

# Лабораторная работа №2

---

Веретенников Д. О.

Российский университет дружбы народов, Москва, Россия

## Информация

---

- Веретенников Дмитрий Олегович
- НКАбд-01-24 ст.билет №1132240687
- Студент 1 Курса
- Российский университет дружбы народов
- [https://github.com/doveretennikov/study\\_2024-2025\\_os-intro](https://github.com/doveretennikov/study_2024-2025_os-intro)

## Вводная часть

---

- Изучить идеологию и применение средств контроля версий.
- Освоить умения по работе с git.
- Установка программного обеспечения.
- Базовая настройка git.
- Настройка рабочего пространства курса.

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.

В классических системах контроля версий используется централизованная модель, предполагающая наличие единого репозитория для хранения файлов. Выполнение большинства функций по управлению версиями осуществляется специальным сервером. 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 момент.

Сервер может сохранять не полную версию изменённых файлов, а производить так называемую дельта-компрессию — сохранять только изменения между последовательными версиями, что позволяет уменьшить объём хранимых данных.

Системы контроля версий поддерживают возможность отслеживания и разрешения конфликтов, которые могут возникнуть при работе нескольких человек над одним файлом. Можно объединить (слить) изменения, сделанные разными участниками (автоматически или вручную), вручную выбрать нужную версию, отменить изменения вовсе или заблокировать файлы для изменения. В зависимости от настроек блокировка не позволяет другим пользователям получить рабочую копию или препятствует изменению рабочей копии файла средствами файловой системы ОС, обеспечивая таким образом, привилегированный доступ только одному пользователю, работающему с файлом.

Системы контроля версий также могут обеспечивать дополнительные, более гибкие функциональные возможности. Например, они могут поддерживать работу с несколькими версиями одного файла, сохраняя общую историю изменений до точки ветвления версий и собственные истории изменений каждой ветви. Кроме того, обычно доступна информация о том, кто из участников, когда и какие изменения вносил. Обычно такого рода информация хранится в журнале изменений, доступ к которому можно ограничить. В отличие от классических, в распределённых системах контроля версий центральный репозиторий не является обязательным. Среди классических VCS наиболее известны CVS, Subversion, а среди распределённых — Git, Bazaar, Mercurial. Принципы их работы схожи, отличаются они в основном синтаксисом используемых в работе команд.



# Выполнение Лабораторной работы №2

---

# Установка программного обеспечения

Устанавливаем git и gh.

```
foot
[doveretennikov@vbox ~]$ dnf install git
Для выполнения запрошенной операции требуются привилегии суперпользователя. Пожалуйста, войдите в систему как пользователь с повышенными правами или используйте опции "--assumeno" или "--downloadonly", чтобы выполнить команду без изменения состояния системы.
[doveretennikov@vbox ~]$ sudo -i
[root@vbox ~]# dnf install git
Обновление и загрузка репозитория:
Репозитории загружены.
Пакет "git-2.48.1-1.fc41.x86_64" уже установлен.

Нечего делать.
[root@vbox ~]# dnf install gh
Обновление и загрузка репозитория:
Репозитории загружены.
Пакет                                Арх.      Версия              Репозиторий         Размер
Установка:
gh                                   x86_64    2.65.0-1.fc41      updates              42.6 MiB

Сводка транзакции:
Установка:      1 пакета

Общий размер входящих пакетов составляет 10 MiB. Необходимо загрузить 10 MiB.
После этой операции будут использоваться дополнительные 43 MiB (установка 43 MiB, удаление 0 B).
Is this ok [y/N]: y
[1/1] gh-0:2.65.0-1.fc41.x86_64                                100% | 146.1 KiB/s | 10.3 MiB | 01m12s
-----
[1/1] Total1                                                    100% | 139.2 KiB/s | 10.3 MiB | 01m16s
Выполнение транзакции
[1/3] Проверить файлы пакета                                100% | 14.0 B/s | 1.0 B | 00m00s
[2/3] Подготовить транзакцию                                100% | 0.0 B/s | 1.0 B | 00m01s
[3/3] Установка gh-0:2.65.0-1.fc41.x86_64                  100% | 9.9 MiB/s | 42.7 MiB | 00m04s
```

