mkdocs-print-site-plugin Docs

MkDocs Plugin allowing your site visitors to *File > Print > Save as PDF* the entire site.

Authors: Tim Vink

Website: http://127.0.0.1:8000/

Repo: https://github.com/timvink/mkdocs-print-site-plugin/

Table of Contents

Customization

pytest passing python 3.5 | 3.6 | 3.7 | 3.8 pypi v0.8 downloads 510/month codecov 86% contributors 1 license MIT

1 mkdocs-print-site-plugin

MkDocs plugin that adds a page to your site combining all pages, allowing your site visitors to *File > Print > Save as PDF* the entire site.

1.1 Installation

Install the plugin using `pip3`:

```
pip3 install mkdocs-print-site-plugin
```

Next, add the following lines to your `mkdocs.yml`:

```
plugins:
  - search
  - print-site
```

> ⚠️ Make sure to put `print-site` to the **bottom** of the plugin list. This is because other plugins might alter your site (like the navigation), and you want these changes included in the print page.
>
> If you have no `plugins` entry in your config file yet, you'll likely also want to add the `search` plugin. MkDocs enables it by default if there is no `plugins` entry set.

2 Options

You can customize the plugin by setting options in `mkdocs.yml`. This site uses the following settings:

```yaml
plugins:
  - print-site:
      add_to_navigation: true
      print_page_title: 'Print Site'
      add_table_of_contents: true
      add_full_urls: false
      enumerate_headings: true
      enumerate_figures: true
      add_cover_page: true
      cover_page_template: ""
```

`add_to_navigation`

Default is `true`. Adds a link 'Print Site' to your site navigation.

`print_page_title`

Default is `'Print Site'`. When `add_to_navigation` is set to `true` this setting controls the name of the print page in the navigation of the site. This setting is ignored when `add_to_navigation` is set to `false`.

`add_table_of_contents`

Default is `true`. Adds a table of contents section at the beginning of the print page.

`add_full_urls`

Default is `false`. When printing a page, you cannot see the target of a link. This option adds the target url in parenthesis behind a link.

For example "google.com" will be replaced by "google.com (https://www.google.com)"

`enumerate_headings`

Default `false`. This will add numbering (enumeration) to all headings as well as the table of contents. Note this will only enumerate the print site page; if you want to enumerate the entire site, you can use mkdocs-enumerate-headings-plugin.

Example "1.2 A chapter subsection".

`enumerate_figures`

Default `false`. This will add numbering to all figure captions (for example "Figure 1: "). Works especially well with mkdocs-img2fig-plugin.

`add_cover_page`

Default `false`. When enabled, a cover page is added to the print page, displaying the `site_title` and other information from the `mkdocs.yml` file. See also Customizing the cover page

`cover_page_template`

Default `""`. The path to a custom cover page template to use. See Customizing the cover page for more info.

3 Adding a print button

You might want to customize your site to include a 'print' button on every page (like the one in the right corner of this page 👆)

MkDocs supports theme extension, an easy way to override parts of a theme. You can use `page.url_to_print_page` to get the link to the site print page.

3.1 Adding a print button to mkdocs-material theme

In `mkdocs-material` theme (see customization), you can create a file for overrides and point to it your `mkdocs.yml`. You might also want to disable adding the print page to the navigation when you already have a button with the link.

Example:

mkdocs.yml **docs/overrides/main.html**

```
theme:
name: material
custom_dir: docs/overrides

plugin:
    - print-site:
        - add_to_navigation: false
```

```
{% extends "base.html" %}

{% block content %}

{% if page.url_to_print_page %}
    <a href="{{ page.url_to_print_page }}" title="Print Site" class="md-content__button md-
icon">
        {% include ".icons/material/printer.svg" %}
    </a>
{% endif %}

{{ super() }}
{% endblock content %}
```

3.2 Adding a print button to mkdocs theme

You can also customize the base mkdocs theme, by creating a file for overrides and pointing to it your `mkdocs.yml`. You might also want to disable adding the print page to the navigation when you already have a button with the link.

