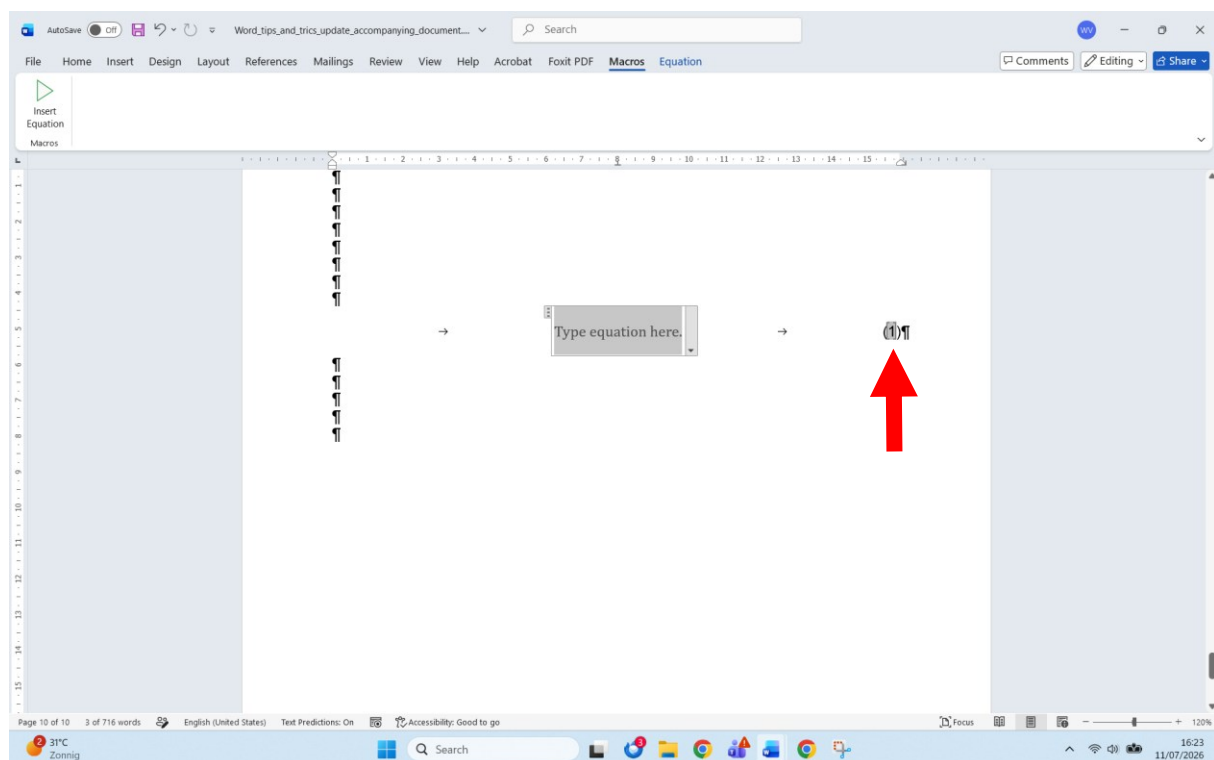


Figure 163 The equation number is now assigned with a bookmark name, indicated by the light grey brackets around the equation number.

12.2. REAL-TIME CHECKING OF USER INPUT IN A DIALOGUE BOX

In the code above, the input of the user about the bookmark name is validated after pressing the OK button. It is even more elegant to check the user input for the bookmark name while typing. In the particular case of the bookmark name, only letters, numbers and underscores are accepted, and when the length exceeds 40 characters, no additional input is accepted.

To do such realtime controls, you need to replace the built-in VBA function *InputBox* with a custom *UserForm*. This is necessary because the standard VBA *InputBox* cannot validate input while the user is typing.

This *UserForm* should be created in the Normal.dotm template, in the Visual Basic Editor. So first open the code of the macro “*Insert_centered_equation*” and make sure you are in the project “Normal”. The project is shown in the left menu of the Visual Basic Editor (Figure 164). Then create the *UserForm* with the command *Insert > UserForm*.

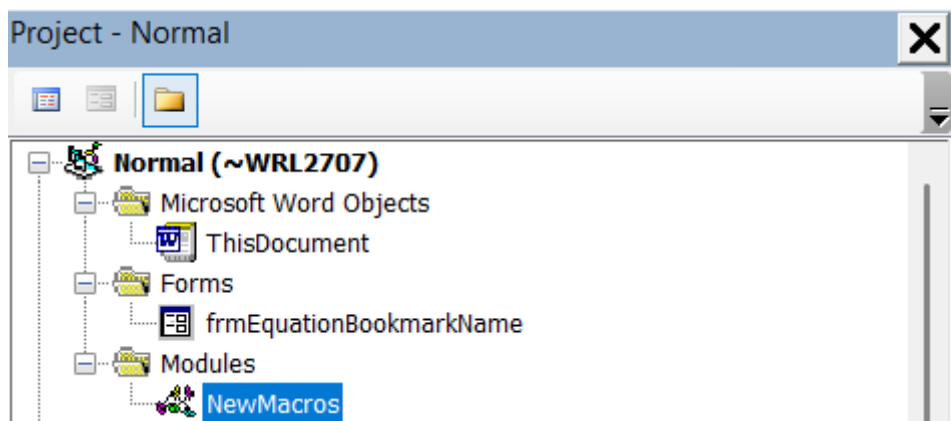


Figure 164 Active location where the Visual Basic project is stored.

Once this form is created, you need to give it the specific name *frmEquationBookmarkName* (because the VBA code below assumes this name for the *UserForm*).

Then you can create the layout of the *UserForm* with for example two *Labels* for user instructions and status of the user input, one *TextBox* for the required input of the bookmark name, and two *CommandButtons* for the OK and Cancel button.

An example of the possible labels and command buttons is shown in Figure 165.

Control type	Name
TextBox	txtBookmarkName
CommandButton	cmdOK
CommandButton	cmdCancel
Label	lblInstruction
Label	lblStatus

Figure 165 Possible labels and buttons for the *UserForm* that asks the user to input the bookmark name.

Since a bookmark name is restricted to 40 characters, you can set the property “*MaxLength*” of the *TextBox* to 40 characters.

The layout may look like Figure 166:

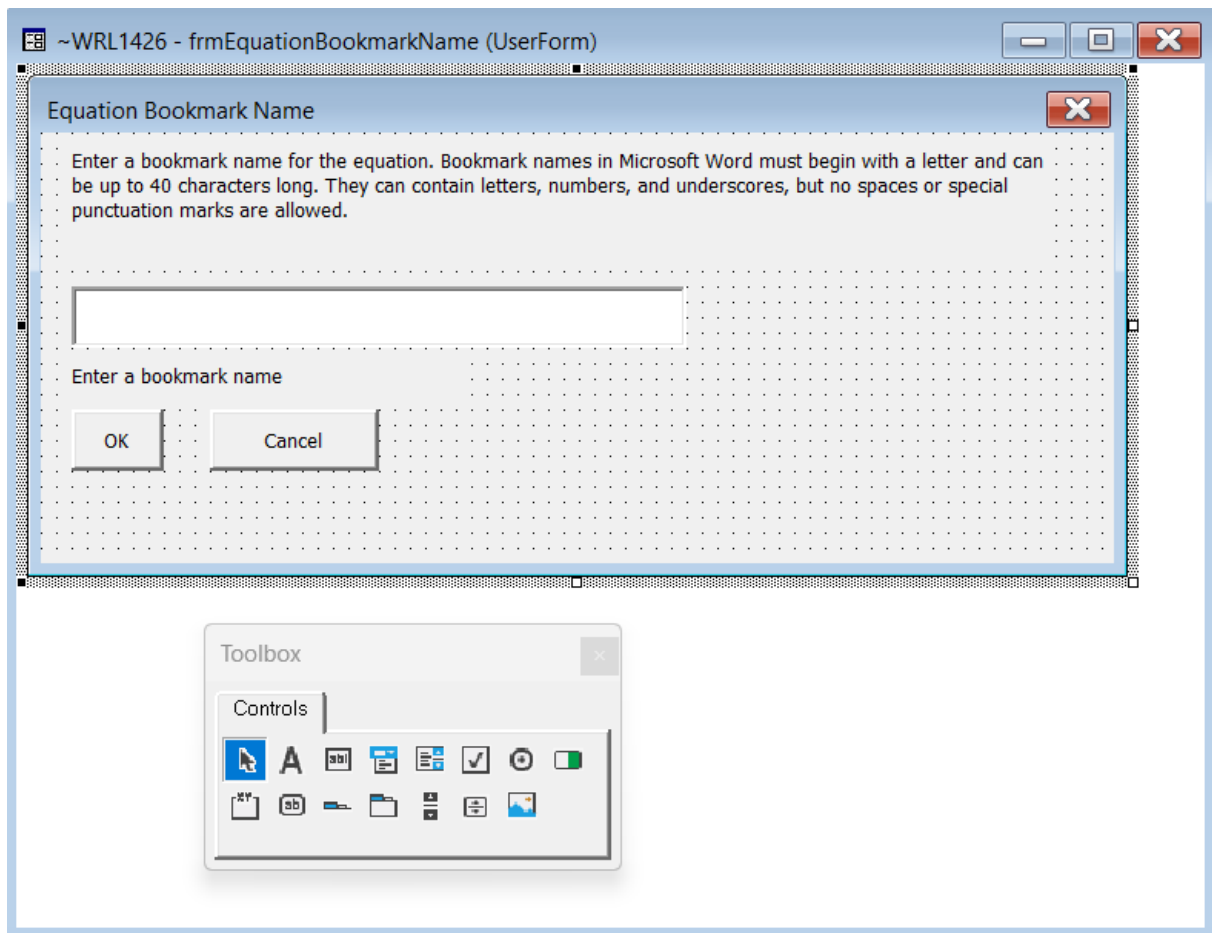


Figure 166 Suggested layout for the *UserForm* that asks the user to input the bookmark name.

Then you have to right-click on the Form *frmEquationBookmarkName* and choose “View code”. The code is still empty.

```
Private Sub UserForm_Click()

End Sub
```

Replace the entire code by this code:

Visual Basic code for *UserForm* with real-time validation of user input

Option Explicit

```
Private m_Cancelled As Boolean
Private m_BookmarkName As String
Private m_InternalUpdate As Boolean
```

```
Public Property Get Cancelled() As Boolean
    Cancelled = m_Cancelled
End Property
```

```

Public Property Get BookmarkName() As String
    BookmarkName = m_BookmarkName
End Property

Private Sub UserForm_Initialize()
    m_Cancelled = True
    m_BookmarkName = ""
    m_InternalUpdate = False

    Me.Caption = "Equation Bookmark Name"

    lblInstruction.Caption = _
        "Enter a bookmark name for the equation." & vbCrLf & vbCrLf & _
        "Bookmark names in Microsoft Word must begin with a letter and can be up to 40"
        "characters long." & vbCrLf & _
        "They can contain letters, numbers, and underscores, but no spaces or special"
        "punctuation marks are allowed."

    lblStatus.Caption = "Enter a bookmark name."

    txtBookmarkName.Text = ""
    txtBookmarkName.MaxLength = 40

    cmdOK.Caption = "OK"
    cmdCancel.Caption = "Cancel"
End Sub

Private Sub txtBookmarkName_KeyPress(ByVal KeyAscii As MSForms.ReturnInteger)
    Dim TypedChar As String

    If KeyAscii = vbKeyBack Then
        Exit Sub
    End If

    If KeyAscii = vbKeyReturn Then
        KeyAscii = 0
        cmdOK_Click
        Exit Sub
    End If

    If KeyAscii >= 32 Then
        TypedChar = Chr$(KeyAscii)

        If Not IsAllowedBookmarkCharacter(TypedChar) Then
            KeyAscii = 0
            Exit Sub
        End If

        If Len(txtBookmarkName.Text) >= 40 And txtBookmarkName.SelLength = 0 Then
            KeyAscii = 0

```



```

        Exit Sub
    End If
End If
End Sub

Private Sub txtBookmarkName_Change()
    Dim OriginalText As String
    Dim CleanText As String
    Dim OriginalSelectionStart As Long

    If m_InternalUpdate Then
        Exit Sub
    End If

    OriginalText = txtBookmarkName.Text
    OriginalSelectionStart = txtBookmarkName.SelStart

    CleanText = CleanBookmarkText(OriginalText)

    If CleanText <> OriginalText Then
        m_InternalUpdate = True

        txtBookmarkName.Text = CleanText

        If OriginalSelectionStart > Len(CleanText) Then
            txtBookmarkName.SelStart = Len(CleanText)
        Else
            txtBookmarkName.SelStart = OriginalSelectionStart
        End If

        m_InternalUpdate = False
    End If

    UpdateStatusMessage
End Sub

Private Sub cmdOK_Click()
    m_BookmarkName = txtBookmarkName.Text
    m_Cancelled = False
    Me.Hide
End Sub

Private Sub cmdCancel_Click()
    m_BookmarkName = ""
    m_Cancelled = True
    Me.Hide
End Sub

Private Sub UserForm_QueryClose(Cancel As Integer, CloseMode As Integer)
    If CloseMode = vbFormControlMenu Then

```

```

        m_BookmarkName = ""
        m_Cancelled = True
    End If
End Sub

Private Function CleanBookmarkText(ByVal RawText As String) As String
    Dim i As Long
    Dim CurrentChar As String
    Dim ResultText As String

    ResultText = ""

    For i = 1 To Len(RawText)
        CurrentChar = Mid$(RawText, i, 1)

        If IsAllowedBookmarkCharacter(CurrentChar) Then
            ResultText = ResultText & CurrentChar
        End If

        If Len(ResultText) >= 40 Then
            Exit For
        End If
    Next i

    CleanBookmarkText = ResultText
End Function

Private Function IsAllowedBookmarkCharacter(ByVal TestChar As String) As Boolean
    IsAllowedBookmarkCharacter = TestChar Like "[A-Za-z0-9_]"
End Function

Private Function StartsWithLetter(ByVal TestText As String) As Boolean
    If Len(TestText) = 0 Then
        StartsWithLetter = False
    Else
        StartsWithLetter = Left$(TestText, 1) Like "[A-Za-z]"
    End If
End Function

Private Sub UpdateStatusMessage()
    If Len(txtBookmarkName.Text) = 0 Then
        lblStatus.Caption = "Enter a bookmark name."
    ElseIf Not StartsWithLetter(txtBookmarkName.Text) Then
        lblStatus.Caption = "The bookmark name must begin with a letter."
    ElseIf Len(txtBookmarkName.Text) >= 40 Then
        lblStatus.Caption = "Maximum length of 40 characters reached."
    Else
        lblStatus.Caption = "Bookmark name format is valid."
    End If
End Sub

```

Finally, also replace the code of the macro *Insert_centered_equation* by the one below, in order to call the *UserForm* and read the user input from the *UserForm*.

Visual Basic code for automatically inserting equations with real-time validation of user input for the bookmark name

```
Sub Insert_centered_equation()
'
' Insert_centered_equation Macro
'
'
    Dim BookmarkName As String
    Dim IsValid As Boolean
    Dim RegEx As Object
    Dim EqNumberField As Field
    Dim EqNumberRange As Range
    Dim BookmarkForm As frmEquationBookmarkName

    Set RegEx = CreateObject("VBScript.RegExp")
    RegEx.Pattern = "[A-Za-z][A-Za-z0-9_]{0,39}$"
    RegEx.IgnoreCase = False
    RegEx.Global = False

    Do
        Set BookmarkForm = New frmEquationBookmarkName
        BookmarkForm.Show vbModal

        If BookmarkForm.Cancelled Then
            Unload BookmarkForm
            Exit Sub
        End If

        BookmarkName = BookmarkForm.BookmarkName
        Unload BookmarkForm

        If BookmarkName = "" Then
            Exit Sub
        End If

        IsValid = RegEx.Test(BookmarkName)

        If IsValid Then
            If ActiveDocument.Bookmarks.Exists(BookmarkName) Then
                IsValid = False
                MsgBox "A bookmark with this name already exists. Please enter another bookmark name.", _
                    vbExclamation, "Bookmark Already Exists"
            End If
        End If
    End Do
End Sub
```

```

If Not IsValid Then
    MsgBox "Invalid bookmark name." & vbCrLf & vbCrLf & _
        "The bookmark name must:" & vbCrLf & _
        "- Begin with a letter" & vbCrLf & _
        "- Be no longer than 40 characters" & vbCrLf & _
        "- Contain only letters, numbers, and underscores" & vbCrLf & _
        "- Not already exist in the document", _
        vbExclamation, "Invalid Bookmark Name"
End If

Loop Until IsValid

Selection.Style = ActiveDocument.Styles("Centered_equation")
Selection.TypeText Text:=vbTab & vbTab & "("

Set      EqNumberField      =      Selection.Fields.Add(Range:=Selection.Range,
Type:=wdFieldEmpty, _
    PreserveFormatting:=False)

Selection.TypeText Text:="SEQ my_equations"
Selection.Fields.Update

Set EqNumberRange = EqNumberField.Result.Duplicate
EqNumberRange.Select
ActiveDocument.Bookmarks.Add Name:=BookmarkName, Range:=EqNumberRange

Selection.EndKey Unit:=wdLine
Selection.TypeText Text:=")"
Selection.HomeKey Unit:=wdLine
Selection.MoveRight Unit:=wdCharacter, Count:=1

Dim objRange As Range
Dim objOMath As OMath

' Get the current cursor location
Set objRange = Selection.Range

' Add an Office Math equation structure to the text range
Set objOMath = objRange.OMaths.Add(objRange).OMaths(1)

' Optional: Convert it to a professionally built-up visual display format
objOMath.BuildUp

' Position the cursor right inside the new equation placeholder
objRange.Select
End Sub

```

Do not forget to press the *Save* button in Visual Basic Editor, before you return to the Word document. When you execute the macro now, the user input is combined with real-time validation of the input, as shown in Figure 167.

