

# **Лабораторная работа №2**

**Операционные системы**

Клюкин Михаил Александрович

# Содержание

<b>1</b>	<b>Цель работы</b>	<b>5</b>
<b>2</b>	<b>Задание</b>	<b>6</b>
<b>3</b>	<b>Теоретическое введение</b>	<b>7</b>
<b>4</b>	<b>Выполнение лабораторной работы</b>	<b>8</b>
4.1	Установка и настройка программного обеспечения . . . . .	8
4.2	Базовая настройка git . . . . .	9
4.3	Настройка рабочего пространства . . . . .	13
<b>5</b>	<b>Выводы</b>	<b>16</b>
<b>6</b>	<b>Ответы на контрольные вопросы</b>	<b>17</b>
<b>7</b>	<b>Список литературы</b>	<b>20</b>

## Список иллюстраций

4.1	Установка git и gh . . . . .	8
4.2	Настройка git . . . . .	9
4.3	Генерация ключа и выбор опций . . . . .	10
4.4	Ввод личной информации . . . . .	10
4.5	Экспорт ключа . . . . .	11
4.6	Экспорт ключа . . . . .	11
4.7	Настройка автоматических подписей коммитов . . . . .	11
4.8	Задание параметров и имени . . . . .	12
4.9	Создание ssh ключа . . . . .	12
4.10	Настройка gh . . . . .	13
4.11	Создание каталога . . . . .	13
4.12	Создание репозитория . . . . .	13
4.13	Клонирование репозитория . . . . .	14
4.14	Настройка репозитория . . . . .	14
4.15	Настройка репозитория . . . . .	15

## Список таблиц

# 1 Цель работы

- Изучить идеологию и применение средств контроля версий
- Научиться работать с git

## 2 Задание

1. Установить программное обеспечение
2. Провести базовую настройку git
3. Создать ключ SSH
4. Создать ключи pgr
5. Провести настройку github
6. Добавить ключ PGP на Github
7. Настроить автоматическое подписывание коммитов git
8. Настроить gh
9. Создать шаблон рабочего пространства

### 3 Теоретическое введение

Системы контроля версий используются при работе нескольких человек над одним проектом.

Системы контроля версий хранят дерево проекта в локальном или удаленном репозитории, к которому имеют доступ все участники проекта.

При внесении изменений в проект система контроля версий позволяет фиксировать эти изменения, совмещать их, возвращаться к более ранним версиям проекта.

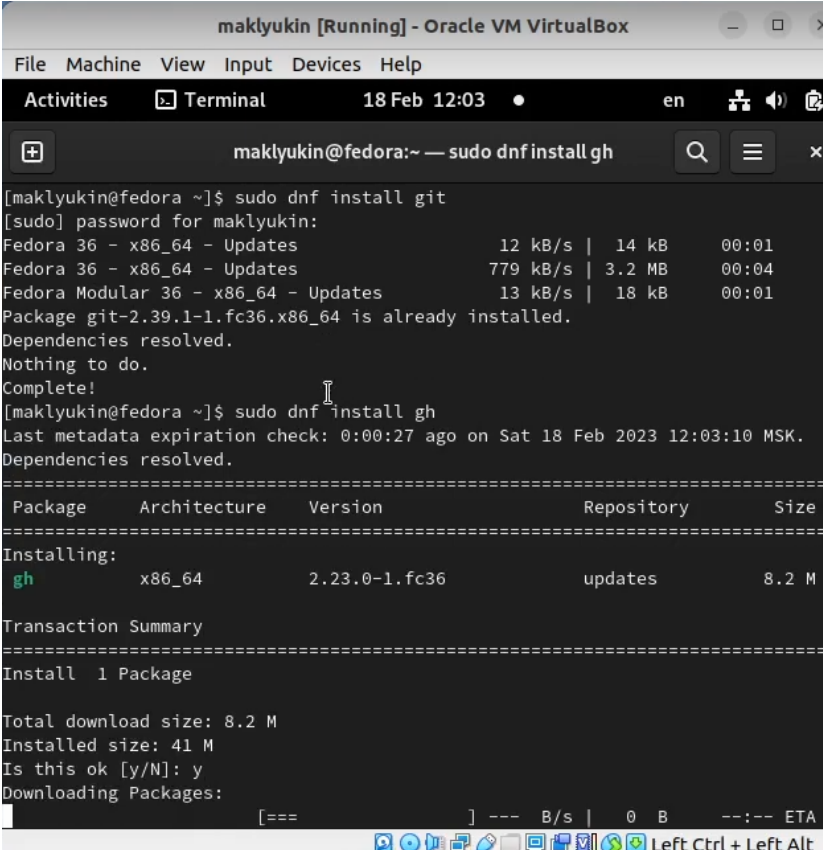
Бывают распределенные и централизованные VCS. Мы будем работать с распределенной системой контроля версий – Git. Также к распределенным системам контроля версий относятся такие VCS как Mercurial, Bazaar, Darcs.