Рис. 1: Процесс установки git и gh

## Базовая настройка git

Задаём имя и email владельца репозитория. Далее настраиваем utf-8 в выводе сообщений git. После этого задаём имя начальной ветки, параметр autocrlf и параметр safecrlf.

```
foot
Пакет "git-2.48.1-1.fc41.x86_64" уже установлен.

Нечего делать.
[root@vbox ~]# dnf install gh
Обновление и загрузка репозитория:
Репозитории загружены.
Пакет
Установка:
  gh
  Арх.      Версия
  x86_64    2.65.0-1.fc41
  Репозиторий
  updates
  Размер
  42.6 MiB

Сводка транзакции:
Установка:      1 пакета

Общий размер входящих пакетов составляет 10 MiB. Необходимо загрузить 10 MiB.
После этой операции будут использоваться дополнительные 43 MiB (установка 43 MiB, удаление 0 B).
Is this ok [y/N]: y
[1/1] gh-0:2.65.0-1.fc41.x86_64
100% | 146.1 KiB/s | 10.3 MiB | 01m12s
-----
[1/1] Total
100% | 139.2 KiB/s | 10.3 MiB | 01m16s
Выполнение транзакции
[1/3] Проверить файлы пакета
100% | 14.0 B/s | 1.0 B | 00m00s
[2/3] Подготовить транзакцию
100% | 0.0 B/s | 1.0 B | 00m01s
[3/3] Установка gh-0:2.65.0-1.fc41.x86_64
100% | 9.9 MiB/s | 42.7 MiB | 00m04s
Завершено!
[root@vbox ~]# git config --global user.name "doveretennikov"
[root@vbox ~]# git config --global user.email "domiswww2018@gmail.com"
[root@vbox ~]# git config --global core.quotepath false
[root@vbox ~]# git config --global init.defaultBranch master
[root@vbox ~]# git config --global core.autocrlf input
[root@vbox ~]# git config --global core.safecrlf warn
[root@vbox ~]#
```

Создаём ключи ssh. С начала ключ по алгоритму rsa, а потом по алгоритму ed25519.

```
root
[2/3] Подготовить транзакцию          100% | 0.0 B/s | 1.0 B | 00m01s
[3/3] Установка gh-0.2.65.0-1.fc41.x86_64 100% | 9.9 MiB/s | 42.7 MiB | 00m04s
Завершено!
[root@vbox ~]# git config --global user.name "doveretennikov"
[root@vbox ~]# git config --global user.email "domiswww2018@gmail.com"
[root@vbox ~]# git config --global core.quotepath false
[root@vbox ~]# git config --global init.defaultBranch master
[root@vbox ~]# git config --global core.autocrlf input
[root@vbox ~]# git config --global core.safecrlf warn
[root@vbox ~]# ssh-keygen -t rsa -b 4096
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa):
Enter passphrase for "/root/.ssh/id_rsa" (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /root/.ssh/id_rsa
Your public key has been saved in /root/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:UcgnZxU41IX6KNLQxserdswUhy9vZCyAEXtbjdEirra root@vbox
The key's random image is:
+---[RSA 4096]-----+
| ..+0+++. |
| =oB ++. |
| ooX.+o.. |
| o =oBo |
| *So.B o |
| E + + + |
| . * = |
| o + . o |
| . . +. |
+-----[SHA256]-----+
[root@vbox ~]#
```