Example:

mkdocs.yml docs/overrides/main.html

```
theme:
    name: mkdocs
    custom_dir: docs/overrides

plugin:
    - print-site:
        - add_to_navigation: false
```

```
{% extends "base.html" %}

{% block repo %}
    {% if page.url_to_print_page %}
        <li class="nav-item">
            <a href="{{ page.url_to_print_page }}" title="Print Site" class="nav-link">
                <i class="fa fa-print"></i> Print
            </a>
        </li>
    {% endif %}

{{ super() }}
{% endblock repo %}
```

4 Adding a PDF button

You might want to make downloading a PDF even easier for your users. You could include a 'PDF' button on every page (like the one in the right corner of this page 👆)

MkDocs supports theme extension, an easy way to override parts of a theme. You can use `page.url_to_pdf` to get the link to site PDF from a page.

> **Info**
>
> While it might be easier for your users, using this option means you need to manually create the PDF everytime you make a change to your website.
>
> If you use this option and you are using version control like git, you might want to also gitignore the PDF file.

4.1 Adding a PDF button to mkdocs-material theme

In `mkdocs-material` theme (see customization), you can create a file for overrides and point to it your `mkdocs.yml`. You might also want to disable adding the print page to the navigation when you already have a button with the link.

Example:

mkdocs.yml **docs/overrides/main.html**

```yaml
theme:
name: material
custom_dir: docs/overrides

plugin:
    - print-site:
        - path_to_pdf: "assets/the_name_of_your_file.pdf"
        - add_to_navigation: false
```

```html
{% extends "base.html" %}

{% block content %}

{% if page.url_to_pdf %}
    <a href="{{ page.url_to_pdf }}" title="Site PDF" class="md-content__button md-icon">
        {% include ".icons/material/pdf-box.svg" %}
    </a>
{% endif %}

{{ super() }}
{% endblock content %}
```

4.2 Adding a print button to mkdocs theme

You can also customize the base mkdocs theme, by creating a file for overrides and pointing to it your `mkdocs.yml`. You might also want to disable adding the print page to the navigation when you already have a button with the link.

Example:

mkdocs.yml **docs/overrides/main.html**

```
theme:
    name: mkdocs
    custom_dir: docs/overrides

plugin:
    - print-site:
        - path_to_pdf: "assets/the_name_of_your_file.pdf"
        - add_to_navigation: false
```

```
{% extends "base.html" %}

{% block repo %}
    {% if page.url_to_pdf %}
        <li class="nav-item">
            <a href="{{ page.url_to_pdf }}" title="Site PDF" class="nav-link">
                <i class="fas fa-file-pdf"></i> PDF
            </a>
        </li>
    {% endif %}

{{ super() }}
{% endblock repo %}
```

5 Customize the cover page

When the `add_cover_page` option is set to true, `mkdocs-print-site-plugin` will generate a cover page when printing. This cover page is quite basic, and you might want to customize it.

You can do that by specifying a custom cover page template in the `mkdocs.yml` file. This is a standard jinja2 template file where you can combine HTML and use the information specified in `mkdocs.yml` as jinja2 variables (see mkdocs project information).

Example:

mkdocs.yml **docs/assets/templates/custom_cover_page.tpl**

```
plugins:
    - print-site:
        add_cover_page: true
        cover_page_template: "docs/assets/templates/custom_cover_page.tpl"
```

```
{% if config.site_name %}
    <h1>{{ config.site_name }}</h1>
{% endif %}
<h2>This is my custom print cover page</h2>
```

As an example, have a look at the default cover_page.tpl.

5.1 Adding extra content

You might want to add some content to your cover page that's not yet specified in your `mkdocs.yml` file. Of course you could just hard-code it in your custom template file, but you could also make use of MkDocs's extra context feature, allowing you to use `{{ config.extra.<your variable> }}`

Example:

mkdocs.yml **docs/assets/templates/custom_cover_page.tpl**

```yaml
plugins:
    - print-site:
        add_cover_page: true
        cover_page_template: "docs/assets/templates/custom_cover_page.tpl"

extra:
    abstract: This is a report about a topic
```

```
{% if config.site_name %}
    <h1>{{ config.site_name }}</h1>
{% endif %}
<p>{{ config.extra.abstract }}</p>
```

5.2 Change the styling

You'll likely also need to change the styling to your liking. You can add your own CSS file using the extra_css option from MkDocs. `mkdocs-print-site-plugin` wraps the cover page in a `<section>` with id `print-site-cover-page`. You should use this in your CSS to ensure not affecting other pages.

