

# **Шаблон отчёта по лабораторной работе 2**

**Markdown**

Валерия Дельгадильо

# Содержание

Цель работы	5
2 Выполнение лабораторной работы	6
Лабораторной работы	7
Контрольные вопросы.	20
Выводы	24
Список литературы	25

## Список иллюстраций

1	Базовая настройка git . . . . .	7
2	Создание ключа по алгоритму rsa с ключём размером 4096 бит . .	7
3	Создание ключа по алгоритму ed25519 . . . . .	8
4	Команда начала генерации ключа . . . . .	8
5	Настройка параметров . . . . .	9
6	Настройка параметров . . . . .	9
7	Создание пароля . . . . .	10
8	Генерация . . . . .	10
9	Вывод списка ключей . . . . .	11
10	Загрузка ключа . . . . .	11
11	Загрузка ключа . . . . .	12
12	Итог . . . . .	12
13	Настройка автоматических подписей коммитов git . . . . .	13
14	Авторизация . . . . .	13
15	Подключение к браузеру . . . . .	14
16	Итог . . . . .	14
17	Создание репозитория на основе шаблона . . . . .	15
18	Репозиторий . . . . .	16
19	Авторизация . . . . .	16
20	Подключение к браузеру . . . . .	17
21	Итог . . . . .	17
22	Переход в каталог, удаление лишних файлов, создание каталогов .	18
23	Отправка файлов на сервер . . . . .	18
24	Отправка файлов на сервер . . . . .	18
25	Отправка файлов на сервер . . . . .	19

## Список таблиц

## Цель работы

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

## 2 Выполнение лабораторной работы

Всю практическую часть я выполнял в первом семестре и детально изложил в файле `report_old.pdf`. Здесь посмотрим только на основные вехи: 1. установка пакета `git` на компьютер 2. Создание аккаунта на `github.com` 3. Создание и подключение `ssh`-ключа 4. создание и развертка репозитория из шаблона.

# Лабораторной работы

1. Базовая настройка git (рис. 4.1).

```
Операционные системы]$ cd ~  
~]$ git config --global user.name "yveljgadiljo"  
~]$ git config --global user.email "yveljgadiljo@yandex.ru"  
~]$ git config --global core.quotepath false  
~]$ git config --global init.defaultBranch master  
~]$ git config --global core.autocrlf input  
~]$ git config --global core.safecrlf warn  
~]$ ssh-keygen -t rsa -b 4096
```

Рис. 1: Базовая настройка git

2. Создание SSH ключа (рис. 4.2, 4.3).

```
[yveljgadiljo@fedora ~]$ ssh-keygen -t rsa -b 4096  
Generating public/private rsa key pair.  
Enter file in which to save the key (/home/yveljgadiljo/.ssh/id_rsa): ~/.ssh/  
Enter passphrase (empty for no passphrase):  
Enter same passphrase again:  
Saving key "~/.ssh/" failed: No such file or directory  
[yveljgadiljo@fedora ~]$ ssh-keygen -t ed25519  
Generating public/private ed25519 key pair.  
Enter file in which to save the key (/home/yveljgadiljo/.ssh/id_ed25519):  
Created directory '/home/yveljgadiljo/.ssh'.  
Enter passphrase (empty for no passphrase):
```

Рис. 2: Создание ключа по алгоритму rsa с ключём размером 4096 бит

```
[yvdeljgadiljo@fedora ~]$ ssh-keygen -t ed25519
Generating public/private ed25519 key pair.
Enter file in which to save the key (/home/yvdeljgadiljo/.ssh/id_ed25519):
Created directory '/home/yvdeljgadiljo/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/yvdeljgadiljo/.ssh/id_ed25519
Your public key has been saved in /home/yvdeljgadiljo/.ssh/id_ed25519.pub
The key fingerprint is:
SHA256:05mj+4tbavXuWmK6n1NcvH/NN5ER2y+94V1KCL20V14 yvdeljgadiljo@fedora
The key's randomart image is:
+--[ED25519 256]--+
|
|      .
|      . . +
|      ..oo ooE|
|      S =+ = ==|
|      o .+ =+=|
|      . =.o.=B|
|      X.= .+B|
|      0=+++ +|
+-----[SHA256]-----+
[yvdeljgadiljo@fedora ~]$
```

