
```
## Front matter title: "Отчёт по лабораторной работе 3" subtitle: "Язык
разметки Markdown" author: "Вишняков Родион Сергеевич"
## Generic options lang: ru-RU toc-title: "Содержание"
## Bibliography bibliography: bib/cite.bib csl:
pandoc/csl/gost-r-7-0-5-2008-numeric.csl
## Pdf output format toc: true # Table of contents toc-depth: 2 lof: true # List
of figures lot: true # List of tables fontsize: 12pt linestretch: 1.5 papersize: a4
documentclass: scrreprt ## I18n polyglossia polyglossia-lang: name: russian
options: - spelling=modern - babelshorthands=true polyglossia-otherlangs:
name: english ## I18n babel babel-lang: russian babel-otherlangs: english ##
Fonts mainfont: IBM Plex Serif romanfont: IBM Plex Serif sansfont: IBM
Plex Sans monofont: IBM Plex Mono mathfont: STIX Two Math
mainfontoptions: Ligatures=Common,Ligatures=TeX,Scale=0.94
romanfontoptions: Ligatures=Common,Ligatures=TeX,Scale=0.94
sansfontoptions:
Ligatures=Common,Ligatures=TeX,Scale=MatchLowercase,Scale=0.94
monofontoptions: Scale=MatchLowercase,Scale=0.94,FakeStretch=0.9
mathfontoptions: ## Biblatex biblatex: true biblio-style: "gost-numeric"
biblatexoptions: - parenttracker=true - backend=biber - hyperref=auto -
language=auto - autolang=other* - citestyle=gost-numeric ##
Pandoc-crossref LaTeX customization figureTitle: "Рис." tableTitle:
"Таблица" listingTitle: "Листинг" lofTitle: "Список иллюстраций" lotTitle:
"Список таблиц" lolTitle: "Листинги" ## Misc options indent: true
header-includes: -
```

Цель работы

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

Задание

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.

Теоретическое введение

Маркдаун, он же markdown — удобный и быстрый способ разметки текста. Маркдаун используют, если недоступен HTML, а текст нужно сделать читаемым и хотя бы немного размеченным (заголовки, списки, картинки, ссылки). Главный

пример использования маркдауна, с которым мы часто сталкиваемся — файлы `readme.md`, которые есть в каждом репозитории на Гитхабе. `md` в имени файла это как раз сокращение от `markdown`. Другой частый пример — сообщения в мессенджерах. Можно поставить звёздочки вокруг текста в Телеграме, и текст станет полужирным.

Выполнение лабораторной работы

Установили программы `randos` и `TexLive` по указаниям в лабораторной работе.

1. Откройте терминал
2. Перейдите в каталог курса сформированный при выполнении лабораторной работы №3: Обновите локальный репозиторий, скачав изменения из удаленного репозитория.
3. Перейдите в каталог с шаблоном отчета по лабораторной работе № 3
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. (рис. [-@fig:001], [-@fig:002], [-@fig:003])
5. Удалите полученный файлы с использованием `Makefile`. Для этого введите команду `make clean` Проверьте, что после этой команды файлы `report.pdf` и `report.docx` были удалены. (рис. [-@fig:004])
6. Откройте файл `report.md` с помощью любого текстового редактора, например `gedit` Внимательно изучите структуру этого файла. (рис. [-@fig:005])
7. Заполните отчет и скомпилируйте отчет с использованием `Makefile`. Проверьте корректность полученных файлов. (рис. [-@fig:007], [-@fig:008]) (Обратите внимание, для корректного отображения скриншотов они должны быть размещены в каталоге `image`)
8. Загрузите файлы на Github.

