

4. Лабораторная работа №4. Язык разметки Markdown

4.1. Цель работы

Целью работы является освоение процедуры оформления отчетов с помощью легковесного языка разметки Markdown.

4.2. Теоретическое введение

4.2.1. Базовые сведения о Markdown

Чтобы создать заголовок, используйте знак #, например:

```
# This is heading 1
## This is heading 2
### This is heading 3
#### This is heading 4
```

Чтобы задать для текста полужирное начертание, заключите его в двойные звездочки:

This text is **bold**.

Чтобы задать для текста курсивное начертание, заключите его в одинарные звездочки:

This text is *italic*.

Чтобы задать для текста полужирное и курсивное начертание, заключите его в тройные звездочки:

This is text is both *****bold and italic*****.

Блоки цитирования создаются с помощью символа >:

> The drought had lasted now for ten million years, and the reign of
↪ the terrible lizards had long since ended. Here on the Equator,
↪ in the continent which would one day be known as Africa, the
↪ battle for existence had reached a new climax of ferocity, and
↪ the victor was not yet in sight. In this barren and desiccated
↪ land, only the small or the swift or the fierce could flourish,
↪ or even hope to survive.

Упорядоченный список можно отформатировать с помощью соответствующих цифр:

1. First instruction
 1. Sub-instruction
 1. Sub-instruction
1. Second instruction

Чтобы вложить один список в другой, добавьте отступ для элементов дочернего списка:

1. First instruction
1. Second instruction
1. Third instruction

Неупорядоченный (маркированный) список можно отформатировать с помощью звездочек или тире:

- * List item 1
- * List item 2
- * List item 3

Чтобы вложить один список в другой, добавьте отступ для элементов дочернего списка:

- List item 1
 - List item A
 - List item B
- List item 2

Синтаксис Markdown для встроенной ссылки состоит из части [link text], представляющей текст гиперссылки, и части (file-name.md) – URL-адреса или имени файла, на который дается ссылка:

[link text](file-name.md)

или

[link text](http://example.com/ "Необязательная подсказка")

Markdown поддерживает как встраивание фрагментов кода в предложение, так и их размещение между предложениями в виде отдельных огражденных блоков. Огражденные блоки кода — это простой способ выделить синтаксис для фрагментов кода. Общий формат огражденных блоков кода:

```
``` language
your code goes in here
```
```

4.2.2. Оформление формул в Markdown

Внутритекстовые формулы делаются аналогично формулам LaTeX. Например, формула $\sin^2(x) + \cos^2(x) = 1$ запишется как

$\sin^2(x) + \cos^2(x) = 1$

Выключение формулы:

$$\sin^2(x) + \cos^2(x) = 1 \quad (4.1)$$

со ссылкой в тексте «Смотри формулу ({-eq. 4.1}).» записывается как

```
$$
\sin^2 (x) + \cos^2 (x) = 1
$$ {#eq:eq1}
```

Смотри формулу (`[-@eq:eq1]`).

4.2.3. Оформление изображений в Markdown

В Markdown вставить изображение в документ можно с помощью непосредственного указания адреса изображения. Синтаксис данной команды выглядит следующим образом:

```
![Подпись к рисунку](/путь/к/изображению.jpg "Необязательная
↪ подсказка"){ #fig:fig1 width=70% }
```

Здесь:

- в квадратных скобках указывается подпись к изображению;
- в круглых скобках указывается URL-адрес или относительный путь изображения, а также (необязательно) всплывающую подсказку, заключённую в двойные или одиночные кавычки.
- в фигурных скобках указывается идентификатор изображения (`#fig:fig1`) для ссылки на него по тексту и размер изображения относительно ширины страницы (`width=90%`)

Ссылка на изображение (рис. 4.1) может быть оформлена следующим образом (рис. `[-@fig:fig1]`)

