

# **Отчёт по лабораторной работе №3**

**Markdown**

Грузинова Елизавета Константиновна

# Содержание

<b>1</b>	<b>Цель работы</b>	<b>4</b>
<b>2</b>	<b>Задание</b>	<b>5</b>
<b>3</b>	<b>Теоретическое введение</b>	<b>6</b>
<b>4</b>	<b>Выполнение лабораторной работы</b>	<b>7</b>
<b>5</b>	<b>Выводы</b>	<b>11</b>
	<b>Список литературы</b>	<b>12</b>

## Список иллюстраций

4.1	Открыв шаблон для лабораторной работы, изменила данные об авторе и названии . . . . .	7
4.2	Продублировала цель работы . . . . .	7
4.3	Добавила список с заданием . . . . .	8
4.4	Ввела теоретическое введение . . . . .	8
4.5	Продублировала выполнение лабораторной работы . . . . .	9
4.6	Завершила отчёт выводами и контрольными вопросами с ответами	9
4.7	Преобразование отчёта в трёх форматах . . . . .	10

# 1 Цель работы

Научиться оформлять отчёты с помощью легковесного языка разметки Markdown.

## 2 Задание

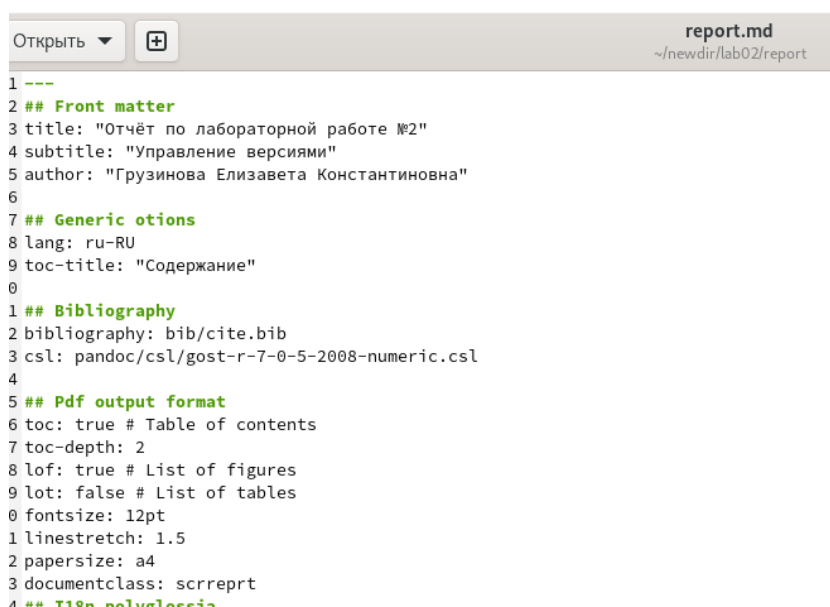
1. Сделайте отчёт по предыдущей лабораторной работе в формате Markdown.
2. В качестве отчёта просьба предоставить отчёты в 3 форматах: pdf, docx и md (в архиве, поскольку он должен содержать скриншоты, Makefile и т.д.)

## 3 Теоретическое введение

Markdown (маркдаун) — это язык упрощённой разметки, который был создан в 2004 году специально для написания наиболее читаемого и удобного для правки текста. При этом markdown может быть перекомпилирован в HTML. Файлы использующие markdown обозначаются как формат .md. [1]

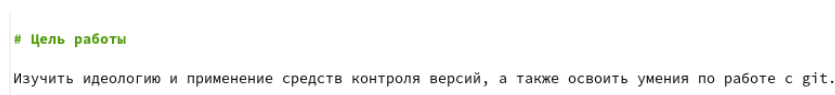
## 4 Выполнение лабораторной работы

1. Сделайте отчёт по предыдущей лабораторной работе в формате Markdown.  
(рис. 4.1, 4.2, 4.3, 4.4, 4.5, 4.6)



```
1 ---
2 ## Front matter
3 title: "Отчёт по лабораторной работе №2"
4 subtitle: "Управление версиями"
5 author: "Грузинова Елизавета Константиновна"
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: false # List of tables
20 fontsize: 12pt
21 linestretch: 1.5
22 papersize: a4
23 documentclass: scrreprt
24 ## Title page class
```

Рис. 4.1: Открыв шаблон для лабораторной работы, изменила данные об авторе и названии



```
# Цель работы
Изучить идеологию и применение средств контроля версий, а также освоить умения по работе с git.
```

Рис. 4.2: Продублировала цель работы

### # Задание

1. Создайте учётную запись на <https://github.com>.
2. Заполните основные данные на <https://github.com>.
3. Установка git-flow в Fedora Linux.
4. Установка gh в Fedora Linux.
5. Базовая настройка git.
6. Создайте ключи ssh.
7. Создайте ключи pgp.
8. Добавление PGP ключа в GitHub.
9. Настройка автоматических подписей коммитов git.
10. Настройка gh.
11. Создание репозитория курса на основе шаблона.
12. Настройка каталога курса.