В распределенных системах контроля версий в локальных репозиториях пользователей хранятся полные копии всего репозитория проекта.

## 4 Выполнение лабораторной работы

### 4.1 Установка и настройка программного обеспечения

Установили git, установили gh (Рис. 4.1).



```
maklyukin [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Activities Terminal 18 Feb 12:03 en
maklyukin@fedora:~ — sudo dnf install gh

[maklyukin@fedora ~]$ sudo dnf install git
[sudo] password for maklyukin:
Fedora 36 - x86_64 - Updates          12 kB/s | 14 kB    00:01
Fedora 36 - x86_64 - Updates          779 kB/s | 3.2 MB  00:04
Fedora Modular 36 - x86_64 - Updates  13 kB/s | 18 kB   00:01
Package git-2.39.1-1.fc36.x86_64 is already installed.
Dependencies resolved.
Nothing to do.
Complete!
[maklyukin@fedora ~]$ sudo dnf install gh
Last metadata expiration check: 0:00:27 ago on Sat 18 Feb 2023 12:03:10 MSK.
Dependencies resolved.
=====
Package      Architecture Version      Repository    Size
=====
Installing:
gh           x86_64      2.23.0-1.fc36 updates      8.2 M
Transaction Summary
=====
Install 1 Package

Total download size: 8.2 M
Installed size: 41 M
Is this ok [y/N]: y
Downloading Packages:
[===          ] --- B/s | 0 B    --:-- ETA
```

Рис. 4.1: Установка git и gh



## 4.2 Базовая настройка git

Задали имя и email владельца репозитория (Рис. 4.1):

```
'git config --global user.name "Name Surname"' git config --global user.email  
"user@email"
```

Настроили utf-8 в выводе сообщений git (Рис. 4.1):

```
git config --global core.quotepath false
```

