

ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ № 3

дисциплина: Архитектура компьютера

Газизянов Владислав Альбертович

Содержание

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

Список иллюстраций

4.1	каталог курса	9
4.2	каталог с шаблоном отчёта с лабораторной работой №3	9
4.3	корректность	10
4.4	Удалим полученные файлы	10
4.5	Проверка	10
4.6	Открываем файл report.md	11
4.7	Заполним и скомпилируем отчет	12
4.8	корректность	12

1 Цель работы

Изучить основные принципы и синтаксис языка разметки Markdown. Научиться использовать Markdown для создания и оформления текстовых документов различного назначения. Ознакомиться с различными реализациями и диалектами Markdown, а также с инструментами для работы с ним. Сравнить преимущества и недостатки Markdown по сравнению с другими языками разметки, такими как HTML, XML, TeX и т.д. Продемонстрировать полученные знания и навыки на практических примерах.

2 Задание

Освоение процедуры оформления отчетов с помощью легковесного языка разметки Markdown.

3 Теоретическое введение

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 или

link text

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:

```
your code goes in here
```

3.2 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:

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

(3.1) со ссылкой в тексте «Смотри формулу ({-eq. 3.1}).» записывается как

$$\sin^2(x) + \cos^2(x) = 1 \tag{3.1}$$

Смотри формулу ([{-eq: eq1}]) 3.1 Оформление изображений в Markdown В Markdown вставить изображение в документ можно с помощью непосредственного указания адреса изображения. Синтаксис данной команды выглядит следующим образом:

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%`)

3.2Обработка 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, $(wildcard *.md)) LATEX_FORMAT = FILTER =
-- filterpandoc -- crossref<"(FILTER) -- o"@".pdf: %.md -pandoc "$<"
$(LATEX_FORMAT) (FILTER) -- o"@".all: $(FILES) [echo?] $(FILES) clean: -rm
$(FILES) ~
```


4 Выполнение лабораторной работы

Открываем терминал переходим в каталог курса и обновляем репозиторий.

(Рис.1)

```
vagazizyanov@vagazizyanov:~$ cd ~/work/study/2023-2024/"Архитектура компьютера"/arch-pc/
vagazizyanov@vagazizyanov:~/work/study/2023-2024/Архитектура компьютера/arch-pc$ git pull
Already up to date.
```

Рис. 4.1: каталог курса

Переходим в каталог с шаблоном отчёта с лабораторной работой №3 и проводим компиляцию. (Рис.2)

```
vagazizyanov@vagazizyanov:~/work/study/2023-2024/Архитектура компьютера/arch-pc$ cd ~/work/study/2023-2024/"Архитектура компьютера"/arch-pc/labs/lab03/report
vagazizyanov@vagazizyanov:~/work/study/2023-2024/Архитектура компьютера/arch-pc/labs/lab03/report$ make
pandoc "report.md" --filter pandoc/filters/pandoc_fignos.py --filter pandoc/filters/pandoc_eqnos.py --filter pandoc/filters/pandoc_tablenos.py --filter pandoc/filters/pandoc_secnos.py --number-sections --citeproc -o "report.docx"

--main--: Bad reference: @fig:001.
pandoc "report.md" --filter pandoc/filters/pandoc_fignos.py --filter pandoc/filters/pandoc_eqnos.py --filter pandoc/filters/pandoc_tablenos.py --filter pandoc/filters/pandoc_secnos.py --pdf-engine=lualatex --pdf-engine-opt=--shell-escape --citeproc --number-sections -o "report.pdf"

--main--: Bad reference: @fig:001.
```

Рис. 4.2: каталог с шаблоном отчёта с лабораторной работой №3

Проверим корректность полученных файлов.(Рис.3)

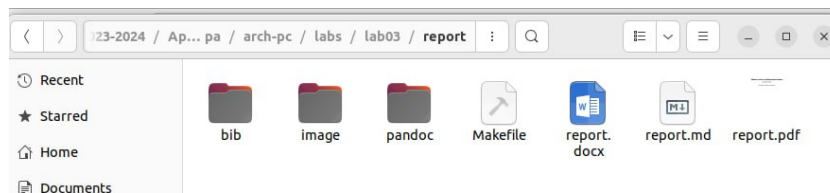


Рис. 4.3: корректность

Удалим полученные файлы с использованием Makefile.(Рис.4)