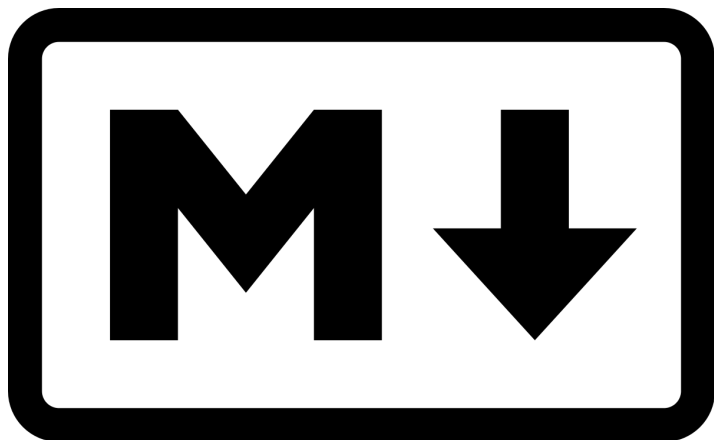


Рис. 4.1. Подпись к рисунку

4.2.4. Обработка файлов в формате Markdown

Преобразовать файл README.md можно следующим образом:

```
pandoc README.md -o README.pdf
```

или так

```
pandoc README.md -o README.docx
```

Для компиляции отчетов по лабораторным работам предлагается использовать следующий Makefile

```
FILES = $(patsubst %.md, %.docx, $(wildcard *.md))  
FILES += $(patsubst %.md, %.pdf, $(wildcard *.md))
```

```
LATEX_FORMAT =  
  
FILTER = --filter pandoc-crossref  
  
%.docx: %.md  
    -pandoc "$<" $(FILTER) -o "$@"  
  
%.pdf: %.md  
    -pandoc "$<" $(LATEX_FORMAT) $(FILTER) -o "$@"  
  
all: $(FILES)  
    @echo $(FILES)  
  
clean:  
    -rm $(FILES) *~
```

4.3. Техническое обеспечение

При выполнении лабораторной работы на своей технике необходимо установить следующее ПО:

- TeX Live (<https://www.tug.org/texlive/>) последней версии.
- Pandoc (<https://pandoc.org/>) версии v2.18
- Pandoc-crossref (<https://github.com/lierdakil/pandoc-crossref/releases>) версии v0.3.13.0

На компьютерах в дисплейных классах факультета физико-математических и естественных наук РУДН все необходимое ПО установлено.

4.4. Порядок выполнения лабораторной работы

1. Откройте терминал
2. Перейдите в каталог курса сформированный при выполнении лабораторной работы №3:

```
cd ~/work/study/2022-2023/"Архитектура компьютера"/arch-pc/
```

Обновите локальный репозиторий, скачав изменения из удаленного репозитория с помощью команды

```
git pull
```

3. Перейдите в каталог с шаблоном отчета по лабораторной работе № 4

```
cd ~/work/study/2022-2023/"Архитектура  
↳ компьютера"/arch-pc/labs/lab04/report
```

4. Проведите компиляцию шаблона с использованием Makefile. Для этого введите команду

```
make
```

При успешной компиляции должны сгенерироваться файлы `report.pdf` и `report.docx`. Откройте и проверьте корректность полученных файлов.

5. Удалите полученные файлы с использованием Makefile. Для этого введите команду

```
make clean
```

Проверьте, что после этой команды файлы `report.pdf` и `report.docx` были удалены.

6. Откройте файл `report.md` с помощью любого текстового редактора, например `gedit`

```
gedit report.md
```

Внимательно изучите структуру этого файла.

7. Заполните отчет и скомпилируйте отчет с использованием Makefile. Проверьте корректность полученных файлов. (Обратите внимание, для корректного отображения скриншотов они должны быть размещены в каталоге image)
8. Загрузите файлы на Github.

```
cd ~/work/study/2022-2023/"Архитектура компьютера"/arch-pc  
git add .  
git commit -am 'feat(main): add files lab-4'  
git push
```

4.5. Задание для самостоятельной работы

1. В соответствующем каталоге сделайте отчёт по лабораторной работе № 3 в формате Markdown. В качестве отчёта необходимо предоставить отчёты в 3 форматах: pdf, docx и md.
2. Загрузите файлы на github.

4.6. Содержание отчёта по выполнению лабораторной работы

- Титульный лист с указанием номера лабораторной работы и ФИО студента.
- Формулировка цели работы.
- Описание результатов выполнения лабораторной работы:
 - описание выполняемого задания;
 - скриншоты (снимки экрана), фиксирующие выполнение заданий лабораторной работы;

- комментарии и выводы по результатам выполнения заданий.
- Описание результатов выполнения заданий для самостоятельной работы:
 - описание выполняемого задания;
 - ссылка на репозиторий Github, содержащий отчёт о выполнении работы №3;
 - комментарии и выводы по результатам выполнения заданий.
- Выводы, согласованные с целью работы.