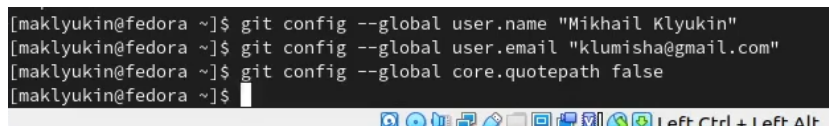
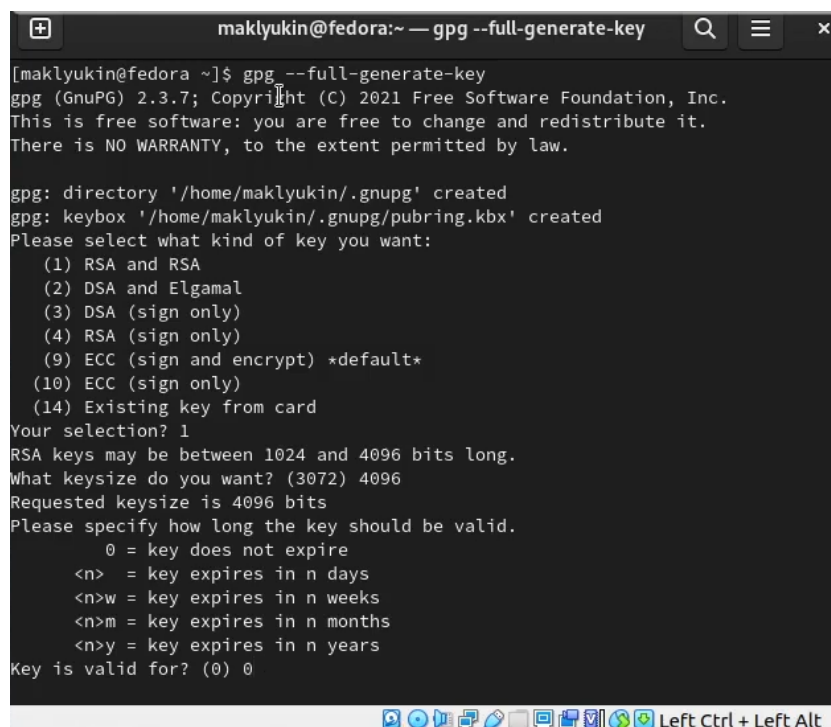
A screenshot of a terminal window with a dark background. It shows three lines of commands being executed: `[maklyukin@fedora ~]$ git config --global user.name "Mikhail Klyukin"`, `[maklyukin@fedora ~]$ git config --global user.email "klumisha@gmail.com"`, and `[maklyukin@fedora ~]$ git config --global core.quotepath false`. The prompt `[maklyukin@fedora ~]$` is visible at the end of each line. At the bottom of the terminal, there is a taskbar with various application icons and the text "Left Ctrl + Left Alt".

Рис. 4.2: Настойка git

Настроили верификацию и подписывание коммитов git:

- Сгенерировали ключ (Рис. 4.1)  
`gpg --full-generate-key`
- Выбрали следующие опции – RSA and RSA, 4096, 0 (Рис. 4.3)

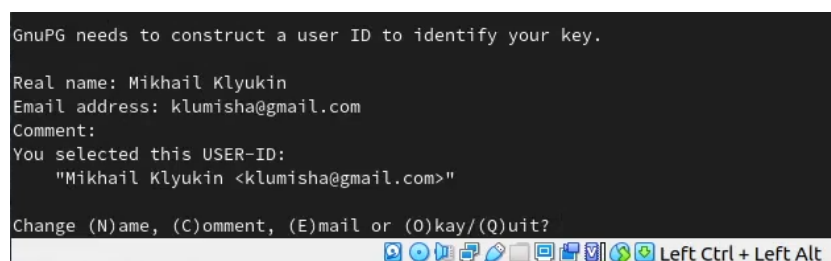


```
[maklyukin@fedora ~]$ gpg --full-generate-key
gpg (GnuPG) 2.3.7; Copyright (C) 2021 Free Software Foundation, Inc.
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.

gpg: directory '/home/maklyukin/.gnupg' created
gpg: keybox '/home/maklyukin/.gnupg/pubring.kbx' created
Please select what kind of key you want:
  (1) RSA and RSA
  (2) DSA and Elgamal
  (3) DSA (sign only)
  (4) RSA (sign only)
  (9) ECC (sign and encrypt) *default*
  (10) ECC (sign only)
  (14) Existing key from card
Your selection? 1
RSA keys may be between 1024 and 4096 bits long.
What keysize do you want? (3072) 4096
Requested keysize is 4096 bits
Please specify how long the key should be valid.
    0 = key does not expire
    <n> = key expires in n days
    <n>w = key expires in n weeks
    <n>m = key expires in n months
    <n>y = key expires in n years
Key is valid for? (0) 0
```