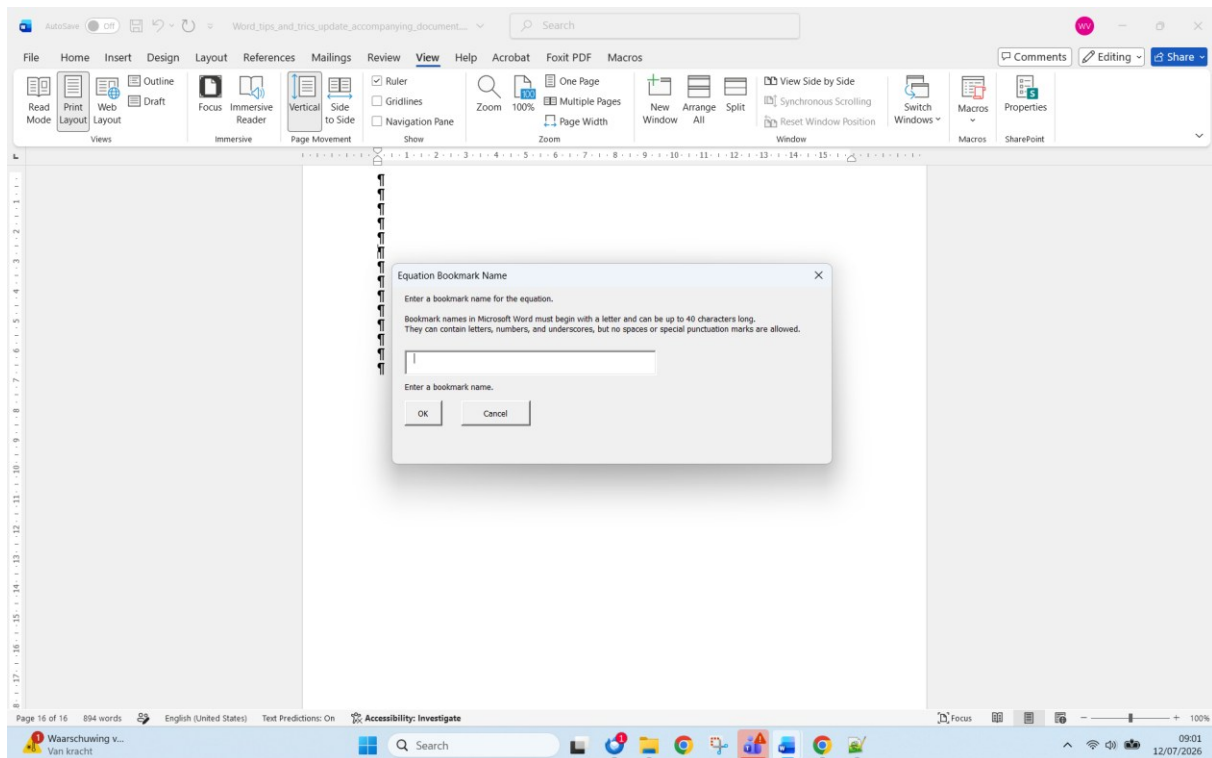


Figure 167 Result of running the updated macro with user-validated input of the bookmark name.

12.3. FIND & REPLACE IN FIELD CODES

Another useful macro is the Find & Replace function to search text strings in field codes and replace them by another text string.

Indeed, if you want to change the parameters of a particular field code, you can just press <Alt> + <F9> to show the field code, and then modify the parameters. However, if you have many field codes that you would like to change all at once, it is more difficult. Word does not allow to do Find & Replace in field codes, even not when the field code texts are made visible with <Alt> + <F9>.

Therefore, you need to use a macro in Visual Basic code, that searches all fields and replaces certain text strings by another.

Here again, it is handy to ask for input of the user which text string to search for, and with which text string to replace it.

The entire Visual Basic code is shown below. You can assign this macro again to the Ribbon Macros.

If you frequently use field codes like SEQ, STYLEREF, REF, etc., this macro is also very handy to change or update many field codes at once.

Visual Basic code for finding and replacing text strings in field codes

```

Public Sub Find_and_replace_in_fields()

    Dim findText As String
    Dim replaceText As String
    Dim response As VbMsgBoxResult

    Dim fieldsSearched As Long
    Dim replacementsDone As Long
    Dim fieldsWithErrors As Long

    Dim doc As Document

    Set doc = ActiveDocument

    'Ask for the text to search for.
    findText = InputBox( _
        Prompt:="Enter the text string to search for in all field codes:" & vbCrLf & vbCrLf & _
        "Example: Figure_4.", _
        Title:="Search Text in Field Codes")

    'InputBox returns an empty string if the user presses Cancel.
    'An empty search string is not meaningful, so abort.
    If Len(findText) = 0 Then
        MsgBox "Operation cancelled or no search text was entered." & vbCrLf & _
            "No field codes were changed.", _
            vbInformation, "Cancelled"
        Exit Sub
    End If

    'Ask for the replacement text.
    replaceText = InputBox( _
        Prompt:="Enter the replacement text string:" & vbCrLf & vbCrLf & _
            "Example: Figure_5." & vbCrLf & vbCrLf & _
            "Leave empty only if you want to remove the search text.", _
        Title:="Replacement Text in Field Codes")

    'Because InputBox cannot distinguish Cancel from an empty replacement string,
    'ask for confirmation if the replacement text is empty.
    If Len(replaceText) = 0 Then
        response = MsgBox( _
            Prompt:="The replacement text is empty." & vbCrLf & vbCrLf & _
                "Do you want to replace each occurrence of:" & vbCrLf & _
                """" & findText & """" & vbCrLf & vbCrLf & _
                "with nothing?", _
            Buttons:=vbYesNoCancel + vbQuestion, _
            Title:="Confirm Empty Replacement")
    End If
End Sub

```

```

    If response <> vbYes Then
        MsgBox "Operation cancelled. No field codes were changed.", _
            vbInformation, "Cancelled"
        Exit Sub
    End If
End If

Application.ScreenUpdating = False

ReplaceInAllStoryRanges doc, findText, replaceText, _
    fieldsSearched, replacementsDone, fieldsWithErrors

Application.ScreenUpdating = True

MsgBox "Field-code replacement completed." & vbCrLf & vbCrLf & _
    "Field codes successfully searched: " & fieldsSearched & vbCrLf & _
    "Text string replacements done: " & replacementsDone & vbCrLf & _
    "Field codes skipped because of errors: " & fieldsWithErrors, _
    vbInformation, "Replace Text in Field Codes"

End Sub

Private Sub ReplaceInAllStoryRanges( _
    ByVal doc As Document, _
    ByVal findText As String, _
    ByVal replaceText As String, _
    ByRef fieldsSearched As Long, _
    ByRef replacementsDone As Long, _
    ByRef fieldsWithErrors As Long)

    Dim storyRange As Range
    Dim currentRange As Range

    For Each storyRange In doc.StoryRanges

        Set currentRange = storyRange

        Do While Not currentRange Is Nothing

            ReplaceInFieldsOfRange currentRange, findText, replaceText, _
                fieldsSearched, replacementsDone, fieldsWithErrors

            Set currentRange = currentRange.NextStoryRange

        Loop

        Next storyRange

    End Sub

```

```

Private Sub ReplaceInFieldsOfRange( _
    ByVal rng As Range, _
    ByVal findText As String, _
    ByVal replaceText As String, _
    ByRef fieldsSearched As Long, _
    ByRef replacementsDone As Long, _
    ByRef fieldsWithErrors As Long)

    Dim fld As Field
    Dim oldCode As String
    Dim newCode As String
    Dim occurrenceCount As Long

    For Each fld In rng.Fields

        oldCode = vbNullString
        occurrenceCount = 0

        On Error Resume Next
        oldCode = fld.Code.Text

        If Err.Number <> 0 Then
            fieldsWithErrors = fieldsWithErrors + 1
            Err.Clear
            On Error GoTo 0
        Else
            On Error GoTo 0

            fieldsSearched = fieldsSearched + 1

            occurrenceCount = CountOccurrences(oldCode, findText)

            If occurrenceCount > 0 Then

                newCode = Replace(oldCode, findText, replaceText, 1, -1, vbBinaryCompare)

                On Error Resume Next
                fld.Code.Text = newCode

                If Err.Number = 0 Then
                    replacementsDone = replacementsDone + occurrenceCount

                    'Update the field result after changing the field code.
                    fld.Update
                Else
                    fieldsWithErrors = fieldsWithErrors + 1
                    Err.Clear
                End If
            End If
        End If
    End For

```



```
        On Error GoTo 0

    End If

End If

Next fld

End Sub

Private Function CountOccurrences( _
    ByVal sourceText As String, _
    ByVal searchText As String) As Long

    Dim position As Long
    Dim count As Long
    Dim startAt As Long

    If Len(searchText) = 0 Then
        CountOccurrences = 0
        Exit Function
    End If

    startAt = 1

    Do
        position = InStr(startAt, sourceText, searchText, vbBinaryCompare)

        If position > 0 Then
            count = count + 1
            startAt = position + Len(searchText)
        Else
            Exit Do
        End If

    Loop

    CountOccurrences = count

End Function
```

In this chapter, we have discussed two macros that automate the tasks of inserting equations and find & replace in field codes with user input.

With the advent of AI assisted coding, you can now automate almost any task in Word with a macro. In this chapter, you have basically seen all functionalities that you need, to replace simple macros with more advanced Visual Basic code, and in this way execute complex tasks in an automated way.

Inserting equations in Word is often seen as the weakest point of Word, where LaTeX claims to outperform Word in many ways. We have already learned many things about equations in Word:

- Assigning a user-defined paragraph style to the equations, to make them centered
- Adding a number to the equations with the field code SEQ, so that after moving the equations to another place in the document, the numbering is automatically updated
- Adding even the chapter number is possible, by using the STYLEREF field code
- Automating these actions into a macro and adding that macro to a Ribbon, so that one mouse click is sufficient to automate these tasks
- Adding the Visual Basic code to also launch the Microsoft Equation Editor automatically
- Asking input from the user to add a bookmark name to the equation
- Replacing the *InputBox* with a *UserForm*, to have real-time validation of the user input and make sure that the bookmark name is always valid.

The last step is to dive into the Microsoft Equation Editor itself, and give some advice how to layout your equations better. That is discussed in the next chapter.

13

Formatting equations

The Microsoft Word Equation editor is frequently considered as a rudimentary editor for mathematical equations, especially by the LaTeX community. Unfortunately, most users of the Microsoft Word equation editor do not fully exploit the capabilities of this editor. For example, one of the major headaches is to nicely align different equations, but as shown in the example equation below, you can solve this issue also with the Microsoft Word equation editor.

$$\begin{aligned} \frac{dD_{11}}{dN} &= \begin{cases} c_1 \cdot \Sigma_{11} \cdot \exp\left(-c_2 \frac{D_{11}}{\sqrt{\Sigma_{11}}}\right) + c_3 \cdot D_{11} \cdot \Sigma_{11}^2 \cdot [1 + \exp(c_5(\Sigma_{11} - c_4))] & \text{if } \sigma_{11} \geq 0 \\ \left[c_1 \cdot \Sigma_{11} \cdot \exp\left(-c_2 \frac{D_{11}}{\sqrt{\Sigma_{11}}}\right) \right]^3 + c_3 \cdot D_{11} \cdot \Sigma_{11}^2 \cdot \left[1 + \exp\left(\frac{c_5}{3}(\Sigma_{11} - c_4)\right) \right] & \text{if } \sigma_{11} < 0 \end{cases} \\ \\ \frac{dD_{22}}{dN} &= \begin{cases} c_1 \cdot \Sigma_{22} \cdot \exp\left(-c_2 \frac{D_{22}}{\sqrt{\Sigma_{22}}}\right) + c_3 \cdot D_{22} \cdot \Sigma_{22}^2 \cdot [1 + \exp(c_5(\Sigma_{22} - c_4))] & \text{if } \sigma_{22} \geq 0 \\ \left[c_1 \cdot \Sigma_{22} \cdot \exp\left(-c_2 \frac{D_{22}}{\sqrt{\Sigma_{22}}}\right) \right]^3 + c_3 \cdot D_{22} \cdot \Sigma_{22}^2 \cdot \left[1 + \exp\left(\frac{c_5}{3}(\Sigma_{22} - c_4)\right) \right] & \text{if } \sigma_{22} < 0 \end{cases} \\ \\ \frac{dD_{12}}{dN} &= c_1 \cdot \Sigma_{12} \cdot \exp\left(-c_2 \frac{D_{12}}{\sqrt{\Sigma_{12}}}\right) + c_3 \cdot D_{12} \cdot \Sigma_{12}^2 \cdot [1 + \exp(c_5(\Sigma_{12} - c_4))] \end{aligned}$$

For this example equation, we will show step by step how to control the layout.

13.1. ALIGNING EQUATIONS

Tables (or equivalent ‘matrices’ in the Microsoft Word Equation Editor) are very powerful to correctly align all terms in the equations.

First you start by inserting a new equation in your document. This can already be done with one mouse click, by using the previously recorded macro “*Insert_centered_equation*”.

After the equation has been inserted, you go to the *Equation* Ribbon in the top menu, so that the toolbar of the Microsoft Word Equation Editor is shown, as illustrated in Figure 168.

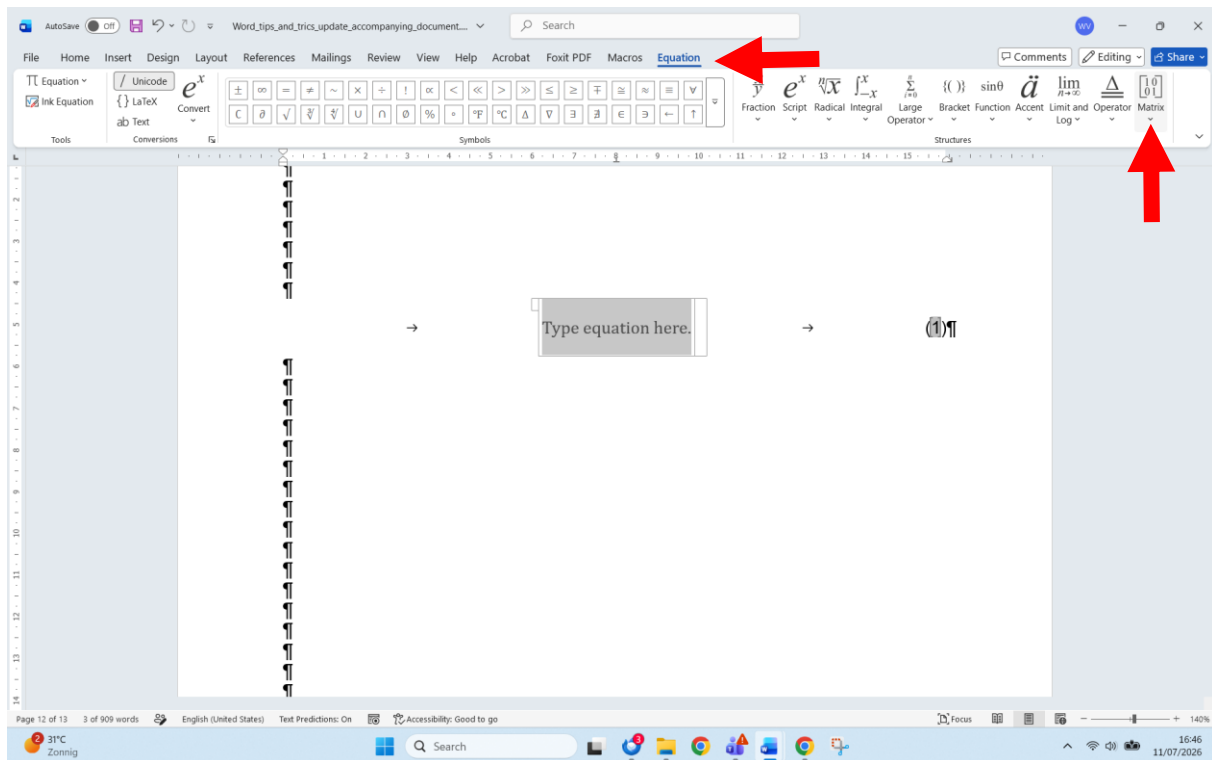


Figure 168 Toolbar of the Microsoft Word Equation Editor.

You select the *Matrix* toolbar (the rightmost button on the toolbar) and you insert a matrix with three rows and three columns in the Equation Editor, as illustrated in Figure 169.

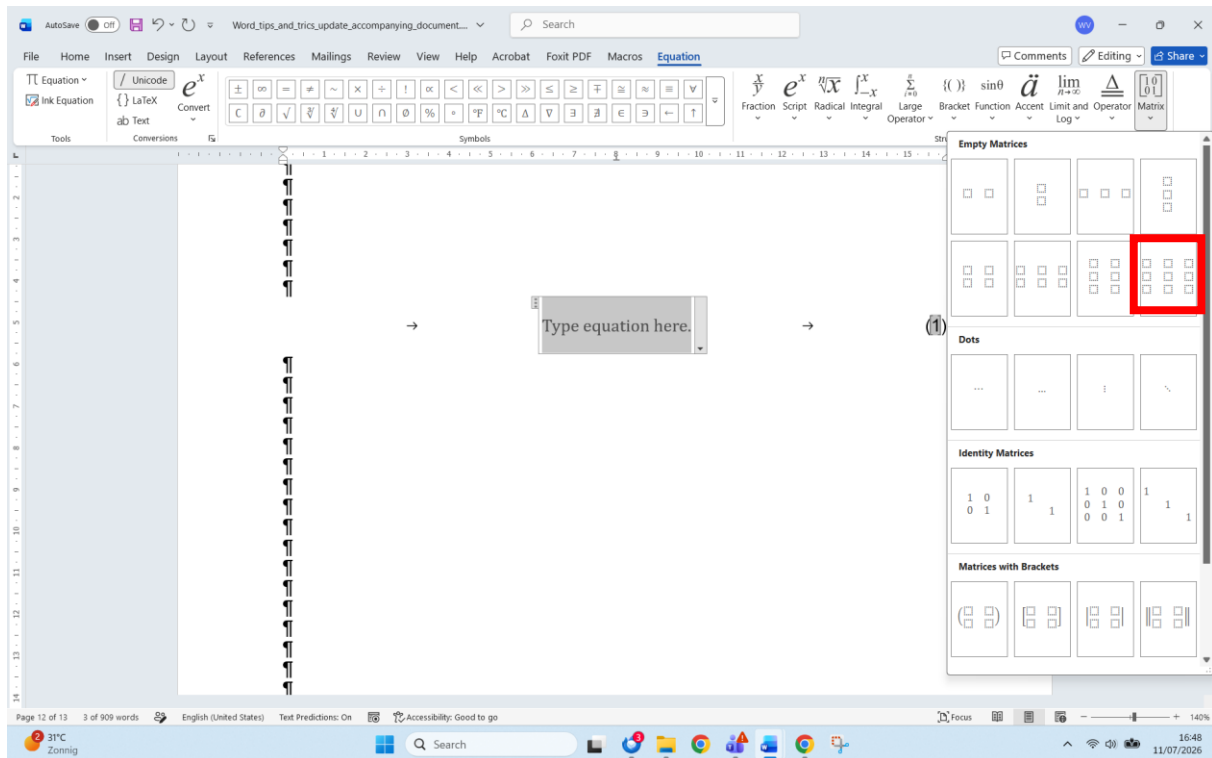


Figure 169 Inserting a 3 x 3 matrix into the Equation Editor.

For the example equation that we want to layout, we extend the 3 x 3 matrix to a 5 x 3 matrix, because the 2 empty rows between the 3 filled rows act as vertical spacing. Since a 3 x 3 matrix is the largest matrix that you can insert through the matrix toolbar directly, you have to change the number of rows and columns afterwards. You do that by right-clicking in a cell of the inserted matrix and choose the option “*Insert > Insert row after*”, as shown in Figure 170.