Рис. 3: Процесс создание ключей ssh

## Создание ключей SSH

Создаём ключи ssh. С начала ключ по алгоритму rsa, а потом по алгоритму ed25519.

```
root
|      =oB ++. |
|      ooX.+o.. |
|      o =oBo  |
|      *So.B o  |
|      E ++ = + |
|      . * =    |
|      o + . o  |
|      . . +.   |
|      +-----+ [SHA256] -----+
|      [root@vbox ~]# ssh-keygen -t ed25519
|      Generating public/private ed25519 key pair.
|      Enter file in which to save the key (/root/.ssh/id_ed25519):
|      Enter passphrase for "/root/.ssh/id_ed25519" (empty for no passphrase):
|      Enter same passphrase again:
|      Your identification has been saved in /root/.ssh/id_ed25519
|      Your public key has been saved in /root/.ssh/id_ed25519.pub
|      The key fingerprint is:
|      SHA256:LEnoI1RFJeqecVspFI82V0bzsp7M+nrQ4gGo+61W+TQ root@vbox
|      The key's randomart image is:
|      +---[ED25519 256]---+
|      | .o*o+ | |
|      | . + B  |
|      | ..O * + |
|      | ..O.+ + + |
|      | .. =.=.S . |
|      | . o +=.o  |
|      | .oEo..+. |
|      | .=. . .+.. |
|      | |=.o. o=*  |
|      | +-----+ [SHA256] -----+
|      [root@vbox ~]#
```

Рис. 4: Процесс создание ключей ssh

Создаём ключ pgr.

```
foot
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: doveretennikov
Адрес электронной почты: domiswww2018@gmail.com
Примечание:
Вы выбрали следующий идентификатор пользователя:
"doveretennikov <domiswww2018@gmail.com>"

Сменить (N)Имя, (C)Примечание, (E)Адрес; (O)Принять/(Q)Выход? 0
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: создан каталог '/root/.gnupg/openpgp-revocs.d'
gpg: сертификат отзыва записан в '/root/.gnupg/openpgp-revocs.d/9D0C61A13D8AFC5E8E29006E6D2B28B637D4588B.rev'.
открытый и секретный ключи созданы и подписаны.

pub   rsa4096 2025-02-25 [SC]
       9D0C61A13D8AFC5E8E29006E6D2B28B637D4588B
uid     doveretennikov <domiswww2018@gmail.com>
sub     rsa4096 2025-02-25 [E]

[root@vbox ~]#
```

Рис. 5: Процесс создание ключа pgr

Выводим список ключей и копируем отпечаток приватного ключа.

```
foot
случайных чисел больше возможностей получить достаточное количество энтропии.
Необходимо получить много случайных чисел. Желательно, чтобы Вы
в процессе генерации выполняли какие-то другие действия (печатать
на клавиатуре, движения мыши, обращения к дискам); это даст генератору
случайных чисел больше возможностей получить достаточное количество энтропии.
gpg: создан каталог '/root/.gnupg/openpgp-revocs.d'
gpg: сертификат отзыва записан в '/root/.gnupg/openpgp-revocs.d/9D0C61A13D8AFC5E8E29006E6D2B28B637D4588B.rev'.
открытый и секретный ключи созданы и подписаны.

pub  rsa4096 2025-02-25 [SC]
     9D0C61A13D8AFC5E8E29006E6D2B28B637D4588B
uid                          doveretennikov <domiswww2018@gmail.com>
sub  rsa4096 2025-02-25 [E]

[root@vbox ~]# 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, 0m, 0f, 1u
[keyboxd]
-----
sec  rsa4096/6D2B28B637D4588B 2025-02-25 [SC]
     9D0C61A13D8AFC5E8E29006E6D2B28B637D4588B
uid      [ абсолютно ] doveretennikov <domiswww2018@gmail.com>
ssb  rsa4096/D0B9D23DE422690D 2025-02-25 [E]

[root@vbox ~]# gpg --armor --export <PGP Fingerprint> | xclip -set clip
-bash: синтаксическая ошибка рядом с неожиданным маркером «|»
[root@vbox ~]# gpg --armor --export <^C
[root@vbox ~]# gpg --armor --export <PGP Fingerprint> | xclip -set clip
-bash: синтаксическая ошибка рядом с неожиданным маркером «|»
[root@vbox ~]#
```

