








→

MD₂PDF

A portable, self-contained script that converts Markdown files to beautifully formatted PDFs.

Features

-  **Image support** - automatically converts SVG icons to PNG when available
-  **Portable** - works from any project directory
-  **Print-optimized** - responsive layout that looks great in PDF format
-  **Page breaks** - supports manual page break comments (`<!-- PAGE-BREAK -->`)
-  **PDF-only content** - special tags for content that only appears in PDF (`<!-- PDF-ONLY` or `<!-- PDF ONLY`)
-  **Table column widths** - precise control over table column sizing with comments (`<!--! col-widths:`)
-  **Token replacement** - dynamic content with configurable tokens (`Ziad Ezzat` , `2025-09-08` , etc.)

Usage

Basic Usage

```
# Convert README.md in current directory
./md2pdf.sh

# Convert a specific markdown file
./md2pdf.sh myfile.md
```

Installation in New Projects

Option 1: Manual Copy

1. Copy the entire `md2pdf` folder to your project root
2. Run from your project directory:

```
./md2pdf/md2pdf.sh
```

Option 2: Using Install Script

1. From within the `md2pdf` directory:

```
./install.sh /path/to/your/project
```

2. The script will copy everything needed and make it executable

Dependencies

The script automatically handles dependencies:

- **Pandoc** - Auto-installed via Homebrew on macOS
- **Google Chrome or Chromium** - Used for PDF generation

Manual Installation

If auto-installation fails:

macOS:

```
brew install pandoc  
# Chrome: Download from https://www.google.com/chrome/
```

Linux:

```
# Ubuntu/Debian  
sudo apt-get install pandoc google-chrome-stable  
  
# CentOS/RHEL  
sudo yum install pandoc google-chrome-stable
```

Output

- Input: `README.md` → Output: `README.pdf`
- Input: `myfile.md` → Output: `myfile.pdf`
- PDFs are created in the same directory as the input file

Styling Features

Typography

- **Headers:** Purple H1 with underline, clean hierarchy for H2-H6
- **Body text:** Inter font family with optimized line spacing
- **Code blocks:** Syntax highlighting with rounded corners
- **Tables:** Professional styling with alternating row colors

Images & Icons

- **SVG Support:** Automatically converts to PNG when available
- **Icon Processing:** Complex HTML icons become simple colored squares
- **Responsive:** Images scale appropriately for print

Print Optimization

- **Page margins:** 0.5 inch on all sides
- **Font scaling:** Optimized sizes for print readability
- **Table formatting:** Fixed column widths for consistent layout
- **Page breaks:** Manual control with HTML comments

Advanced Features

Manual Page Breaks

Add page breaks in your Markdown by including HTML comments:

```
## Section 1

Content here...

<!-- PAGE-BREAK -->

## Section 2

Content on new page...
```

PDF-Only Content Tags

You can include content that only appears in the PDF version using special HTML comments:

This content will only appear in the PDF, not in web/GitHub rendering.

- Perfect for print-specific instructions
- Copyright notices for printed versions
- Page-specific formatting notes

Alternative format with space instead of dash - both work identically

The script automatically processes these tags:

- `<!-- PDF-ONLY` - Content block that only appears in PDF (with dash)
- `<!-- PDF ONLY` - Alternative format with space instead of dash
- Content is processed as **both HTML and Markdown** during PDF generation
- Completely invisible in web/GitHub rendering
- Supports complex formatting including lists, links, and styling

Common Use Cases:

```
**Print Version - Generated:** January 2025  
**Copyright:** © 2025 Your Company Name  
**Internal Use Only** - Not for distribution  
  
## Project Documentation  
  
> 📄 **Note for printed version:** This document contains interactive links  
> that are not clickable in print. See the digital version for full functionality.  
  
For technical support, contact: support@company.com
```

Advanced Features:

- **Dual processing:** Content is converted from Markdown to HTML, then embedded in PDF
- **Rich formatting:** Supports all Markdown features including tables, code blocks, and links
- **Flexible syntax:** Use either `PDF-ONLY` or `PDF ONLY` - both formats work identically
- **Clean integration:** No impact on web rendering or GitHub display

Token Replacement System

The script supports dynamic token replacement to automatically populate content like developer names, dates, and project information. Tokens use the format `{{TOKEN_NAME}}` and are replaced during PDF generation.

Configuration File

Create a `md2pdf.config` file in either:

- The same directory as `md2pdf.sh` (script directory)
- The current working directory where you run the script

```
# md2pdf.config example
DEVELOPER_NAME=John Doe
GITHUB_HANDLE=johndoe
DEVELOPER_EMAIL=john@example.com
COMPANY_NAME=Acme Corp
PROJECT_VERSION=2.1.0
LICENSE=MIT License
```

Available Tokens

User-Defined Tokens (from config file):

- `Ziad Ezzat` - Developer's full name
- `feralcreative` - GitHub username
- `ziad@feralcreative.co` - Email address
- `Feral Creative` - Company/organization
- `1.0.1` - Project version
- `MIT License` - License information
- Any custom tokens you define

Automatic Date/Time Tokens:

- `2025-09-08` - Current date (2025-01-08)
- `September 08, 2025` - Long date format (January 08, 2025)
- `17:35` - Current time (14:30)
- `2025-09-08 17:35` - Date and time (2025-01-08 14:30)
- `2025`, `09`, `08` - Individual date components
- `1757378132` - Unix timestamp

System Information Tokens:

- `feral-studio` - System hostname
- `ziad` - Current user
- `/Users/ziad/www/ezzat/md2pdf` - Current working directory
- `/Users/ziad/www/ezzat/md2pdf` - Script location
- `md2pdf` - Auto-generated from directory name (or from config)