Выводы

Изучили синтаксис языка разметки `Markdown`, получили отчет из шаблона при помощи `Makefile`.

```
[rodvish@vbox report]$ git
usage: git [--version] [--help] [-C <path>] [-c <name>=<value>]
        [--exec-path[=<path>]] [--html-path] [--man-path] [--info-path]
        [-p | --paginate | -P | --no-pager] [--no-replace-objects] [--bare]
        [--git-dir=<path>] [--work-tree=<path>] [--namespace=<name>]
        [--super-prefix=<path>] [--config-env=<name>=<envvar>]
        <command> [<args>]

Стандартные команды Git используемые в различных ситуациях:

создание рабочей области (смотрите также: git help tutorial)
  clone      Клонирование репозитория в новый каталог
  init       Создание пустого репозитория Git или переинициализация существующего

работа с текущими изменениями (смотрите также: git help everyday)
  add        Добавление содержимого файла в индекс
  mv         Перемещение или переименование файла, каталога или символической ссылки
  restore    Restore working tree files
  rm         Удаление файлов из рабочего каталога и индекса

просмотр истории и текущего состояния (смотрите также: git help revisions)
  bisect     Выполнение двоичного поиска коммита, который вносит ошибку
  diff       Вывод разницы между коммитами, коммитом и рабочим каталогом и т.д.
  grep       Вывод строк, соответствующих шаблону
  log        Вывод истории коммитов
  show       Вывод различных типов объектов
  status     Вывод состояния рабочего каталога

выращивание, отметка и настройка вашей общей истории
  branch     Вывод списка, создание или удаление веток
  commit     Запись изменений в репозиторий
  merge      Объединение одной или нескольких историй разработки вместе
  rebase     Повторное применение коммитов над вершущей другой ветки
  reset      Сброс текущего состояния HEAD на указанное состояние
  switch     Switch branches
  tag        Создание, вывод списка, удаление или проверка метки, подписанной с помощью GPG

совместная работа (смотрите также: git help workflows)
  fetch      Загрузка объектов и ссылок из другого репозитория
  pull       Извлечение изменений и объединение с другим репозиторием или локальной веткой
  push       Обновление внешних ссылок и связанных объектов

'git help -a' and 'git help -g' list available subcommands and some
concept guides. See 'git help <command>' or 'git help <concept>'
to read about a specific subcommand or concept.
See 'git help git' for an overview of the system.
```

Figure 1: Make запускает компиляцию

```
[rodvish@vbox ~]$ git config --global user.name "Rodion Vishnyakov"
[rodvish@vbox ~]$ git config --global user.email "1132241588@pfur.ru"
[rodvish@vbox ~]$ git config --global core.quotepath false
[rodvish@vbox ~]$ git config --global init.defaultBranch master
[rodvish@vbox ~]$ git config --global core.autocrlf input
[rodvish@vbox ~]$ git config --global core.safecrlf warn
```

Figure 2: Получен файл в docx

```

[rodvish@vbox ~]$ ssh-keygen -t rsa -b 4096
Generating public/private rsa key pair.
Enter file in which to save the key (/home/rodvish/.ssh/id_rsa):
Created directory '/home/rodvish/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/rodvish/.ssh/id_rsa
Your public key has been saved in /home/rodvish/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:cnM5u39E6wh5y4LY7lRoa7pxs1nTTQh0jpZgLuag4E rodvish@vbox
The key's randomart image is:
+---[RSA 4096]-----+
|      o . . |
|      o o = |
|      . . = . |
|      . + + ... |
| E . . S *.....|
|      o = +o=..oo |
|      . +o..B+o=. . |
|      ..o*.*+.o |
|      o+*o.. |
+-----[SHA256]-----+

```

Figure 3: Получен файл в pdf

```

[rodvish@vbox ~]$ ssh-keygen -t ed25519
Generating public/private ed25519 key pair.
Enter file in which to save the key (/home/rodvish/.ssh/id_ed25519):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/rodvish/.ssh/id_ed25519
Your public key has been saved in /home/rodvish/.ssh/id_ed25519.pub
The key fingerprint is:
SHA256:Fsg7qT6T6Bj0dcuWU3U6POR2vkFe4c6cBUkfu2jgEhw rodvish@vbox
The key's randomart image is:
+---[ED25519 256]--+
|      E      .. |
|      . . . . . oo|
|      o .o + .oo.|
|      o .B + o.o|
|      . .+.So 0 = +.|
|      . . .oo+ o 0 = o|
|      . o.. *      + = |
|      o..+ . .      o |
|      ... .o      . |
+-----[SHA256]-----+