Рис. 6: Список ключей

Копирую сгенерированный PGP ключ в буфер обмена.

```
foot
-bash: синтаксическая ошибка рядом с неожиданным маркером «|»
[root@vbox ~]# gpg --armor --export <^C
[root@vbox ~]# gpg --armor --export <PGP Fingerprint> | xclip -set clip
-bash: синтаксическая ошибка рядом с неожиданным маркером «|»
[root@vbox ~]# gpg --armor --export <PGP Fingerprint>
-bash: синтаксическая ошибка рядом с неожиданным маркером «newline»
[root@vbox ~]# gpg --armor --export 602B28B637D4588B | xclip -sel clip
-bash: xclip: команда не найдена
gpg: [stdout]: write error: Обрыв канала
gpg: filter_flush failed on close: Обрыв канала
[root@vbox ~]# gpg --armor --export 602B28B637D4588B
-----BEGIN PGP PUBLIC KEY BLOCK-----

mQINBGe9StABEACTDbcqxud+7hihyf5JikWUJZIVs/cVqAz1s/rS8MTbq/0ZJ6XM
6JmMoA/QISKqG4/v9YVS8A01gxrxm1eM972/5LMt9IOvuk+MMx1sJNh3yDYkduAd
oIK+YPgdalNouBVVehEQE1mxRBoj/09zsZMoKdos0pHUTCvv2MwPaszm0BCui3Ef
HVjaTeL5AQaa33D2twVZcPQhgm1CBA0/eJUjyRrWgOV6NoOnfph3nCSb0vEea/y
FyS8juM/xWzCaQqxmG+KfwzKD27E7Ktdw9Iuc+hel+Bf2pAP13GCgiDu0PZCK3it
Sc03MGoUfjDnB4cvx1w192CdoDn9Sc6W3q1TIBNffj4754QNsJE0UH+6UvAHmTG1
fg16pQeFemxtjEeTdRLyAHMI0F2nEKEVPhRoMvn+35wFNLMSXc1ewdmIay3N7D2
fASj9b+ssSta8Q19ocdY9GbQNm2TT6cSEgy31BYXGwEN3g9Myexvts5v04qHLEc
omG4x7mNd75Yu11JLVSDASvd1Fh+wsmwSjIymPbcOTOF0W5aScR0u38AfScsW0
qgxZ8C3D90E6cTMzu8ZdyYq2bVLI9rRg4vVhv8dU0sX/db0LY0wcDL0qB2CT1M5h
NDu0SEBASY/ECC5IwoNOAqudH0SBech9pHKbInkodCHFXnu3uSsvy1vU9wARAQAB
tCdkb3ZlcmV0ZW5ualwtdiA8ZG9taXN3d3cyMDE4Q0dtYWI1sLmNvbT6JA1EEWEI
ADSwIQSdQdGhPy8Xo4pAGStKy12N9RY1wUCZ73m8A1bAwULCQgHAgI1AgYVCgkI
CwIEFgIDAQIeBwIXgAAKRbtkKy12N9RY14J00/9102bd17RMBRG31shS1TOAFEv
pc9p/45pbHVWg20GgRDS08vk2Td1Qhsn6cPwXpV19/zh3jPHKSJQd7NPM0RvbpL
vRm74kR45Zw91jIOPCyZDR9Vem11oSreQnwzcFL1XhU2GdGIXj1RNVTDby8ZNMXL
72hzyrvaz2BvWV5190jcAE61fjXgz3N90dTz/R/1Hxbk4jUtR4bpE+1EN3zyJu14u
1zwvOWgtS9JmKqGSGwE7J2kod/upMpwE1uyxx9qJ6no+B+pZzYsv1fDQtcCeeFN
```