Рис. 4.3: Генерация ключа и выбор опций

- Указали личную информацию – имя, адрес электронной почты (Рис. 4.4)



```
GnuPG needs to construct a user ID to identify your key.

Real name: Mikhail Klyukin
Email address: klumisha@gmail.com
Comment:
You selected this USER-ID:
    "Mikhail Klyukin <klumisha@gmail.com>"

Change (N)ame, (C)omment, (E)mail or (O)kay/(Q)uit?
```

Рис. 4.4: Ввод личной информации

- Произвели экспорт ключа и добавили его на Github (Рис. 4.5, 4.6) `gpg --list-secret-keys --keyid-format LONG gpg --armor --export`

```
[maklyukin@fedora ~]$ gpg --list-secret-keys --keyid-format LONG
gpg: checking the trustdb
gpg: marginals needed: 3 completes needed: 1 trust model: pgp
gpg: depth: 0 valid: 1 signed: 0 trust: 0-, 0q, 0n, 0m, 0f, 1u
/home/maklyukin/.gnupg/pubring.kbx
-----
sec   rsa4096/188B68190E1F5042 2023-02-18 [SC]
      80C772BABB252D2988C5847B188B68190E1F5042
uid           [ultimate] Mikhail Klyukin <klumisha@gmail.com>
ssb   rsa4096/539936823C31883E 2023-02-18 [E]

[maklyukin@fedora ~]$ gpg --armor --export 188B68190E1F5042
```

Рис. 4.5: Экспорт ключа

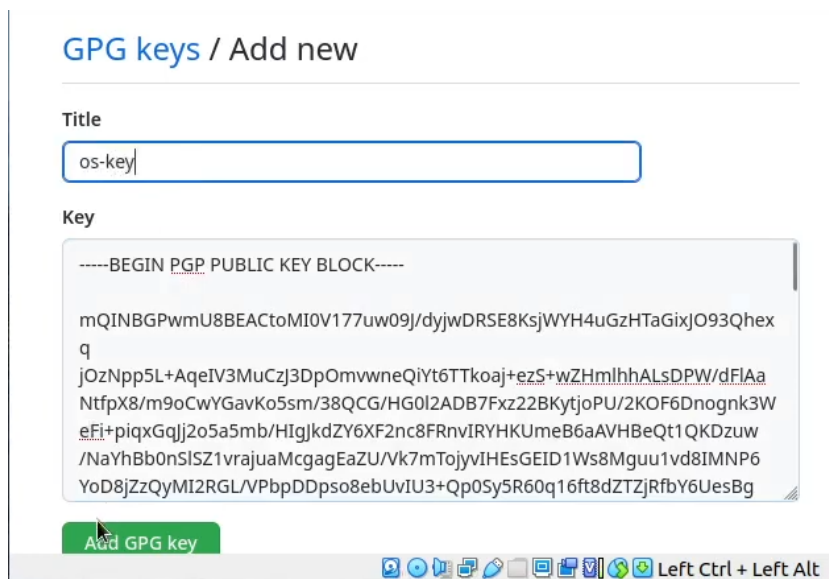


Рис. 4.6: Экспорт ключа

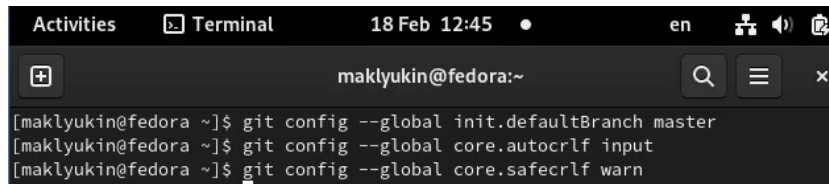
- Произвели нвстройку автоматических подписей коммитов git (Рис. 4.7)

```
Activities Terminal 18 Feb 12:43 en
maklyukin@fedora:~
[maklyukin@fedora ~]$ git config --global user.signingkey 188B68190E1F5042
[maklyukin@fedora ~]$ git config --global commit.gpgsign true
[maklyukin@fedora ~]$ git config --global gpg.program $(which gpg2)
```

Рис. 4.7: Настройка автоматических подписей коммитов

Задали имя начальной ветки, параметр autocrlf, параметр safecrlf (Рис. 4.8)