Usage Example

```
# md2pdf Documentation

**Created by:** Ziad Ezzat ([@feralcreative](https://github.com/feralcreative))
**Generated:** September 08, 2025 at 17:35
**Version:** 1.0.1





## Contact

For questions about this project, contact ziad@feralcreative.co.

---

_Document generated on 2025-09-08 17:35 by ziad@feral-studio_
```

Features:

-  **Automatic fallbacks:** Missing config file doesn't break the script
-  **Smart defaults:** `PROJECT_NAME` auto-generates from directory name
-  **Warning system:** Alerts about unprocessed tokens
-  **Flexible placement:** Config file can be in script or working directory

Configuration File Format

Create a `md2pdf.config` file with this format:

```
# Comments start with # and are ignored
# Empty lines are also ignored

# Developer Information
DEVELOPER_NAME=Your Full Name
GITHUB_HANDLE=yourusername
DEVELOPER_EMAIL=you@example.com
COMPANY_NAME=Your Company
DEVELOPER_ROLE=Your Role

# Project Information
PROJECT_NAME=                # Leave empty to auto-generate from directory
PROJECT_VERSION=1.0.0
PROJECT_DESCRIPTION=Your project description
PROJECT_REPO=https://github.com/yourusername/yourproject

# Custom tokens
LICENSE=MIT License
CONTACT_INFO=For questions, contact ziad@feralcreative.co
FOOTER_TEXT=Generated on 2025-09-08 by Ziad Ezzat

# Add your own custom tokens
CUSTOM_TOKEN=Custom Value
ANOTHER_TOKEN=Another Value
```

Advanced Token Features

Nested Tokens

Tokens can contain other tokens, which are processed in multiple passes:

```
# In config file:
CONTACT_INFO=For questions, contact ziad@feralcreative.co
FOOTER_TEXT=Generated on 2025-09-08 by Ziad Ezzat

# In markdown:
For questions, contact ziad@feralcreative.co
# Becomes: For questions, contact john@example.com

Generated on 2025-09-08 by Ziad Ezzat
# Becomes: Generated on 2025-01-08 by John Doe
```

Special Characters

The system properly handles URLs, paths, and special characters:

```
PROJECT_REPO=https://github.com/user/repo-name
PWD_PATH=/Users/ziad/www/ezzat/md2pdf/subfolder
```

Config File Locations

The script looks for `md2pdf.config` in this order:

1. **Script directory:** Same folder as `md2pdf.sh`
2. **Working directory:** Where you run the script from

This allows for:

- **Global config:** Place in script directory for all projects
- **Project-specific config:** Place in each project directory

Common Use Cases

Document Headers

```
# md2pdf Documentation

**Version:** 1.0.1
**Author:** Ziad Ezzat ([@feralcreative](https://github.com/feralcreative))
**Company:** Feral Creative
**Generated:** September 08, 2025
```

Contact Information

```
## Support

For technical support:

- **Email:** ziad@feralcreative.co
- **GitHub:** [@feralcreative](https://github.com/feralcreative)
- **Repository:** https://github.com/feralcreative/md2pdf
```

Document Footers

```
---

_Generated on 2025-09-08 by Ziad Ezzat_

**System Info:** Generated by ziad on feral-studio at 2025-09-08 17:35
```

PDF-Only Content with Tokens


```
## Print Information
```

```
**Document generated:** 2025-09-08 17:35
```

```
**System:** ziad@feral-studio
```

```
**Version:** 1.0.1
```

```
**Contact:** ziad@feralcreative.co
```

Token Processing Order

The script processes tokens in this specific order to handle nested tokens correctly:

1. **Config file tokens** (multiple passes to resolve nested tokens)
2. **Automatic date/time tokens** (processed last to handle tokens within config values)
3. **System information tokens**

This means you can use automatic tokens inside your config file values:

```
# This works - DATE_TODAY will be replaced after FOOTER_TEXT
```

```
FOOTER_TEXT=Generated on 2025-09-08 by Ziad Ezzat
```

Troubleshooting Tokens

Unprocessed Token Warnings

If you see warnings like:

```
⚠ Warning: Found unprocessed tokens:  
  {{SOME_TOKEN}}
```

Solutions:

- Add the token to your `md2pdf.config` file
- Check for typos in token names (case-sensitive)
- Ensure the config file is in the correct location

Config File Not Found

If you see:

```
ℹ No config file found (looked for md2pdf.config in script and current directories)
```

Solutions:

- Create `md2pdf.config` in the script directory (global)
- Create `md2pdf.config` in your project directory (project-specific)
- The script will still work without a config file (no token replacement)

Special Characters in Values

For values containing special characters, the script automatically escapes them:

```
# These work correctly:
PROJECT_REPO=https://github.com/user/repo-name
SPECIAL_PATH=/path/with/slashes
REGEX_PATTERN=[a-z]+$
```

Troubleshooting

Common Issues

“Pandoc not found”

- Install manually: `brew install pandoc` (macOS) or use your package manager

“Chrome not found”

- Install Google Chrome from <https://www.google.com/chrome/>
- Or install Chromium as alternative

“PDF generation failed”






- Check that input Markdown file exists
- Ensure Chrome/Chromium is properly installed
- Check `/tmp/md_styled.html` for debugging

Debug Mode

The script saves the intermediate HTML file at `/tmp/md_styled.html` for inspection if issues occur.

Portability

This script is designed to be completely portable:

-  No Node.js dependencies
-  No project-specific configurations
-  Works from any directory
-  Self-contained styling and logic
-  Cross-platform (macOS, Linux)

Simply copy the `md2pdf` folder to any project and run the script.

License

This script is provided as-is for personal or commercial use or whatever.