Рис. 3: Создание ключа по алгоритму ed25519

### 3. Создание PGP ключа (рис. 4.4, 4.5, 4.6, 4.7, 4.8, 4.9).

```
[yvdeljgadiljo@fedora ~]$ gpg --full-generate-key
gpg (GnuPG) 2.4.0; Copyright (C) 2021 Free Software Foundation, Inc.
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.

gpg: создан каталог '/home/yvdeljgadiljo/.gnupg'
gpg: создан щит с ключами '/home/yvdeljgadiljo/.gnupg/pubring.kbx'
Выберите тип ключа:
  (1) RSA and RSA
  (2) DSA and Elgamal
  (3) DSA (sign only)
  (4) RSA (sign only)
  (9) ECC (sign and encrypt) +default+
  (10) ECC (только для подписи)
  (14) Existing key from card
Ваш выбор? 1
длина ключей RSA может быть от 1024 до 4096.
Какой размер ключа Вам необходим? (3072) 4096
```

Рис. 4: Команда начала генерации ключа



```

Выберите тип ключа:
(1) RSA and RSA
(2) DSA and Elgamal
(3) DSA (sign only)
(4) RSA (sign only)
(9) ECC (sign and encrypt) +default+
(10) ECC (только для подписи)
(14) Existing key from card
Ваш выбор? 1
длина ключей RSA может быть от 1024 до 4096.
Какой размер ключа Вам необходим? (3072) 4096
Запрошенный размер ключа - 4096 бит
Выберите срок действия ключа.
    0 = не ограничен
    <n> = срок действия ключа - n дней
    <n>w = срок действия ключа - n недель
    <n>m = срок действия ключа - n месяцев
    <n>y = срок действия ключа - n лет
Срок действия ключа? (0) 0
Срок действия ключа не ограничен
Все верно? (y/N) y

GnuPG должен составить идентификатор пользователя для идентификации ключа.
Ваше полное имя: 

```

Рис. 5: Настройка параметров

```

(14) Existing key from card
Ваш выбор? 1
длина ключей RSA может быть от 1024 до 4096.
Какой размер ключа Вам необходим? (3072) 4096
Запрошенный размер ключа - 4096 бит
Выберите срок действия ключа.
    0 = не ограничен
    <n> = срок действия ключа - n дней
    <n>w = срок действия ключа - n недель
    <n>m = срок действия ключа - n месяцев
    <n>y = срок действия ключа - n лет
Срок действия ключа? (0) 0
Срок действия ключа не ограничен
Все верно? (y/N) y

GnuPG должен составить идентификатор пользователя для идентификации ключа.

Ваше полное имя: Valeria Delgadillo
Адрес электронной почты: 1032229098@pfur.ru
Примечание:
Вы выбрали следующий идентификатор пользователя:
    "Valeria Delgadillo <1032229098@pfur.ru>"

Сменить (N)Имя, (C)Примечание, (E)Адрес; (0)Принять/(Q)Выход? 

