

The Beauty of Sharing Knowledge

Max Mustermann

‘Ilm (Urdu: ﴿) is the Urdu term for knowledge. In its general usage, ‘ilm may refer to knowledge of any specific thing or any form of “learning”. Subsequently, the term came to be used to refer to various categories of “sciences”, especially when used in its plural form (‘ulum).

March 19, 2024

Thank you for using this template ❤️,
I hope you like it 😊

Contents

1. Layout	4
1.1. Cover	4
1.1.1. Multiple authors	4
1.1.2. Date format	4
1.1.3. Customizing the cover page	5
1.2. Preface	5
1.3. Table of Contents	6
1.4. Body	6
1.5. Appendices	6
1.6. Bibliography	7
1.7. Indices	7
1.8. Footer	7
1.8.1. Customizing page numbering	8
2. Text	9
2.1. External links	9
2.2. Blockquotes	9
2.3. Small- and all caps	9
2.4. Tables	10
3. Code	11
3.1. Custom font and size	11
Bibliography	12
Index of Tables	13
Index of Listings	13

1. Layout

The template uses `A4` as its page size, you can specify a different paper size string^o using:

```
#show: ilm.with(  
    paper-size: "us-letter",  
)
```

‘Ilm display’s its content in the following order:

1. Cover page (unless disabled)
2. Preface page (if defined)
3. Table of contents (unless disabled)
4. Body (your main content)
5. Appendix (if defined)
6. Bibliography (if defined)
7. Indices (if enabled) — index of figures (images), tables, or listings (code blocks)

1.1. Cover

By default, ‘Ilm generates a cover/title page with a title, author(s), date, and abstract:

```
#show: ilm.with(  
    title: [Your Title],  
    authors: "Author Name",  
    date: datetime(year: 2024, month: 03, day: 19),  
    abstract: [Your content goes here],  
)
```

Only the `title` and `authors` fields are necessary; `date` and `abstract` are optional.

1.1.1. Multiple authors

You can specify multiple authors by providing an array. Authors will be displayed on separate lines on the cover page, with font size automatically adjusted based on the number of authors:

```
#show: ilm.with(  
    title: [Your Title],  
    authors: ("John Doe", "Jane Smith", "Max Mustermann"),  
)
```

The `authors` parameter accepts either a string (single author) or an array of strings (multiple authors).

1.1.2. Date format

By default, the date is shown in the format: `MMMM DD, YYYY`. You can change the date format by specifying a different format string:

```
#show: ilm.with(  
    date-format: "[month repr:long] [day padding:zero], [year repr:full]",  
)
```

See Typst’s official documentation^o for more info on how date format strings are defined.

1.1.3. Customizing the cover page

You have full control over the cover page behavior:

No cover page:

```
#show: ilm.with(  
    cover-page: none,  
)
```

Custom cover page:

```
#show: ilm.with(  
    cover-page: [  
        #align(center + horizon)[  
            #text(4em)[*My Custom Title*]  
            #v(2em)  
            #text(2em)[Subtitle]  
            #v(1em)  
            #text(1.2em)[Author Name]  
        ]  
    ],  
)
```

When you provide custom content, ‘ilm will automatically wrap it in a `page()` for you. The `title`, `authors`, `date`, and `abstract` parameters are still used for document metadata even when a custom cover page is provided.

 Tip: if your custom cover page is complex, define it in a separate file and import it:

```
#show: ilm.with(  
    cover-page: [#include "custom-cover.typ"],  
)
```

1.2. Preface

The preface content is shown on its own separate page after the cover page.

You can define it using:

```
#show: ilm.with(  
    preface: [  
        = Preface Heading  
        Your content goes here.  
    ],  
)
```

 Tip: if your preface is quite long then you can define it in a separate file and import it in the template definition like so:

```
#show: ilm.with(  
    // Assuming your file is called `preface.typ` and is  
    // located in the same directory as your main Typst file.  
    preface: [#include "preface.typ"],  
)
```

1.3. Table of Contents

By default, ‘ilm’ display a table of contents before the body (your main content). You can disable this behavior using:

```
#show: ilm.with(  
  table-of-contents: none,  
)
```

The `table-of-contents` option accepts the result of a call to the `outline()` function, so if you want to customize the behavior of table of contents then you can specify a custom `outline()` function:

```
#show: ilm.with(  
  table-of-contents: outline(title: "custom title"),  
)
```

See Typst’s official documentation^o for more information.