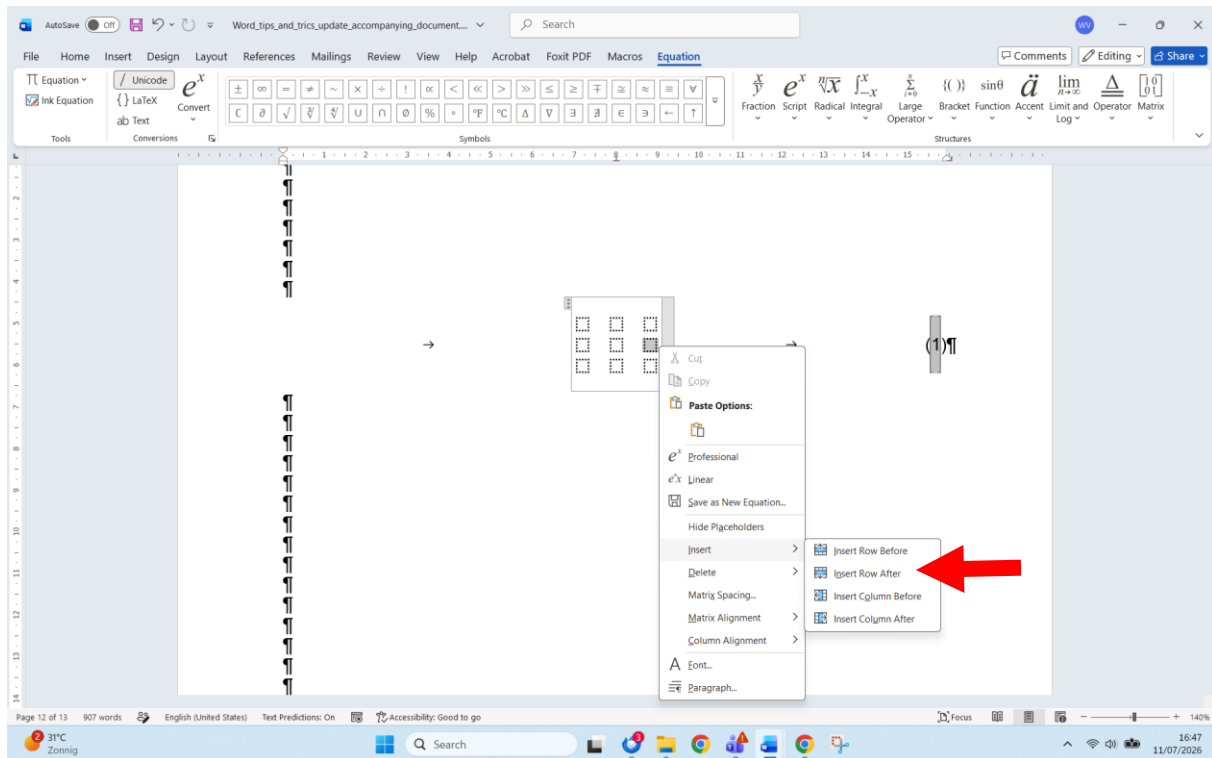


Figure 170 Expanding the 3 x 3 matrix to a 5 x 3 matrix.

Figure 171 shows the partially filled matrix. By using the *Matrix* layout, the “=” signs in the second column are already aligned and there is some vertical spacing between the three differential equations by the empty row 2 and 4, and the vertical bars for the third cell of the first and third row are already inserted.

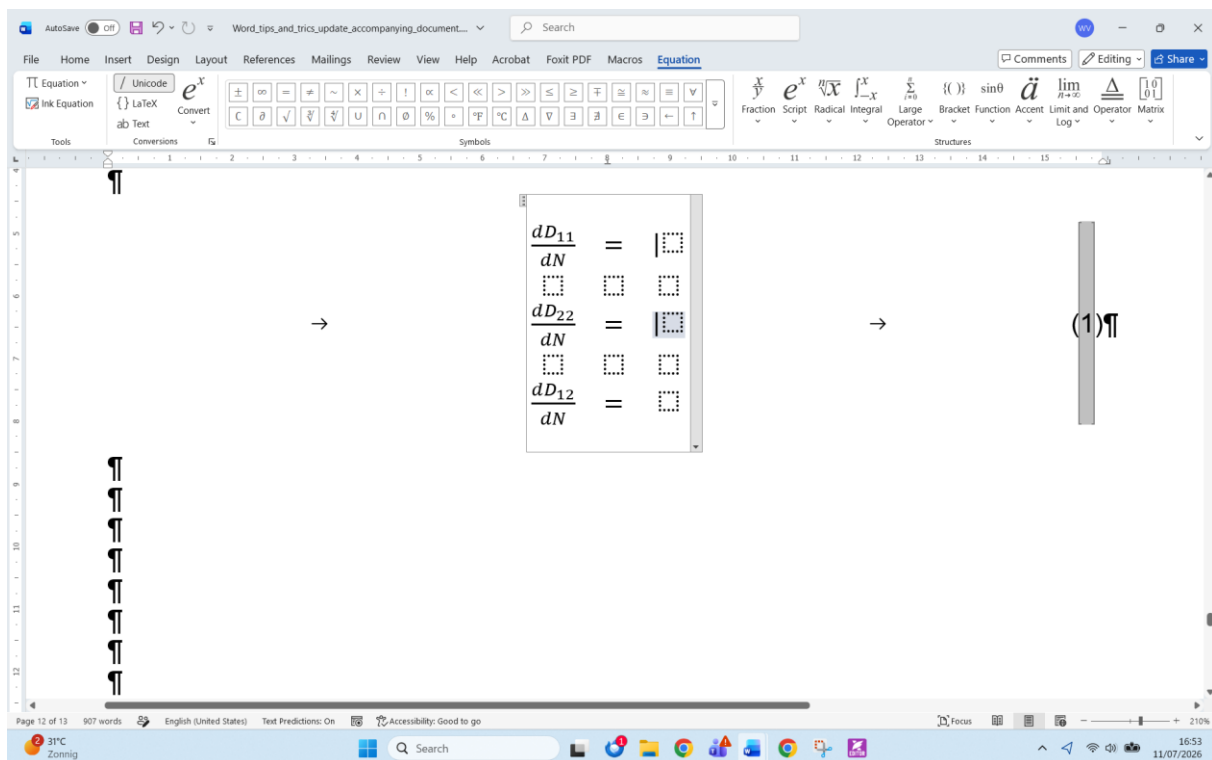


Figure 171 Partially completed matrix for aligning the three differential equations.

Now, precisely in this third cell of the first and third row (the ones with the vertical bar), you insert a sub-matrix with two rows and three columns, as shown in Figure 172.

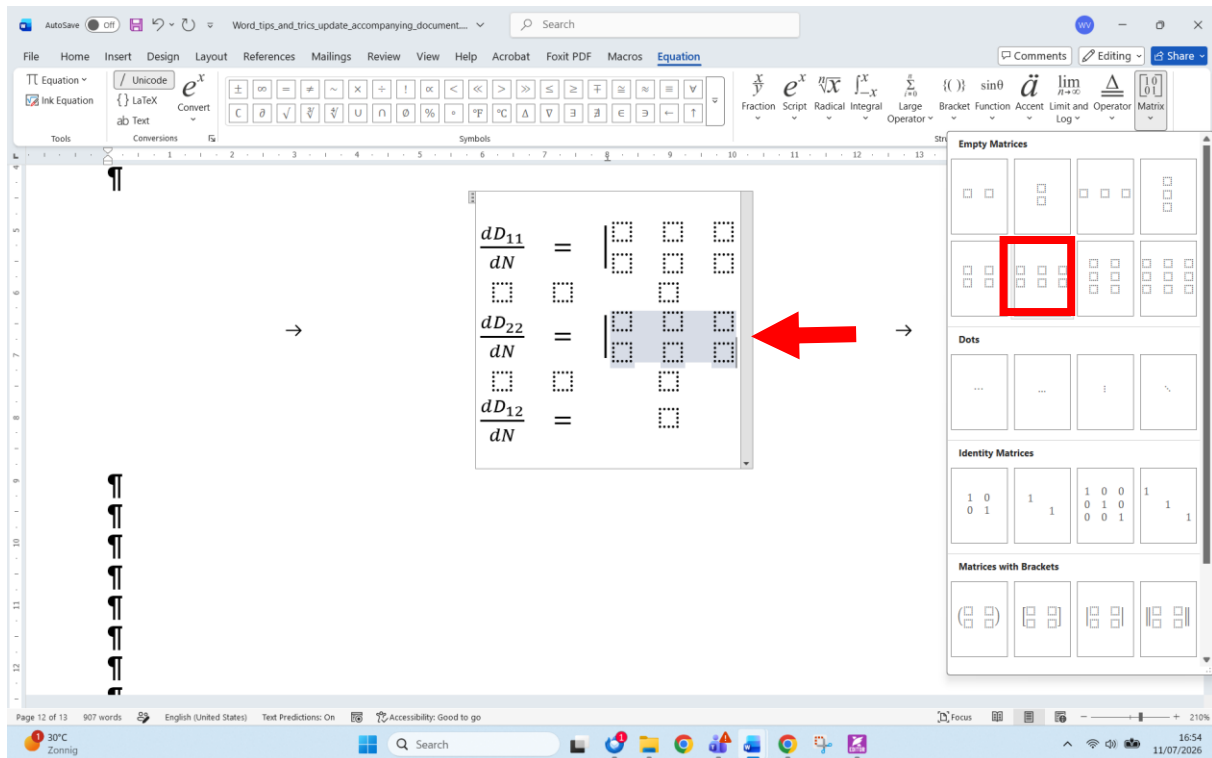


Figure 172 Inserting a sub-matrix in one cell of the matrix.

Figure 173 shows the further completion of the matrix layout.

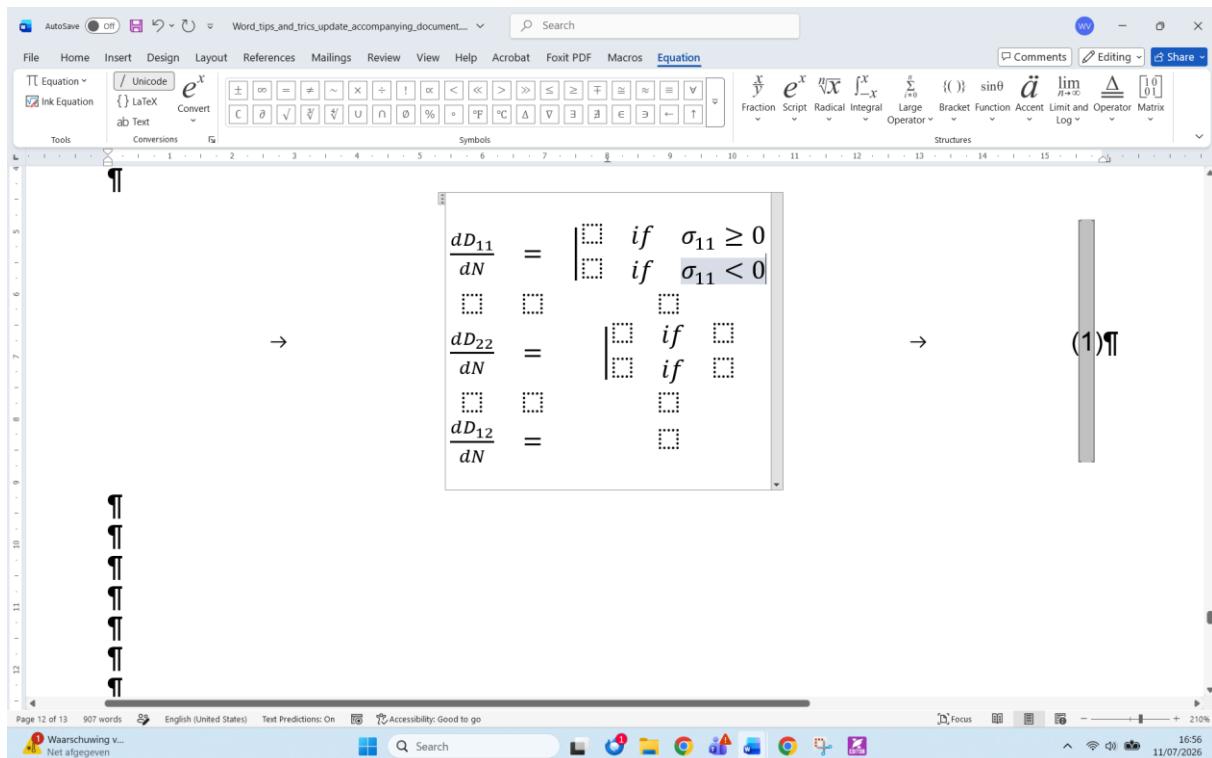


Figure 173 Further completion of the matrix with the equations.

Now that you know how to align your equations, the next section discusses some more advanced *Math* commands to format your equations even better.

13.2. USING MATH COMMANDS

You can still add a lot more advanced control of the equations, by typing so called “*Math commands*”. These *Math commands* are replaced by the intended character, by using the Word *Autocorrect* function. You can find a description of all those *Math commands*, by opening the global menu *File > Options > Proofing > Autocorrect options...* In that menu, you click the button “*Autocorrect options...*”, as shown in Figure 174.

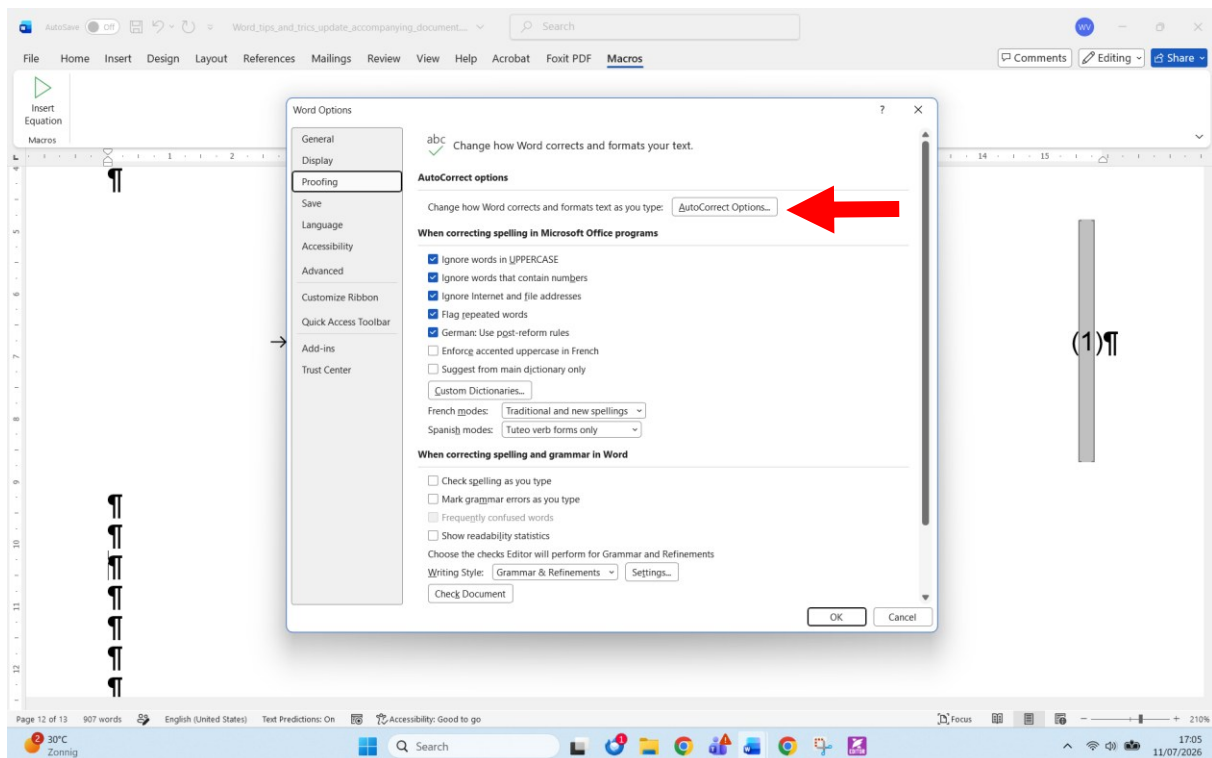


Figure 174 Accessing the *Autocorrect* options in the Word proofing tools dialogue box.

Under the tab “*Math Autocorrect*”, you can find all the *Autocorrect* commands that you can use in the *Math* editor (Figure 175). All these commands are autocorrected by writing the command, followed by pressing the spacebar.

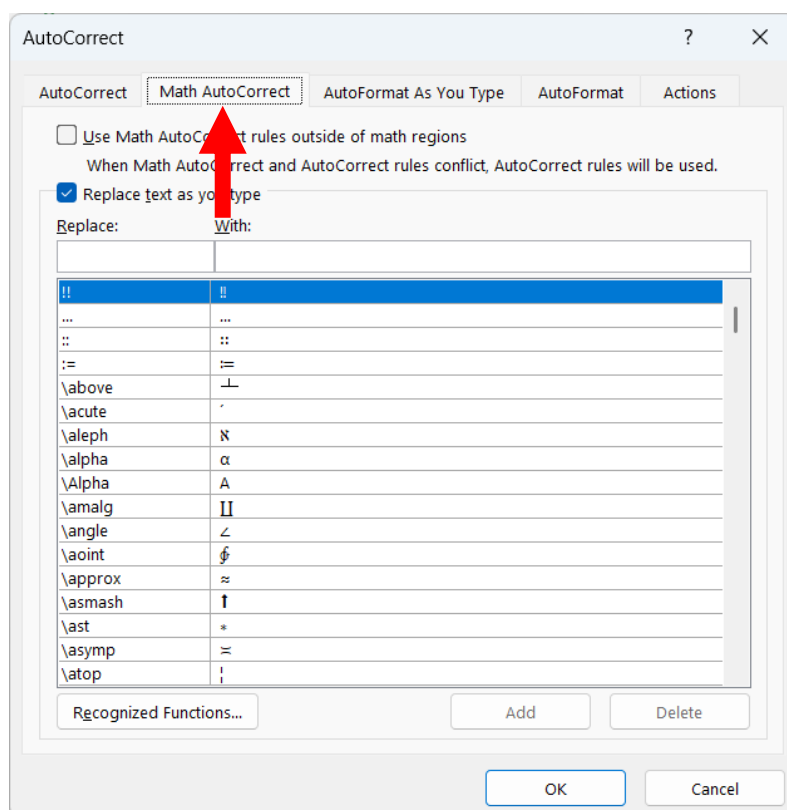


Figure 175 Math Autocorrect window with all Math commands.

In the formula example that we treated before, you can for example add some standardized horizontal spacing in the third columns to separate the vertical bars from the subsequent equations.

The *Math* command for that is `\thicksp` (thick space). To convert it to the actual space symbol, you type `\thicksp`, followed by pressing the Spacebar. Then the command is replaced by a narrow rectangle, as shown in Figure 176.