```

Рис. 6: Настройка параметров

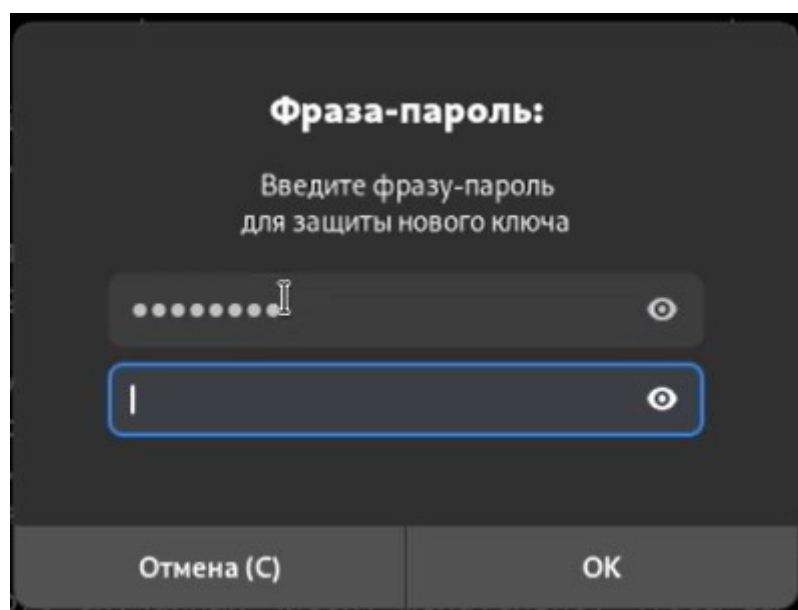


Рис. 7: Создание пароля

```

Вы выбрали следующий идентификатор пользователя:
"Valeria Delgadillo <1032229098@pfur.ru>"

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/yvdeljgadillo/.gnupg/trustdb.gpg: создана таблица доверия
gpg: создан каталог '/home/yvdeljgadillo/.gnupg/openpgp-revocs.d'
gpg: сертификат отзыва записан в '/home/yvdeljgadillo/.gnupg/openpgp-revocs.d/9F9987D3E684
3632D2C9B1FF99DAAEE096857DCD.rev'.
открытый и секретный ключи созданы и подписаны.

pub  rsa4096 2024-02-19 [SC]
     9F9987D3E6843632D2C9B1FF99DAAEE096857DCD
uid                Valeria Delgadillo <1032229098@pfur.ru>
sub  rsa4096 2024-02-19 [E]

[yvdeljgadillo@fedora ~]$

```

Рис. 8: Генерация

```
[yvdeljgadiljo@fedora ~]$ 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, 0w, 0f, 1u
/home/yvdeljgadiljo/.gnupg/pubring.kbx
-----
sec   rsa4096/99DAEE096857DCD 2024-02-19 [SC]
      9F9987D3E6843632D2C9B1FF99DAEE096857DCD
uid    [ абсолютно ] Valeria Delgadillo <1032229098@pfur.ru>
ssb    rsa4096/6DA480DB5978A7E3 2024-02-19 [E]
```

Рис. 9: Вывод списка ключей

4. Добавление PGP ключа в GitHub (рис. 4.10, 4.11, 4.12).

## GPG keys

New GPG 

There are no GPG keys associated with your account.

Learn how to [generate a GPG key and add it to your account](#).

Рис. 10: Загрузка ключа

## GPG keys / Add new

Title

gpg ts[anina p]

Key

```
-----BEGIN PGP PUBLIC KEY BLOCK-----
mQINBzTLRchFjaM6klwwCu5zS53G6hOwcLCJNtMfKynOpm7c36CaycqQVJPA/zn83Zy
QEPzRNVSnr28wWSUpI7zvr9UOzu2sh4Wk0b9upV1YZ5gW2JIo6qX00hxVFZCZ8Yd
hLQzPCP7JhXaBNOj9CP8+zGKh4+KEhxDPG2mO+dULTZuIn3zpgpp+WfXkCfHmRjW
jw==
=Ufnf
-----END PGP PUBLIC KEY BLOCK-----
```


Add GPG key

Рис. 11: Загрузка ключа

## GPG keys

New GPG key

This is a list of GPG keys associated with your account. Remove any keys that you do not recognize.

