

Отчёт по лабораторной работе №3

Межеловский Александр Игоревич

Содержание

1	Цель работы	5
2	Задание	6
3	Теоретическое введение	7
4	Выполнение лабораторной работы	9
5	Выполнение самостоятельной работы	13
6	Выводы	14

Список иллюстраций

3.1	Разметка текста в Markdown (1)	7
3.2	Разметка текста в Markdown (2)	8
4.1	Переход в каталог курса с помощью команды <code>cd</code>	9
4.2	Использование команды <code>git pull</code>	10
4.3	Использование команды <code>make</code>	11
4.4	Использование команды <code>make clean</code>	12
4.5	Использование текстового редактора <code>gedit</code>	12

Список таблиц

1 Цель работы

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

2 Задание

1. Практика создания отчета с помощью Makefile
2. Создание отчета в Markdown
3. Загрузка на Github

3 Теоретическое введение

3.2.1. 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***.
```

Рис. 3.1: Разметка текста в Markdown (1)

3.2.3. 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. 3.1) может быть оформлена следующим образом (рис. [-@fig:fig1])

Рис. 3.2: Разметка текста в Markdown (2)

4 Выполнение лабораторной работы

Перехожу в каталог с курса, куда был клонирован репозиторий.

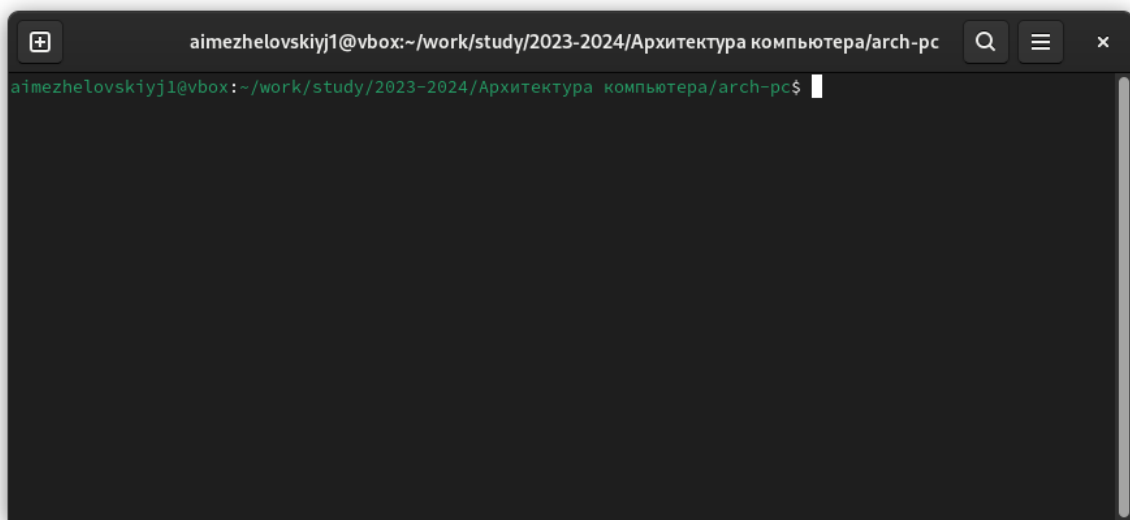


Рис. 4.1: Переход в каталог курса с помощью команды `cd`

Обновляю локальный репозиторий, скачав изменения из удалённого репозитория с помощью команды `git pull`.

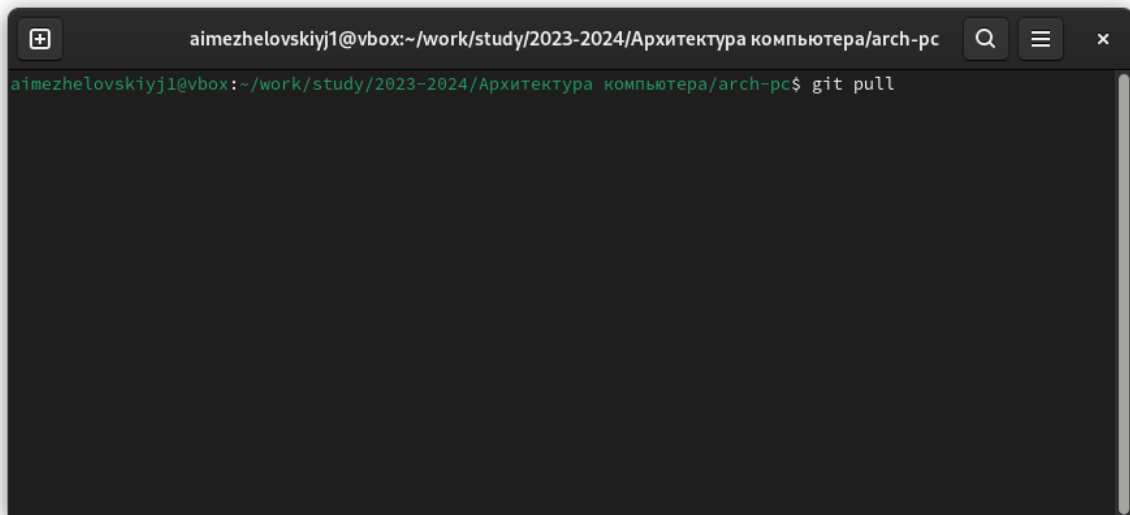
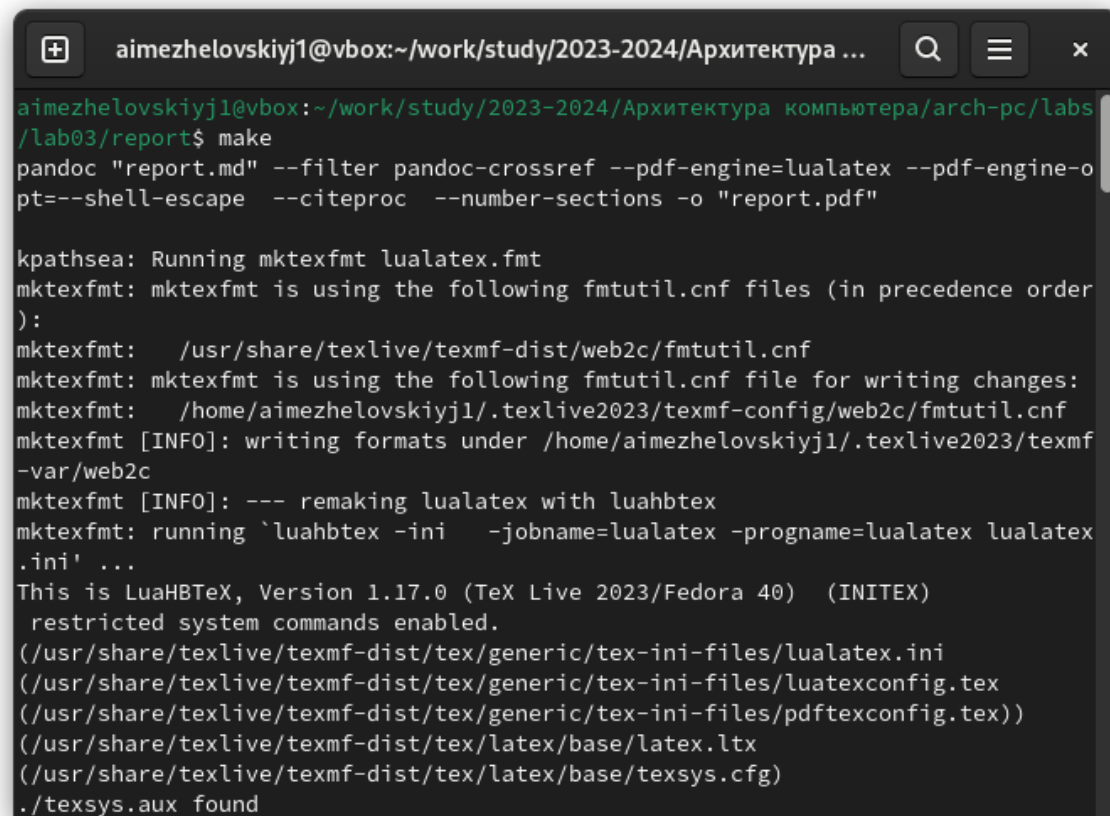


Рис. 4.2: Использование команды git pull

Получаю сообщение “Уже актуально”. Перехожу в каталог с 3 лабораторной работой. Создаю шаблон отчёта с помощью команды make.

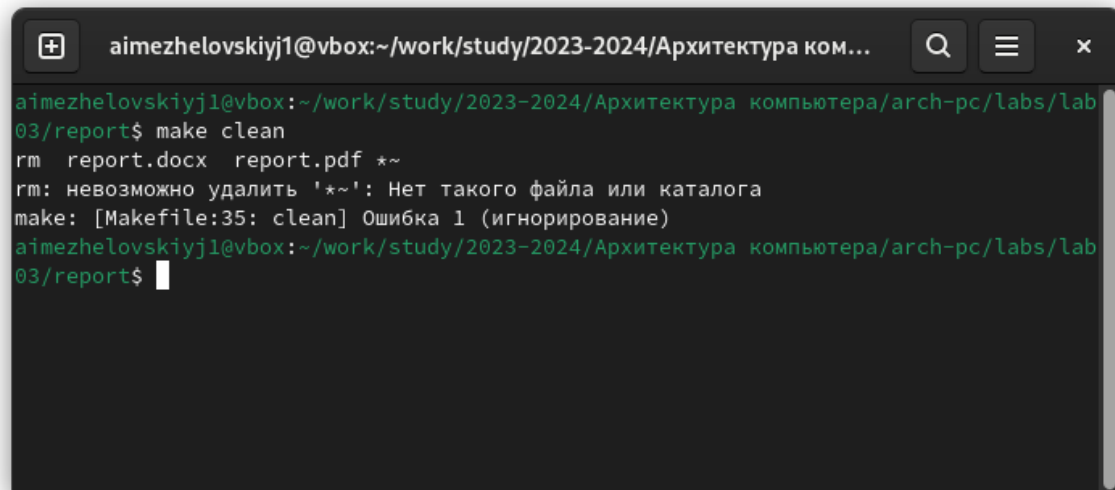