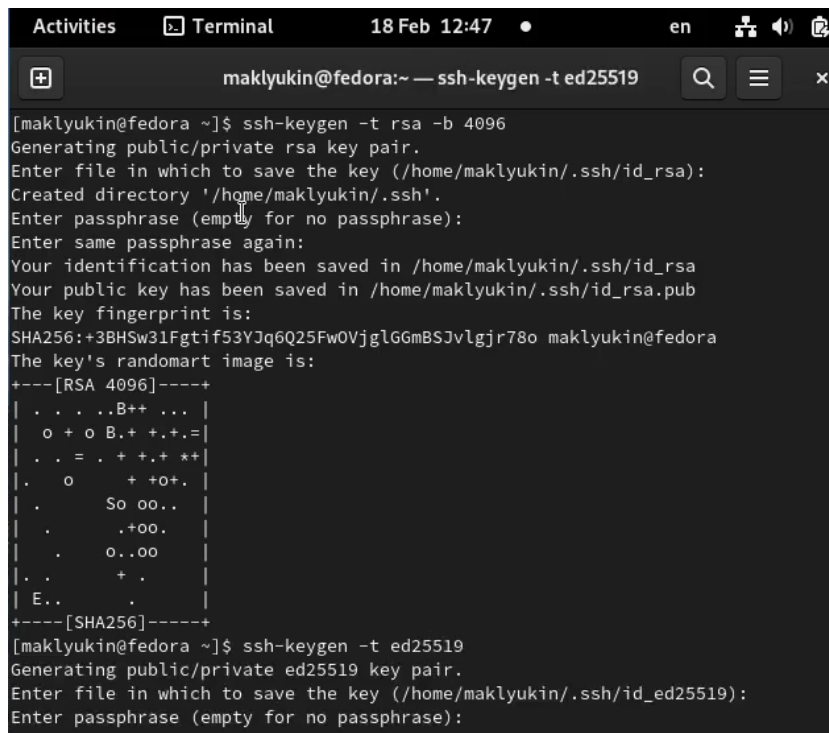
git config --global init.defaultBranch master  
git config --global core.autocrlf input  
git config --global core.safecrlf warn

A terminal window titled 'maklyukin@fedora:~' showing three lines of commands: 'git config --global init.defaultBranch master', 'git config --global core.autocrlf input', and 'git config --global core.safecrlf warn'. The terminal has a dark background and standard Linux window controls at the top.

```
maklyukin@fedora:~  
[maklyukin@fedora ~]$ git config --global init.defaultBranch master  
[maklyukin@fedora ~]$ git config --global core.autocrlf input  
[maklyukin@fedora ~]$ git config --global core.safecrlf warn
```

Рис. 4.8: Задание параметров и имени

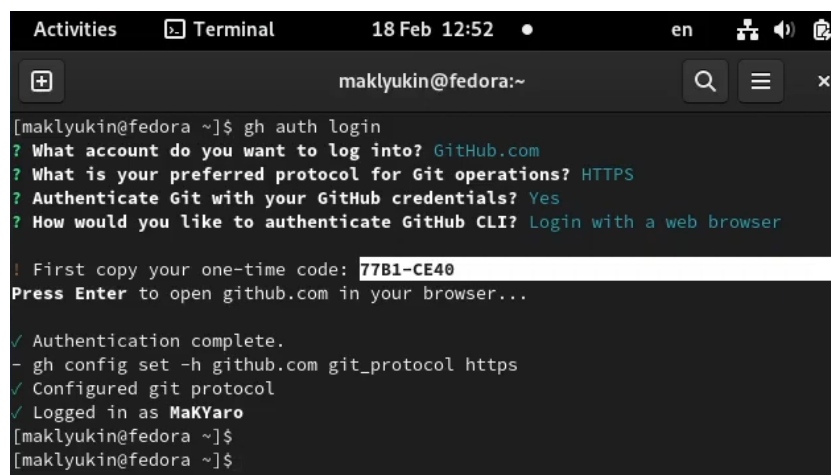
Создали ключ ssh по алгоритму rsa с размером 4096 бит, по алгоритму ed25519  
(Рис. 4.9)