  
GPG

**GPG Valeria**

Email address: 1032229098@pfur.ru

Key ID: 99DAAEE096857DCD

Subkeys: 6DA40DD85978A7E3

Added on Feb 19, 2024

Delete

Рис. 12: Итог

5. Настройка автоматических подписей коммитов git (рис. 4.13).

```
[yvdeljgadiljo@fedora ~]$ git config --global user.signingkey 99DAAEE0968570CD
[yvdeljgadiljo@fedora ~]$ git config --global commit.gpgsign true
[yvdeljgadiljo@fedora ~]$ git config --global gpg.program $(which gpg2)
[yvdeljgadiljo@fedora ~]$ gh auth login
```

Рис. 13: Настройка автоматических подписей коммитов git

6. Настройка gh (рис. 4.14, 4.15, 4.16).

```
[yvdeljgadiljo@fedora Операционные системы]$ 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? No
? How would you like to authenticate GitHub CLI? Paste an authentication token
Tip: you can generate a Personal Access Token here https://github.com/settings/tokens
The minimum required scopes are 'repo', 'read:org'.
? Paste your authentication token:

[yvdeljgadiljo@fedora Операционные системы]$ 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: 2484-B717
```

Рис. 14: Авторизация

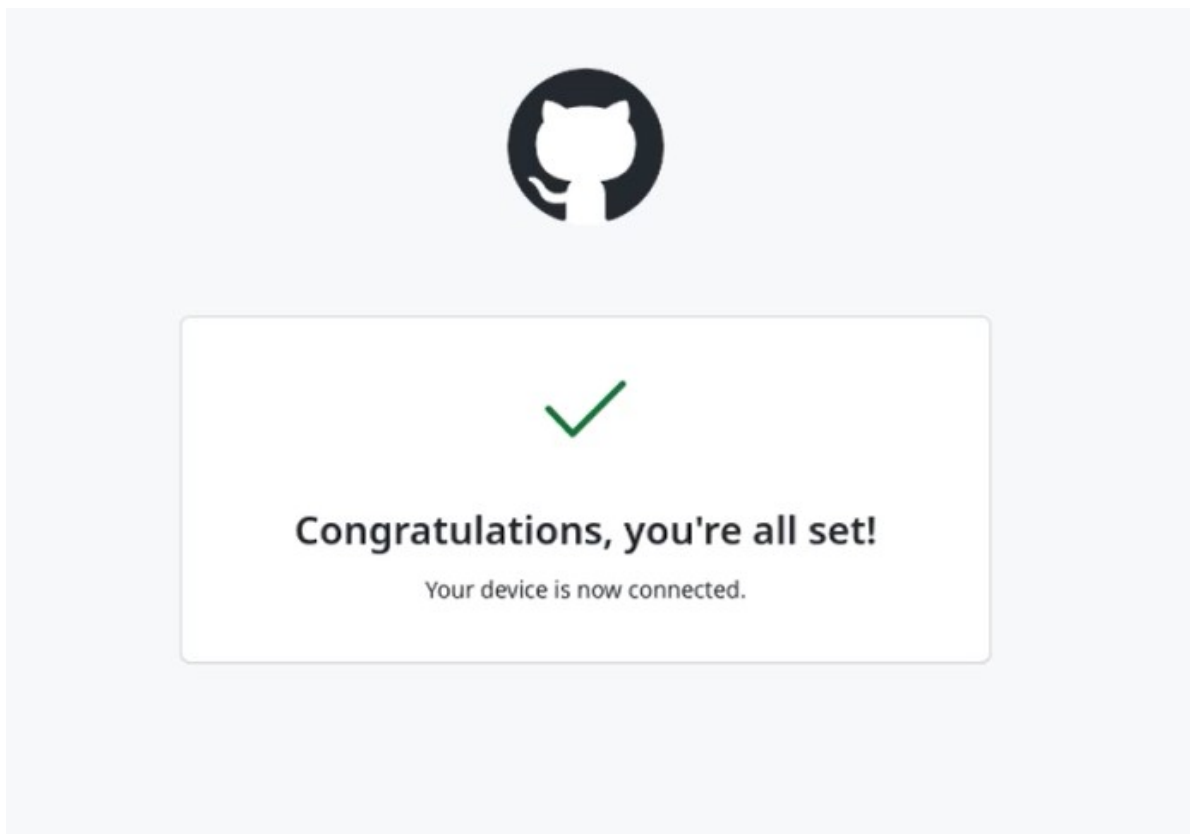


Рис. 15: Подключение к браузеру

```
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 yvdeljgadiljo
```

Рис. 16: Итог

7. Шаблон для рабочего пространства (рис. 4.17, 4.18).