Рис. 7: Копирование ключа PGP



## Настройка автоматических подписей коммитов git

Используя введенный email, указываем git применять его при подписи коммитов.

```
foot
q2z+6btN72cgnaG9m8W0Z19ZFnK2HtG5LEDBBKlw7ea+U9mz001WBxqKv0vKvW
1qeXeB2Ds jXEV4i+119t1yYbrXDR3Dm0BDCCvy1HnQLL2Xt2600RGc+z0/c0sTio
DjReNLr21KH2syM1+TTVf5FMckemcm3ZT+Un7H44RvUSm7wJ/M+gCr1TRa00UKYM
8xaf5S5sjVD04PyyZyuqmjnk1iVnTdpGjLZ0xjmay44sHsD+09w+Zk1syUV08LL0Q
K2PPsNPpdjR3b19Qhym0L0wBxwQNhuhz1Sjy42Fnhpe3w05YLhV82uaHk+xGh0x
ESRJqCRQYCEHJ1UTnodmESBQZ9k4a6MGmQaNUg/6uMd3XYtvxYZREfLpQQt0wtj22
20PBE1BEqGL0wPx1X98+WMpmheJz47zH1G2QmM1bagXUuJc4+H7FdSuwi j6PBPHj
Yk6G1L003yxV1dEC2xZh3uAmpyS71B4m5GHEzhW2ZQ1+chs jMZNzMNKjLxCDNA+
IbXXJB757F8AEQEAAYKCNqQYAQgAIBYHBJ0MYaE9ivxej1kAbm0zKLY31FiLBQJn
vebQAhSMAAoJEG0zKLY31FiLmYIP+gNG8+sA0Loc/cEu5YKYNya/9NegeHVPLwm
Yg9EUJQo0LE9Hud59Bt6I/19rJ4RKUFvIa001JG1aqTt8+NdeL121RUC2N5whAk
WXX/8Cj2c8nETTKYmqF83cn/mNfVTe7ueLdANePNBQXPcUnj jbp1DABPm9f65MNz
8N8gzodALDj/10vKqo0smmNK2RTk5IxbKtAb2xLB1j8e1wq15/57MqnZBdydF2a
+Nz1BmwprUXCACExpPRnywS1HvSoYQsa9uXXFf0Q3w51zRzpFxa7PuY57b0FKR6N
0Yxd2uBa+NM1IaxgYUK67Pb0V4/F2ja8mo/4w+gFLIFB2Q1QdmMMw7v0yMiGM5q
h/SM0uzB1awNkX652k8NwaqpjMcS19AxtzIXXsgVBQa/GpA18FmT541FBggeAygDy
rJmZKwNBZ1X07J//u/uF9H/1n2F/P/zkYrahTug9i2uqGVkKPlcQvThubHfjbbK/
r6VbXoqSq9ndya016x7Hw3T+SW/ Jx3BINm1ZyM01g1/j1559e49WtQ84nSI+y3
dKzTFcCATJ23VC122C1x20g2pZB9KUq8yC0Y+GRDam1a8mVshTmJEF0G10m/0Fm
NBnEg0rGqz4cda7Fn7zIeaoytCJ0dFAY1cx2U7kgcQ+SdI2INV5Sk//ey+jd5GYK
5p4cAo4+
=yzZM
-----END PGP PUBLIC KEY BLOCK-----
[root@vbox ~]# git config --global user.signkey 602828863704588B
[root@vbox ~]# git config --global commit.gpgsign true
[root@vbox ~]# git config --global gpg.program $(which gpg2)
[root@vbox ~]# gh auth login
? Where do you use GitHub? [Use arrows to move, type to filter]
> GitHub.com
  Other
```