A terminal window titled 'maklyukin@fedora:~ — ssh-keygen -t ed25519' showing the output of 'ssh-keygen -t rsa -b 4096'. It includes prompts for file location and passphrase, followed by a detailed ASCII art representation of the RSA key. Then it shows the start of 'ssh-keygen -t ed25519' with prompts for file location and passphrase.

```
maklyukin@fedora:~ — ssh-keygen -t ed25519  
[maklyukin@fedora ~]$ ssh-keygen -t rsa -b 4096  
Generating public/private rsa key pair.  
Enter file in which to save the key (/home/maklyukin/.ssh/id_rsa):  
Created directory '/home/maklyukin/.ssh'.  
Enter passphrase (empty for no passphrase):  
Enter same passphrase again:  
Your identification has been saved in /home/maklyukin/.ssh/id_rsa  
Your public key has been saved in /home/maklyukin/.ssh/id_rsa.pub  
The key fingerprint is:  
SHA256:+3BHSw31Fgtif53YJq6Q25FwOVjglGmBSJvlgjr78o maklyukin@fedora  
The key's randomart image is:  
+---[RSA 4096]-----+  
| . . . .B++ ... |  
| o + o B.+ +.+.=|  
| . . = . + +.+ **|  
| . o + +o+. |  
| . So oo.. |  
| . .+oo. |  
| . o..oo |  
| . . + . |  
| E.. . |  
+---[SHA256]-----+  
[maklyukin@fedora ~]$ ssh-keygen -t ed25519  
Generating public/private ed25519 key pair.  
Enter file in which to save the key (/home/maklyukin/.ssh/id_ed25519):  
Enter passphrase (empty for no passphrase):
```

Рис. 4.9: Создание ssh ключа

Создали учетную запись на github и провели базовую настройку профиля.  
Провели настройку gh (Рис. 4.10)



```
Activities Terminal 18 Feb 12:52 en
maklyukin@fedora:~
[maklyukin@fedora ~]$ 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: 77B1-CE40
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 MaKVaro
[maklyukin@fedora ~]$
[maklyukin@fedora ~]$
```

Рис. 4.10: Настройка gh

## 4.3 Настройка рабочего пространства

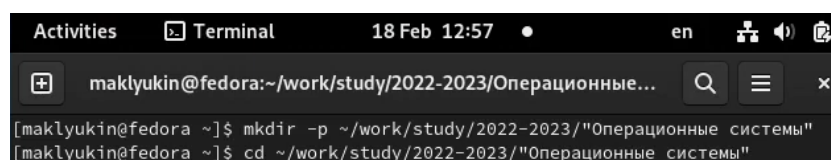
Создали шаблон рабочего пространства (Рис. 4.11, 4.12, 4.13)

```
mkdir -p ~/work/study/2022-2023/“Операционные системы”
```

```
cd ~/work/study/2022-2023/“Операционные системы”
```

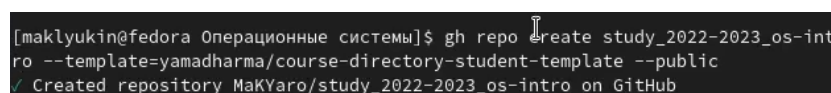
```
gh repo create study_2022-2023_os-intro --template=yamadharma/course-directory-
student-template --public
```

```
git clone --recursive
```