## Create a new repository from course-directory-student-template

The new repository will start with the same files and folders as [yamadharm/course-directory-student-template](#).

Owner \*

 tsganina ▾

Repository name \*

study\_2022-2023\_os-intro ✓

Great repository names are short and memorable. Need inspiration? How about [stunning-guide?](#)

Description (optional)

☒  **Public**

Anyone on the internet can see this repository. You choose who can commit.

☐  **Private**

You choose who can see and commit to this repository.

☐ **Include all branches**

Copy all branches from yamadharm/course-directory-student-template and not just master.

 You are creating a public repository in your personal account.


Creating repository 

Рис. 17: Создание репозитория на основе шаблона

master	1 branch	0 tags	Go to file	Add file	Code
tsganina Initial commit			c788d86	now	1 commit
config	Initial commit	now			
template	Initial commit	now			
.gitattributes	Initial commit	now			
.gitignore	Initial commit	now			
.gitmodules	Initial commit	now			
CHANGELOG.md	Initial commit	now			
COURSE	Initial commit	now			
LICENSE	Initial commit	now			
Makefile	Initial commit	now			
README.en.md	Initial commit	now			
README.git-flow.md	Initial commit	now			
README.md	Initial commit	now			
package.json	Initial commit	now			

Рис. 18: Репозиторий

- Создание локального репозитория курса на основе шаблона (рис.4.19, 4.20, 4.21).

```
[yvdeljgadi@fedora ~]$ mkdir -p ~/work/study/2023-2024/"Операционные системы"
[yvdeljgadi@fedora ~]$ cd ~/work/study/2023-2024/"Операционные системы"
```

