

Отчёт по лабораторной работе №2

Управление версиями

Сморчков Дмитрий НБИбд-01-21

Содержание

1	Цель работы	4
2	Выполнение лабораторной работы	5
3	Вывод	10
4	Контрольные вопросы	11
	Список литературы	15

List of Figures

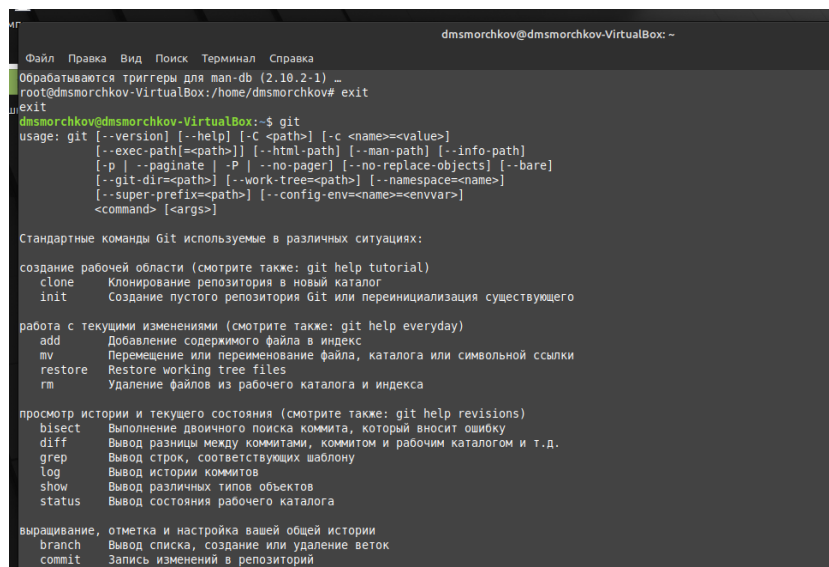
2.1	Загрузка пакетов	5
2.2	Параметры репозитория	6
2.3	rsa-4096	6
2.4	ed25519	7
2.5	GPG ключ	7
2.6	GPG ключ	8
2.7	Параметры репозитория	8
2.8	Связь репозитория с аккаунтом	9
2.9	Загрузка шаблона	9
2.10	Первый коммит	9

1 Цель работы

Целью данной работы является изучение идеологии и применения средств контроля версий и освоение умений работать с git.

2 Выполнение лабораторной работы

Устанавливаем git, git-flow и gh.



```
dmsmorchkov@dmsmorchkov-VirtualBox: ~  
Файл Правка Вид Поиск Терминал Справка  
Обрабатываются триггеры для man-db (2.10.2-1) ...  
root@dmsmorchkov-VirtualBox:/home/dmsmorchkov# exit  
exit  
dmsmorchkov@dmsmorchkov-VirtualBox:~$ git  
usage: git [-version] [-help] [-C <path>] [-c <name>=<value>]  
          [--exec-path=<path>] [--html-path] [--man-path] [--info-path]  
          [-p | --paginate] [-P | --no-pager] [--no-replace-objects] [--bare]  
          [--git-dir=<path>] [--work-tree=<path>] [--namespace=<name>]  
          [--super-prefix=<path>] [--config-env=<name>=<envvar>]  
          <command> [<args>]  
  
Стандартные команды Git используемые в различных ситуациях:  
  
создание рабочей области (смотрите также: git help tutorial)  
  clone  Клонирование репозитория в новый каталог  
  init   Создание пустого репозитория Git или переинициализация существующего  
  
работа с текущими изменениями (смотрите также: git help everyday)  
  add    Добавление содержимого файла в индекс  
  mv     Перемещение или переименование файла, каталога или символической ссылки  
  restore Restore working tree files  
  rm     Удаление файлов из рабочего каталога и индекса  
  
просмотр истории и текущего состояния (смотрите также: git help revisions)  
  bisect Выполнение двоичного поиска коммита, который вносит ошибку  
  diff   Вывод разницы между коммитами, коммитом и рабочим каталогом и т.д.  
  grep   Вывод строк, соответствующих шаблону  
  log    Вывод истории коммитов  
  show   Вывод различных типов объектов  
  status Вывод состояния рабочего каталога  
  
выращивание, отметка и настройка вашей общей истории  
  branch Вывод списка, создание или удаление веток  
  commit  Запись изменений в репозиторий
```