Отчёт по выполнению лабораторной работы оформляется в формате Markdown. В качестве отчёта необходимо предоставить отчёты в 3 форматах: pdf, docx и md. Файлы необходимо загрузить на странице курса в ТУИС в задание к соответствующей лабораторной работе и загрузить на Github.

4.7. Контрольные вопросы для самопроверки

1. Что такое Markdown?
2. Как в Markdown задается начертание шрифтов?
3. Как в Markdown оформляются списки?
4. Как в Markdown оформляются изображения и ссылки на них?
5. Как в Markdown оформляются математические формулы и ссылки на них?

4.8. Дополнительная информация

4.8.1. Установка TeX Live

1. На странице официального сайта TeX Live <https://www.tug.org/texlive/acquire-netinstall.html> скачать архив `install-tl-unx.tar.gz`.

```
cd /tmp
```

```
wget https://mirror.ctan.org/systems/texlive/tlnet/install-tl-
```

```
↪ unx.tar.gz
```

2. Распаковать архив:

```
zcat install-tl-unx.tar.gz | tar xf -
```

3. Перейти в распакованную папку:

```
cd install-tl-*
```

4. Запустить скрипт install-tl с root правами

```
sudo perl ./install-tl --no-interaction
```

5. Добавить /usr/local/texlive/2022/bin/x86_64-linux в ваш PATH для текущей и будущих сессий.

```
export PATH=$PATH:/usr/local/texlive/2022/bin/x86_64-linux
```

4.8.2. Установка Pandoc и pandoc-crossref

Внимание! Версия pandoc должна соответствовать версии pandoc-crossref.

На странице с релизами pandoc-crossref (<https://github.com/lierdakil/pandoc-crossref/releases>) нужно скачать последнюю версию, и проверить какой версии pandoc она соответствует.

Например, pandoc-crossref версии v0.3.13.0 соответствует версия pandoc v2.18 (рис. 4.2)

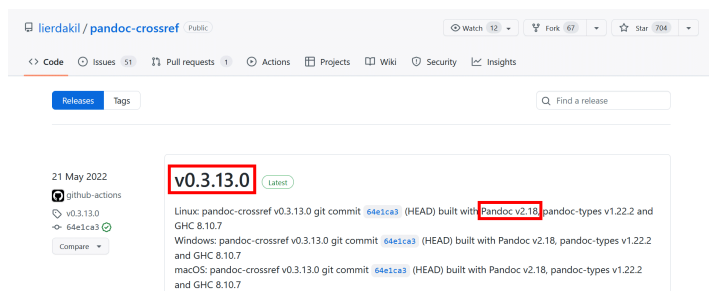


Рис. 4.2. Обратите внимание на соответствие версий pandoc и pandoc-crossref

1. Скачать архивы с исходными файлами.

Скачать архив pandoc (<https://github.com/jgm/pandoc/releases>):

```
wget https://github.com/jgm/pandoc/releases/download/2.19/pandoc-  
↳ 2.19-linux-amd64.tar.gz
```

Скачать архив pandoc-crossref (<https://github.com/lierdakil/pandoc-crossref/releases>):

```
wget https://github.com/lierdakil/pandoc-  
↳ crossref/releases/download/v0.3.13.0/pandoc-crossref-  
↳ Linux.tar.xz
```

2. Распаковать архивы:

```
tar -xf pandoc-2.18-linux-amd64.tar.gz  
tar -xf pandoc-crossref-Linux.tar.xz
```

3. Скопировать файлы pandoc и pandoc-crossref в каталог /usr/local/bin/:

```
sudo cp /tmp/pandoc-2.18/bin/pandoc /usr/local/bin/  
sudo cp /tmp/pandoc-crossref /usr/local/bin/
```

С помощью команды `ls` можно проверить корректность выполненных действий:

```
user@dk4n31:~$ ls /usr/local/bin/  
pandoc  pandoc-crossref  
user@dk4n31:~$
```