Рис. 19: Авторизация



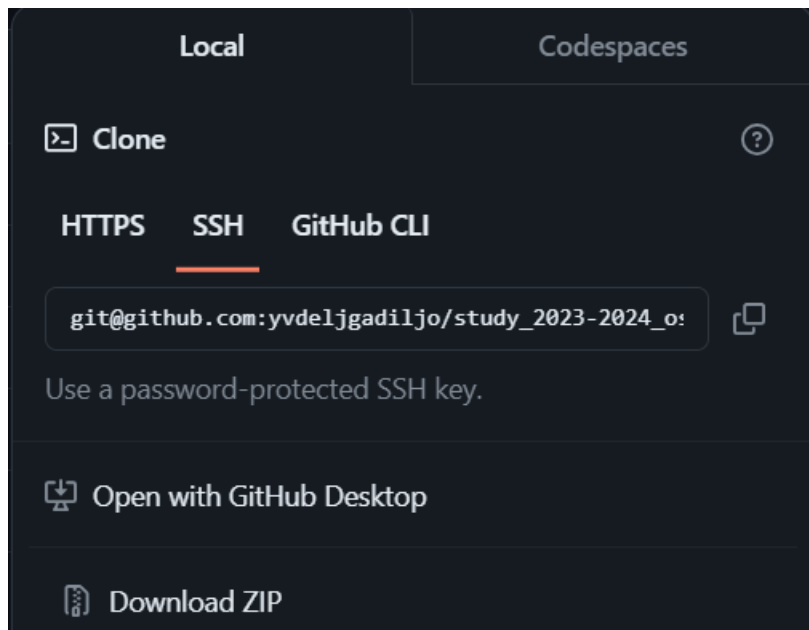


Рис. 20: Подключение к браузеру

```

Файл  Правка  Вид  Терминал  Вкладки  Справка
tation-markdown-template.git) зарегистрирован по пути «template/presentation»
Подмодуль «template/report» (https://github.com/yamadharma/academic-laboratory-r
eport-template.git) зарегистрирован по пути «template/report»
Клонирование в «/home/tsganina/work/study/2022-2023/Операционные системы/os-intro
o/template/presentation»...
remote: Enumerating objects: 82, done.
remote: Counting objects: 100% (82/82), done.
remote: Compressing objects: 100% (57/57), done.
remote: Total 82 (delta 28), reused 77 (delta 23), pack-reused 0
Получение объектов: 100% (82/82), 92.90 КиБ | 923.00 КиБ/с, готово.
Определение изменений: 100% (28/28), готово.
Клонирование в «/home/tsganina/work/study/2022-2023/Операционные системы/os-intro
o/template/report»...
remote: Enumerating objects: 101, done.
remote: Counting objects: 100% (101/101), done.
remote: Compressing objects: 100% (70/70), done.
remote: Total 101 (delta 40), reused 88 (delta 27), pack-reused 0
Получение объектов: 100% (101/101), 327.25 КиБ | 1.92 МиБ/с, готово.
Определение изменений: 100% (40/40), готово.
Submodule path 'template/presentation': checked out 'b1be380ee91f5809264cb755d3
16174540b753e'
Submodule path 'template/report': checked out '1d1b61dcac9c287a83917b82e3aef11a3
3b1e3b2'

```

Рис. 21: Итог

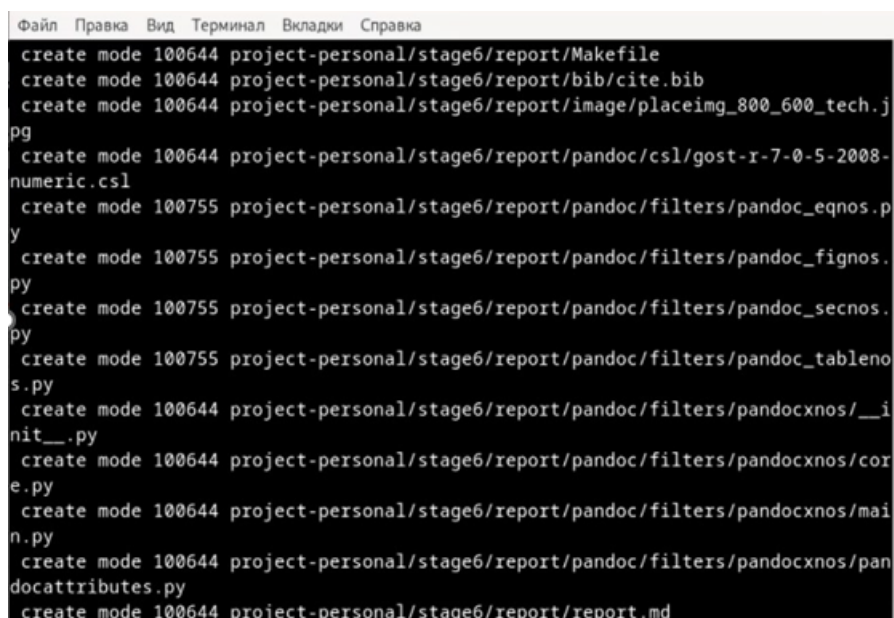
9. Настройка каталога курса (рис. 4.22, 4.23, 4.24, 4.25).

```

~]$ cd ~/work/study/2022-2023/"Операционные системы"/os-intro
os-intro]$ rm package.json
os-intro]$ make COURSE=os-intro
os-intro]$ git add /

```

Рис. 22: Переход в каталог, удаление лишних файлов, создание каталогов



```

Файл  Правка  Вид  Терминал  Вкладки  Справка
create mode 100644 project-personal/stage6/report/Makefile
create mode 100644 project-personal/stage6/report/bib/cite.bib
create mode 100644 project-personal/stage6/report/image/placeimg_800_600_tech.jpg
create mode 100644 project-personal/stage6/report/pandoc/csl/gost-r-7-0-5-2008-numeric.csl
create mode 100755 project-personal/stage6/report/pandoc/filters/pandoc_eqnos.py
create mode 100755 project-personal/stage6/report/pandoc/filters/pandoc_fignos.py
create mode 100755 project-personal/stage6/report/pandoc/filters/pandoc_secnos.py
create mode 100755 project-personal/stage6/report/pandoc/filters/pandoc_tablenos.py
create mode 100644 project-personal/stage6/report/pandoc/filters/pandocxnos/__init__.py
create mode 100644 project-personal/stage6/report/pandoc/filters/pandocxnos/core.py
create mode 100644 project-personal/stage6/report/pandoc/filters/pandocxnos/main.py
create mode 100644 project-personal/stage6/report/pandoc/filters/pandocxnos/pandocattributes.py
create mode 100644 project-personal/stage6/report/report.md