1.4. Body

By default, the template will insert a pagebreak^o before each chapter, i.e. first-level heading. You can disable this behavior using:

```
#show: ilm.with(  
  chapter-pagebreak: false,  
)
```

1.5. Appendices

The template can display different appendix, if you enable and define it:

```
#show: ilm.with(  
  appendix: (  
    enabled: true,  
    title: "Appendix", // optional  
    heading-numbering-format: "A.1.1.", // optional  
    body: [  
      = First Appendix  
      = Second Appendix  
    ],  
  ),  
)
```

The `title` and `heading-numbering-format` options can be omitted as they are optional and will default to predefined values.

 Tip: if your appendix is quite long then you can define it in a separate file and import it in the template definition like so:

```
#show: ilm.with(  
  appendix: (  
    enabled: true,  
    // Assuming your file is called `appendix.typ` and is  
    // located in the same directory as your main Typst file.  
    body: [#include "appendix.typ"],
```

```
 ),  
 )
```

1.6. Bibliography

If your document contains references and you want to display a bibliography/reference listing at the end of the document but before the indices then you can do so by defining `bibliography` option:

```
#show: ilm.with(  
  // Assuming your file is called `refs.bib` and is  
  // located in the same directory as your main Typst file.  
  bibliography: bibliography("refs.bib"),  
)
```

The `bibliography` option accepts the result of a call to the `bibliography()` function, so if you want to customize the behavior of table of contents then you can do so by customizing the `bibliography()` function that you specify here. See Typst's official documentation⁹ for more information.

1.7. Indices

The template also displays an index of figures (images), tables, and listings (code blocks) at the end of the document, if you enable them:

```
#show: ilm.with(  
  figure-index: (  
    enabled: true,  
    title: "Index of Figures" // optional  
,  
  table-index: (  
    enabled: true,  
    title: "Index of Tables" // optional  
,  
  listing-index: (  
    enabled: true,  
    title: "Index of Listings" // optional  
,  
)
```

The `title` option can be omitted as it is optional and will default to predefined values.

1.8. Footer

By default, 'Ilm displays page numbers in the footer with alternating alignment (left on even pages, right on odd pages). If a page does not begin with a chapter, the chapter's name is shown alongside the page number.

Look at the page numbering for the current page. It should show “LAYOUT” next to the page number because the current subheading *Footer* is part of the *Layout* chapter.

When we say chapter, we mean the the first-level or top-level heading which is defined using a single equals sign (=).

1.8.1. Customizing page numbering

You can customize the page numbering style in footer or disable it entirely:

Disable page numbering:

```
#show: ilm.with(  
  footer: none,  
)
```

Different styles:

```
// Alternating sides with chapter name (default)  
#show: ilm.with(  
  footer: "page-number-alternate-with-chapter",  
)  
  
// Left-aligned with chapter name  
#show: ilm.with(  
  footer: "page-number-left-with-chapter",  
)  
  
// Right-aligned with chapter name  
#show: ilm.with(  
  footer: "page-number-right-with-chapter",  
)  
  
// Centered page number only (no chapter name)  
#show: ilm.with(  
  footer: "page-number-center",  
)  
  
// Left-aligned page number only (no chapter name)  
#show: ilm.with(  
  footer: "page-number-left",  
)  
  
// Right-aligned page number only (no chapter name)  
#show: ilm.with(  
  footer: "page-number-right",  
)
```

2. Text

Typst defaults to English for the language of the text. If you are writing in a different language then you need to define your language before the 'Ilm template is loaded, i.e. before the `#show: ilm.with()` like so:

```
#set text(lang: "de")
#show: ilm.with(
    // 'Ilm's options defined here.
)
```

By defining the language before the template is loaded, 'Ilm will set title for bibliography and table of contents as per your language settings as long as you haven't customized it already.

2.1. External links

'Ilm adds a small maroon circle to external (outgoing) links like so^o.

This acts as a hint for the reader so that they know that a specific text is a hyperlink. This is far better than underlining a hyperlink or making it a different color. Don't you agree?

