

Лабораторная работа №3

Дисциплина: Операционные системы

Паращенко Антонина Дмитриевна

Содержание

1	Цель работы	5
2	Ход лабораторной работы:	6
3	Вывод:	11

List of Figures

2.1	Оформление основы отчёта	6
2.2	Заголовки	7
2.3	Ход работы	7
2.4	Синтаксис вставки изображений	8
2.5	Вывод и контрольные вопросы	8
2.6	Конвертация	8
2.7	Формат docx	9
2.8	Формат pdf	10

List of Tables

1 Цель работы

Научиться оформлять отчёты с помощью легковесного языка разметки Markdown.

2 Ход лабораторной работы:

1) Используем шаблон из github

```
---  
# Front matter  
lang: ru-RU  
title: "Лабораторная работа №2"  
subtitle: "Дисциплина: Операционные системы"  
author: "Парашенко Антонина Дмитриевна"  
  
# Formatting  
toc-title: "Содержание"  
toc: true # Table of contents  
toc_depth: 2  
lof: true # List of figures  
lot: true # List of tables  
fontsize: 12pt  
linestretch: 1.5  
papersize: a4paper  
documentclass: scrreprt  
polyglossia-lang: russian  
polyglossia-otherlangs: english  
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  
indent: true  
pdf-engine: lualatex  
header-includes:  
  - \linepenalty=10 # the penalty added to the badness of each line within a paragraph (no associated  
penalty node) Increasing the value makes tex try to have fewer lines in the paragraph.  
  - \interlinepenalty=0 # value of the penalty (node) added after each line of a paragraph.  
  - \hyphenpenalty=50 # the penalty for line breaking at an automatically inserted hyphen  
  - \exhyphenpenalty=50 # the penalty for line breaking at an explicit hyphen  
  - \binoppenalty=700 # the penalty for breaking a line at a binary operator  
  - \relpenalty=500 # the penalty for breaking a line at a relation  
  - \clubpenalty=150 # extra penalty for breaking after first line of a paragraph  
  - \widowpenalty=150 # extra penalty for breaking before last line of a paragraph  
  - \displaywidowpenalty=50 # extra penalty for breaking before last line before a display math  
  - \brokenpenalty=100 # extra penalty for page breaking after a hyphenated line  
  - \prelispdisplaypenalty=10000 # penalty for breaking before a display  
  - \postdisplaypenalty=0 # penalty for breaking after a display  
  - \floatingpenalty = 20000 # penalty for splitting an insertion (can only be split footnote in  
standard LaTeX)
```

Figure 2.1: Оформление основы отчёта

2) Создаём заголовки

Цель работы

Изучить идеологию и применени

Ход лабораторной работы:

Figure 2.2: Заголовки

3) Прописываем ход выполнения лабораторной работы

```
# Ход лабораторной работы:
1) Настройка github
![[Аккаунт на github](image/1.png){ #fig:001 width=70% }
2) Задаём основную информацию владельца репозитория
![[Имя](image/2.png){ #fig:002 width=70% }
![[Базовая настройка git](image/3.png){ #fig:003 width=70% }
3) Создание ключа ssh
![[Генерация и подключение ключа SSH](image/4.png){ #fig:004 width=70% }
4) Создание ключа gpg
![[Генерация следа](image/5.png){ #fig:005 width=70% }
![[Ключ](image/6.png){ #fig:006 width=70% }
![[Генерация и подключение ключа GPG](image/7.png){ #fig:007 width=70% }
5) Переносим репозиторий из шаблона
![[Команды переноса](image/8.png){ #fig:008 width=70% }
![[Создание репозитория курса на основе шаблона](image/9.png){ #fig:009 width=70% }
6) Настройка каталога курса
![[Перенос с компьютера в github](image/10.png){ #fig:010 width=70% }
![[Отправка файлов на сервер](image/11.png){ #fig:011 width=70% }
# Вывод:
```

Figure 2.3: Ход работы

4) Вставка изображений в отчёт

```
![[Генерация следа](image/5.png){ #fig:005 width=70% }
![[Ключ](image/6.png){ #fig:006 width=70% }
![[Генерация и подключение ключа GPG](image/7.png){ #fig:007 width=70% }
```