Figure 2.1: Загрузка пакетов

Зададим имя и email владельца репозитория, кодировку и прочие параметры.

```
дмсморчков@dmsmorchkov-VirtualBox: ~
Файл Правка Вид Поиск Терминал Справка
diff Вывод разницы между коммитами, коммитом и рабочим каталогом и т.д.
grep Вывод строк, соответствующих шаблону
log Вывод истории коммитов
show Вывод различных типов объектов
status Вывод состояния рабочего каталога

выращивание, отметка и настройка вашей общей истории
branch Вывод списка, создание или удаление веток
commit Запись изменений в репозиторий
merge Объединение одной или нескольких историй разработки вместе
rebase Повторное применение коммитов над вершущей другой ветки
reset Сброс текущего состояния HEAD на указанное состояние
switch Switch branches
tag Создание, вывод списка, удаление или проверка метки, подписанной с помощью GPG

совместная работа (смотрите также: git help workflows)
fetch Загрузка объектов и ссылок из другого репозитория
pull Извлечение изменений и объединение с другим репозиторием или локальной веткой
push Обновление внешних ссылок и связанных объектов

'git help -a' and 'git help -g' list available subcommands and some
concept guides. See 'git help <command>' or 'git help <concept>'
to read about a specific subcommand or concept.
See 'git help git' for an overview of the system.
дмсморчков@dmsmorchkov-VirtualBox:~$
дмсморчков@dmsmorchkov-VirtualBox:~$
дмсморчков@dmsmorchkov-VirtualBox:~$
дмсморчков@dmsmorchkov-VirtualBox:~$ git config --global user.name "dmsmorchkov"
дмсморчков@dmsmorchkov-VirtualBox:~$ git config --global user.email "1032210090@pfur.ru"
дмсморчков@dmsmorchkov-VirtualBox:~$ git config --global core.quotepath false
дмсморчков@dmsmorchkov-VirtualBox:~$ git config --global init.defaultBranch master
дмсморчков@dmsmorchkov-VirtualBox:~$ git config --global core.autocrlf input
дмсморчков@dmsmorchkov-VirtualBox:~$ git config --global core.safecrlf warn
дмсморчков@dmsmorchkov-VirtualBox:~$
```

Figure 2.2: Параметры репозитория

Создаем SSH ключи

```
дмсморчков@dmsmorchkov-VirtualBox: ~
Файл Правка Вид Поиск Терминал Справка
to read about a specific subcommand or concept.
See 'git help git' for an overview of the system.
дмсморчков@dmsmorchkov-VirtualBox:~$
дмсморчков@dmsmorchkov-VirtualBox:~$
дмсморчков@dmsmorchkov-VirtualBox:~$
дмсморчков@dmsmorchkov-VirtualBox:~$ git config --global user.name "dmsmorchkov"
дмсморчков@dmsmorchkov-VirtualBox:~$ git config --global user.email "1032210090@pfur.ru"
дмсморчков@dmsmorchkov-VirtualBox:~$ git config --global core.quotepath false
дмсморчков@dmsmorchkov-VirtualBox:~$ git config --global init.defaultBranch master
дмсморчков@dmsmorchkov-VirtualBox:~$ git config --global core.autocrlf input
дмсморчков@dmsmorchkov-VirtualBox:~$ git config --global core.safecrlf warn
дмсморчков@dmsmorchkov-VirtualBox:~$ ssh-keygen -t rsa -b 4096
Generating public/private rsa key pair.
Enter file in which to save the key (/home/dmsmorchkov/.ssh/id_rsa):
Created directory '/home/dmsmorchkov/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/dmsmorchkov/.ssh/id_rsa
Your public key has been saved in /home/dmsmorchkov/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:+ktJ0hhKBjQJv/VjfiM7J0/9DZIr7gfGDzXqZHTcx04 дмсморчков@dmsmorchkov-VirtualBox
The key's randomart image is:
+---[RSA 4096]-----+
|
|  o+
| = . . o
| o . = +
| o o oS+ =
| . + + +0oE..
| o .B=+++ .
| o+*+ o+ o
| o@*+ . .
+---[SHA256]-----+
дмсморчков@dmsmorchkov-VirtualBox:~$
```