Example:

mkdocs.yml **docs/assets/css/my_cover_page.css**

```yaml
plugins:
    - print-site:
        add_cover_page: true

extra_css:
    - docs/assets/css/my_cover_page.css
```

```css
#print-site-cover-page h1 {
    color: blue;
}
```

6 Demo Content

This content is here to demonstrate what it looks like while printing.

👉 Go ahead and visit the print page and check it out!

6.1 Links to other pages

`mkdocs-print-site-plugin` will fix internal links when combining all the pages into one. Try navigating to other site pages using these internal links:

- Home
- Options
- Contributing

6.2 Links to other sections

When combining all pages into one, `mkdocs-print-site-plugin` will also ensure anchor links keep working (also to anchor links on other pages). Try them out:

- Dummy section lower down this demo page
- The Manual Testing in the contributing guide

6.3 Markdown extensions

MkDocs has support for many markdown extensions (see mkdocs-material reference). Below is a quick showcase so you can see how they print.

> ✎ **Phasellus posuere in sem ut cursus**
>
> Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nulla et euismod nulla. Curabitur feugiat, tortor non consequat finibus, justo purus auctor massa, nec semper lorem quam in massa.

> Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nulla et euismod nulla. Curabitur feugiat, tortor non consequat finibus, justo purus auctor massa, nec semper lorem quam in massa.

Note

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nulla et euismod nulla. Curabitur feugiat, tortor non consequat finibus, justo purus auctor massa, nec semper lorem quam in massa.

```python
def bubble_sort(items):
    for i in range(len(items)):
        for j in range(len(items) - 1 - i):
            if items[j] > items[j + 1]:
                items[j], items[j + 1] = items[j + 1], items[j]
```

Nunc eu odio eleifend, blandit leo a, volutpat sapien. Phasellus posuere in sem ut cursus. Nullam sit amet tincidunt ipsum, sit amet elementum turpis. Etiam ipsum quam, mattis in purus vitae, lacinia fermentum enim.

Note

This is a collapsible block, that is collapsed by default.

Example of a button

Primary button

With icon

```python
import tensorflow as tf
```

```python
def bubble_sort(items):
    for i in range(len(items)):
        for j in range(len(items) - 1 - i):
            if items[j] > items[j + 1]:
                items[j], items[j + 1] = items[j + 1], items[j]
```

```python
def bubble_sort(items):
    for i in range(len(items)):
        for j in range(len(items) - 1 - i):
            if items[j] > items[j + 1]:
                items[j], items[j + 1] = items[j + 1], items[j]
```

The `range()` function is used to generate a sequence of numbers.

`⌃ Ctrl`+`⌥ Alt`+`⌦ Del`

C C++

```
#include <stdio.h>

int main(void) {
  printf("Hello world!\n");
  return 0;
}
```

```
#include <iostream>

int main(void) {
  std::cout << "Hello world!" << std::endl;
  return 0;
}
```

Another tabbed content

Unordered list Ordered list

- Sed sagittis eleifend rutrum
- Donec vitae suscipit est
- Nulla tempor lobortis orci

1. Sed sagittis eleifend rutrum
2. Donec vitae suscipit est
3. Nulla tempor lobortis orci

Embedding content:

Example

Unordered List Ordered List

Example:

```
* Sed sagittis eleifend rutrum
* Donec vitae suscipit est
* Nulla tempor lobortis orci
```

Result:

- Sed sagittis eleifend rutrum
- Donec vitae suscipit est
- Nulla tempor lobortis orci

Example:

```
1. Sed sagittis eleifend rutrum
2. Donec vitae suscipit est
3. Nulla tempor lobortis orci
```