```
vagazizyanov@vagazizyanov:~/work/study/2023-2024/Архив  
labs/lab03/report$ make clean  
rm report.docx report.pdf *~  
rm: cannot remove '*~': No such file or directory  
make: [Makefile:34: clean] Error 1 (ignored)
```

Рис. 4.4: Удалим полученные файлы

Проверим, что файлы были удалены(Рис.5)

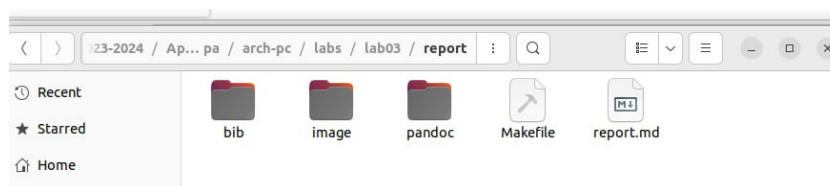
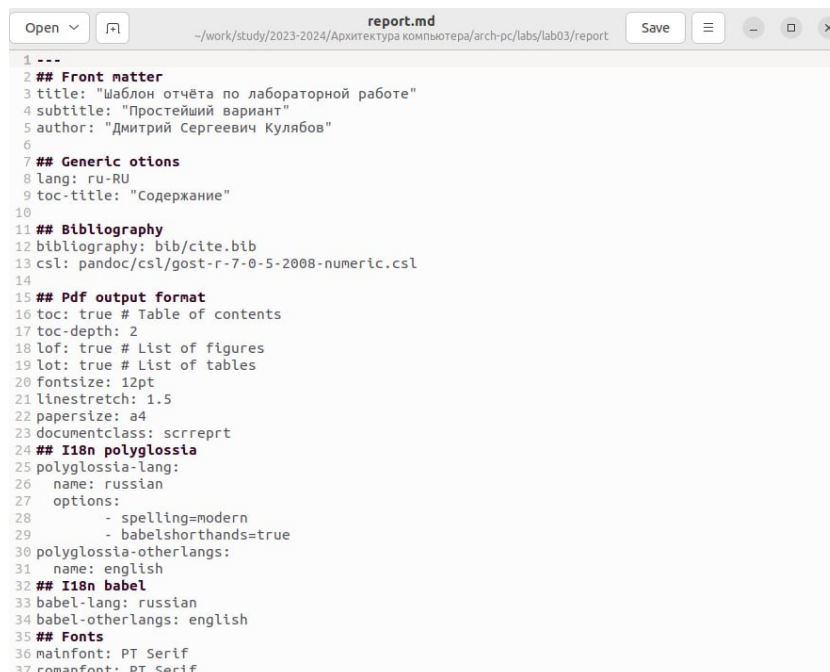


Рис. 4.5: Проверка

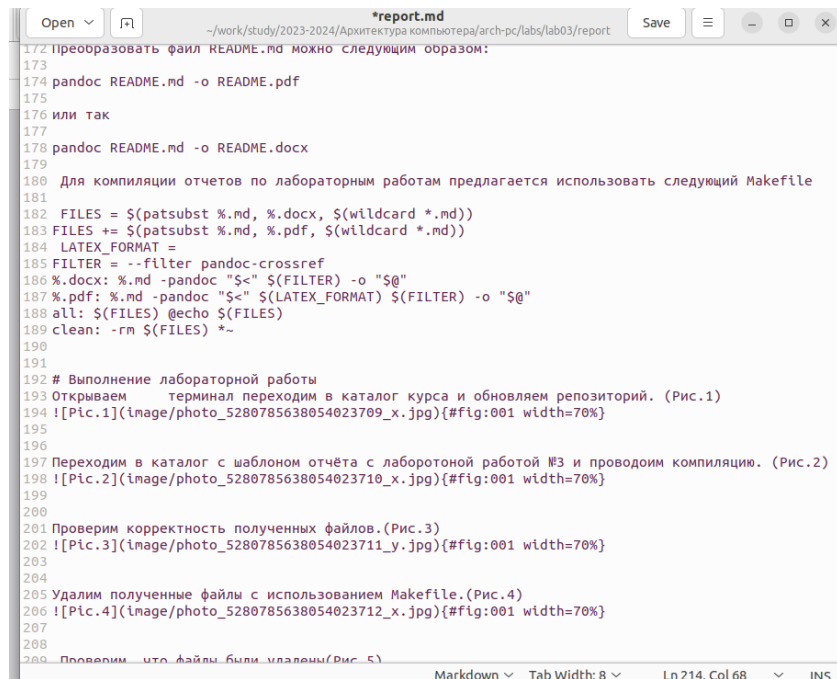
Открываем файл report.md с помощью любого текстового редактора, например gedit(Рис.6)