Figure 2.3: rsa-4096

```
dmsmorchkov@dmsmorchkov-VirtualBox: ~
Файл Правка Вид Поиск Терминал Справка
The key's randomart image is:
+---[RSA 4096]-----+
|
|o+
|= . . o
|o . = +
|o o oS+ =
|. + + +0oE..
|o . B=+++ .
|o*** o+ o
|o@*=. . .
+---[SHA256]-----+
dmsmorchkov@dmsmorchkov-VirtualBox:~$ ssh-keygen -t ed25519
Generating public/private ed25519 key pair.
Enter file in which to save the key (/home/dmsmorchkov/.ssh/id_ed25519):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/dmsmorchkov/.ssh/id_ed25519
Your public key has been saved in /home/dmsmorchkov/.ssh/id_ed25519.pub
The key fingerprint is:
SHA256:BhZ5OyLM7k6dgKk8IWJCLe2Tx4FIZB2uUt4YTzLbvKE dmsmorchkov@dmsmorchkov-VirtualBox
The key's randomart image is:
+---[ED25519 256]---+
|o+=.o .
|. +. = .o..
|. B. +++. o
|++o/o+=.+
|++o0. +So
|. +. oo.o
|. E..o
|.
+---[SHA256]-----+
dmsmorchkov@dmsmorchkov-VirtualBox:~$
```

Figure 2.4: ed25519

Создаем GPG ключ

```
dmsmorchkov@dmsmorchkov-VirtualBox: ~
Файл Правка Вид Поиск Терминал Справка
Срок действия ключа не ограничен
Все верно? (y/N) y
GnuPG должен составить идентификатор пользователя для идентификации ключа.
Ваше полное имя: dmsmorchkov
Адрес электронной почты: 1032210090@pfur.ru
Примечание:
Вы выбрали следующий идентификатор пользователя:
"dmsmorchkov <1032210090@pfur.ru>"
Сменить (N)Имя, (C)Примечание, (E)Адрес; (O)Принять/(O)Выход? O
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.
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: /home/dmsmorchkov/.gnupg/trustdb.gpg: создана таблица доверия
gpg: ключ B8ED422E174AFB1D помечен как абсолютно доверенный
gpg: создан каталог '/home/dmsmorchkov/.gnupg/openpgp-revocs.d'
gpg: сертификат отзыва записан в '/home/dmsmorchkov/.gnupg/openpgp-revocs.d/DA310BD10FF2A68D8F31C4F588ED422E174AFB1D.rev'.
открытый и секретный ключи созданы и подписаны.

pub   rsa4096 2022-08-26 [SC]
      DA310BD10FF2A68D8F31C4F588ED422E174AFB1D
uid    dmsmorchkov <1032210090@pfur.ru>
sub    rsa4096 2022-08-26 [E]

dmsmorchkov@dmsmorchkov-VirtualBox:~$
dmsmorchkov@dmsmorchkov-VirtualBox:~$
```

Figure 2.5: GPG ключ


Добавляем GPG ключ в аккаунт

Check out our guide to [generating SSH keys](#) or troubleshoot [common SSH problems](#).

GPG keys

[New GPG key](#)