Figure 2.4: Синтаксис вставки изображений

5) Оформляем вывод и контрольные вопросы

```
# Вывод:
Изучила идеологию и применение средств контроля версий, а также освоила команды по работе с git.

# Контрольные вопросы:
1. Системы контроля версий (Version Control System, VCS) применяются при работе нескольких человек над одним проектом. 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.
2. -
3. Централизованная система (CVS, Subversion) предполагает наличие единого репозитория для хранения файлов, однако для децентрализованных систем (Git, Bazaar, Mercurial) это необязательно.
4. Предварительная конфигурация, настроить utf-8, инициализация локального репозитория.
5. Работа пользователя со своей веткой начинается с проверки и получения изменений из центрального репозитория, затем можно вносить изменения в локальном дереве и/или ветке.
6. Благодаря тому, что Git является распределённой системой контроля версий, резервную копию локального хранилища можно сделать простым копированием или архивацией.
7. Система контроля версий Git представляет собой набор программ командной строки. Доступ к ним можно получить из терминала посредством ввода команды git с различными опциями.
8. Участник проекта (пользователь) перед началом работы посредством определённых команд получает нужную ему версию файлов. После внесения изменений, пользователь размещает новую версию в хранилище. При этом предыдущие версии не удаляются из центрального хранилища и к ним можно вернуться в любой
```

Figure 2.5: Вывод и контрольные вопросы

6) С помощью makefile и консоли прописываем команды, чтобы сконвертировать файл типа md в типы pdf и docx

```
adparathenko@dk6n61: ~/work/study/2021-2022/Операционные системы/os-intro/labs/lab03/report $ make
pandoc "report.md" -F pandoc-crossref --number-sections --citeproc -o "report.docx"
pandoc "report.md" -F pandoc-crossref --pdf-engine=lualatex --pdf-engine-opt=--shell-escape --citeproc --number-sections -o "report.pdf"
```

Figure 2.6: Конвертация

7) Проверяем файл в pdf и docx форматах

Лабораторная работа №2

Дисциплина: Операционные системы

Паращенко Ангелина Дмитриевна

Содержание

1 Цель работы

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

2 Ход лабораторной работы:

1) Настройка `github`

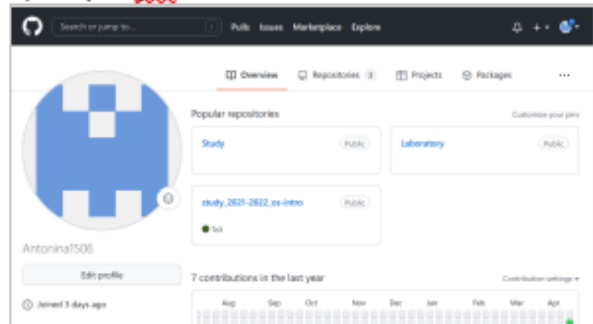


Figure 1: Аккаунт на `github`

2) Задаём основную информацию владельца репозитория

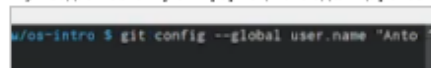


Figure 2: Имя

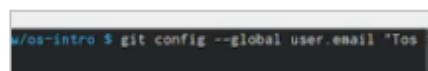


Figure 3: Базовая настройка `git`

3) Создание ключа `ssh`

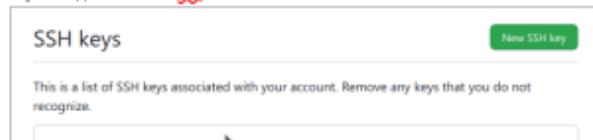


Figure 2.7: Формат docx

2 Ход лабораторной работы:

1) Настройка github

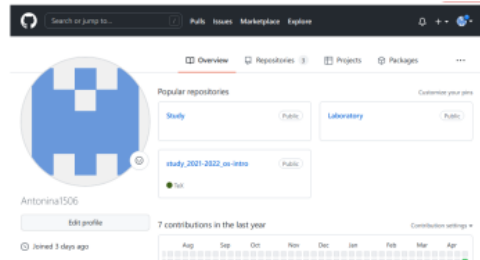


Figure 2.1: Аккаунт на github

2) Задаём основную информацию владельца репозитория



Figure 2.2: Имя

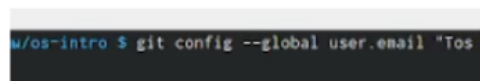


Figure 2.3: Базовая настройка git

6

Figure 2.8: Формат pdf

3 Вывод:

Научилась оформлять отчёты с помощью легковесного языка разметки Markdown и в качестве практики оформила отчёт к лабораторной работе №2.