In this document, I learn more about formating in Typst.

These first two paragraphs are writen before any rule is set, so they are simply using the default formating.

## Setting rules

This paragraph, is justified. That's because I used the "par" function (called using "#par"), with the argument "justify" set to "true", and surrounded this paragraph by this function call. It is inconvenient to do it this way; indeed, if I wanted to have my whole document justified, I would need to surround it entirely with many functions.

If I write another paragraph, like this one, out of the "par" function, then it returns to default formating, where paragraphs are not justified but left aligned. To justify all paragraphs in my document, I can set a rule about paragraphs.

In order to do so, I use the "set" keyword, called like a function using "#set". Let's set a rule to justify all the paragraphs in this document.

Now that this rule is set, I can write freely, because when Typst compiler will call the "par" function, the "justify" argument will always be set to "true" by default. You can clearly see that this paragraph is justified even though it is not surrounded by anything. Pretty intuitive! Setting several rules at the top of the document should allow me to dictate how this document should be formated and it makes changing the formating super easy.

## Setting up the page

There are many rules that can be set, mainly for the following functions:

- text
- page
- par
- heading
- document

All of them are pretty self-explanatory, except for "document", which allows me to set metadata in the compiled pdf.

Let's set some rules, following the example in the Typst tutorial about formatting [1].

Would you look at that! A totally new format for the text, the page and the paragraphs! All of this in the same document.

## Section title

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore.

Lorem Ipsum [2] is very convenient for testing formatting and the Typst function “lorem” can quickly generate sample text. These formated paragraphs are justified and use 52% font height for separation.

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magnam aliquam quaerat voluptatem. Ut enim aeque doleamus.

Notice how the rule we set change the page format (it’s now using A6 format), and the margin size. All of this without importing a single package, and by using only a few lines of intuitive code!

We can also set attributes about headers format, such as their numbering. Headers are currently unnumbered, which is uncommon and not really practical for any serious work. I'll set a rule to number headers, using arabic numbers for the first level header and letters for second level headers.

# 1 First level section title

See!

Note that the numbering starts at one even though we already defined other sections, because we only set the numbering rule before this sections.

## 1.a Second level section title

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magnam aliquam quaerat.

## 1.b Another second level section title

What will happen if I use a level three header? My rule does not set how it should be numbered...

### 1.b.a Third level section title

Looks like it's using the second level header numbering rule.

### 1.b.b Another third level section title

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magnam aliquam quaerat.

### 1.b.c Final third level section title

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magnam aliquam quaerat.

Figure 1: An angry looking cat

Sorry, I could not resist the urge to put a cat picture here... At least I can check that I remember how to create figures with captions and include images, something we learned in the previous chapter.

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magnam aliquam quaerat voluptatem. Ut enim aeque doleamus animo, cum corpore dolemus, fieri.

# 2 Show rule

The show rule is unique since it allows me to set a way to display things. For instance, if I wanted to display “bonjour” as an *italicized* word each time I use it, instead of surrounding it with underscores each time, I could use the show rule in the following way.

Let’s test it out: *bonjour*! Seems like it works perfectly fine, I can now type *bonjour* without worying about it’s format. Even cooler, I can very easily change how *bonjour* is displayed, simply by changing the rule. Any change to this show rule will affect anything that comes after.

The example given in the documentation [1] is pretty cool because it demonstrates how this rule can be used to always display a logo next to a specific name.

# Bibliography

[1] Typst, “Tutorial - Formating – Typst Documentation.” 2024.

[2] Lipsum, “Lorem Ipsum - All the facts.” 2024.