```
aimezhelovskiyj1@vbox:~/work/study/2023-2024/Архитектура ...
/ lab03/report$ make
pandoc "report.md" --filter pandoc-crossref --pdf-engine=lualatex --pdf-engine-o
pt=--shell-escape --citeproc --number-sections -o "report.pdf"

kpathsea: Running mktexfmt luatex.fmt
mktexfmt: mktexfmt is using the following fmtutil.cnf files (in precedence order
):
mktexfmt: /usr/share/texlive/texmf-dist/web2c/fmtutil.cnf
mktexfmt: mktexfmt is using the following fmtutil.cnf file for writing changes:
mktexfmt: /home/aimezhelovskiyj1/.texlive2023/texmf-config/web2c/fmtutil.cnf
mktexfmt [INFO]: writing formats under /home/aimezhelovskiyj1/.texlive2023/texmf
-var/web2c
mktexfmt [INFO]: --- remaking luatex with luahbtex
mktexfmt: running `luahbtex -ini -jobname=lualatex -programe=lualatex lualatex
.ini' ...
This is LuaHBTeX, Version 1.17.0 (TeX Live 2023/Fedora 40) (INITEX)
restricted system commands enabled.
(/usr/share/texlive/texmf-dist/tex/generic/tex-ini-files/luatex.ini
(/usr/share/texlive/texmf-dist/tex/generic/tex-ini-files/luatexconfig.tex
(/usr/share/texlive/texmf-dist/tex/generic/tex-ini-files/pdftexconfig.tex))
(/usr/share/texlive/texmf-dist/tex/latex/base/latex.ltx
(/usr/share/texlive/texmf-dist/tex/latex/base/texsys.cfg)
./texsys.aux found
```

Рис. 4.3: Использование команды make

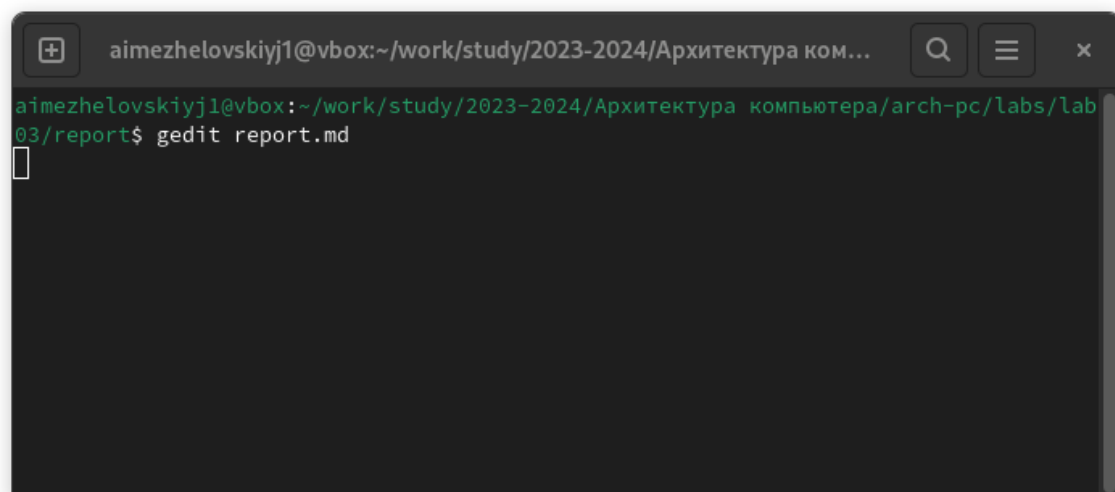
Далее использую команду make clean чтобы удалить файлы report.pdf и report.docx.

A terminal window with a dark background. The title bar shows the user 'aimezhelovskiyj1@vbox' and the current directory '~/work/study/2023-2024/Архитектура ком...'. The terminal text shows the user running 'make clean', which results in 'rm report.docx report.pdf *~' and an error message from 'rm' stating it cannot delete '*' because no file or directory exists. The 'make' command then reports an error and ignores it. The prompt returns to the user.

```
aimezhelovskiyj1@vbox:~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab03/report$ make clean
rm report.docx report.pdf *~
rm: невозможно удалить '*~': Нет такого файла или каталога
make: [Makefile:35: clean] Ошибка 1 (игнорирование)
aimezhelovskiyj1@vbox:~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab03/report$
```

Рис. 4.4: Использование команды make clean

Далее создаю отчет с помощью текстового редактора gedit.

A terminal window with a dark background. The title bar shows the user 'aimezhelovskiyj1@vbox' and the current directory '~/work/study/2023-2024/Архитектура ком...'. The terminal text shows the user running 'gedit report.md', which opens a new window for editing the file. The prompt returns to the user.

```
aimezhelovskiyj1@vbox:~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab03/report$ gedit report.md
```

Рис. 4.5: Использование текстового редактора gedit

5 Выполнение самостоятельной работы

Аналогично этой лабораторной работе, создаю также в markdown отчет по 2 лабораторной работе и использую команду make чтобы создать отчет в форматах pdf и docx.

6 Выводы

В результате проделанной работы я освоил процедуру оформления отчетов с помощью легковесного языка разметки Markdown.