

Introduction

This zine helps in navigating the `zine_maker` code, a small software tool derived from `pyFPDF` library. The code is written in Python, and provides a few scripts for creating covers, colophons, content, and also signatures for preparing a PDF for the printer. The scripts run with python version 3.x.x and we can feed to the scripts inputs and outputs. As an input, we give the path to a text file, which shall create the text and images content of the zine, and as an output we give a the path/filename of our choice. We can also run the scripts with no input nor output, in which case, they take as default parameters the readme text from the 'text/' folder and produce an output to either of the folders 'covers', 'colophons', 'body', or 'zines', depending on which script we are running each time.

Get the code

Download the source code

The code is under development by the author and can be cloned or downloaded from https://git.systemserver.net/mara/zine_maker.

To download the project, near the top of the gitlab page, click on the download icon next to "Find file". Once downloaded, right-click to extract the files, or from terminal run:

```
tar -xvf zine_maker.tar -C /home/user/destination
unzip zine_maker.zip -d /home/user/destination
```

Or with git clone:

```
export username=zine
export token=DskM_8XtKt-Wym1xHd1
export repo=git.systemserver.net/mara/zine_maker.git
git clone https://$username:$token@$repo
cd zine_maker
```

Requirements

Basic requirements are listed here.

Python3 should be installed on the computer

<https://www.python.org/downloads/>

And also the pip command if it didn't get installed with Python

<https://pip.pypa.io/en/stable/installation/>

Once these are installed, from within zine_maker folder run:

```
pip install -r requirements.txt
```

Miscellaneous

Fonts

The source code comes with some fonts under the 'fonts' folder. You can use your fonts of preference by adding them either in the 'fonts' folder and edit the files `cover.py`, `colophon.py` and `doc_pdf.py` to give the new names. Or add your absolute font path directly to the python scripts.

Text

The input texts should be clean from characters added by some text editors or Operating Systems. Use the `cat` command to check your text is ready as input with:

```
cat --show-nonprinting input.txt
```

Characters such as `M-oM-;M` or `^M` (carriage Return / line feed) need to be removed. A cool tool for that is `dos2unix`, which is available as command line, but needs to be installed:

```
dos2unix filename
```

Or with the `sed` command:

```
sed -e "s/\r//g" file > newfile
```

Extensive info can be find at:

<https://www.cyberciti.biz/faq/sed-remove-m-and-line-feeds-under-unix-linux-bsd-appleosx/>

Layout

All the font styling happens in the `zine_maker` function `create_pages()`. The input text is parsed for specific tags or symbols in the beginning of each line and changes to the font color and size happen accordingly. We can add more or edit existing rules, directly in the `zine_maker` code.

Parameters

The python scripts `cover.py`, `colophon.py` and `doc_pdf.py` take a text input and an output filename. If we give no input/output, the default input is the related readme files under `'covers/'`, `'body/'`, `'colophons/'` folders.

Merge

For merging the cover, body and colophon pdf files, there are many pdf merger tools. One that is command line based and is used in this tutorial is `pdfunite` (<http://linux-commands-examples.com/pdfunite>).

Print

For shuffling the final pdf and prepare it for printing you need the pdfseparate command

(<http://www.linux-commands-examples.com/pdfseparate>).

it is used inside the shuffle_pdf.py file (see details at the end of this README).

Images

For making use of images in the script image magick needs to be installed (<http://www.imagemagick.org/>)

Run the code!

