

Модуль markdown-pdf

Бесплатный, с открытым исходным кодом Python модуль `markdown-pdf` позволит создать PDF файл из вашего контента в формате `markdown`.

При создании PDF файла вы можете:

- Использовать в `markdown` текст на любом языке в кодировке UTF-8
- Встраивать используемые в `markdown` картинки
- Разбивать текст на страницы в нужном порядке
- Создавать оглавление (bookmarks) из заголовков `markdown`
- Оформлять нужные элементы при помощи вашего CSS кода
- Использовать разные размеры страниц внутри одного pdf
- Создавать таблицы в `markdown`

Модуль использует функции двух замечательных библиотек.

- [markdown-it-py](#) для преобразования `markdown` в `html`.
- [PyMuPDF](#) для преобразования `html` в `pdf`.

Установка

```
pip install markdown-pdf
```

Использование

Создаем pdf с оглавлением (bookmarks) из заголовков до 2 уровня.

```
from markdown_pdf import MarkdownPdf
```

```
pdf = MarkdownPdf(toc_level=2)
```

Добавляем в pdf первую секцию. Заголовок не включаем в оглавление.

```
from markdown_pdf import Section
```

```
pdf.add_section(Section("# Title\n", toc=False))
```

Добавляем вторую секцию. В pdf-файле она начинается с новой страницы. Заголовок центрируется при помощи CSS, включается в оглавление pdf-файла и на страницу встраивается изображение из файла `img/python.png`.

```
pdf.add_section(  
    Section("# Head1\n\n! [python] (img/python.png) \n\nbody\n"),  
    user_css="h1 {text-align:center;}"  
)
```

Добавляем третью секцию. Два заголовка разного уровня из этой секции включаются в оглавление pdf-файла. Секция имеет альбомную ориентацию страниц A4.

```
pdf.add_section(Section("## Head2\n\n### Head3\n\n", paper_size="A4-L"))
```

Добавляем четвертую секцию с таблицей.

```
text = """# Section with Table

|TableHeader1|TableHeader2|
|--|--|
|Text1|Text2|
|ListCell|<ul><li>FirstBullet</li><li>SecondBullet</li></ul>|
"""

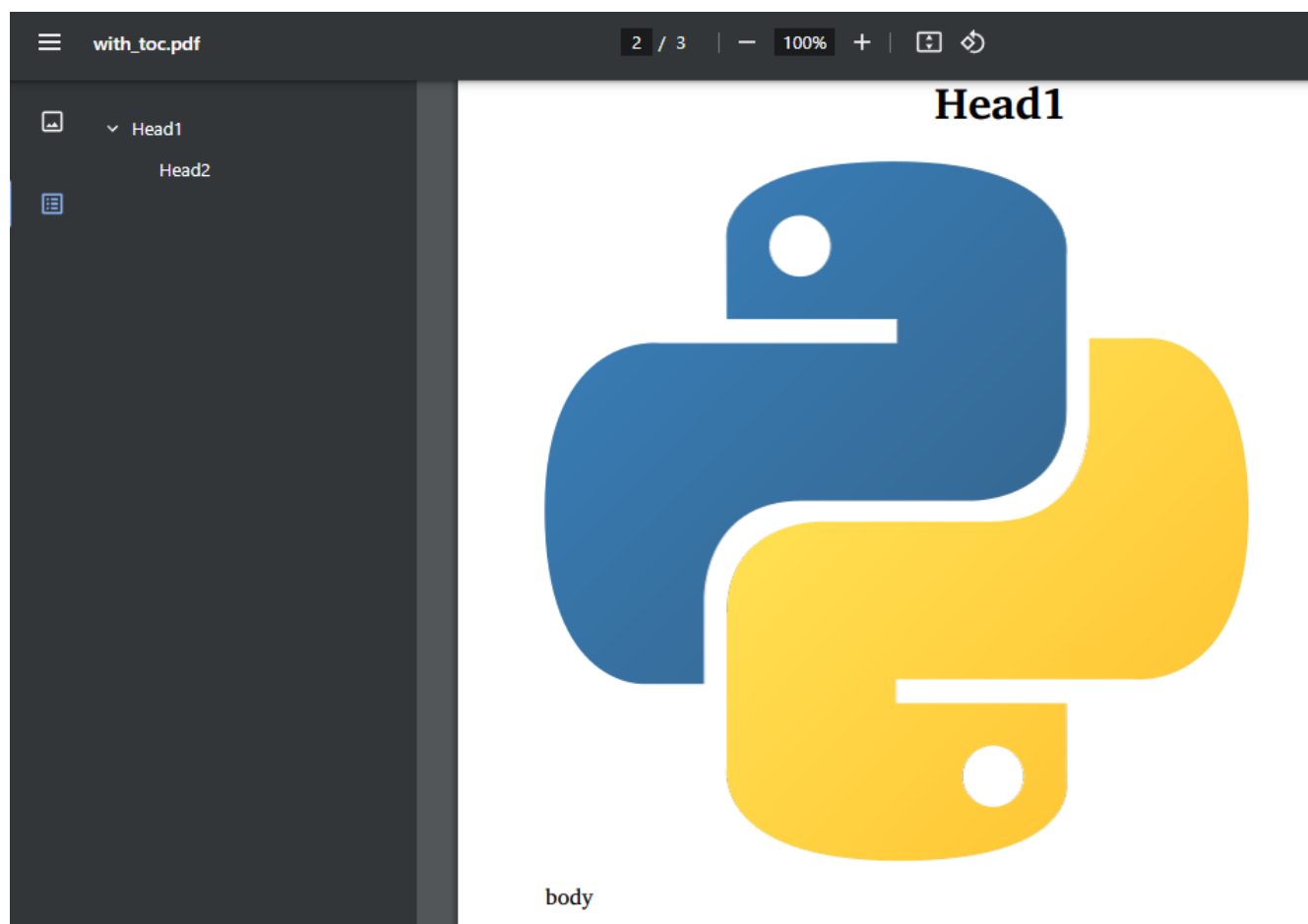
pdf.add_section(Section(text))
```

Устанавливаем свойства pdf документа.

```
pdf.meta["title"] = "Руководство пользователя"
pdf.meta["author"] = "Виталий Богомолов"
```

Сохраняем в файл.

```
pdf.save("guide.pdf")
```



Настройки и опции

Класс `Section` задает порцию данных markdown, которые обрабатываются по единым правилам. Данные следующего `Section` начинаются с новой страницы.

У класса `Section` можно задавать следующие атрибуты.

- `toc`: нужно ли включать заголовки `<h1>` - `<h6>` этой секции в ТОС. По умолчанию `True`.
- `root`: имя корневого каталога, от которого начинаются пути файлов картинок в `markdown`. По умолчанию `"."`.
- `paper_size`: название размера бумаги, [как описано здесь](#). По умолчанию `"A4"`.
- `borders`: размер полей. По умолчанию `(36, 36, -36, -36)`.

Для присвоения доступны следующие свойства документа (словарь `MarkdownPdf.meta`) с указанными значениями по умолчанию.

- `creationDate`: текущая дата
- `modDate`: текущая дата
- `creator`: `"PyMuPDF library: https://pypi.org/project/PyMuPDF"`
- `producer`: `""`
- `title`: `""`
- `author`: `""`
- `subject`: `""`
- `keywords`: `""`

Пример

В качестве примера вы можете загрузить созданный из этого `md` файла [pdf файл](#). Для создания `pdf` файла использовался [этот скрипт](#) Python.