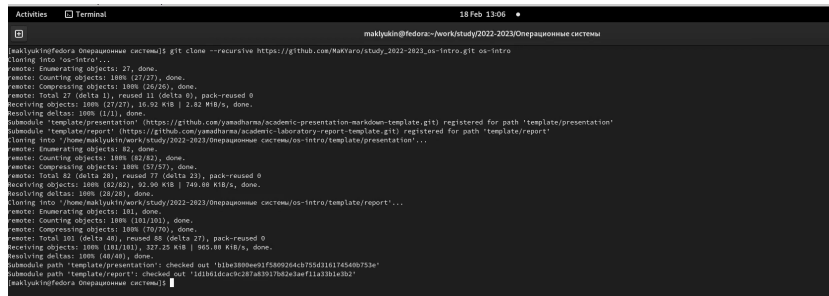
```
Activities Terminal 18 Feb 12:57 en
maklyukin@fedora:~/work/study/2022-2023/Операционные...
[maklyukin@fedora ~]$ mkdir -p ~/work/study/2022-2023/"Операционные системы"
[maklyukin@fedora ~]$ cd ~/work/study/2022-2023/"Операционные системы"
```

Рис. 4.11: Создание каталога



```
[maklyukin@fedora Операционные системы]$ gh repo create study_2022-2023_os-int
ro --template=yamadharma/course-directory-student-template --public
✓ Created repository MaKVaro/study_2022-2023_os-intro on GitHub
```

Рис. 4.12: Создание репозитория

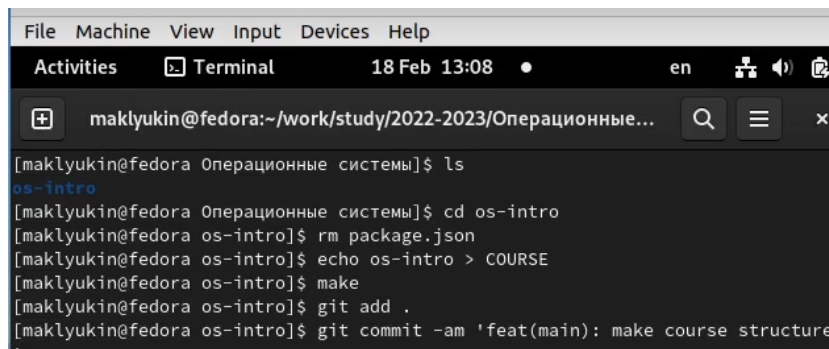


```
maklyukin@fedora:~/work/study/2022-2023/Операционные системы$ git clone --recursive https://github.com/Maklyukin/study_2022-2023-os-intro.git os-intro
maklyukin@fedora:~/work/study/2022-2023/Операционные системы$ cd os-intro
maklyukin@fedora:~/work/study/2022-2023/Операционные системы/os-intro$ rm package.json
maklyukin@fedora:~/work/study/2022-2023/Операционные системы/os-intro$ echo os-intro > COURSE
maklyukin@fedora:~/work/study/2022-2023/Операционные системы/os-intro$ make
maklyukin@fedora:~/work/study/2022-2023/Операционные системы/os-intro$ git add .
maklyukin@fedora:~/work/study/2022-2023/Операционные системы/os-intro$ git commit -am 'feat(main): make course structure'
maklyukin@fedora:~/work/study/2022-2023/Операционные системы/os-intro$ git push
maklyukin@fedora:~/work/study/2022-2023/Операционные системы/os-intro$
```

Рис. 4.13: Клонирование репозитория

Произвели настройку каталога курса. Для этого перешли в каталог курса, удалили лишние файлы, создали необходимые каталоги, отправили файлы на сервер (Рис. 4.14, 4.15)

```
cd ~/work/study/2022-2023/“Операционные системы”/os-intro
rm package.json
echo os-intro > COURSE
make
git add .
git commit -am 'feat(main): make course structure'
git push
```



```
maklyukin@fedora:~/work/study/2022-2023/Операционные системы$ ls
os-intro
maklyukin@fedora:~/work/study/2022-2023/Операционные системы$ cd os-intro
maklyukin@fedora:~/work/study/2022-2023/Операционные системы/os-intro$ rm package.json
maklyukin@fedora:~/work/study/2022-2023/Операционные системы/os-intro$ echo os-intro > COURSE
maklyukin@fedora:~/work/study/2022-2023/Операционные системы/os-intro$ make
maklyukin@fedora:~/work/study/2022-2023/Операционные системы/os-intro$ git add .
maklyukin@fedora:~/work/study/2022-2023/Операционные системы/os-intro$ git commit -am 'feat(main): make course structure'
maklyukin@fedora:~/work/study/2022-2023/Операционные системы/os-intro$
```