Рис. 8: Настройки автоматических подписей коммитов git

Авторизовываюсь через gh.

```
root
rJmZWKwB21X07J/U/uF9H/1n2F/P/zkYrahTUq9i2uqGVkKPlcQvThubhFjbbk/
r6VBXoqSQq9mDya016x7Hw3T+SW/Jx3B1Nm12yMOigI/j1S59e49WtQ8f4nS1+y3
dKrTFcCATJ23VC122Clx20g2pZB9KUq8yC0Y+GRDam1a8mVshTmuEFOGP10m/0Fm
NBnEg0rGqz4cda7Fn7zIeaoytCJ0dFAV1cx2U7kgcQ+sdI2INV5Sk//ey+jd5GYK
Sp4cAo4+
=yzZM
-----END PGP PUBLIC KEY BLOCK-----
[root@vbox ~]# git config --global user.signkey 602B28B637D4588B
[root@vbox ~]# git config --global commit.gpgsign true
[root@vbox ~]# git config --global gpg.program $(which gpg2)
[root@vbox ~]# gh auth login
? Where do you use GitHub? GitHub.com
? What is your preferred protocol for Git operations on this host? SSH
? Upload your SSH public key to your GitHub account? Skip
? How would you like to authenticate GitHub CLI? Login with a web browser

! First copy your one-time code: 458A-AF18
Press Enter to open https://github.com/login/device in your browser...
restorecon: SELinux: Could not get canonical path for /root/.mozilla/firefox/* /gmp-widevinecdm/* restorecon: No such
file or directory.
Authorization required, but no authorization protocol specified

Error: cannot open display: :0

✓ Authentication complete.
- gh config set -h github.com git_protocol ssh
✓ Configured git protocol
! Authentication credentials saved in plain text
✓ Logged in as doveretennikov
[root@vbox ~]#
[root@vbox ~]#
```

Рис. 9: Настройка gh

# Создание репозитория курса на основе шаблона

Создаю репозиторий курса на основе шаблона.

```
foot
~yzzM
-----END PGP PUBLIC KEY BLOCK-----
[root@vbox ~]# git config --global user.signkey 602B28B637D4588B
[root@vbox ~]# git config --global commit.gpgsign true
[root@vbox ~]# git config --global gpg.program $(which gpg2)
[root@vbox ~]# gh auth login
? Where do you use GitHub? GitHub.com
? What is your preferred protocol for Git operations on this host? SSH
? Upload your SSH public key to your GitHub account? Skip
? How would you like to authenticate GitHub CLI? Login with a web browser

! First copy your one-time code: 458A-AF19
Press Enter to open https://github.com/login/device in your browser...
restorecon: SELinux: Could not get canonical path for /root/.mozilla/firefox/*/gmp-widevinecdm/* restorecon: No such
file or directory.
Authorization required, but no authorization protocol specified

Error: cannot open display: :0

✓ Authentication complete.
- gh config set -h github.com git_protocol ssh
✓ Configured git protocol
! Authentication credentials saved in plain text
✓ Logged in as doveretennikov
[root@vbox ~]#
[root@vbox ~]# mkdir -p ~/work/study/2024-2025/"Операционные системы"
[root@vbox ~]# cd ~/work/study/2024-2025/"Операционные системы"
[root@vbox Операционные системы]# gh repo create study_2024-2025_os-intro --template=yamadaxma/course-directory-student-template --public
✓ Created repository doveretennikov/study_2024-2025_os-intro on GitHub
https://github.com/doveretennikov/study_2024-2025_os-intro
```

Рис. 10: Процесс создания репозитория курса на основе шаблона

Клонирую шаблон репозитория.

```
foot
-Intro
Клонирование в «os-intro»...
git@github.com: Permission denied (publickey).
fatal: Не удалось прочитать из внешнего репозитория.

Удостоверьтесь, что у вас есть необходимые права доступа
и репозиторий существует.
[root@vbox Операционные системы]# mc

[root@vbox .ssh]# cat id_ed25519.pub
ssh-ed25519 AAAAC3NzaC1lZD11NTE5AAAAIJiY77bJghz2RX341f0ygh+04D9WvORkucum1Wn+4Da/ root@vbox

[root@vbox Операционные системы]# git clone --recursive git@github.com:doveretennikov/study_2024-2025_os-intro.git os-
-Intro
Клонирование в «os-intro»...
remote: Enumerating objects: 36, done.
remote: Counting objects: 100% (36/36), done.
remote: Compressing objects: 100% (35/35), done.
remote: Total 36 (delta 1), reused 21 (delta 0), pack-reused 0 (from 0)
Получение объектов: 100% (36/36), 19.37 Киб | 19.37 Миб/с, готово.
Определение изменений: 100% (1/1), готово.
Подмодуль «template/presentation» (https://github.com/yamadharma/academic-presentation-markdown-template.git) зарегистрирован по пути «template/presentation»
Подмодуль «template/report» (https://github.com/yamadharma/academic-laboratory-report-template.git) зарегистрирован по пути «template/report»
Клонирование в «/root/work/study/2024-2025/Операционные системы/os-intro/template/presentation»...
remote: Enumerating objects: 111, done.
remote: Counting objects: 100% (111/111), done.
remote: Compressing objects: 100% (77/77), done.
remote: Total 111 (delta 42), reused 100 (delta 31), pack-reused 0 (from 0)
Получение объектов: 100% (111/111), 102.17 Киб | 1.23 Миб/с, готово.
```

