



# Minimal Résumé

## Manual

Simple and professional résumé for professional people.

---

min-resume

0.1.0

MIT

Maycon F. Melo\*

## QUICK START

```
#import "@preview/min-resume:0.1.0": resume
#show: manual.resume(
  name: "Your Name",
  title: "Title or Main Occupation",
  photo: image("photo.png"),
  personal: "Relevant personal info",
  birth: (1997, 05, 19),
  address: "Your Address (no street nor house number)",
  email: "example@email.com",
  phone: "+1 (000) 000-0000",
)
```

## DESCRIPTION

Generate modern and direct to the point résumé, fit for today's Human Resources demands of assertiveness. There is no colorful designs, figures, creative fonts, nor anything that diverts attention when reading the document: is just plain old black sans-serif text in white paper. In fact, if one sees only the resulting résumé, may say it was written in Word — but it was written with all of Typst's benefits and conveniences.

As this package was written by a Brazilian, it was made using some of the Brazilian practices when writing a résumé — but it is simple and minimalistic even to Brazilian standards. Therefore, if some information are missing or unnecessary to you, feel free to adapt it to your needs.

---

\*<https://github.com/mayconfmelo>

This manual will be updated only when new versions break or modify something; otherwise, it will be valid to all newer versions starting by the one documented here.

## OPTIONS

Those are the full list of options available and its default values:

```
#import "@preview/min-manual:0.1.0": manual
#show: manual.with(
  name: none,
  title: none,
  photo: none,
  personal: none,
  birth: none,
  address: none,
  email: none,
  phone: none,
  date: datetime.today(),
  show-country-code: false,
  letter: (
    enterprise: none,
    dept: auto,
  ),
  skills: (
    display: "inline",
    sep: " " + sym.bullet + " ",
  ),
  paper: "a4",
  lang: "en",
  lang-data: toml("assets/lang.toml"),
  justify: true,
  line-space: 0.65em,
  par-margin: 1.2em,
  margin: 0.75in,
  font: "Arial",
  font-size: 11.5pt,
)
```

Seems like an awfull lot to start with, but let's just break down all this to understand it better, shall we?

**name:** `string` `content` (required)

The name of the résumé owner; generally your name, or someone you know.

**title:** string content

The title or main occupation of the résumé owner; keep it short and concise, just one line is enough.

**photo:** none

A photo of the résumé owner. It is not a trending practice nowadays, but if needed is supported.

**personal:** string content

Some general personal information, like nationality, or social status, or really anything that describes the résumé owner a little bit.

**birth:** array datetime

Date of birth of résumé owner; an array, in format (yyyy, mm, dd); or a proper #datetime.

**address:** string content

General address — no street name nor house number needs to be disclosed.

**email:** string content

Email address of the résumé owner. Printed with a *mailto:* hyperlink to conveniently send emails.

**phone:** string

Phone number of the résumé owner, starting with a country code in format +CODE where CODE is a number, separated from the phone number itself. Spaces, parenthesis, and dashes can be freely used as separators inside the phone number.

US phone example: +1 (000) 000-0000.

**date:** array datetime

Defines the document date — used only to set #document(date) option.

**show-country-code:** boolean

Defines whether the phone number will be printed as is or without the country code. For a domestic résumé this option can be false, but it is better to set it true when creating a résumé for some international job offer.

show-country-code: false example: (000) 000-0000.

**letter:** dictionary string boolean

Global letter configuration; can take a dictionary like:

```
(enterprise: ENTERPRISE, dept: DEPARTMENT)
```

If the value is STRING, passes (enterprise: STRING); and if the value is a boolean, just hide or show the letter using the default configurations.

**skills:** dictionary string

Global skills configuration; can take a dictionary like:

```
(display: MODE, sep: SEPARATOR)
```

If the value is STRING, passes (display: STRING), where the string must be whether "inline" or "list".

**paper:** string

Defines the page paper type — and its size therefore.

**lang:** string

Defines the language of the written text.

**justify:** boolean

Defines if the text will have justified alignment.

**line-space:** length

Defines the space between lines in the document.

**par-margin:** length

Defines the document margin space after each paragraph. Set it the same as line-space to remove get paragraphs without additional space in between.

**margin:** length

Defines the document margins.

**font:** string array

Defines the font families used for the text: a principal font and its fallbacks.

**font-size:** length

Defines the size of the text in the document.

## LETTER COMMAND

```
#letter(  
  body  
)
```

Adds a professional letter, in its own page. Receives configuration from `#resume(letter)` and a letter body content.