```

Рис. 23: Отправка файлов на сервер

config	<a href="#">Initial commit</a>	6 minutes ago
labs	feat(main): make course structure	now
presentation	feat(main): make course structure	now
project-personal	feat(main): make course structure	now

Рис. 24: Отправка файлов на сервер

```
Файл Правка Вид Терминал Вкладки Справка
py
create mode 100755 project-personal/stage6/report/pandoc/filters/pandoc_tableno
s.py
create mode 100644 project-personal/stage6/report/pandoc/filters/pandocxnos/__i
nit__.py
create mode 100644 project-personal/stage6/report/pandoc/filters/pandocxnos/cor
e.py
create mode 100644 project-personal/stage6/report/pandoc/filters/pandocxnos/mai
n.py
create mode 100644 project-personal/stage6/report/pandoc/filters/pandocxnos/pa
docattributes.py
create mode 100644 project-personal/stage6/report/report.md
[tsganina@tsganina os-intro]$ git push
Перечисление объектов: 38, готово.
Подсчет объектов: 100% (38/38), готово.
При сжатии изменений используется до 5 потоков
Сжатие объектов: 100% (30/30), готово.
Запись объектов: 100% (37/37), 343.00 КиБ | 2.52 МиБ/с, готово.
Всего 37 (изменений 4), повторно использовано 0 (изменений 0), повторно использо
вано пакетов 0
remote: Resolving deltas: 100% (4/4), completed with 1 local object.
To github.com:tsganina/study_2022-2023_os-intro.git
c788d86..78203dc master -> master
```

Рис. 25: Отправка файлов на сервер

# Контрольные вопросы.

1. Что такое системы контроля версий (VCS) и для решения каких задач они предназначаются?

Система контроля версий - программное обеспечение для облегчения работы с изменяющейся информацией.

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.

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

- Хранилище - репозиторий, хранилище версий, в нем хранятся все документы вместе с их историей, и другой служебной информацией.
- Commit - отслеживание изменений, сохранение разницы в рабочих изменениях.
- История - сохранение всех изменений в проекте и при необходимости возможность обратиться к старым данным.

- Рабочая копия - копия проекта, связанная с репозиторием, текущее состояние файлов проекта, основанное на их последней версии из хранилища.
3. Что представляют собой и чем отличаются централизованные и децентрализованные VCS? Приведите примеры VCS каждого вида.
- Централизованные VCS - одно основное хранилище всего проекта, каждый пользователь копирует себе необходимые файлы из репозитория и изменяет их, а потом может добавить обратно. Примеры: - Subversion; - CVS; - TFS; -VAULT; -AccuRev;
  - Децентрализованные VCS - У каждого пользователя свой вариант репозитория, есть возможность добавлять и забирать версии из любого репозитория.

Например: - Git; - Mercurial; - Bazaar;

Обычно используются централизованные системы контроля версий, с одним общим репозиторием.

4. Опишите действия с VCS при единоличной работе с хранилищем. Сначала нужно создать удалённый репозиторий, после - подключить его. Затем по мере того, как проект будет выполняться, нужно отправлять данные на сервер.
5. Опишите порядок работы с общим хранилищем VCS. В начале при помощи последовательности команд пользователь получает нужную ему версию данных. Потом он работает с ними, вносит некоторые изменения, и уже после этого он может разместить новую версию в хранилище. При этом предыдущие версии не удаляются из хранилища, что является очень удобным - к ним можно вернуться в любой момент.
6. Каковы основные задачи, решаемые инструментальным средством git?
- Хранить информацию обо всех изменениях, производимых в проекте.

- Обеспечить командную работу.