Рис. 11: Клонирования шаблона репозитория

## Настройка каталога курса

Настраиваю каталог курса, удаляю лишние файлы и создаю необходимые каталоги, затем отправляю всё на сервер.

```
root
[root@vbox Операционные системы]# rm package.json
rm: невозможно удалить 'package.json': Нет такого файла или каталога
[root@vbox Операционные системы]# cd
[root@vbox ~]# cd ~/work/study/2024-2025/"Операционные системы"/os-intro
[root@vbox os-intro]# rm package.json
rm: удалить обычный файл 'package.json'?
[root@vbox os-intro]# echo os-intro > COURSE
[root@vbox os-intro]# make
Usage:
  make <target>

Targets:
  list           List of courses
  prepare       Generate directories structure
  submodule      Update submodules

[root@vbox os-intro]# git add .
[root@vbox os-intro]# git commit -am 'feat(main): make course structure'
(master 3cc4ddd) feat(main): make course structure
1 file changed, 1 insertion(+)
[root@vbox os-intro]# git push
Перечисление объектов: 5, готово.
Подсчет объектов: 100% (5/5), готово.
При сжатии изменений используется до 4 потоков
Сжатие объектов: 100% (2/2), готово.
Запись объектов: 100% (3/3), 949 байтов | 949.00 Киб/с, готово.
Total 3 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To github.com:doveretennikov/study_2024-2025_os-intro.git
  5665798..3cc4ddd master -> master
[root@vbox os-intro]#
```

Рис. 12: Настройка каталога курса

## Ответы на контрольные вопросы

---

1. 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. Системы контроля версий поддерживают возможность отслеживания и разрешения конфликтов, которые могут возникнуть при работе нескольких человек над одним файлом. Можно объединить (слить) изменения, сделанные разными участниками (автоматически или вручную), вручную выбрать нужную версию, отменить изменения вовсе или заблокировать файлы для изменения. В зависимости от настроек блокировка не позволяет другим пользователям получить рабочую копию или препятствует изменению рабочей копии файла средствами файловой системы ОС, обеспечивая таким образом, привилегированный доступ только одному пользователю,

2. В классических системах контроля версий используется централизованная модель, предполагающая наличие единого репозитория для хранения файлов. Выполнение большинства функций по управлению версиями осуществляется специальным сервером. 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 момент. Сервер может сохранять не полную версию изменённых файлов, а производить так называемую дельта-компрессию — сохранять только изменения между последовательными версиями, что позволяет уменьшить объём хранимых данных.



3. В классических системах контроля версий используется централизованная модель, предполагающая наличие единого репозитория для хранения файлов. В отличие от классических, в распределённых системах контроля версий центральный репозиторий не является обязательным. Среди классических VCS наиболее известны CVS, Subversion, а среди распределённых — Git, Bazaar, Mercurial.
4. 4.1 получение файлов с сервера 2. изменения файлов 3. отправка файлов на сервер.
5. 5.1 получение нужной версии файлов 2. изменения файлов 3. отправка файлов на сервер, объединение изменений если возникают конфликты при работе над файлом несколькими людьми.

## Результаты

---

После выполнения данной лабораторной работы я освоил навыка настройки и работы git, изучил идеологию и применения контроля версий.

## Список литературы

---

Курс Архитектура компьютеров и операционные системы. Раздел “Операционные системы”  
Лабораторная работа № 2 ТУИС РУДН.