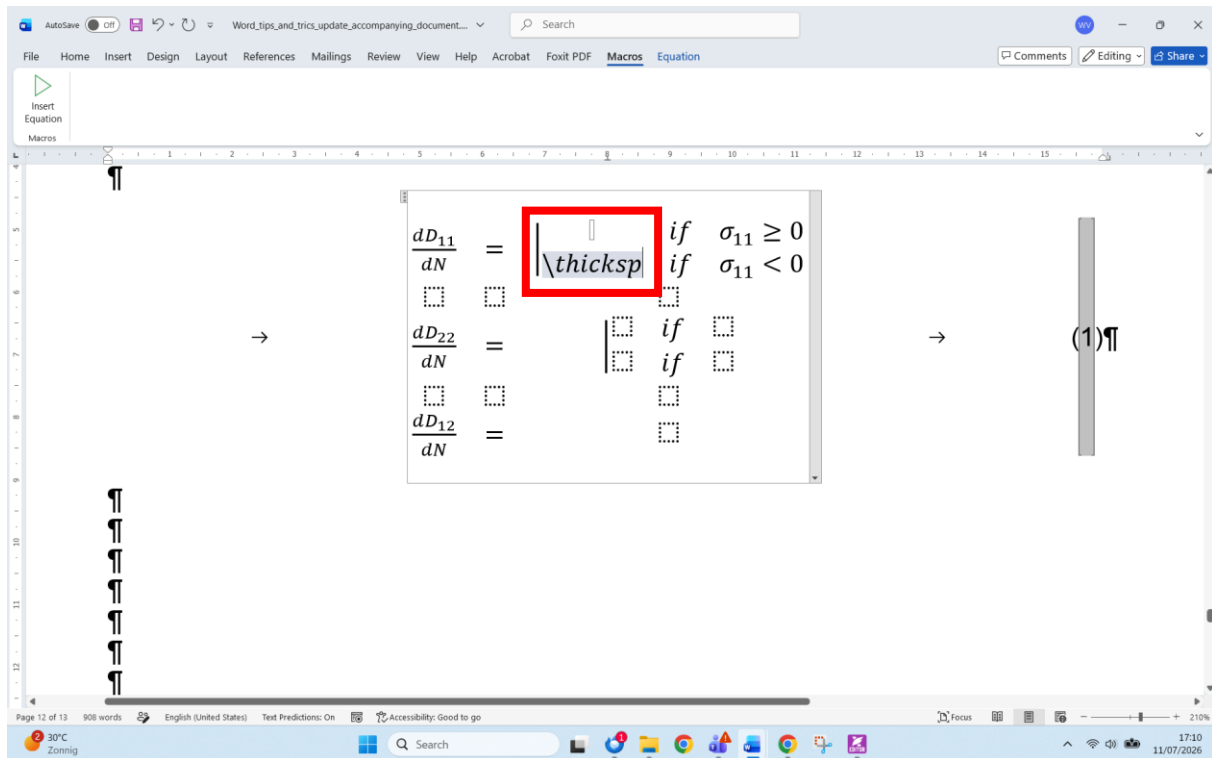


Figure 176 Adding some standardized horizontal spacing between the vertical bars and the equations.

Finally, it is important to note that these *Math* commands only work if the typing mode is set to accept Unicode characters. This is the default setting in the *Conversions* toolbar of the *Equation* ribbon, as illustrated in Figure 177.

In the same figure, you can see that if you click the little expand button of the *Conversions* toolbar, the *Equation options* window appears. In that window, you can also click the button “*Math Autocorrect...*” and then you are taken to the same menu as through the menu *File > Options > Proofing > Autocorrect options...*

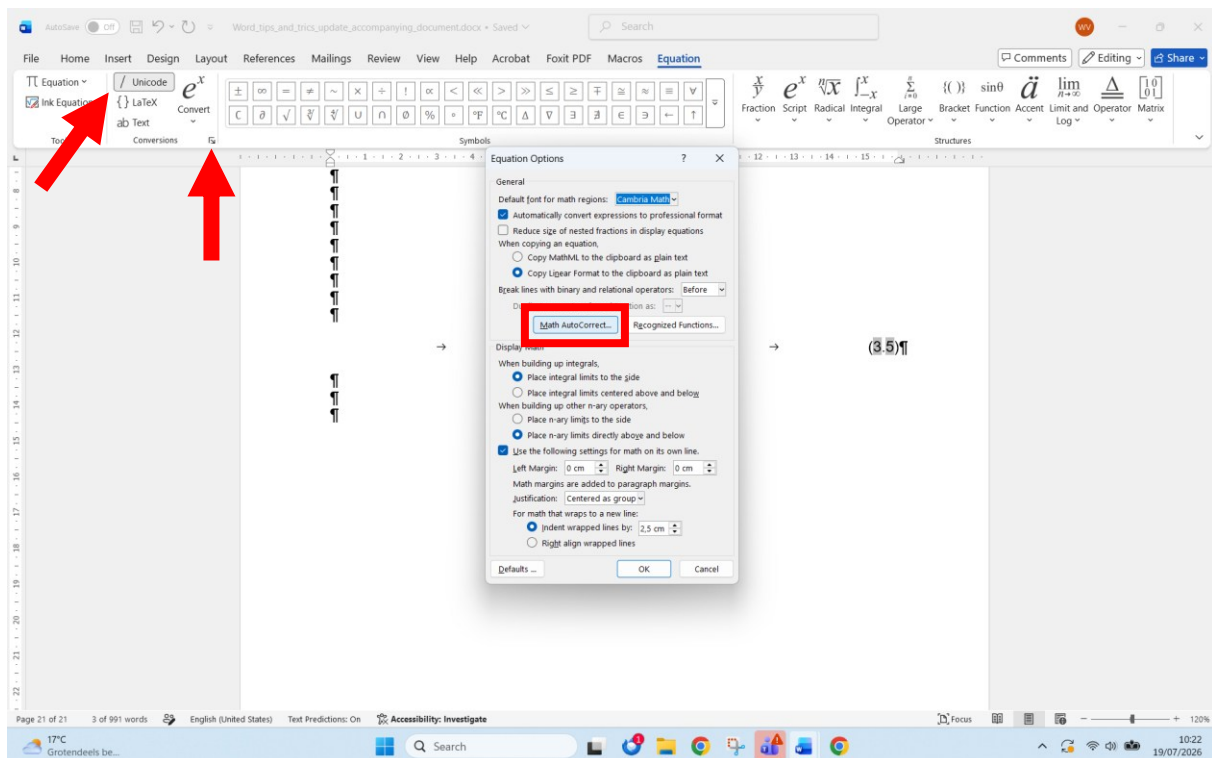


Figure 177 Accessing the *Math AutoCorrect* options from the *Equation* toolbar.

If you search for dedicated tutorials about the Microsoft Equation Editor on the Internet, it is confusing to find many old tutorials and webpages that still refer to the Microsoft Equation Editor 3.0. This Equation Editor was popular in Word 2007 and later releases, but got obsolete because of security holes. Microsoft removed Equation Editor 3.0 in January 2018 due to severe security vulnerabilities that left users open to remote code execution. Because it was originally a legacy, third-party component licensed from Design Science, Microsoft chose to completely retire it rather than rewrite the code.

In particular for our example of the thick space *\thicksp*, old tutorials refer to the “*Spacing*” toolbar where you could insert all kinds of spaces, but that is a toolbar from the old Microsoft Equation Editor 3.0, and this *Spacing* toolbar has disappeared in the current Microsoft Equation Editor.

If you want to deepen your knowledge on the Equation Editor and *Math AutoCorrect* commands, you can search on the Internet for these references:

- Microsoft’s Quick start guide to *Math AutoCorrect* commands and symbols
- Daisy consortium’s List of Commonly Used *Math AutoCorrect* Entries in Word 365
- If you want to specialize in writing equations in a LaTeX like style, there is an excellent webpage about *Math AutoCorrect* commands and Microsoft Equation Editor by Ms. Jennifer Marsala, Instructional Designer at the University of Houston, Department of Mathematics (<https://online.math.uh.edu/typing-math-in-ms-word/>).

14

Managing references and citations

14.1. IMPORTANCE OF MANAGING REFERENCES

Managing references is particularly important when writing a MSc or PhD thesis, or a scientific publication.

From the very beginning of your research, it is important to keep a database of all articles you have gathered. If you write your own publications or you write your project proposal or MSc or PhD thesis, you will refer to the articles from other researchers. At that point, it saves a lot of time if these citations are managed by a reference manager. The main tasks of reference management software are:

- storing all bibliographic information of your collection of articles (title, authors, publication year, journal, volume/issue,...),
- inserting citations into your text documents written in Microsoft Word or LaTeX,
- automatically building the reference list at the end of your text document,
- renumbering the citations in your text if you move parts of your text up or down,
- changing the format of your references depending on the required format of the publisher where you submit your publication.

Why should you use this reference management software ? Suppose that I have written the text fragment shown below (for a conference paper, article or thesis):

Most one-dimensional damage models for fibre-reinforced composites only account for the effect of damage on the stiffness [1,2]. The degradation of the Poisson's ratio is not included in these models. Nevertheless this degradation has been observed and is not negligible [3,4]. Bandoh et al. showed that the Poisson's ratio of a carbon/epoxy UD laminate can drop by 50 % under static tensile loading [3], while Pidaparti and Vogt proved that the Poisson's ratio is a very sensitive parameter whilst monitoring fatigue damage in human bone [4].

Reference list:

- [1] Sidoroff, F. and Subagio, B. (1987). Fatigue damage modelling of composite materials from bending tests. In : Matthews, F.L., Buskell, N.C.R., Hodgkinson, J.M. and Morton, J. (eds.). Sixth International Conference on Composite Materials (ICCM-VI) & Second European Conference on Composite Materials (ECCM-II) : Volume 4. Proceedings, 20-24 July 1987, London, UK, Elsevier, pp. 4.32-4.39.
- [2] Brondsted, P., Lilholt, H. and Andersen, S.I. (1997). Fatigue damage prediction by measurements of the stiffness degradation in polymer matrix composites. In : Degallaix, S., Bathias, C. and Fougères, R. (eds.). International Conference on fatigue of composites. Proceedings, 3-5 June 1997, Paris, France, La Société Française de Métallurgie et de Matériaux, pp. 370-377.
- [3] Bandoh, S., Matsumura, K., Zako, M., Shiino, T. and Kurashiki, T. (2001). On the detection of fatigue damage in CFRP by measuring Poisson's ratio. In: Hui, D. (ed.). Eighth International Conference on Composites Engineering (ICCE/8). Proceedings. Tenerife, Spain, 5-11 August 2001, pp. 55-56.
- [4] Pidaparti, R.M. and Vogt, A. (2002). Experimental investigation of Poisson's ratio as a damage parameter for bone fatigue. *Journal of Biomedical Materials Research Part A*, 59 (2): 282-287.

If I want to insert a preceding paragraph with some additional background on damage mechanisms in composites, and this preceding paragraph also includes a number of references, then all subsequent references must be renumbered.

If you have inserted all references in your text automatically with reference management software, a reference number is inserted in your text for every reference and the full reference is added at the end of your document in the reference list.

In case of inserting the preceding paragraph, the renumbering is done automatically, and the resulting new text fragment is shown below:

Damage in fibre-reinforced composites can take many forms [1,2]: (i) matrix cracks, (ii) fibre-matrix interface failure, (iii) fibre pull-out, (iv) delaminations, (v) fibre fracture. This damage affects the value of the elastic properties at an early stage. Especially in fatigue, the damage initiation phase can cause a pronounced drop of the elastic modulus of 5 to 10 %. In the next damage propagation phase, the stiffness continues to decrease gradually, ranging from a few percent for unidirectionally reinforced carbon composites to several tens of percents for multidirectional glass laminates [3].

Most one-dimensional damage models for fibre-reinforced composites only account for the effect of damage on the stiffness [4,5]. The degradation of the Poisson's ratio is not included in these models. Nevertheless this degradation has been observed and is not negligible [6,7]. Bandoh et al. showed that the Poisson's ratio of a carbon/epoxy UD laminate can drop by 50 % under static tensile loading [6], while Pidaparti and Vogt proved that the Poisson's ratio is a very sensitive parameter whilst monitoring fatigue damage in human bone [7].

Reference list:

- [1] Mallick, P.K. (1997). *Composites Engineering Handbook*. New York, Marcel Dekker Inc.
- [2] Herakovich, C.T. (1998). *Mechanics of fibrous composites*. New York, John Wiley & Sons, Inc.
- [3] Schulte, K., Baron, Ch., Neubert, H., Bader, M.G., Boniface, L., Wevers, M., Verpoest, I. and de Charentenay, F.X. (1985). Damage development in carbon fibre epoxy laminates : cyclic loading. In : Proceedings of the MRS-symposium "Advanced Materials for Transport", November 1985, Strassbourg, 8 p.
- [4] Sidoroff, F. and Subagio, B. (1987). Fatigue damage modelling of composite materials from bending tests. In : Matthews, F.L., Buskell, N.C.R., Hodgkinson, J.M. and Morton, J. (eds.). Sixth International Conference on Composite Materials (ICCM-VI) & Second European Conference on Composite Materials (ECCM-II) : Volume 4. Proceedings, 20-24 July 1987, London, UK, Elsevier, pp. 4.32-4.39.
- [5] Brondsted, P., Lilholt, H. and Andersen, S.I. (1997). Fatigue damage prediction by measurements of the stiffness degradation in polymer matrix composites. In : Degallaix, S., Bathias, C. and Fougères, R. (eds.). International Conference on fatigue of composites. Proceedings, 3-5 June 1997, Paris, France, La Société Française de Métallurgie et de Matériaux, pp. 370-377.
- [6] Bandoh, S., Matsumura, K., Zako, M., Shiino, T. and Kurashiki, T. (2001). On the detection of fatigue damage in CFRP by measuring Poisson's ratio. In: Hui, D. (ed.). Eighth International Conference on Composites Engineering (ICCE/8). Proceedings. Tenerife, Spain, 5-11 August 2001, pp. 55-56.
- [7] Pidaparti, R.M. and Vogt, A. (2002). Experimental investigation of Poisson's ratio as a damage parameter for bone fatigue. *Journal of Biomedical Materials Research Part A*, 59 (2): 282-287.

Finally, it is possible that you have to change the citation style from [1], [2], [3],... to (Mallick, 1997), (Herakovich, 1998), (Schulte et al., 1985),... The first citation style is sometimes referred to as the IEEE style, the second one is called the “*Harvard system*” (or BSI standard). More information can be found on <http://en.wikipedia.org/wiki/Citations>. By just changing the citation style in your reference manager, you can automatically update the citations in your text. The result then looks like this:

Damage in fibre-reinforced composites can take many forms (Mallick, 1997; Herakovich, 1998): (i) matrix cracks, (ii) fibre-matrix interface failure, (iii) fibre pull-out, (iv) delaminations, (v) fibre fracture. This damage affects the value of the elastic properties at an early stage. Especially in fatigue, the damage initiation phase can cause a pronounced drop of the elastic modulus of 5 to 10 %. In the next damage propagation phase, the stiffness continues to decrease gradually, ranging from a few percent for unidirectionally reinforced carbon composites to several tens of percents for multidirectional glass laminates (Schulte et al., 1985).

Most one-dimensional damage models for fibre-reinforced composites only account for the effect of damage on the stiffness (Sidoroff & Subagio, 1987; Brondsted et al., 1997). The degradation of the Poisson's ratio is not included in these models. Nevertheless this degradation has been observed and is not negligible (Bandoh et al., 2001; Pidaparti & Vogt, 2002). Bandoh et al. (2001) showed that the Poisson's ratio of a carbon/epoxy UD laminate can drop by 50 % under static tensile loading, while Pidaparti and Vogt (2002) proved that the Poisson's ratio is a very sensitive parameter whilst monitoring fatigue damage in human bone.

Reference list:

- Bandoh, S., Matsumura, K., Zako, M., Shiino, T. and Kurashiki, T. (2001). On the detection of fatigue damage in CFRP by measuring Poisson's ratio. In: Hui, D. (ed.). Eighth International Conference on Composites Engineering (ICCE/8). Proceedings. Tenerife, Spain, 5-11 August 2001, pp. 55-56.
- Brondsted, P., Lilholt, H. and Andersen, S.I. (1997). Fatigue damage prediction by measurements of the stiffness degradation in polymer matrix composites. In: Degallaix, S., Bathias, C. and Fougères, R. (eds.). International Conference on fatigue of composites. Proceedings, 3-5 June 1997, Paris, France, La Société Française de Métallurgie et de Matériaux, pp. 370-377.
- Herakovich, C.T. (1998). Mechanics of fibrous composites. New York, John Wiley & Sons, Inc.
- Mallick, P.K. (1997). Composites Engineering Handbook. New York, Marcel Dekker Inc.
- Pidaparti, R.M. and Vogt, A. (2002). Experimental investigation of Poisson's ratio as a damage parameter for bone fatigue. Journal of Biomedical Materials Research Part A, 59 (2): 282-287.
- Schulte, K., Baron, Ch., Neubert, H., Bader, M.G., Boniface, L., Wevers, M., Verpoest, I. and de Charentenay, F.X. (1985). Damage development in carbon fibre epoxy laminates : cyclic loading. In: Proceedings of the MRS-symposium "Advanced Materials for Transport", November 1985, Strassbourg, 8 p.
- Sidoroff, F. and Subagio, B. (1987). Fatigue damage modelling of composite materials from bending tests. In: Matthews, F.L., Buskell, N.C.R., Hodgkinson, J.M. and Morton, J. (eds.). Sixth International Conference on Composite Materials (ICCM-VI) & Second European Conference on Composite Materials (ECCM-II) : Volume 4. Proceedings, 20-24 July 1987, London, UK, Elsevier, pp. 4.32-4.39.

14.2. REFERENCE MANAGEMENT SOFTWARE

Most universities offer commercial campus software for reference management, such as EndNote and RefWorks.

However, there are a lot of free alternatives that are very good:

- **Zotero**: Probably the most popular open-source reference manager today. Excellent browser integration and PDF annotation.
- **Mendeley**: Very popular in science and medicine. Free basic use; cloud storage limits apply.
- **BibTex**: this format can also be exchanged with LaTeX.

It is strongly recommended to install one of those professional reference management softwares and use them while writing your project proposal, MSc or PhD thesis.

But if you only have a few references or you do not have access to any reference management software, we will explain in the next section how you can generate your own reference list and citation generation in Word. In fact, we will only use components that you already know: field codes, bookmarks and cross-references.

14.3. MANAGING REFERENCES AND CITATIONS WITH ONLY WORD

If you use one of the reference management softwares above, they are handling the citations and reference list for you. But with the knowledge that we built up in the previous chapters, you can build your little own reference manager in Word. Sometimes, this method is handy if you work with many people on the same scientific document and the other colleagues copy the references for their pieces of text in review comments or in plain text, instead of using the same reference manager as you do.

What we want to achieve in this example, is a reference list at the end of the document with the style of Figure 178. The text above the reference list gives an example how we would like to cite these references. It is important to mention that this approach has limitations. You cannot automatically switch between Harvard (Author–Date Citation Style) and IEEE citation style (Numeric Citation Style).