7. Назовите и дайте краткую характеристику командам git.

git init - Создание основного дерева репозитория git pull - Получение обновлений (изменений) текущего дерева из центрального репозитория

git push - Отправка всех произведённых изменений локального дерева в центральный репозиторий

git status - Просмотр списка изменённых файлов в текущей директории изменений

git add . / git add <имя файла / git rm <имя файла - Сохранение текущих изменений git commit / git commit -am "описание коммита" – Сохранение добавленных изменений

git checkout -b имя\_ветки - Создание новой ветки, базирующейся на текущей

git checkout имя\_ветки - Переключение на некоторую ветку git push origin имя\_ветки - Отправка изменений конкретной ветки в центральный репозиторий

git merge --no-ff имя\_ветки - Слияние ветки с текущим деревом git branch -d имя\_ветки -Удаление локальной уже слитой с основным деревом ветки git branch -D имя\_ветки - Принудительное удаление локальной ветки git push origin :имя\_ветки -Удаление ветки с центрального репозитория.

8. Приведите примеры использования при работе с локальным и удалённым репозиториями. git push -all (push origin master/любой branch)

9. Что такое и зачем могут быть нужны ветви (branches)? Ветвь - один из параллельных участков истории проекта в одном хранилище. Все ветви исходят из одной версии-точки ветвления. Обычно ветви делятся на master и trunk. Между ветками возможно и слияние. Ветки нужны для разработки новых функций.

Как и зачем можно игнорировать некоторые файлы при commit? Игнорировать некоторые файлы можно прописав шаблон .gitignore специально для игнорируемых файлов. Зачем это нужно? Чтобы в репозиторий не попали “лишние” файлы,

которые неминуемо будут возникать при работе над проектом. Это могут быть временные файлы, объектные файлы.

## Выводы

- Я изучила идеологию и применение средств контроля версий.
- Освоила умения по работе с git.



## Список литературы

- GDB: The GNU Project Debugger. — URL: <https://www.gnu.org/software/gdb/>.
- GNU Bash Manual. — 2016. — URL: <https://www.gnu.org/software/bash/manual/>.
- Midnight Commander Development Center. — 2021. — URL: <https://midnight-commander.org/>.
- NASM Assembly Language Tutorials. — 2021. — URL: <https://asmtutor.com/>.
- Newham C. Learning the bash Shell: Unix Shell Programming. — O'Reilly Media, 2005. — 354 с. — (In a Nutshell). — ISBN 0596009658. — URL: <http://www.amazon.com/Learningbash-Shell-Programming-Nutshell/dp/0596009658>.
- Robbins A. Bash Pocket Reference. — O'Reilly Media, 2016. — 156 с. — ISBN 978-1491941591.
- The NASM documentation. — 2021. — URL: <https://www.nasm.us/docs.php>.
- Zarrelli G. Mastering Bash. — Packt Publishing, 2017. — 502 с. — ISBN 9781784396879.
- Колдаев В. Д., Lupin С. А. Архитектура ЭВМ. — М. : Форум, 2018.
- Куляс О. Л., Никитин К. А. Курс программирования на ASSEMBLER. — М. : Солон-Пресс, 2017.
- Новожилов О. П. Архитектура ЭВМ и систем. — М. : Юрайт, 2016.
- Расширенный ассемблер: NASM. — 2021. — URL: <https://www.opennet.ru/docs/RUS/nasm/>.

- Робачевский А., Немнюгин С., Стесик О. Операционная система UNIX. — 2-е изд. — БХВПетербург, 2010. — 656 с. — ISBN 978-5-94157-538-1.
- Столяров А. Программирование на языке ассемблера NASM для ОС Unix. — 2-е изд. — М. : МАКС Пресс, 2011. — URL: [http://www.stolyarov.info/books/asm\\_unix](http://www.stolyarov.info/books/asm_unix).
- Таненбаум Э. Архитектура компьютера. — 6-е изд. — СПб. : Питер, 2013. — 874 с. — (Классика Computer Science).
- Таненбаум Э., Бос Х. Современные операционн