Result:

1. Sed sagittis eleifend rutrum
2. Donec vitae suscipit est
3. Nulla tempor lobortis orci

Method	Description
`GET`	✓ Fetch resource
`PUT`	✓✓ Update resource
`DELETE`	✕ Delete resource

Lorem ipsum[1] dolor sit amet, consectetur adipiscing elit.[2]

- 👤 – `.icons/material/account-circle.svg`
- 😆 – `.icons/fontawesome/regular/laugh-wink.svg`
- 🐱 – `.icons/octicons/octoface-16.svg`
- Ⓜ – Medium

- 🐦 – Twitter
- f – Facebook

😄

6.3.1 Images

`mkdocs-print-site-plugin` supports enumerating figure captions (which can be added easily using the img2fig plugin):

Figure 1: Image caption

Figure 2: Another image caption

6.3.2 Lists

1. Vivamus id mi enim. Integer id turpis sapien. Ut condimentum lobortis sagittis. Aliquam purus tellus, faucibus eget urna at, iaculis venenatis nulla. Vivamus a pharetra leo.
    a. Vivamus venenatis porttitor tortor sit amet rutrum. Pellentesque aliquet quam enim, eu volutpat urna rutrum a. Nam vehicula nunc mauris, a ultricies libero efficitur sed.
    b. Morbi eget dapibus felis. Vivamus venenatis porttitor tortor sit amet rutrum. Pellentesque aliquet quam enim, eu volutpat urna rutrum a.
        i. Mauris dictum mi lacus
        ii. Ut sit amet placerat ante
        iii. Suspendisse ac eros arcu

`Lorem ipsum dolor sit amet`

> Sed sagittis eleifend rutrum. Donec vitae suscipit est. Nullam tempus tellus non sem sollicitudin, quis rutrum leo facilisis.

`Cras arcu libero`

> Aliquam metus eros, pretium sed nulla venenatis, faucibus auctor ex. Proin ut eros sed sapien ullamcorper consequat. Nunc ligula ante.
>
> Duis mollis est eget nibh volutpat, fermentum aliquet dui mollis. Nam vulputate tincidunt fringilla. Nullam dignissim ultrices urna non auctor.

- ☑ Lorem ipsum dolor sit amet, consectetur adipiscing elit
- □ Vestibulum convallis sit amet nisi a tincidunt
    - ☑ In hac habitasse platea dictumst
    - ☑ In scelerisque nibh non dolor mollis congue sed et metus
    - □ Praesent sed risus massa
- □ Aenean pretium efficitur erat, donec pharetra, ligula non scelerisque

$$\operatorname{ker} f = \{g \in G : f(g) = e_H\}.$$

The homomorphism f is injective if and only if its kernel is only the singleton set e_G, because otherwise $\exists a, b \in G$ with $a \neq b$ such that $f(a) = f(b)$00.

6.4 Dummy section

This section has an incoming anchor link, at the top of this page

6.5 Some lorem ipsum

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut eget mi lacinia arcu ultrices rutrum eget a lacus. Etiam erat mi, sodales at nisl vel, bibendum tempor nunc. Cras in ultrices augue. Cras aliquam in mauris et semper. In hac habitasse platea dictumst. Proin dignissim scelerisque risus, consectetur ornare justo ultrices eu. Quisque tempus, elit eu ullamcorper interdum, metus augue pulvinar lectus, nec dictum mi dolor non turpis. Sed condimentum vulputate pretium.

Nulla at nisl tortor. Praesent vitae turpis sit amet sem condimentum fermentum eget nec dolor. Maecenas et imperdiet ante, at ultrices orci. Nunc ornare sodales enim. Sed tempor vitae mi et faucibus. Nunc aliquam est sit amet mauris tempus varius. Aenean blandit vel nibh nec sagittis. Sed vehicula nunc a nunc vehicula viverra. Proin risus justo, ullamcorper ac sem a, vulputate ornare justo. Sed facilisis pharetra elit, vitae dignissim nibh iaculis eu. Suspendisse potenti. Curabitur quis arcu ac est faucibus suscipit vel non lacus.