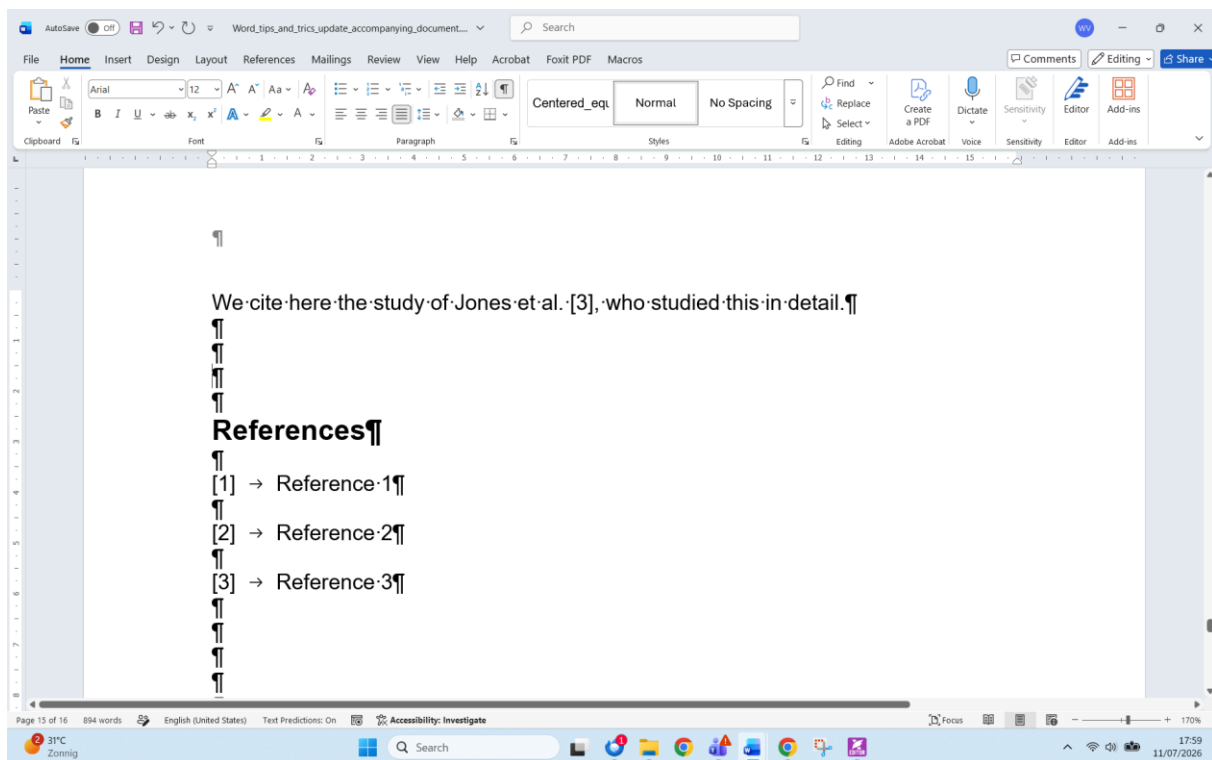


Figure 178 Desired style of the reference list and citations in the text.

With the knowledge from the previous chapters, it is not so difficult to implement your own reference management in Word:

1. Every reference in the reference list gets a number in front of it, like [1], [2], [3], etc. and these numbers are field codes of the type SEQ. In this way, if you insert references in between and you update the fields with <F9>, the numbers are renumbered.

2. The number preceding every reference, gets a bookmark name attached to it. That bookmark name should not be a name like “*Reference_1*”, because the numbering can change. Moreover, bookmark names in Microsoft Word must begin with a letter and can be up to 40 characters long. They can contain letters, numbers, and underscores, but no spaces or special punctuation marks are allowed. So the easiest name for the bookmark is the first author name, then an underscore, then the year of publication, then another underscore, and then the title words filling up till the 40 characters are spent. For example, bookmark names can be “*Jones_2019_Modelling_fatigue_damage_into*” which is exactly 40 characters.
3. When we insert a citation in the text, we insert a cross-reference to the bookmark name.

This procedure works perfectly fine, but there is one major caveat that you need to know, to avoid a lot of frustration with this approach. Initially, I used an invisible table with two columns, one column with the reference numbers and one column with the publication records, just to have a nice alignment of all records. However, if “*Track changes*” is enabled for your document, bookmarks get simply lost when you move table rows around.

1. Word with “*Track changes*” enabled cannot copy bookmark names within a table. If you inserted your references in a two-column table with the left column containing the reference numbers, and the right column the references themselves, and you try to move references around, the bookmark names get lost.
2. Also, even if you do not use a table, but just a numbered list with indented paragraphs, you cannot copy this reference list to another document without losing the bookmark names. The bookmark names are lost when copying the references to another document.

So these are important limitations that I only found out the hard way, because this bug seems nowhere documented on the Internet.

The problem is immediately solved if you stop “*Tracking changes*”, but in collaborative documents, this is not always possible.

Respecting these limitations, you can create a reference list as shown in Figure 179. This reference list was inserted in a Word document for a collaborative project, which was put on a Sharepoint and was edited by many users. The reference list has the following features:

- A paragraph style was created for every reference, with an indented paragraph. The use of a two-column table is avoided, because of the problem with “*Track changes*” that I mentioned above.
- The number of every publication is a field code SEQ.
- Each number has been assigned a bookmark name, as shown in Figure 180. All bookmark names are preceded by the word “*Ref_*”, because I also used bookmark names for equations in that document. The rest of the bookmark name is composed of the author names, year of publication and first part of the title, till the 40 character limit is reached.
- All steps are automated with a macro that inserts a number for the reference and asks for the bookmark name, very similar as our example for the centered and numbered equation in the previous chapters.

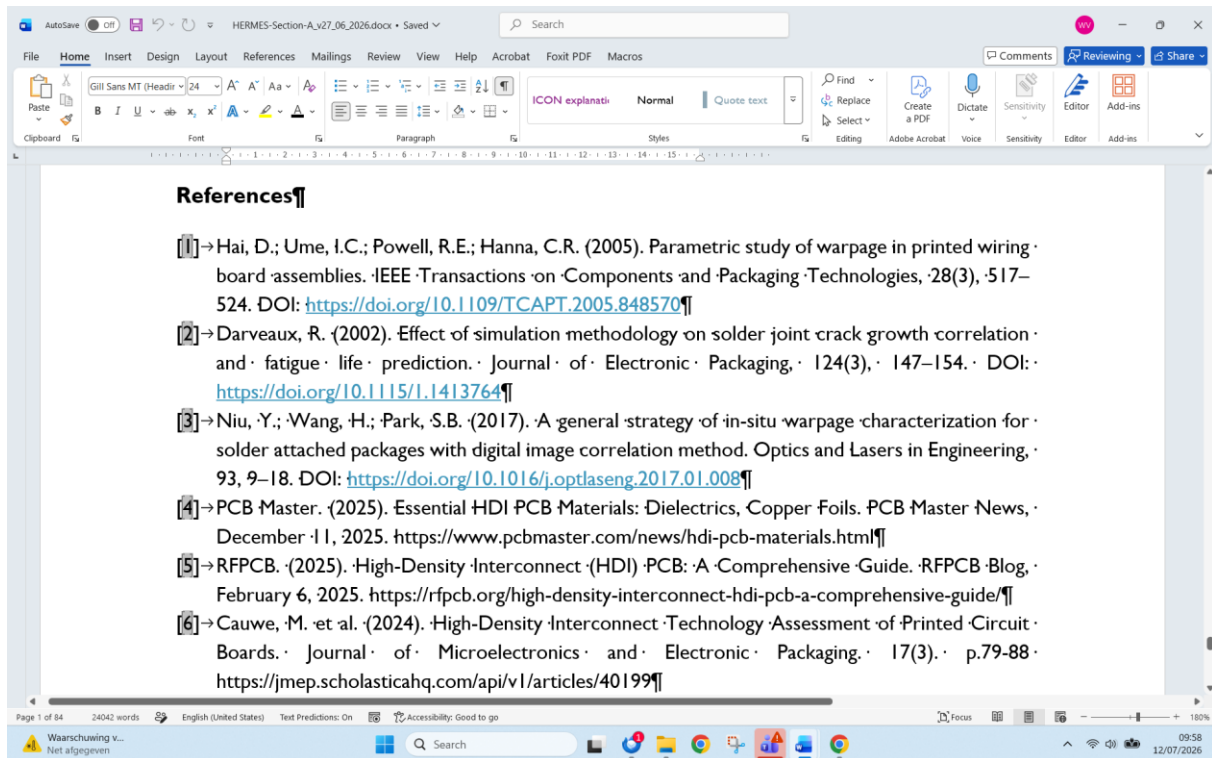


Figure 179 Example of reference list in a Word document for a collaborative project.

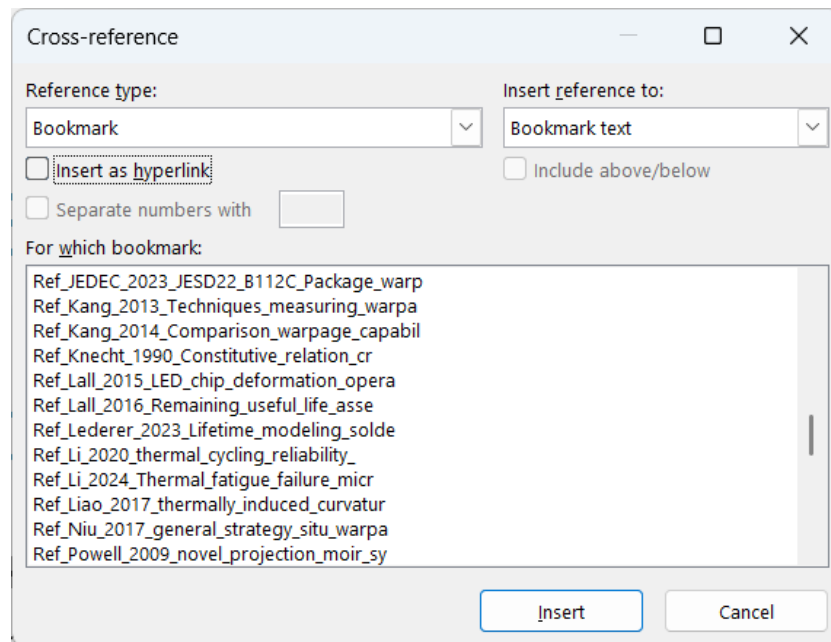


Figure 180 Screenshot of the bookmarks names that are associated with the numbers of the references in the reference list.

In this chapter, we have seen a simple implementation of a reference manager in Word, but I would still strongly recommend to use a dedicated reference manager for managing your literature and references. Softwares like Zotero, Mendeley and EndNote offer so many more functionalities than you can implement with a simple system of field codes and bookmarks.

15

Solving layout issues

This section shortly discusses a few more tips for layout issues that can be helpful when structuring a document.

15.1. ALIGNING LISTS AND PARAGRAPHS

Inserting leading and trailing spaces is still a widely used technique to align lists or paragraphs. This method is quite bad, because as soon as the font size or the number of words changes, the width of the leading or trailing spaces changes.

Therefore the best solution is to use tab stops. A tab stop is a location on the horizontal ruler that indicates how far to indent text or where to begin a column of text. There are five different tab stops:

-  A **Left Tab** stop sets the left end of the text line. As you type, the text fills toward the right.
-  A **Center Tab** stop sets the position at the middle of the text line. As you type, the text centers on this position.
-  A **Right Tab** stop sets the right end of the text line. As you type, the text fills toward the left.
-  A **Decimal Tab** stop aligns numbers around a decimal point. Without regard to the number of digits, the decimal point remains in the same position. (When you use this type of tab stop, you must align the numbers around a decimal character. You cannot use any other type of character — such as a comma, a hyphen, or an ampersand symbol.)
-  A **Bar Tab** places a solid vertical line through selected text.

You can change the type of tab stop on the leftmost side of the horizontal ruler, as shown in Figure 181.

Next you put a tab stop by just clicking a location on the horizontal ruler. This tab stop will be active for all new paragraphs that are created after the current one.

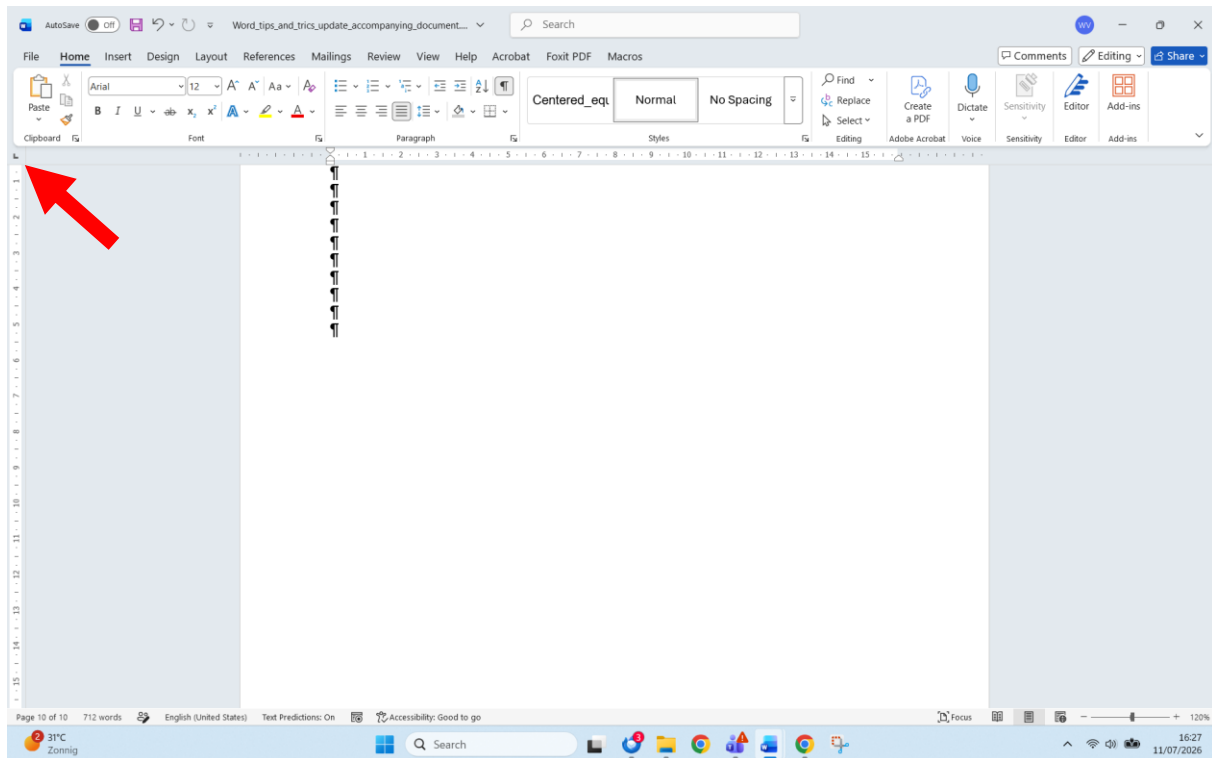


Figure 181 Selecting the type of tab stop (left, centre, right, decimal or bar).

If you want to set a tab stop at a precise location, it is better to use the *Tabs* dialogue box by clicking on the small arrow in the bottom-right corner of the *Paragraph* group in the Home ribbon, as shown in Figure 182, and then selecting the bottom left button “*Tabs...*” in the Paragraph dialogue box.

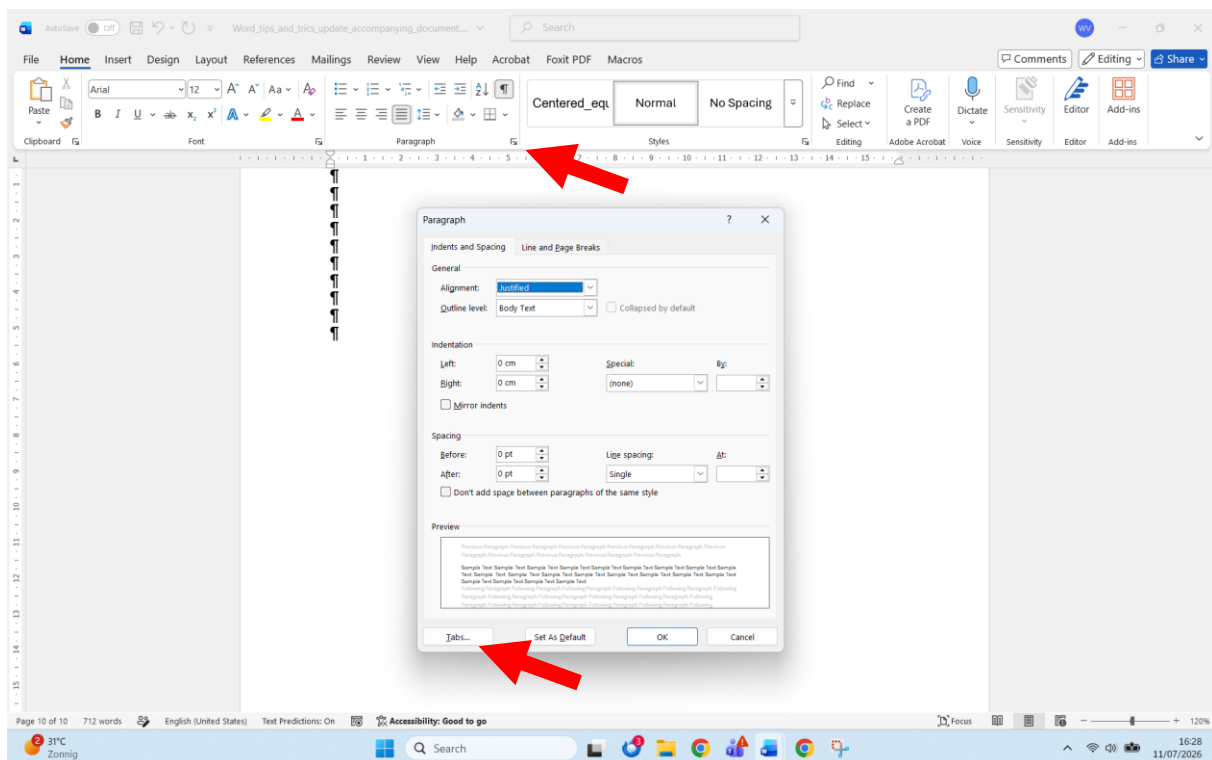


Figure 182 Opening the Tabs window from the Paragraph toolbar.

For example, you can set a tab stop at positions 2, 6 and 10 cm, as shown in Figure 183.

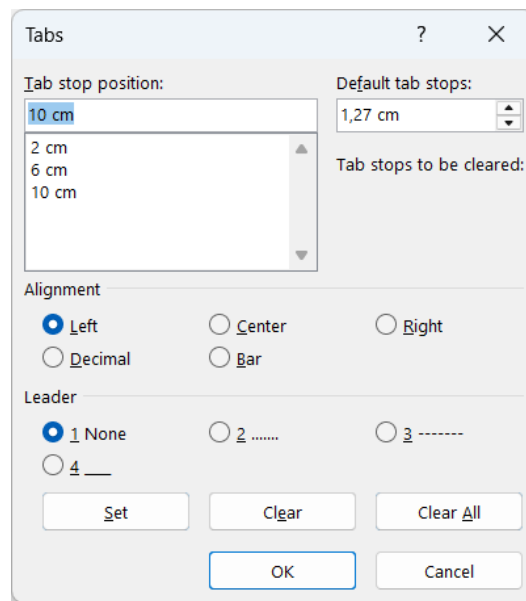


Figure 183 Setting tab stops at precise locations.

When you would now insert a list with the twelve months of the year, you can align them very easily, as shown in Figure 184.