```
1 ---
2 ## Front matter
3 title: "Шаблон отчёта по лабораторной работе"
4 subtitle: "Простейший вариант"
5 author: "Дмитрий Сергеевич Кулябов"
6
7 ## Generic options
8 lang: ru-RU
9 toc-title: "Содержание"
10
11 ## Bibliography
12 bibliography: bib/cite.bib
13 csl: pandoc/csl/gost-r-7-0-5-2008-numeric.csl
14
15 ## Pdf output format
16 toc: true # Table of contents
17 toc-depth: 2
18 lof: true # List of figures
19 lot: true # List of tables
20 fontsize: 12pt
21 linestretch: 1.5
22 papersize: a4
23 documentclass: scrreprt
24 ## I18n polyglossia
25 polyglossia-lang:
26   name: russian
27   options:
28     - spelling=modern
29     - babelshorthands=true
30 polyglossia-otherlangs:
31   name: english
32 ## I18n babel
33 babel-lang: russian
34 babel-otherlangs: english
35 ## Fonts
36 mainfont: PT Serif
37 romanfont: PT Serif
```

Рис. 4.6: Открываем файл report.md

Внимательно изучаем структуру этого файла. Заполним отчет и скомпилируем отчет с использованием Makefile. (изучаем структуру, Заполним отчет и скомпилируем отчет)



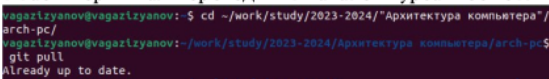
```
172 Преобразовать файл README.md можно следующим образом:
173
174 pandoc README.md -o README.pdf
175
176 или так
177
178 pandoc README.md -o README.docx
179
180 Для компиляции отчетов по лабораторным работам предлагается использовать следующий Makefile
181
182 FILES = $(patsubst %.md, %.docx, $(wildcard *.md))
183 FILES += $(patsubst %.md, %.pdf, $(wildcard *.md))
184 LATEX_FORMAT =
185 FILTER = --filter pandoc-crossref
186 %.docx: %.md -pandoc "$@" $(FILTER) -o "$@"
187 %.pdf: %.md -pandoc "$@" $(LATEX_FORMAT) $(FILTER) -o "$@"
188 all: $(FILES) @echo $(FILES)
189 clean: -rm $(FILES) *~
190
191
192 # Выполнение лабораторной работы
193 Открываем терминал переходим в каталог курса и обновляем репозиторий. (Рис.1)
194 ![Pic.1](image/photo_5280785638054023709_x.jpg){#fig:001 width=70%}
195
196
197 Переходим в каталог с шаблоном отчёта с лабораторной работой №3 и проводим компиляцию. (Рис.2)
198 ![Pic.2](image/photo_5280785638054023710_x.jpg){#fig:001 width=70%}
199
200
201 Проверим корректность полученных файлов. (Рис.3)
202 ![Pic.3](image/photo_5280785638054023711_y.jpg){#fig:001 width=70%}
203
204
205 Удалим полученные файлы с использованием Makefile. (Рис.4)
206 ![Pic.4](image/photo_5280785638054023712_x.jpg){#fig:001 width=70%}
207
208
209 Проверим, что файлы были удалены. (Рис.5)
```

Рис. 4.7: Заполним и скомпилируем отчет

Проверим корректность полученных файлов. (Рис.8)

4 Выполнение лабораторной работы

Открываем терминал переходим в каталог курса и обновляем репозиторий.



```
vagazizyanov@vagazizyanov:~$ cd ~/work/study/2023-2024/Архитектура компьютера/
arch-pc/
vagazizyanov@vagazizyanov:~/work/study/2023-2024/Архитектура компьютера/arch-pc$
git pull
Already up to date.
```

(Рис.1)

Рис. 4.8: корректность

5 Выводы

Язык разметки Markdown является мощным инструментом для создания отформатированных текстовых документов. Он прост в использовании, легко читается и пишется, и может быть преобразован во множество форматов, включая HTML, PDF и др. Markdown особенно полезен для написания технической документации, создания заметок, блогов и комментариев на форумах. Он также широко используется в системах контроля версий, таких как GitHub. В целом, освоение Markdown может значительно упростить процесс оформления текстовых документов и улучшить вашу продуктивность.