



Minimal Résumé

Manual

Simple and professional résumé for professional people

typst

min-resume^{*}

0.2.0

MIT-0

Maycon F. Melo[†]

Contents

Quick Start	2
Description	2
Options	2
Custom Settings	3
Commands	4
Letter	4
Custom List	5
Data Entry	5
Linkedin QR Code	6
YAML Data Structure	6
Copyright	7

^{*}<https://github.com/mayconfmelo/min-resume>

[†]<https://github.com/mayconfmelo>

Quick Start

```
#import "@preview/min-resume:0.2.0": resume
#show: manual.resume(
  name: "Name",
  title: "Academic Title and/or Occupation",
  photo: image("photo.png"),
  personal: "Relevant personal info",
  birth: (1997, 05, 19),
  address: "Public Address",
  email: "example@email.com",
  phone: "+1 (000) 000-0000",
)
```

Description

Generate a modern and straightforward résumé that meets today's Human Resources demands for assertiveness. There are no colorful designs, images, creative fonts, or anything that distracts from reading the document: it's just plain black sans-serif text on white paper. In fact, someone who sees only the resulting résumé might think it was written in Word — but it was actually created with all of Typst's benefits and conveniences.

The package was created with Brazilian trends and practices for Human Resources in mind; thus, if any information is missing or unnecessary for you, feel free to adapt it to your needs.

Options

```
#show: resume.with(
  name: none,
  title: none,
  photo: none,
  info: none,
  birth: none,
  address: none,
  email: none,
```

```
    phone: none,
    data: (),
    cfg: (:),
)
```

name: string content *(required)*

Full name.

title: string content

Academic title or occupation.

photo: image content

Photo or other identification (like a #linkedin QR code).

info: string content

Basic personal information.

birth: array dictionary

(year, month, day)

Birthday date.

address: string content *(required)*

Public address (no street nor house number).

email: string

Email address.

phone: string

Phone number, started by *country code*.¹

data: yaml dictionary

Generate YAML-based document (see YAML Data Structure section).

cfg: dictionary

Custom settings (see Custom Settings section).

Custom Settings

```
cfg: (
  country-code: false,
  letter-show: true,
```

¹<https://www.countrycode.org/>

```

lists: par,
entry-period: true,
entry-dates: "MMM/yyyy",
data-assets: (),
transl: yaml("assets/lang.yaml"),
typst-defaults: false,
)

```

country-code: boolean

Show the + country code in final result.

letter-show: boolean

Show #letter content.

lists: par list enum

Display #list content as inline, unnumbered, or numbered topics.

entry-period: boolean

Show period between dates of #entry(time) commands.

entry-dates: string

Date pattern used in #entry.

data-assets: dictionary

(name: asset)

Expose assets inside YAML-based document scope.

transl: yaml toml dictionary

Translation data (see src/assets/lang.yaml file).

typst-defaults: boolean

Revert back Typst defaults instead of those defined by *min-resume*.

Commands

Letter

```
#letter(
  to: none,
```

```
body
)
```

to: string content

Letter receiver.

Custom List

[#list](#)(body)

Generate custom lists of topics; by default, an inline bullet list (see `cfg.lists` option in Custom Settings).

Data Entry

```
#entry(
  title: none,
  organization: none,
  location: none,
  time: (:),
  skills: none,
)
```

Insert a professional experience or academic degree.

title: string content

Occupation or course name.

organization: string content

Organization related to the entry (enterprise, university, etc.).

location: string context

Organization location.

time: dictionary of arrays dictionary of datetimes array

(from: (yyyy, mm, dd), to: (yyyy, mm, dd)) (yyyy, mm, dd)

Start and end of entry; omit end date to insert a current one.

skills: list string

Related skills and topics (the same as `#list`).

Linkedin QR Code

```
#linkedin(user)
```

user string

www.linkedin.com/in/user

Linkedin username.

YAML Data Structure

```
- header: String
- code: String
- letter:
  to: String
  body: String
- list: Content
- entry:
  title: String
  organization: String
  location: String
  time:
    from: [1997, 5]
    to: [2000, 1]
  skills: Array or String
- linkedin: String
```

The file must be a YAML array, and each array item must be a dictionary with one and only one key — mind the indentation for options; the content is generated in the same order as the array (see `tests/data/data.yaml` file). Multiline YAML string values are allowed.

header: Generate a section heading.

code: Evaluate Typst code from string.

letter: Generate the professional letter.

list: Generate a special inline lists.

entry: Generate a job experience/academic formation entry.

linkedin: Generate a Linkedin profile QR code.

Copyright

Copyright © 2026 Maycon F. Melo.

This manual is licensed under MIT.

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.