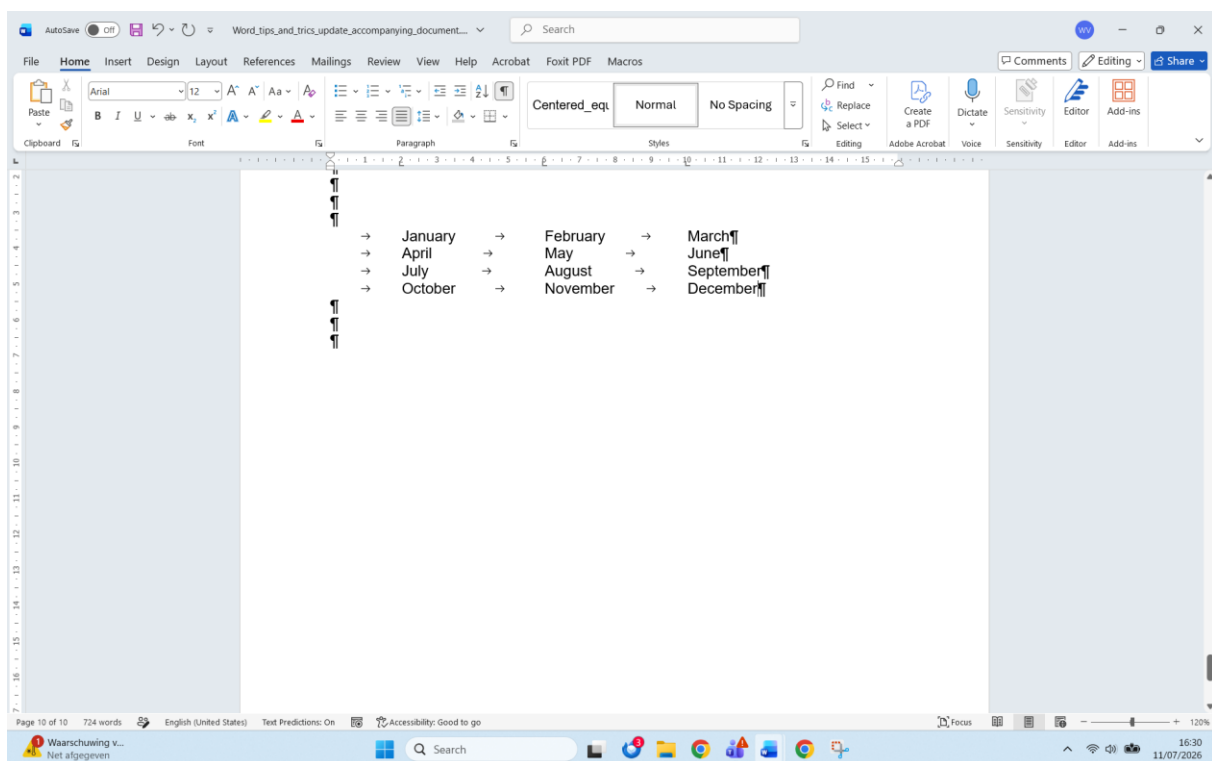


Figure 184 Aligning lists with tab stops.

These tab stops are also very useful for aligning paragraphs. To that purpose, you combine the tab stops with the *First line indent* marker or the *right indent* marker. These markers are the two triangles pointing to each other in the left margin of the horizontal ruler, as indicated in Figure 185.

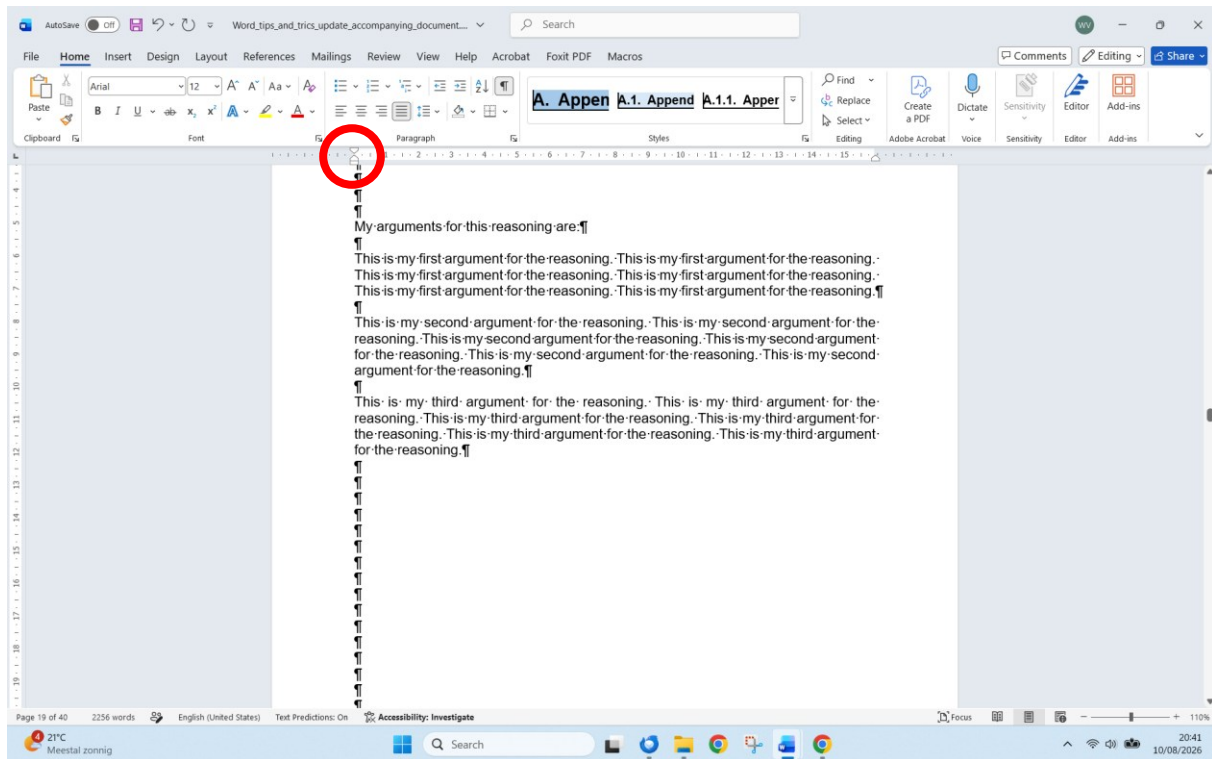


Figure 185 *First line indent* marker and *Right indent* marker on the horizontal ruler.

For example, if you want to list three arguments for your reasoning, you define a tab stop at 1.0 cm and you drag the *Right indent* marker (the lower triangle) until it coincides with the tab stop position. This means that all lines of every next paragraph must be indented to the first tab position. The text from Figure 185 is now formatted as illustrated in Figure 186.

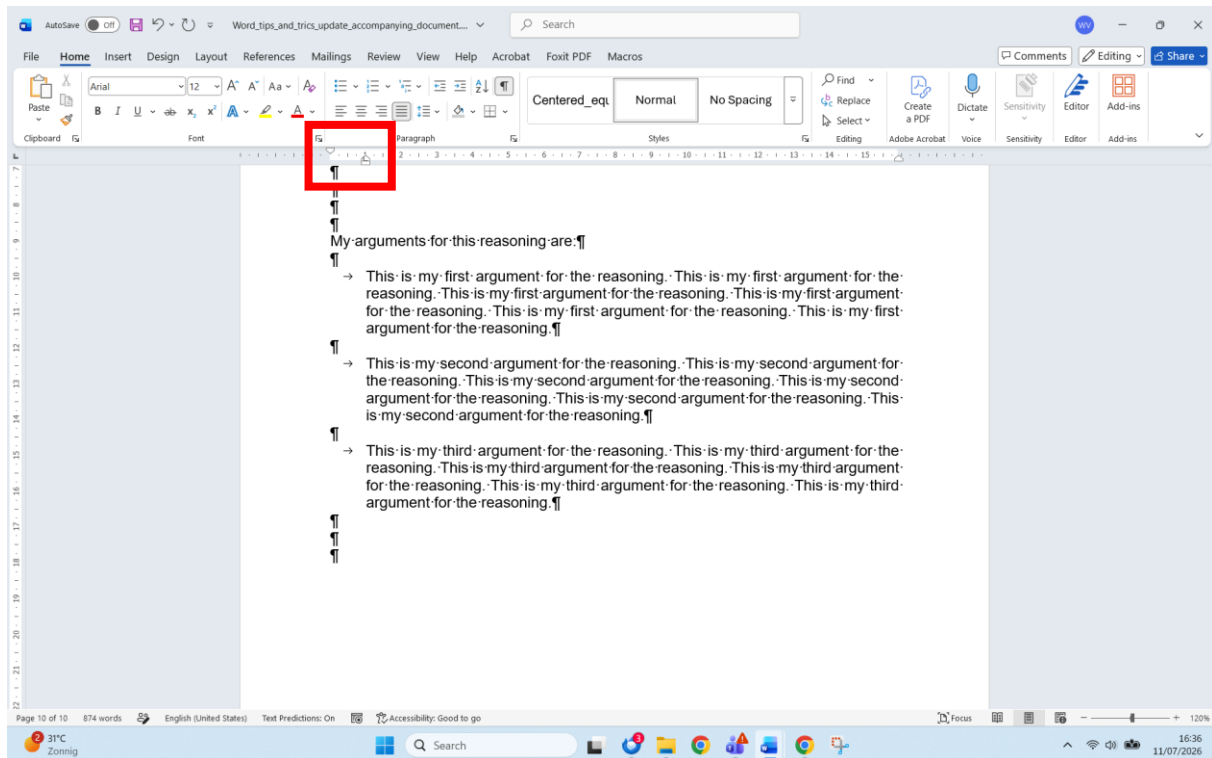


Figure 186 Indenting paragraphs with the *Right indent* marker and a tab stop.

You can use the combination of tab stops and indent markers also to nicely align the captions of different figures, as illustrated in Figure 187. Of course, with what we learned before, it is much more efficient to change the paragraph style *Caption* for all figures together.

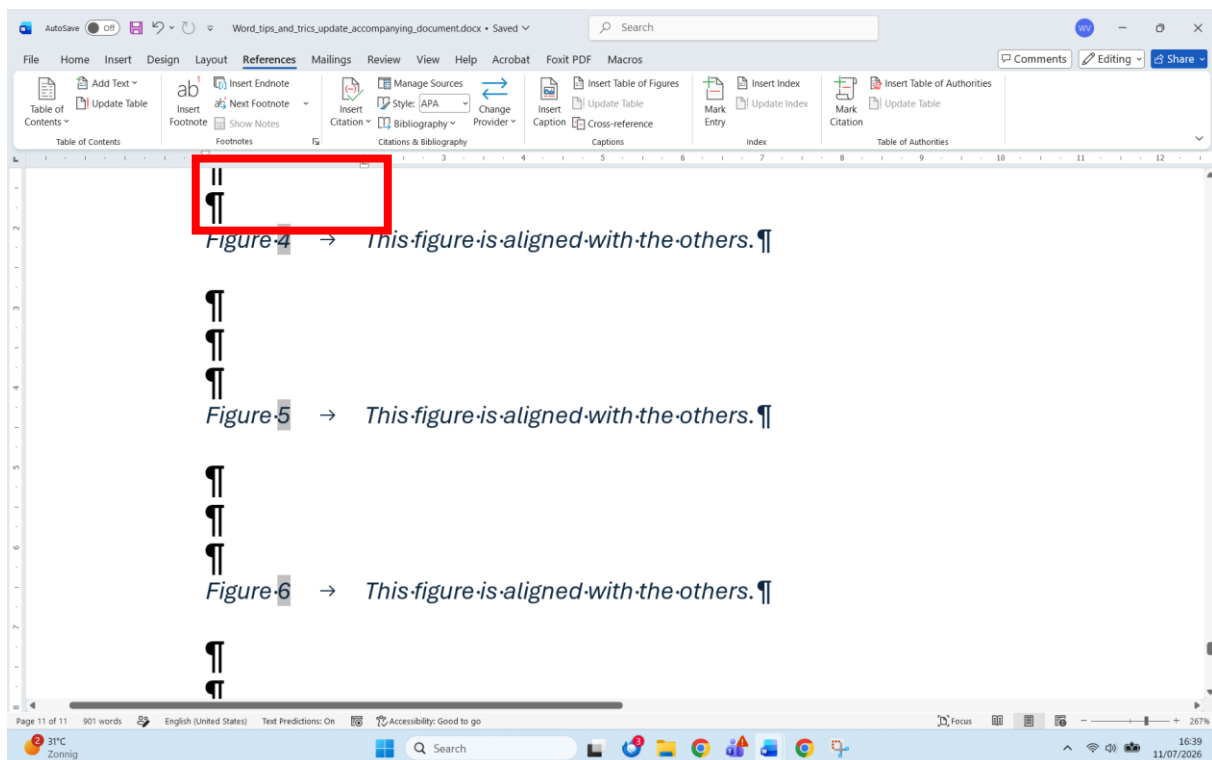


Figure 187 Aligning the captions of different figures with the *Right indent* marker and a tab stop.

15.2. FORMATTING TABULAR DATA

In the previous paragraph, it was already illustrated that you can align a list of data very easily with tab stops. If you have a longer list of tabular data, the best way to format them is using tables. This sounds very logical, but still many people align tabular data with a lot of leading or trailing spaces.

The best way to start, is to insert an empty table that has as many columns as the maximum number of columns that appears in any row, and as many rows as the maximum number of rows that appear in any column. After that, you can use the *Table > Merge cells* command to combine several cells in one row or one column.

An example of a table for a project planning is shown in Figure 188.

It is true that Excel is often better suited to make GANTT charts and more complex tables, but for simple tables, this trick can do.

Task description	Year 1				Year 2				Year 3				Year 4			
Task 1																
1.1																
1.2																
Task 2																
2.1																
2.1																
2.3																
2.4																
Task 3																
3.1																
3.2																
Task 4																
4.1																
4.2																
4.3																

Figure 188 Example of a table for a project planning, using *Merge cells* to combine different cells.

15.3. GROUPING OR SPLITTING WORDS

If you want two words to keep together and not to be broken up on two lines, you can insert a nonbreaking space with <Ctrl>+<Shift>+<Spacebar>.

For example:

Without inserting a nonbreaking space between “item” and “1”, the term “Research item 1” will be broken up just before the number 1.

By inserting now a nonbreaking space between “item” and “1”, the term “Research item 1” will never be broken up just before the number.

This is particularly useful if you use a space to separate the thousands in a large number. The number 3 000 000 (three million) should not be broken up on two different lines.

On the other hand, if you insert a hyphen (-) in a word and this word is at the end of the line, MS Word will break up the word at the position of the hyphen. You can prevent this by inserting a nonbreaking hyphen (<Ctrl>+<Shift>+<->).

For some words in foreign languages, MS Word does not know where to split up the word correctly. Therefore you can insert an optional hyphen (<Ctrl>+<->) where MS Word is allowed to split up the word if necessary.

16

Importance of Word templates

Microsoft Word works with template files. If you create a new file, this file is based by default on the template “*Normal.dotm*”.

The *Normal.dotm* template is stored in the hidden folder *AppData* on Windows. You can easily access it by typing `%appdata%\Microsoft\Templates` in the File Explorer address bar. The exact path is `C:\Users\username\AppData\Roaming\Microsoft\Templates`.

If you are using a Mac, it is located at `/Users/username/Library/Application Support/Microsoft/Office/User Templates/`.

You can open this template file and add/change a lot of things in this template file:

- Define default body and heading fonts, character spacing, and font colours.
- Set custom paper sizes, orientation, gutter margins, and line numbers.
- Map specific Macros, Styles, or Word commands to personalized hotkeys or Ribbons globally.
- Create custom default table styles, cell margins, border colours, and shaded fills so every newly inserted table matches your formatting guidelines.
- If you build custom UI tabs or toolbars, storing them here makes your custom menus available in every document you create.

The macros that we generated in the previous sections, are stored by default in the *Normal.dotm* file and are available to all your documents.

It is wise to generate specific templates for publications, MSc. Thesis, PhD thesis, etc.

In order to do that, you first create a regular document, add all the functionalities you want, and save this as a macro-enabled template file.

Therefore, you choose *File > Save as...*, give it a clear filename and select the file extension “*Word Macro-Enabled Template (*.dotm)*”, as shown in Figure 189.

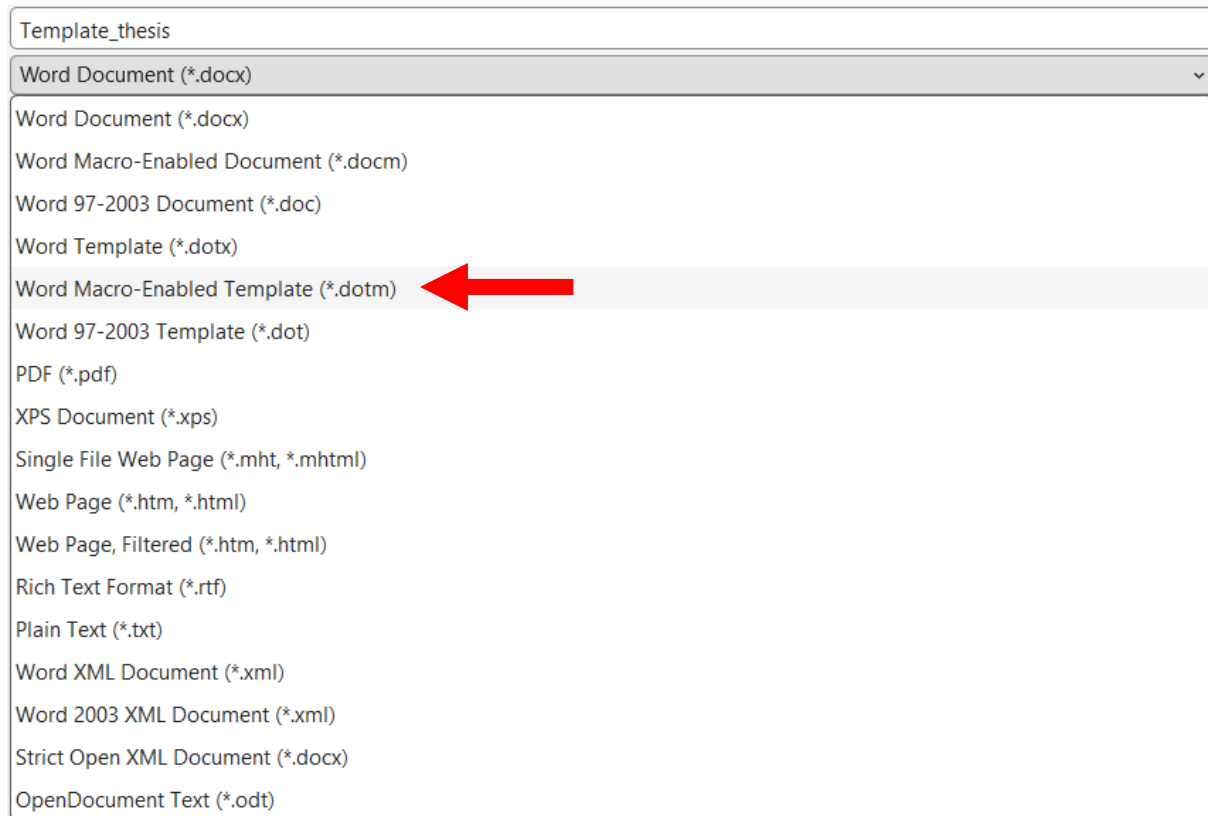


Figure 189 Document format “*Word Macro-Enabled Template (*.dotm)*” in the *File > Save as...* dialogue box.

17

Word Data Structure

This final chapter discusses some more advanced tools for debugging issues with Word. It gives insights in the data structure of Word documents and how object linking and embedding is done.

You will normally rarely need these tools, but when you import old documents into Word 365 or you copy parts of old .doc documents from Word 2003, Word 2007, etc. into a new Word document format, it can be useful to understand these data structures.

The main reason is that integrating parts of old documents into new documents, can suddenly slow down the document saving and field updating tremendously. This is often linked with broken links from embedded objects and badly imported pictures. This can also happen when Excel tables are inserted as embedded objects.

17.1. XML DATA STRUCTURE

The filename of a normal Word document has the extension .docx. In fact, this document hides a collection of XML files and embedded media files. You can see this underlying data structure by executing the following steps:

- Making a copy of your Word document
- Renaming the file extension of that copy from .docx to .zip
- Unzipping that ZIP file into a folder

Now you can see something similar to Figure 190 below. There is an .xml file and several subdirectories.