Ut tincidunt sapien sed sem auctor, et pellentesque erat tristique. Nunc porttitor lacus diam, eu malesuada nibh venenatis in. Donec sit amet enim eget enim facilisis placerat nec eget tortor. Etiam imperdiet, felis ac posuere dignissim, nulla sapien auctor mauris, ac aliquam orci leo nec dui. Donec efficitur turpis quis enim efficitur, eu ornare nisi consectetur. Sed ac arcu at orci pretium lobortis luctus non augue. Duis posuere purus at semper fringilla. Pellentesque facilisis libero vestibulum elit varius iaculis. Donec dapibus pretium scelerisque.

Suspendisse non orci vitae lorem placerat pretium vitae a ex. Nunc facilisis aliquam risus in vehicula. Fusce sodales bibendum lectus id ultricies. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. In cursus blandit quam ac bibendum. Interdum et malesuada fames ac ante ipsum primis in faucibus. Ut commodo tellus vel interdum vulputate. Nulla in turpis tellus. Mauris at semper ex. Nulla varius leo eu libero placerat, quis euismod orci euismod. Phasellus pulvinar ut sapien nec elementum. Maecenas vel mi eros. Interdum et malesuada fames ac ante ipsum primis in faucibus. Phasellus volutpat massa vel purus interdum imperdiet. Curabitur congue turpis eget faucibus varius. Aenean eleifend placerat lorem vel vestibulum.

Nullam in posuere urna. Sed cursus est porta maximus dignissim. Etiam id ante libero. Curabitur ac rhoncus turpis. Cras eu ipsum lacus. Aliquam ac rutrum elit. Donec pharetra in arcu feugiat interdum. Nam sed libero semper, sollicitudin urna vel, tincidunt nulla. Curabitur dapibus massa lectus, vulputate fermentum est finibus et. Ut efficitur velit nec justo varius tempor. Nullam aliquet commodo enim eget lobortis. Nullam sit amet nunc viverra, iaculis sem non, scelerisque mauris. Vivamus eu finibus lacus, dignissim luctus elit.

---

1. Lorem ipsum dolor sit amet, consectetur adipiscing elit. ↩
2. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nulla et euismod nulla. Curabitur feugiat, tortor non consequat finibus, justo purus auctor massa, nec semper lorem quam in massa. ↩

7 Contribution Guidelines

Thanks for considering to contribute to this project! Some guidelines:

- Go through the issue list and if needed create a relevant issue to discuss the change design. On disagreements, maintainer(s) will have the final word.
- You can expect a response from a maintainer within 7 days. If you haven't heard anything by then, feel free to ping the thread.
- This package tries to be as simple as possible for the user (hide any complexity from the user). Options are only added when there is clear value to the majority of users.
- When issues or pull requests are not going to be resolved or merged, they should be closed as soon as possible. This is kinder than deciding this after a long period. Our issue tracker should reflect work to be done.

7.1 Unit Tests

Make sure to install an editable version before running tests:

```
pip install -r tests/test_requirements.txt
pip install -e .
pytest --cov=mkdocs_print_site_plugin --cov-report term-missing tests/
```

If it makes sense, writing tests for your PRs is always appreciated and will help get them merged.

In addition, this project uses pyflakes for static code checking:

```
pip install pyflakes
pyflakes tests/ mkdocs_print_site_plugin/
```

7.2 Manual testing

To quickly serve a website with your latest changes to the plugin use the sites in our tests suite. For example:

```
pip install -r tests/test_requirements.txt
pip install -e .
mkdocs serve -f tests/fixutures/basic/mkdocs.yml
```

Tip: If you use google chrome, you can also view the print version of a page inside the browser by setting the renderer.

7.3 Code Style

Make sure your code *roughly* follows PEP-8 and keeps things consistent with the rest of the code. I recommended using black to automatically format your code.

We use google-style docstrings.

7.4 Documentation

Is in `docs/`. To deploy the docs, run:

```
mkdocs gh-deploy
```