Make the content of the pdf

The default parameters included in the script would create a zine from this readme:

```
python doc_pdf.py
```

OR you can experiment with the other sample text found in this repository under the 'text' folder.

```
python doc_pdf.py text/images.txt body/images.pdf
```

OR get real and add your own text file and replace respectively the input and output filenames.

```
python doc_pdf.py text/<your_file>.txt body/<output-name>.pdf
```

Make the cover of the pdf

Same, the default parameters included in the script would create the cover for the zine_maker:

```
python cover.py
```

OR try-out the other cover samples:

```
python cover.py text/cover.txt covers/cover.pdf
```

OR add your cover text file and replace respectively:

```
python cover.py text/<your-cover>.txt
```

```
covers/<cover-name>.pdf
```

Make the colophon of the pdf

Same, the default parameters included in the script would create the colophon for the zine_maker:

```
python colophon.py
```

OR try-out the other colophon samples:

```
python colophon.py text/colophon.txt colophons/colophon.pdf
```

OR add your own colophon text file and replace respectively:

```
python colophon.py text/<your-colophon>.txt
```

```
colophons/<output-colophon>.pdf
```


Make a screen PDF

For the final screen version

from the terminal run:

```
pdfunite covers/cover.pdf body/body.pdf
```

```
colophons/colophon.pdf zines/final.pdf
```

You shall substitute the file paths to your own corresponding

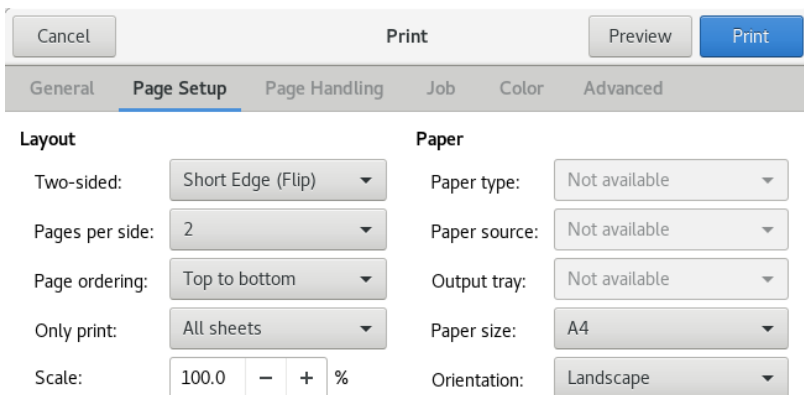
/path/filenames

See this readme as a zine:

open the file zines/zinemaker.pdf

Make a zine

Prepare signatures for printing



Cancel Print Preview Print

General **Page Setup** Page Handling Job Color Advanced

Layout

Two-sided: Short Edge (Flip) ▾

Pages per side: 2 ▾

Page ordering: Top to bottom ▾

Only print: All sheets ▾

Scale: 100.0 - + %

Paper

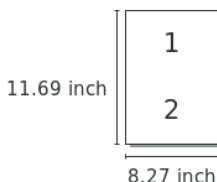
Paper type: Not available ▾

Paper source: Not available ▾

Output tray: Not available ▾

Paper size: A4 ▾

Orientation: Landscape ▾



The script `shuffle.py` creates signatures for printing and folding the A4 in two, to create a zine. Default parameters: as input

"zines/zinemaker_screen.pdf" and as output

"zinemaker{random_number}.pdf"

Run it for your own pdf files as following:

```
python shuffle_pdf.py zines/<input_file>.pdf
```

[zines/<output_file>.pdf](#)

Note: your input file needs to have an even number of pages, divisible by 4, which is the number of pages needed for one signature. We can add blank pages in the beginning and end of the body pdf file we created before if we need to make the total page number divisible by 4.

In the printer settings opt-in for the following:

- A4 Landscape
- Two pages per side
- Double side - short edge

References

code repository:

https://git.systerserver.net/mara/zine_maker

install python:

<https://www.python.org/downloads/>

install pip:

<https://pip.pypa.io/en/stable/installation/>

clean the text file for parsing:

<https://www.cyberciti.biz/faq/sed-remove-m-and-line-feeds-under-unix-linux-bsd-appleosx/>

merge and split pdf files:

<http://linux-commands-examples.com/pdfunite>

<http://www.linux-commands-examples.com/pdfseparate>

process images:

<http://www.imagemagick.org/>