Name	Date modified	Type	Size
Earlier this week			
docProps	04/08/2026 13:19	File folder	
word	04/08/2026 13:19	File folder	
_rels	04/08/2026 13:19	File folder	
Unspecified			
[Content_Types].xml		Microsoft Edge HT...	9 KB

Figure 190 Directory structure after unzipping of the renamed ZIP file of the Word document.

If you open the subdirectory “word”, the directory structure in Figure 191 becomes visible. The main document is “document.xml”. The subdirectory “media” contains the embedded figures, graphs, screenshots, etc. The subdirectory “embeddings” contains embedded objects.

Name	Date modified	Type	Size
Earlier this week			
embeddings	04/08/2026 13:19	File folder	
theme	04/08/2026 13:19	File folder	
media	04/08/2026 13:19	File folder	
_rels	04/08/2026 13:19	File folder	
Unspecified			
document.xml		Microsoft Edge HT...	1.681 KB
endnotes.xml		Microsoft Edge HT...	4 KB
fontTable.xml		Microsoft Edge HT...	4 KB
footer1.xml		Microsoft Edge HT...	4 KB
footer2.xml		Microsoft Edge HT...	4 KB
footer3.xml		Microsoft Edge HT...	4 KB
footer4.xml		Microsoft Edge HT...	4 KB
footer5.xml		Microsoft Edge HT...	4 KB
footnotes.xml		Microsoft Edge HT...	4 KB
header1.xml		Microsoft Edge HT...	8 KB
header2.xml		Microsoft Edge HT...	13 KB
header3.xml		Microsoft Edge HT...	8 KB
header4.xml		Microsoft Edge HT...	13 KB
header5.xml		Microsoft Edge HT...	14 KB
header6.xml		Microsoft Edge HT...	3 KB
header7.xml		Microsoft Edge HT...	13 KB
header8.xml		Microsoft Edge HT...	14 KB
header9.xml		Microsoft Edge HT...	3 KB
header10.xml		Microsoft Edge HT...	13 KB

Figure 191 Content of the subdirectory “word” of the unzipped data structure of the Word document.

If you open the subdirectory “*embeddings*”, you see the contents in Figure 192. This shows that this particular document has a lot of embedded objects, and one of them is much larger than the others, being “*oleObject27.bin*”.

Name	Date modified	Type	Size
Unspecified			
oleObject27.bin		BIN File	449 KB
oleObject37.bin		BIN File	80 KB
oleObject35.bin		BIN File	70 KB
oleObject31.bin		BIN File	68 KB
oleObject275.bin		BIN File	31 KB
oleObject276.bin		BIN File	19 KB
oleObject251.bin		BIN File	17 KB
oleObject248.bin		BIN File	5 KB
oleObject13.bin		BIN File	4 KB
oleObject25.bin		BIN File	4 KB
oleObject63.bin		BIN File	4 KB
oleObject121.bin		BIN File	4 KB
oleObject136.bin		BIN File	4 KB
oleObject137.bin		BIN File	4 KB
oleObject175.bin		BIN File	4 KB
oleObject176.bin		BIN File	4 KB
oleObject190.bin		BIN File	4 KB
oleObject191.bin		BIN File	4 KB
oleObject200.bin		BIN File	4 KB
oleObject269.bin		BIN File	4 KB
oleObject283.bin		BIN File	4 KB
oleObject30.bin		BIN File	4 KB
oleObject38.bin		BIN File	4 KB
oleObject40.bin		BIN File	4 KB

Figure 192 Content of the subdirectory “*embeddings*” of the unzipped data structure of the Word document.

To get a clue which object this is, you can open the file “*document.xml.rels*” in the subdirectory “*_rels*” (see Figure 193). This file defines all relationships within the document. You can open this .xml file with Notepad or Notepad++ and search for the name “*oleObject27.bin*”.

Name	Date modified	Type	Size
Unspecified			
document.xml.rels		XML Document	92 KB

Figure 193 File “*document.xml.rels*” in the subdirectory “*_rels*”.

In this particular case, it finds this text string:

```
<Relationship Id="rId63"
Type="http://schemas.openxmlformats.org/officeDocument/2006/relationships/oleObject"
Target="embeddings/oleObject27.bin"/>
```

Next, you open the main document “*document.xml*” (see Figure 191) with Notepad or Notepad++. Searching for the text string “*rId63*” returns the following result:

```
<o:OLEObject Type="Embed" ProgID="CorelDRAW.Graphic.9" ShapeID="_x0000_i1051"
DrawAspect="Content" ObjectID="_1847355878" r:id="rId63"/>
```

So this object with ID = rId63 is an embedded CorelDRAW.Graphic.9 object, which is a very old vector graphics object. Word 365 cannot natively render or process CorelDRAW 9 vector graphics. Every single time you save, Word has to call a legacy Object Linking and Embedding (OLE) Windows subsystem to pull cached vector metadata from that 450 KB binary file. Because the software is 25+ years old, the process hangs and slows your entire save cycle to a crawl.

Once this OLEObject text string is located in the “*document.xml*” file, look directly before or after that XML tag in the text. Scroll left and right to read the nearby plain text. You will see words, sentences, or section headers that are right next to the broken object. If you see which text is close to this object in the “*document.xml*” file, you can search for that text in the original .docx document and then find the object.

You can then cut the object and paste it again as a completely flat picture (JPEG or PNG). This completely eliminates the ancient CorelDRAW dependency.

However, as indicated in Figure 191, there are still 290 other *oleObject.bin* files which is far too much. When Word saves a file, it does not just write a single text block. Because it contains 290 embedded OLE elements, Word has to pause and execute a heavy binary serialization loop—re-validating, packaging, and zipping nearly 300 discrete legacy object streams into the archive container single file.

Using the same method as before, we can discover that these 290 objects can be traced back to old equations that have been inserted with the Microsoft Equation Editor 3.0.

As explained before, this Equation Editor was popular in Word 2007 and later releases, but got obsolete because of security holes. Microsoft removed Equation Editor 3.0 in January 2018 due to severe security vulnerabilities that left users open to remote code execution. Because it was originally a legacy, third-party component licensed from Design Science, Microsoft chose to completely retire it rather than rewrite the code.

Modern versions of Word (365) natively use Office Math (MathML XML nodes inside *document.xml*). However, this file in Figure 191 is packed with *oleObject.bin* structures. These indicate equations written via the legacy Microsoft Equation Editor 3.0 (or an older version of MathType).

Every single time that this document is saved, Word 365 cannot process these equations natively. It has to activate a background OLE wrapper subsystem to calculate the spatial bounding and binary structure for all 290 elements. If even a handful of those OLE structures have broken links or incomplete data streams, Word hangs waiting for internal loop timeouts before completing the save.

There are a few options to handle these very old equations:

- Double-clicking them, will pop-up a dialogue box that asks if you want to convert these equations to modern Office Math objects. This conversion goes quite well for most equations.
- Sometimes, if the link to the original OLE object is broken, Word does not manage to convert the equation and it replaces it by a picture. However, this picture is actually still a field code, which can be proven by pressing <Alt>+<F9>. You will then see the field code { EMBED Equation.3 }.
- The best solution in that case is either retyping the equation in the new Microsoft Equation Editor, or if the equation will not change anymore, decouple its field code by pressing <Ctrl>+<Shift>+<F9>. The equation becomes then a regular picture and the embedded object is gone.

Cleaning up these 291 objects resulted in a reduction of saving time of the .docx Word document from 40 seconds to 2 seconds.

17.2. SELECTION PANE

A less invasive way to find out if there are embedded objects in your document, is to activate the selection pane. To open that pane, click in the Home Ribbon on the drop down button “*Select*” and choose the bottom option “*Selection Pane...*”, as shown in Figure 194.

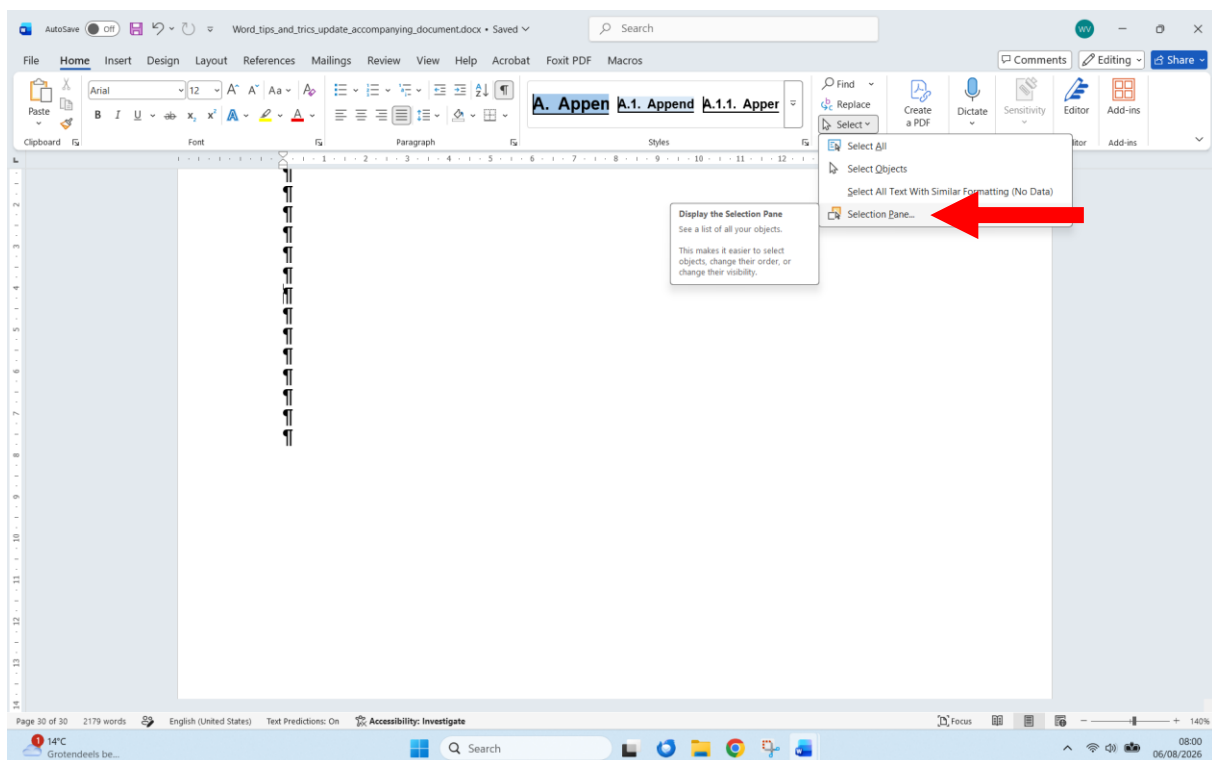


Figure 194 Activating the Selection Pane via the Home Ribbon and the *Select* dropdown menu.

The Selection Pane appears at the right side of the screen, as shown in Figure 195. It shows all objects, pictures, shapes, etc. on the current page. As you can see in this example, there are a lot of embedded objects on this single page. In fact, all grey shaded symbols (like the coordinate axis vectors in the text) are old embedded Microsoft Equation Editor 3.0 objects. These slow down the saving of the document a lot.

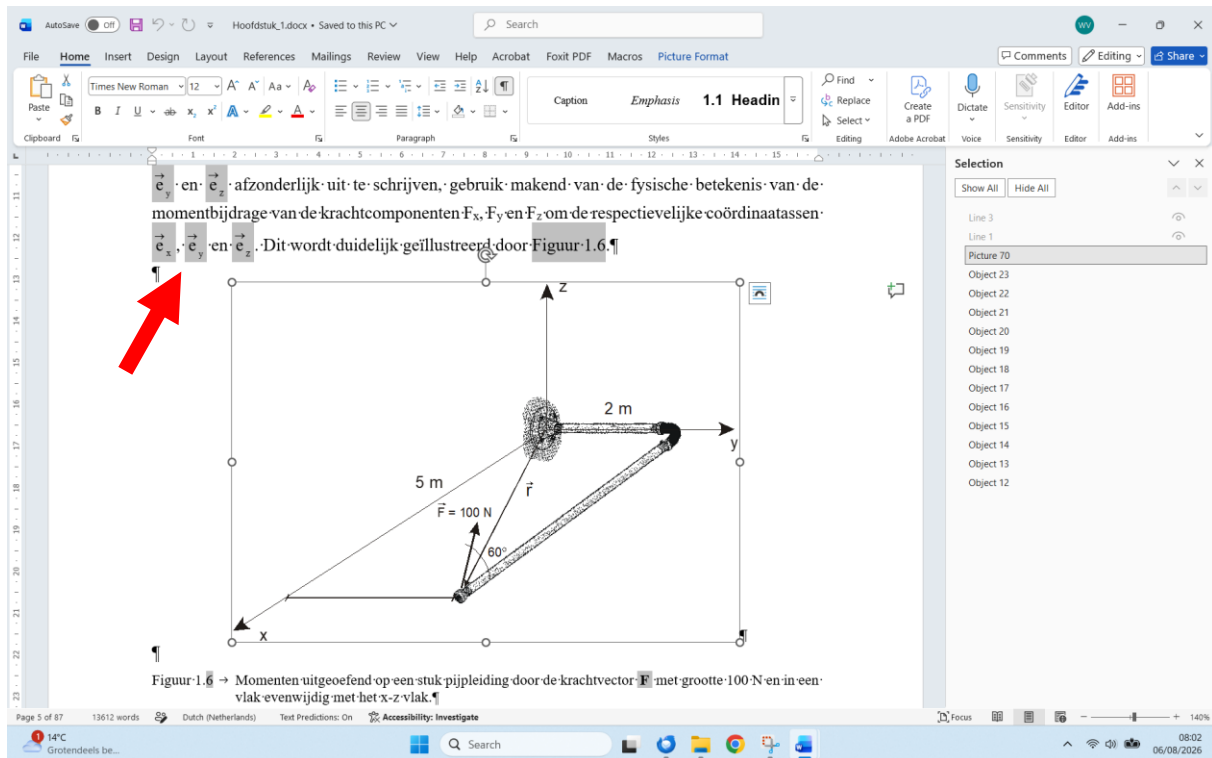


Figure 195 Embedded objects from Microsoft Equation Editor 3.0 in a page of a document.

A very fast way to scan your document for such objects is putting your cursor on the first page of your document with the Selection Pane active on the right side of the screen. You keep pushing the <Arrow down> key all the time, while the pages fly by. Every time your cursor moves to a next page, the Selection Pane is updated with all objects, pictures, shapes, etc. on that page. In this way, you can quickly screen your document for unwanted objects. The best way is always to cut them to the Clipboard and paste them again as a flat figure.

17.3. REVEAL FORMATTING

A last tool which can be handy in debugging, is the “*Reveal Formatting*” pane. This is simply activated by the key combination <Shift>+<F1>. It shows for any place where your cursor is, what the font formatting and paragraph formatting are, as shown in Figure 196.

This can be handy to see hidden formatting marks or subtle differences in paragraph styles.

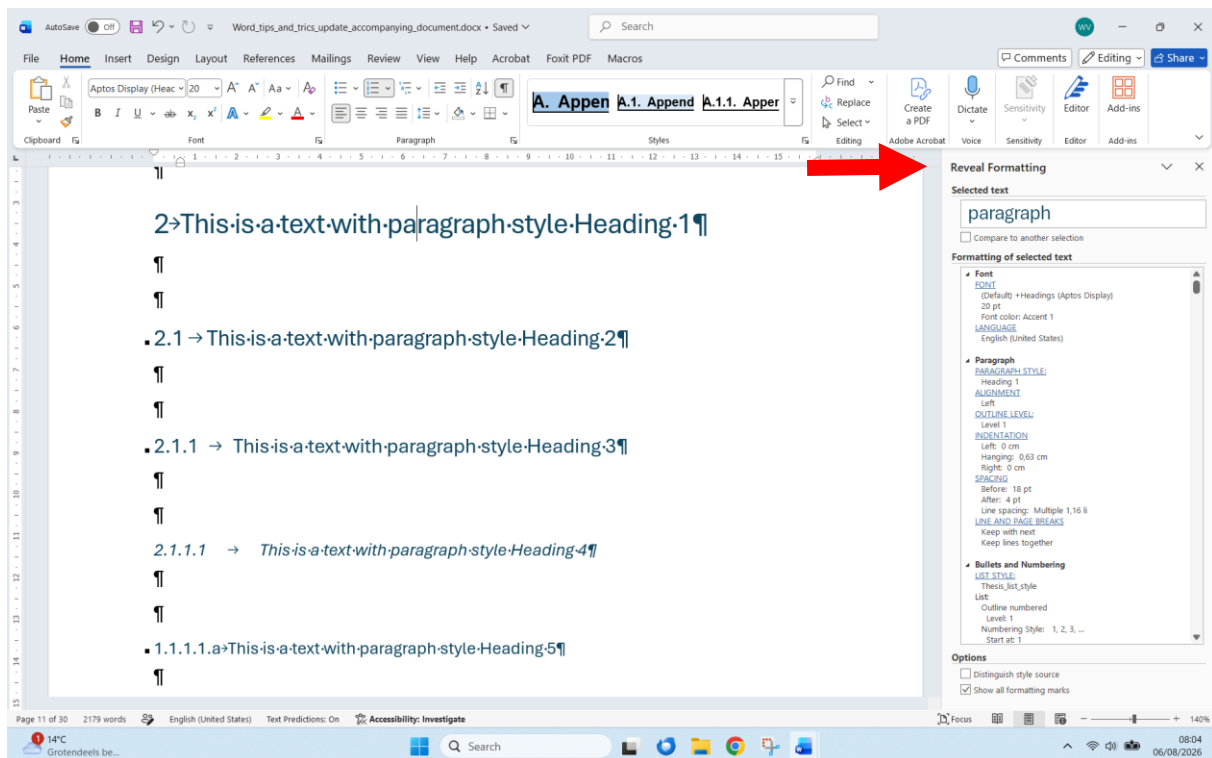


Figure 196 Activating the “Reveal formatting” pane at the right side of the document window with **<Shift>+<F1>**.

17.4. SAVE AS PDF

When you save a document in PDF, the standard format is PDF. But there exists also the PDF/A format, where A stands for “Archival”.

The Adobe website explains it as follows:

PDF/A is the standard format used around the world for archiving electronic documents. It allows documents to be viewed in their original appearance across different software. A PDF/A file differs from a standard PDF file in that it forbids features that aren't suitable for long-term archiving.

It is also self-contained, with all of the elements such as content, fonts and colour information embedded within the file. This means the document can be viewed and reproduced exactly as it was originally created, regardless of the device being used.

So if you want all your fonts and colour information to be embedded in the file and ensure maximum portability to other softwares, the PDF/A format is a safe choice.

To make a good export of your document to PDF/A format, you go as follows.

First, it is a good practice to give a title to your document. This title is shown in Adobe Acrobat Readers in the header of the file. The title can be filled in in the menu *File > Info*, as shown in Figure 197. You click on the empty space next to the label “Title” and you fill in the title.