If you want to disable this behavior then you can do so by setting the concerning option to `false`:

```
#show: ilm.with(
    external-link-circle: false,
)
```

2.2. Blockquotes

'Ilm also exports a `blockquote` function which can be used to create blockquotes. The function has one argument: `body` of the type content and can be used like so:

```
#blockquote[
    A wizard is never late, Frodo Baggins. Nor is he early. He arrives precisely when he means to.
    — Gandalf
]
```

The above code will render the following:

A wizard is never late, Frodo Baggins. Nor is he early. He arrives precisely when he means to. — Gandalf

2.3. Small- and all caps

'Ilm also exports functions for styling text in small caps and uppercase, namely: `smallcaps` and `upper` respectively.

These functions will overwrite the standard `smallcaps`^o and `upper`^o functions that Typst itself provides. This behavior is intentional as the functions that 'Ilm exports fit in better with the rest of the template's styling.

Here is how Typst's own `SMALLCAPS` and `UPPER` look compared to the 'Ilm ones:

`SMALLCAPS` and `UPPER`

They both look similar, the only difference being that 'Ilm uses more spacing between individual characters.

If you prefer Typst's default spacing then you can still use it by prefixing `std-` to the functions:

```
#std-smallcaps[your content here]  
#std-upper[your content here]
```

2.4. Tables

In order to increase the focus on table content, we minimize the table's borders by using thin gray lines instead of thick black ones. Additionally, we use small caps for the header row. Take a look at the table below:

Quantity	Symbol	Unit
length	<i>l</i>	m
mass	<i>m</i>	kg
time	<i>t</i>	s
electric current	<i>I</i>	A
temperature	<i>T</i>	K
amount of substance	<i>n</i>	mol
luminous intensity	<i>I_v</i>	cd

Table 1: 'Ilm's styling

For comparison, this is how the same table would look with Typst's default styling:

Quantity	Symbol	Unit
length	<i>l</i>	m
mass	<i>m</i>	kg
time	<i>t</i>	s
electric current	<i>I</i>	A
temperature	<i>T</i>	K
amount of substance	<i>n</i>	mol
luminous intensity	<i>I_v</i>	cd

Table 2: Typst's default styling

3. Code

3.1. Custom font and size

'Ilm uses the *Iosevka*[1] font for raw text instead of the default *Fira Mono*. If Iosevka is not installed then the template will fall back to Fira Mono.

For comparison, here is what `code` in Fira Mono looks like:

```
fn main() {
    let user = ("Adrian", 38);
    println!("User {} is {} years old", user.0, user.1);

    // tuples within tuples
    let employee = (("Adrian", 38), "die Mobiliar");
    println!("User {} is {} years old and works for {}", employee.0.1, employee.0.1, employee.1);
}
```

Listing 1: Code snippet typeset in Fira Mono font

and here is how the same `code` looks in Iosevka:

```
fn main() {
    let user = ("Adrian", 38);
    println!("User {} is {} years old", user.0, user.1);

    // tuples within tuples
    let employee = (("Adrian", 38), "die Mobiliar");
    println!("User {} is {} years old and works for {}", employee.0.1, employee.0.1, employee.1);
}
```

Listing 2: Code snippet typeset in Iosevka font

In the case that both code snippets look identical then it means that Iosevka is not installed on your computer.

You can use Typst's default raw text formatting by setting the `raw-text` option to a special string:

```
#show: ilm.with(
    raw-text: "use-typst-default",
)
```

Alternatively, you can specify your own custom font and size using a dictionary:

```
#show: ilm.with(
    raw-text: (
        // font takes a list of fonts in order of priority.
        font: ("JetBrains Mono", "Cascadia Mono"),
        size: 10pt,
    ),
)
```

Bibliography

- [1] “Iosevka.” Accessed: June 18, 2024. [Online]. Available: <https://en.wikipedia.org/w/index.php?title=Iosevka&oldid=1217127968> ◉

Index of Tables

Table 1 ‘Ilm’s styling	10
Table 2 Typst’s default styling	10

Index of Listings

Listing 1 Code snippet typeset in Fira Mono font	11
Listing 2 Code snippet typeset in Iosevka font	11