

# Markdown

## Лабораторная работа № 3

---

Петросян Э. М.

23 февраля 2023

Российский университет дружбы народов, Москва, Россия

## Информация

---

- Петросян Эмиль Манукович
- студент 1 курса, группа НММбд-03-22
- Российский университет дружбы народов



## Вводная часть

---

- pandoc
- markdown

- Научиться оформлять отчёты с помощью легковесного языка разметки Markdown.

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

## Выполнение лабораторной работы №3

---



```
---  
## Front matter  
title: "Отчёт по лабораторной работе № 2"  
subtitle: "Операционные системы"  
author: "Петросян Эмиль Манукович"
```

## # Цель работы

- Изучить идеологию и применение средств контроля версий.
- Освоить умения по работе с `git`.

## # Задание

- Установить и настроить ПО для работы с `git`.

# Структурная составляющая отчета

## # Выполнение лабораторной работы

### ## Установка программного обеспечения

Установили git: (рис. [-efig:001])

![-](image/1.png) (#fig:001 width=70%)

Установили gh: (рис. [-efig:002])

![-](image/2.png) (#fig:002 width=70%)

### ## Базовая настройка git

Задали имя и email владельца репозитория: (рис. [-efig:003])

![-](image/3.png) (#fig:003 width=70%)

Настроили utf-8 в выводе сообщений git: (рис. [-efig:004])

![-](image/4.png) (#fig:004 width=70%)

Настроили верификацию и подписание `commit` git.

Задали имя начальной ветки (будет называться «`master`»): (рис. [-efig:005])

![-](image/5.png) (#fig:005 width=70%)

Параметр `autocrlf`: (рис. [-efig:006])

![-](image/6.png) (#fig:006 width=70%)

Параметр `safecrlf`: (рис. [-efig:007])

## # Выводы

В ходе выполнения данной лабораторной работы была изучена идеология и применение средств контроля версий и основы умения по работе с git.

## # Ответы на контрольные вопросы

1. Что такое системы контроля версий (VCS) и для решения каких задач они предназначаются? Система управления версиями (также используется определение «система контроля версий», от англ. *Version Control System*, VCS или *Revision Control System*) – программное обеспечение для обеспечения работы с изменяющейся информацией. Система управления версиями позволяет хранить несколько версий одного и того же документа, при необходимости возвращаться к более ранним версиям, определять, кто и когда сделал то или иное изменение, и многое другое.

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

Хранилище (*repository*), или *репозиторий*, – место хранения файлов и их версий, служебной информации.

Версия (*revision*), или *ревизия*, – состояние всего хранилища или отдельных файлов в конкретное время («пункт истории»).

`Commit` («трудовой вклад», не переводится) – процесс создания новой версии; иногда синоним версии.

Рабочая копия (*working copy*) – текущее состояние файлов проекта (любых версий), полученных из хранилища и, возможно, измененных.

3. Что представляет собой и чем отличается централизованное и децентрализованное VCS? Приведите примеры VCS каждого вида.

Децентрализованное VCS: у каждого пользователя свой вариант (возможно не один) репозитория

Присутствует возможность добавлять и забирать изменения из любого *репозитория*

(Git, Mercurial, Bazaar)

```
## 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
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: PT Serif
romanfont: PT Serif
sansfont: PT Sans
monofont: PT Mono
mainfontoptions: Ligatures=TeX
romanfontoptions: Ligatures=TeX
sansfontoptions: Ligatures=TeX,Scale=MatchLowercase
monofontoptions: Scale=MatchLowercase,Scale=0.9

## 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"
lolTitle: "Листинги"

## Misc options
indent: true
header-includes:
  - \usepackage[indentfirst]
  - \usepackage(float) # keep figures where there are in the text
  - \floatplacement{figure}{H} # keep figures where there are in the text
---
```

Установили gh:(рис. [-@fig:002])

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

## # Задание

- Установить и настроить ПО для работы с `git`.

### # Выполнение лабораторной работы

#### ## Установка программного обеспечения

Установили `git`: (рис. [-@fig:001])

 { #fig:001 width=70%}

Установили `gh`: (рис. [-@fig:002])

 { #fig:002 width=70%}

#### ## Базовая настройка `git`

Задали имя и `email` владельца репозитория: (рис. [-@fig:003])

 { #fig:003 width=70%}

Настроили `utf-8` в выводе сообщений `git`: (рис. [-@fig:004])

 { #fig:004 width=70%}

Настроили верификацию и подписание коммитов `git`.

Задали имя начальной ветки (будем называть её `master`). (рис. [-@fig:005])

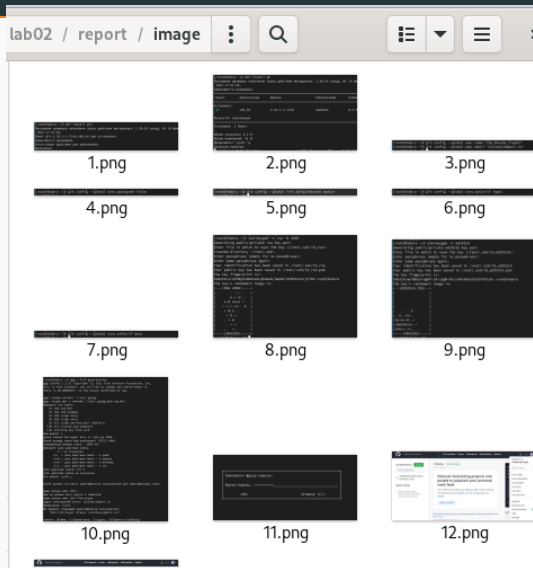
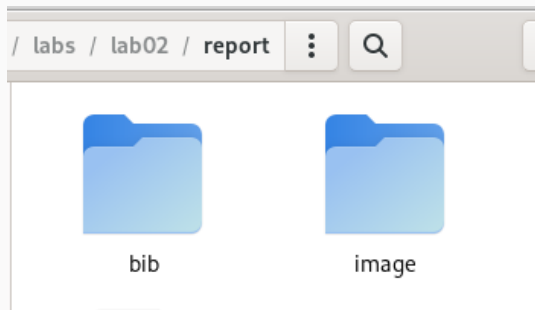
 { #fig:005 width=70%}

Параметр `autocrlf`: (рис. [-@fig:006])

 { #fig:006 width=70%}

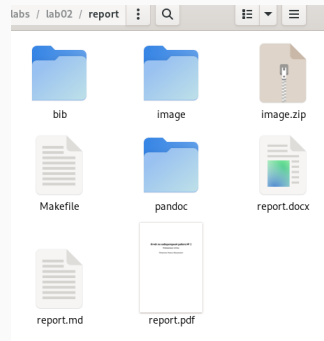
Параметр `safecrlf`: (рис. [-@fig:007])

# Техническая составляющая отчета



```
[empetrosoyan@fedora ~]$ cd ~/work/study/2022-2023/"Операционные системы"/study_1  
2022-2023_os-intro/labs/lab02
```

```
[empetrosoyan@fedora lab02]$ make
```



## Результаты

---

В ходе выполнения лабораторной работы были изучены способы оформления отчётов с помощью легковесного языка разметки Markdown.