Рис. 4.14: Настройка репозитория

```
[maklyukin@fedora os-intro]$ git push
Enumerating objects: 40, done.
Counting objects: 100% (40/40), done.
Compressing objects: 100% (30/30), done.
Writing objects: 100% (38/38), 343.04 KiB | 3.43 MiB/s, done.
Total 38 (delta 4), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (4/4), completed with 1 local object.
To https://github.com/MaKYaro/study_2022-2023_os-intro.git
   c01586c..9dec3cf  master -> master
[maklyukin@fedora os-intro]$
```

Рис. 4.15: Настройка репозитория

## 5 Выводы

Изучили идеологию и научились применять системы контроля версий. Научились работать с git



## 6 Ответы на контрольные вопросы

1. Что такое системы контроля версий? Для решения каких задач они используются?

Системы контроля версий – это системы, записывающая изменения в файл или набор файлов в течение времени и позволяющая вернуться к определенной версии.

VCS используются для работы нескольких человек над одним проектом.

2. Что такое хранилище, коммит, история, рабочая копия? Как они связаны между собой?

Хранилище – это место, в котором хранятся и поддерживаются файлы.

Коммит – это сохранение изменений.

История – все изменения в проекте.

Рабочая копия – текущее состояние файлов проекта

3. Что представляют собой централизованные и распределенные системы контроля версий?

Централизованные системы имеют одно хранилище всего проекта, каждый пользователь копирует необходимые файлы, изменяет их и добавляет изменения обратно (Subversion, CVS, TFS, AccuRev).

В распределенных системах контроля версий у каждого пользователя свой вариант репозитория. В них присутствует возможность добавлять изменения из любого репозитория (Git, Mercurial, Bazaar).

4. Опишите действия с VCS при единоличной работе с хранилищем.

Создаем и подключаем рабочий репозиторий. В ходе работы над проектом отправляем все изменения на сервер.

5. Опишите порядок работы с общим хранилищем.

Получаем нужную версию файлов, вносим изменения, размещаем новую версию в хранилище. Предыдущие версии не удаляются, к ним можно всегда вернуться.

6. Каковы основные задачи, решаемые инструментальным средством git?

Хранит информацию о всех изменениях, обеспечивает комфортную командную работу над проектом.

7. Назовите и дайте краткую характеристику командам git/

- git init – создает основное дерево репозитория
- git pull – вытягивает обновления текущего дерева из центрального репозитория
- git push – отправляет все произведенные изменения локального дерева в центральный репозиторий
- git status – позволяет просмотреть список измененных файлов в текущей директории
- git diff – позволяет посмотреть текущие изменения
- git add – добавляет все измененные или (и) созданные каталоги (файлы)
- git rm – удаляет файл и (или) каталог из индекса репозитория
- git commit – сохраняет добавленные изменения
- git checkout – используется для переключения на ветку

8. Приведите примеры использования при работе с локальным и удаленным репозиториями.

```
git push -all  
git push origin main
```

9. Что такое и зачем нужны ветки?

Ветки – параллельные участки истории изменений  
Ветки используются при разработке новых функций

10. Как и зачем можно игнорировать некоторые файлы?

Для того, чтобы игнорировать файлы, их названия нужно добавить в файл .gitignore.

Некоторые файлы не требуется добавлять в репозиторий. Именно такие файлы и игнорируются.

## 7 Список литературы

1. Демидова А. В. Лабораторная работа №3. Система контроля версий Git – Методическое пособие
2. Система контроля версий git [электронный ресурс]: <https://git-scm.com/book/en/v2>