```

Figure 4: Удалены компилированные docx и pdf

```

Ваше полное имя: rodvish
Адрес электронной почты: 1132241588@pfur.ru
Примечание:
Вы выбрали следующий идентификатор пользователя:
"rodvish <1132241588@pfur.ru>"

Сменить (N)Имя, (C)Примечание, (E)Адрес; (O)Принять/(Q)Выход?
Сменить (N)Имя, (C)Примечание, (E)Адрес; (O)Принять/(Q)Выход? o
Необходимо получить много случайных чисел. Желательно, чтобы Вы
в процессе генерации выполняли какие-то другие действия (печать
на клавиатуре, движения мыши, обращения к дискам); это даст генератору
случайных чисел больше возможностей получить достаточное количество энтропии.
Необходимо получить много случайных чисел. Желательно, чтобы Вы
в процессе генерации выполняли какие-то другие действия (печать
на клавиатуре, движения мыши, обращения к дискам); это даст генератору
случайных чисел больше возможностей получить достаточное количество энтропии.
gpg: /home/rodvish/.gnupg/trustdb.gpg: создана таблица доверия
gpg: создан каталог '/home/rodvish/.gnupg/openpgp-revocs.d'
gpg: сертификат отзыва записан в '/home/rodvish/.gnupg/openpgp-revocs.d/0A5072F4155E360A86E61590D70387FEBB76AC70.rev'.
открытый и секретный ключи созданы и подписаны.

pub   rsa4096 2025-09-15 [SC]
      0A5072F4155E360A86E61590D70387FEBB76AC70
uid           [абсолютно] rodvish <1132241588@pfur.ru>
sub   rsa4096 2025-09-15 [E]

```

Figure 5: Шаблон отчета преподавателя

```

[rodvish@vbox ~]$ gpg --list-secret-keys --keyid-format LONG
gpg: проверка таблицы доверия
gpg: marginals needed: 3 completes needed: 1 trust model: pgp
gpg: глубина: 0 достоверных: 1 подписанных: 0 доверие: 0-, 0q, 0n, 0f, 1u
/home/rodvish/.gnupg/pubring.kbx
-----
sec   rsa4096/D70387FEBB76AC70 2025-09-15 [SC]
      0A5072F4155E360A86E61590D70387FEBB76AC70
uid           [абсолютно] rodvish <1132241588@pfur.ru>
ssb   rsa4096/E68AA262319BCA1 2025-09-15 [E]

[rodvish@vbox ~]$ gpg --armor --export D70387FEBB76AC70 | xclip -sel clip

```

Figure 6: Шаблон презентации преподавателя

```

[rodvish@vbox ~]$ gpg --armor --export D70387FEBB76AC70 | xclip -sel clip
[rodvish@vbox ~]$ git config --global user.signingkey D70387FEBB76AC70
[rodvish@vbox ~]$ git config --global commit.gpgsign true
[rodvish@vbox ~]$ git config --global gpg.program $(which gpg2)

```

Figure 7: Заполним шаблон для отчета

```

[rodvish@vbox ~]$ gh auth login
? What account do you want to log into? GitHub.com
? What is your preferred protocol for Git operations? HTTPS
? Authenticate Git with your GitHub credentials? Yes
? How would you like to authenticate GitHub CLI? Login with a web browser

! First copy your one-time code: 9935-E172
Press Enter to open github.com in your browser...
✓ Authentication complete.
- gh config set -h github.com git_protocol https
✓ Configured git protocol
✓ Logged in as rodvish

```

Figure 8: Заполним шаблон для презентации