

# pdf

A Rust CLI tool and library for converting Markdown files to PDF. Uses Typst as an embedded library with bundled fonts. No external dependencies required.

---

## Features

- Headings (H1-H6)
  - Paragraphs with bold, italic, and inline code
  - Code blocks with syntax highlighting
  - Ordered and unordered lists (with nesting)
  - Task lists / checklists
  - Tables
  - Horizontal rules
  - Smart page breaks (keeps headings with content, avoids widows/orphans)
  - Internal and external links
  - Manual page breaks
  - Beautiful typography with embedded Libertinus Serif font
- 

## Manual Page Breaks

Insert a manual page break anywhere in your markdown:

```
---pagebreak---
```

This must be on its own line as a paragraph.

---

## CLI Usage

```
# Convert markdown to PDF (output: input.pdf)
pdf input.md
```

```
# Specify output file
pdf input.md -o output.pdf
```

---

## Library Usage

Add to your Cargo.toml:

```
[dependencies]
pdf = { path = "path/to/pdf" }
```

Basic usage:

```
use std::fs;

fn main() {
    let markdown = "# Hello\n\nThis is bold and italic.";
    let pdf_bytes = pdf::markdown_to_pdf(markdown).expect("failed to create PDF");
    fs::write("output.pdf", pdf_bytes).expect("failed to write file");
}
```

You can also access the intermediate Typst markup:

```
fn main() {  
    let markdown = "# Title\n\nParagraph text."  
    let typst_markup = pdf::markdown_to_typst(markdown);  
    println!("{}", typst_markup);  
}
```

---

## Building

`cargo build --release`