Рис. 4.3: Добавила список с заданием

### # Теоретическое введение

Системы контроля версий (Version Control System, VCS) применяются при работе нескольких человек над одним проектом. Обычно основное дерево проекта хранится в локальном или удалённом репозитории, к которому настроен доступ для участников проекта. При внесении изменений в содержание проекта система контроля версий позволяет их фиксировать, совмещать изменения, произведённые разными участниками проекта, производить откат к любой более ранней версии проекта, если это требуется.

Рис. 4.4: Ввела теоретическое введение



```

# Выполнение лабораторной работы

1. Создайте учётную запись на https://github.com. (рис. [-@fig:001])

! [Регистрация учетной записи на github.com] (image/1.png) { #fig:001 width=70% }

2. Заполните основные данные на https://github.com. (рис. [-@fig:002])

! [Заполнение основных данных] (image/2.png) { #fig:002 width=70% }

3. Установка git-flow в Fedora Linux. (рис. [-@fig:003;-@fig:004])

! [Ручная установка git-flow] (image/3.png) { #fig:003 width=70% }

! [Git-flow успешно установлен] (image/4.png) { #fig:004 width=70% }

4. Установка gh в Fedora Linux. (рис. [-@fig:005])

! [Команда для установки gh в Fedora Linux] (image/5.png) { #fig:005 width=70% }

```

Рис. 4.5: Продублировала выполнение лабораторной работы

```

# Выводы

В процессе выполнения лабораторной работы я изучила идеологию и применение средств контроля версий, а также освоила умения по работе с git.

# Контрольные вопросы

1. Что такое системы контроля версий (VCS) и для решения каких задач они предназначены?
Система контроля версий – программное обеспечение, которое обеспечивает командную работу в рамках одного или нескольких проектов. Команда разработчиков взаимодействует с консольным или браузерным инструментом для загрузки кода на сервер, скачивания его на рабочий компьютер и изменения структуры. Она хранит все версии проекта и обеспечивает к ним доступ. Любой член команды может взаимодействовать с основной «веткой» проекта или создавать новые.

2. Объясните следующие понятия VCS и их отношения: хранилище, commit, история, рабочая копия.


- Хранилище (repository, сокр. repo), или репозиторий, – место хранения всех версий и служебной информации.
- Версия (revision), или ревизия, – состояние всех файлов на определенный момент времени, сохраненное в репозитории, с дополнительной

```

Рис. 4.6: Завершила отчёт выводами и контрольными вопросами с ответами

2. В качестве отчёта просьба предоставить отчёты в 3 форматах: pdf, docx и md (в архиве, поскольку он должен содержать скриншоты, Makefile и т.д.) (рис. 4.7)

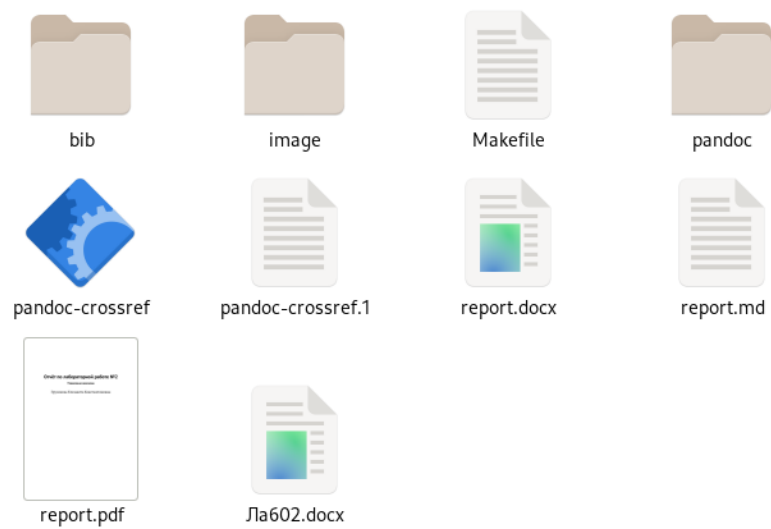


Рис. 4.7: Преобразование отчёта в трёх форматах

## 5 Выводы

При выполнении лабораторной работы научилась оформлять отчёты с помощью легковесного языка разметки Markdown.

# Список литературы

1. Markdown - язык разметки и его синтаксис [Электронный ресурс]. URL: <https://v3c.ru/obo-vsjom/markdown>.