**letter.enterprise** `string` `boolean`

Received from `#resume`. The name of the letter receiver — an enterprise. If the résumé is created to no enterprise in particular, just set it to `true` to generate a letter without receiver; `false` hides the letter.

**letter.dept:** `auto` `string` `content`

Received from `#resume`. The name used for the enterprise's department which will receive the letter. If set as `auto`, try to automatically retrieve the appropriate name for *Human Resources Department* in `text.lang`, or fallback to English if not found.

**body** `content`

The letter content.

## JOB EXPERIENCE COMMAND

```
#xp(  
  role: none,  
  place: none,  
  time: none,  
  skills: none,  
  config: none,  
)
```

Adds a job experience entry. Receives configuration from `#resume(skills)` and some arguments on its own.

**role:** `string` `content` (*required*)

The occupation name or role played in this job.

**place:** `string` `content` (*required*)

The place of work or enterprise name.

**time:** `array` *(required)*

The duration of this work experience, in format (YYYY-INITIAL, MM-INITIAL, YYYY-FINAL, MM-FINAL), where at least the YYYY-INITIAL must be provided, and all omitted values will fallback to the current year and month — thus defining it a current job.

**skills:** `string` `content`

Skills learned and used, as well as goals accomplished. Can be a string or a content with a unnumbered list (topics) inside, shown as inline topics by default.

**config:** `dictionary` `none`

Locally override any global configurations received by `#resume(skills)`. Applicable to the current `#xp` only.

**skills.display:** `string`

Received from `#resume(skills)`. Make the skills list an "inline" topic list (saves space) or a proper "list". Can be overridden using the `config: (display)` argument.

**skills.sep:** `string` `content`

Received from `#resume(skills)`. Defines the separator for each inline topic item (when `display: "inline"`). Can be overridden using the `config: (sep)` argument.

## EDUCATION COMMAND

```
#edu(
  course: none,
  place: none,
  time: none,
  skills: none,
  config: none,
)
```

Adds an education entry, like a course or graduation. Receives configuration from `#resume(skills)` and some arguments on its own.

**course:** `string` `content` *(required)*

The course or graduation name.

**place:** `string` `content` *(required)*

The educational institution where the course or graduation take place.

**time:** `array` (*required*)

The duration of this course or graduation, in format (YYYY-INITIAL, MM-INITIAL, YYYY-FINAL, MM-FINAL), where at least the YYYY-INITIAL must be provided, and all omitted values will fallback to the current year and month — thus defining it a current course or graduation.

**skills:** `string` `content`

Skills learned and used, as well as goals accomplished. Can be a string or a content with a unnumbered list (topics) inside, shown as inline topics by default.

**config:** `dictionary` `none`

Locally override any global configurations received by `#resume(skills)`. Applicable to the current `#xp` only.

**skills.display:** `string`

Received from `#resume(skills)`. Make the skills list an "inline" topic list (saves space) or a proper "list". Can be overridden using the `config: (display)` argument.

**skills.sep:** `string` `content`

Received from `#resume(skills)`. Defines the separator for each inline topic item (when `display: "inline"`). Can be overridden using the `config: (sep)` argument.

## LINKEDIN QR CODE COMMAND

```
#linkedin-qrcode(
  user,
  size
)
```

Generates a Linkedin profile QR code.

**user** `string`

The Linkedin username, not the URL. Can be obtained from the profile URL, in format `https://www.linkedin.com/in/USERNAME`.

**size** `length`

The size of the QR code created.

## SKILLS INPUT COMMAND

```
#skills(  
  config: none,  
  skill-list,  
)
```

Allows to insert an arbitrary skill list. Receives configuration from `#resume(skills)` and some arguments on its own.

**config:** dictionary none

Locally override any global configurations received by `#resume(skills)`. Applicable to the current `#skills` only.

**skills.display:** string

Received from `#resume(skills)`. Make the skills list an "inline" topic list (saves space) or a proper "list". Can be overridden using the `config: (display)` argument.

**skills.sep:** string content

Received from `#resume(skills)`. Defines the separator for each inline topic item (when `display: "inline"`). Can be overridden using the `config: (sep)` argument.

**skill-list:** string content

Skills learned and used, as well as goals accomplished. Can be a string or a content with a unnumbered list (topics) inside, shown as inline topics by default.

## COPYRIGHT

Copyright © 2025 Maycon F. Melo.

This manual is licensed under MIT terms and rights.

The manual source code is free software: you are free to change and redistribute it. There is NO WARRANTY, to the extent permitted by law.

The logo was obtained from Flaticon website.