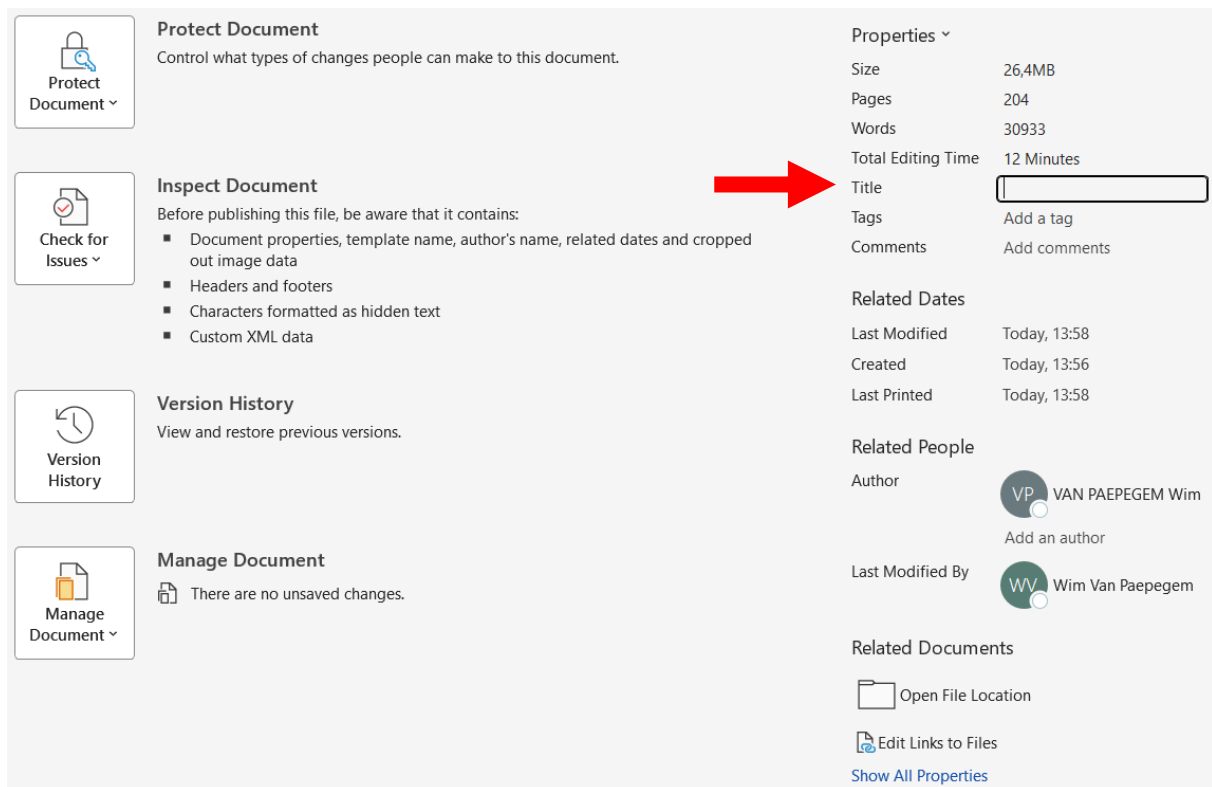


Figure 197 Setting a title for your document in the *File > Info* dialogue box.

Next, you activate the embedding of all used fonts in the document. To this purpose, you click *File > Options...* to open the *Options* dialogue box. Select the *Save* tab and the options in Figure 198 appear. Check the option *Embed fonts in the file*.

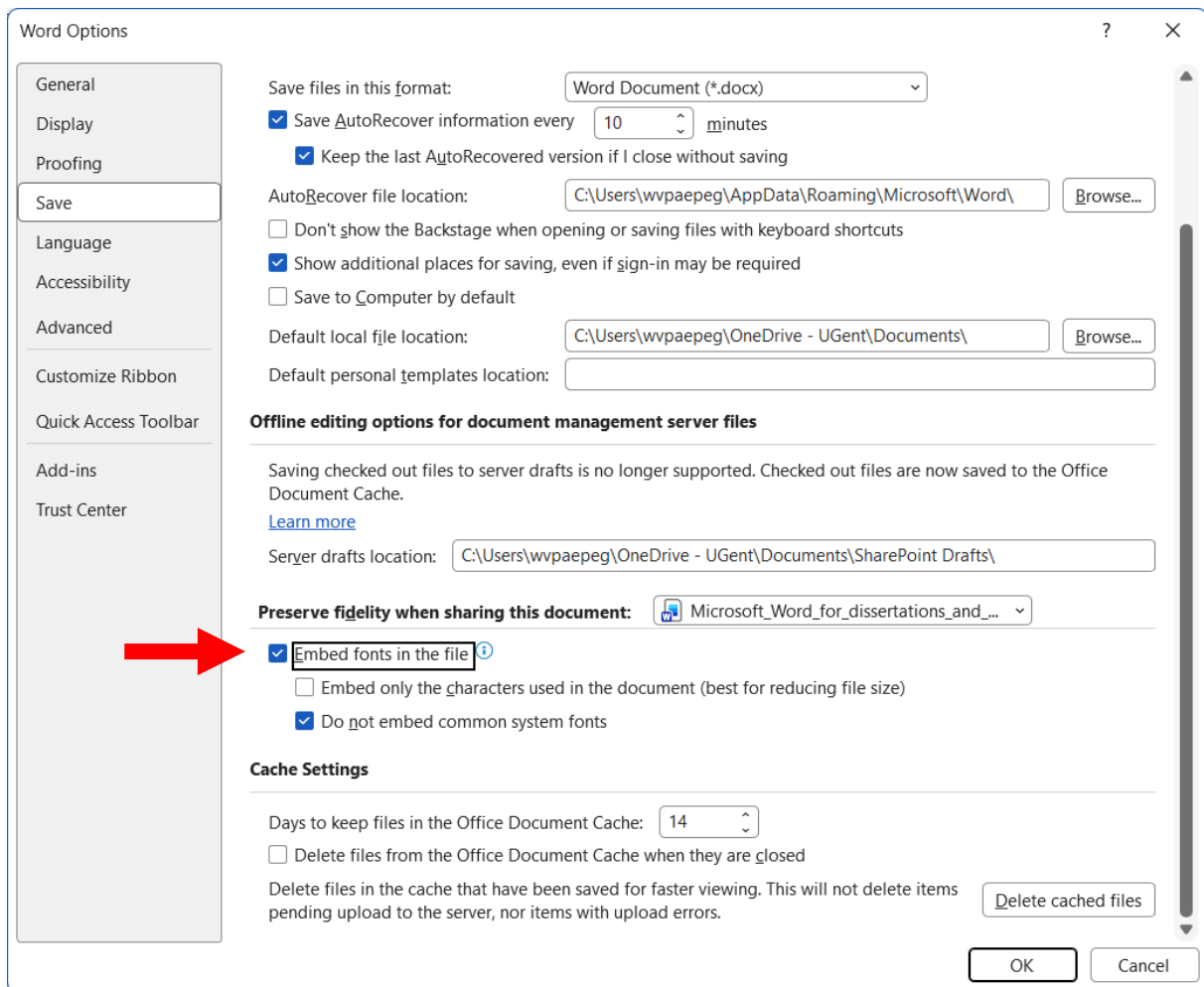


Figure 198 Embedding the used fonts in the .docx file.

Then you are ready to save your document in PDF/A format. Click *File > Save as...* to open the *Save as* window. Enter the filename and select the file type PDF (*.pdf). Select the button “*More options...*”, as shown in Figure 199.

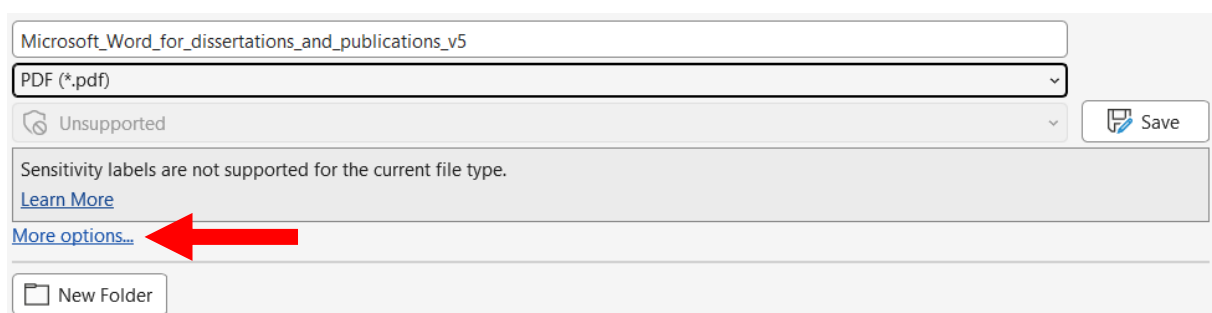


Figure 199 Choosing “*More options...*” in the *File > Save as...* window.

In the *Save as* dialogue box, you select again the button “*Options...*”, as shown in Figure 200.

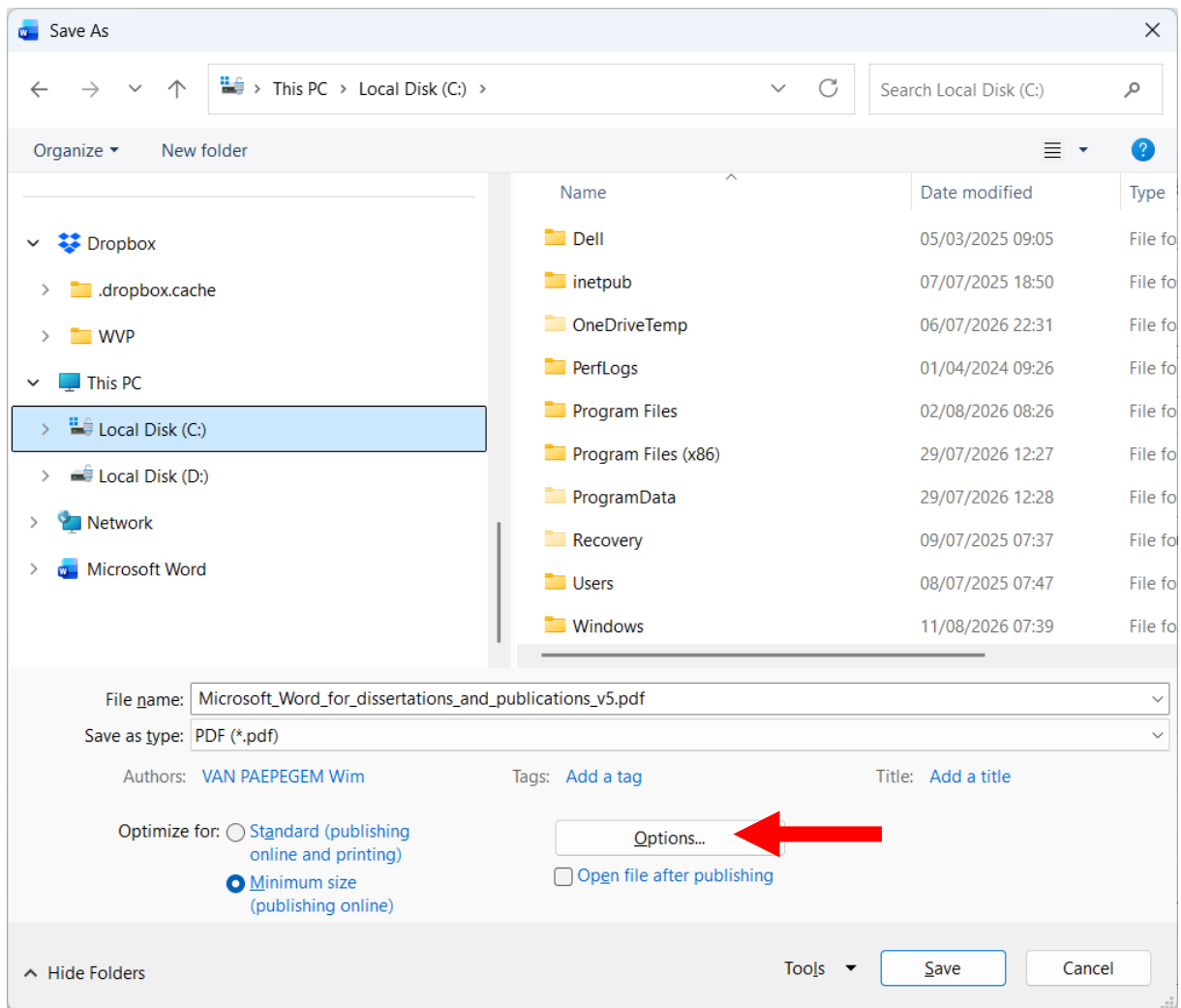


Figure 200 Choosing “Options...” in the dialogue box “More options...” of saving as PDF file.

Now, the dialogue box in Figure 201 appears. There, you can select the option *PDF/A compliant*. If you now save your .docx file as a PDF file, it can be opened on any platform and it will still show the correct fonts and colours, as well as the correct title of the document. Figure 202 shows how the title is correctly shown in the window top bar and in the document tab, and the PDF editor also shows a pop-up that this is a PDF/A compliant document.

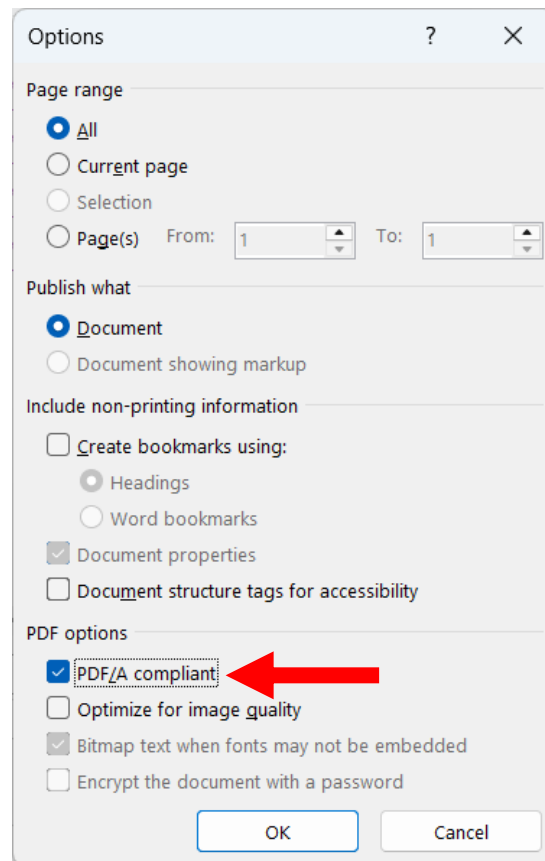


Figure 201 Checking the option for “PDF/A compliant” saving.

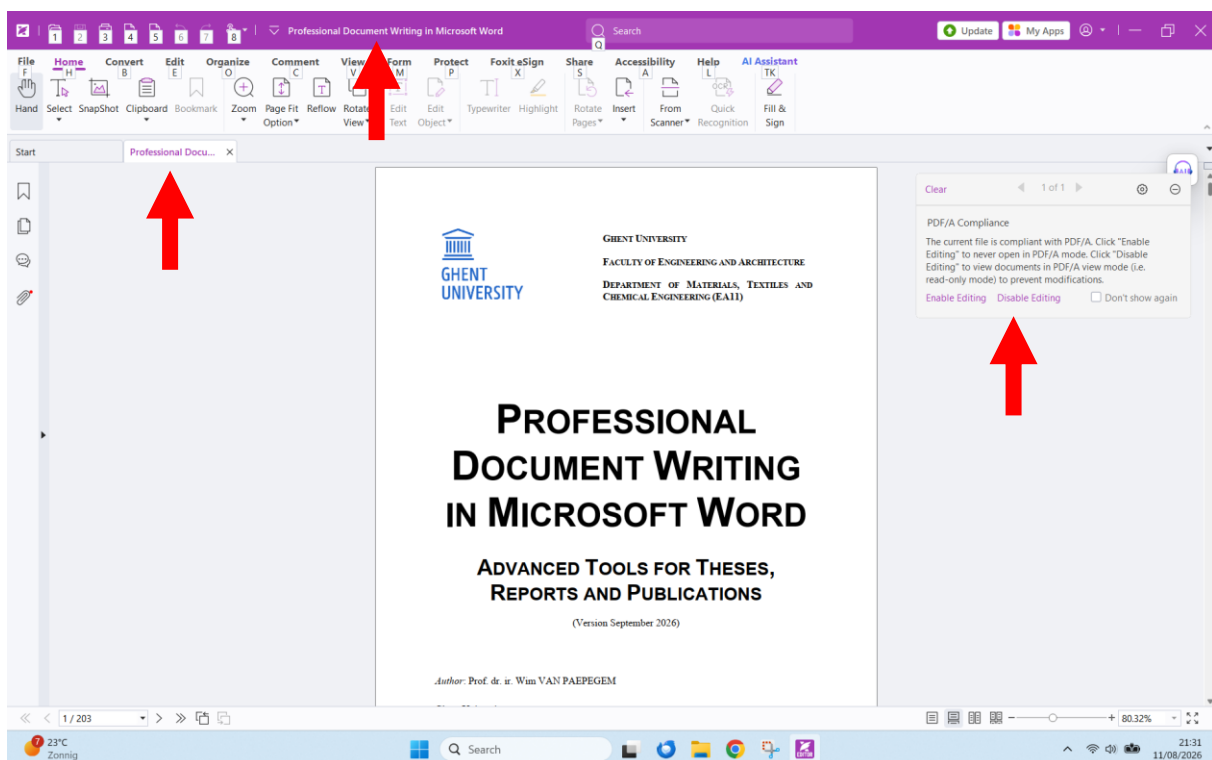


Figure 202 Opening of the PDF document in a PDF editor (title and PDF/A format are recognized).

18

Conclusions

Congratulations, you made it till the end of this document !

If you want to start implementing some of these Word features, I would recommend to start practicing in a dummy document, where you can mess up any text or style. Many of the screenshots in this tutorial also come from a dummy document where I tried out all features.

Once you feel confident with the features that you like, it is absolutely worth the effort to define your own template document that contains some essential definitions:

- The page layout and margins,
- The default font and line spacing you want to use,
- The paragraph styles that you want to apply for the different heading levels (unless you keep using the built-in *Heading* styles or you modify those),
- Create the Multi-Level list to define the hierarchy and numbering of your heading levels. This is one of the most useful features to include.
- If you like to use macros, create a new toolbar in the top menu and start with a few macros that are simple but effective. You can also just copy the Visual Basic source code of the macros that I provided in Chapter 11 and 12.
- For a more complex document template, already create the different sections for the front matter and chapters, insert section breaks, define page numbering and unlink the section headers from each other. If you use your own paragraph styles for the different heading levels, insert at the start of each chapter an invisible built-in *Heading 1* paragraph, so that numbering of figures and equations per chapter is still possible.
- Save this template into a *.dotm* template document, so that every document that you create from this template, has already all these features implemented.

Further, even if you are not a skilled programmer, the advent of AI assisted coding can help you a lot to automate Word tasks into macros, or get help on advanced topics in Word. Most macros in Word are only a few tens to a few hundreds lines of Visual Basic code, and can easily be generated with a free version of Gemini, Copilot or ChatGPT. If you record the basic version

of your macro with a sequence of recorded events in Word, you can ask the AI assistant to add additional functionalities or further tune it to your wishes.

Finally, you can send any suggestions/criticisms/comments on this tutorial. If you have any tips or tricks yourself, or you have a beautiful style template or very useful macros that you have programmed, do not hesitate to share your input by e-mailing to wim.vanpaepegem@ugent.be. I will include those in future updates of the document.