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



1

Email address: 1032210090@pfur.ru

Key ID: B8ED422E174AFB1D

Subkeys: 62A50D174EAE9C2A

Added on 26 Aug 2022

Delete

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

Figure 2.6: GPG ключ

Настройка автоматических подписей коммитов git

```
DBjUZJ2MOCXktj38UHHb9tjyot0rEozwi0beJwqrFuGSCTSFqGGszWAZtMNTD0AR
AQAB1QI2B8BgCgAgFiEE2jEL80/yp02PMCT1u01CLhdK+x8FAMi0JACGwACgkQ
u01CLhdK+x32ug/9GPC0Kt0amT9Jx86f0htCJYtL1M4/DYpyD52X8eKiMzxUDr0s
uv202DSVe0KVwEUEp88doo751cWDlgQXZ8vqBF7woNj7qb6PrUuoNefsV4DYqxy
cNdXdmKmsb6yIvFd5j0917wFcSzdVwfZtIMgKj92kuAjPYfwaVcWSOR10rh3m9
jbl0/sAJxhRjget6oFy177RS9u33k1CX6NoKEK91gIbPr6QI8dcvT8Snl7x7F6e+
QGaVLG9tcrzvIGFRvbqVnszx2q1QH5wmvK03k0U6E6DSbYbnPmUfRwxV8w7pa99
YM0I/BV0HMFmouDhpR+985BDSAk88yl8kQmNgEPFuz0NFj6BxZrY84G8812AzkRH
TogIGgc/IA6VS/4tr3ZFfYgFBeoCOKB80c3LzDF2CwLLOSydNX6u0cHl0nb+Qw
USCBuR0/9wMghyzUKTMe2wXpB00et7/rbU//0yntYANqeS59922qrlVU8gH0a00ap
KDM7FAKygZ57ndh/gQW0JHsvzqN2hzNCqX0x82t0YJaqrjaYK4s5X+4W5rp22qD
4VU91cnseQSA3EamVczfKte3wGN8fgeB/dQfc0wmV81vtz8B9ttto5RzfesMXvQj
ADMeNvxom1ETROqesf15fBRouVpjLYXT2jdsHtuFERLYR+Qx7rlo898ink=
=qYtn
-----END PGP PUBLIC KEY BLOCK-----
dmsmorchkov@dmsmorchkov-VirtualBox:~$
dmsmorchkov@dmsmorchkov-VirtualBox:~$
dmsmorchkov@dmsmorchkov-VirtualBox:~$ git config --global user.signingkey B8ED422E174AFB1D
dmsmorchkov@dmsmorchkov-VirtualBox:~$ git config --global commit.gpgsign true
dmsmorchkov@dmsmorchkov-VirtualBox:~$ git config --global gpg.program $(which gpg2)
dmsmorchkov@dmsmorchkov-VirtualBox:~$
```

Figure 2.7: Параметры репозитория

Настройка gh


```
ADMeNwXom1ETR0qesf15fBRouVpJLYXKT2jdsHtufERLYR+Qx7rlo898ink=
=qYIn
----END PGP PUBLIC KEY BLOCK-----
dmsmorchkov@dmsmorchkov-VirtualBox:~$
dmsmorchkov@dmsmorchkov-VirtualBox:~$
dmsmorchkov@dmsmorchkov-VirtualBox:~$ git config --global user.signingkey B8ED422E174AFB1D
dmsmorchkov@dmsmorchkov-VirtualBox:~$ git config --global commit.gpgsign true
dmsmorchkov@dmsmorchkov-VirtualBox:~$ git config --global gpg.program $(which gpg2)
dmsmorchkov@dmsmorchkov-VirtualBox:~$ gh auth login
? What account do you want to log into? GitHub.com
? What is your preferred protocol for Git operations? SSH
? Upload your SSH public key to your GitHub account? /home/dmsmorchkov/.ssh/id_rsa.pub
? Title for your SSH key: GitHub CLI
? How would you like to authenticate GitHub CLI? Login with a web browser
? First copy your one-time code: 22BE-37C7
Press Enter to open github.com in your browser...
2022-08-26T10:33:18Z ERROR viaduct::backend::ffi Missing HTTP status
2022-08-26T10:33:18Z ERROR viaduct::backend::ffi Missing HTTP status
Authentication complete.
- gh config set -h github.com git_protocol ssh
Configured git protocol
Uploaded the SSH key to your GitHub account: /home/dmsmorchkov/.ssh/id_rsa.pub
Logged in as dmsmorchkov
dmsmorchkov@dmsmorchkov-VirtualBox:~$
```

Figure 2.8: Связь репозитория с аккаунтом

Загрузка шаблона репозитория и синхронизация

```
dmsmorchkov@dmsmorchkov-VirtualBox:~/work/study/2021-2022/Операционные системы$ git clone --recursive git@github.com:magorunov/study_2021-2022_os-intro.git
os-intro
Клонирование в «os-intro»...
The authenticity of host 'github.com [140.82.121.3]' can't be established.
ED25519 key fingerprint is SHA256:0LY3wvV6TujHbpZisf/zLDARzPMSVHdkr4uVCQqu.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'github.com' (ED25519) to the list of known hosts.
remote: Enumerating objects: 315, done.
remote: Counting objects: 100% (69/69), done.
remote: Compressing objects: 100% (56/56), done.
remote: Total 315 (delta 11), reused 64 (delta 11), pack-reused 246
Получение объектов: 100% (315/315), 89.68 МБ/s | 7.64 МБ/с, готово.
Определение изменений: 100% (41/41), 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 «/home/dmsmorchkov/work/study/2021-2022/Операционные системы/os-intro/template/presentation»...
remote: Enumerating objects: 59, done.
remote: Counting objects: 100% (53/53), done.
remote: Compressing objects: 100% (39/39), done.
remote: Total 59 (delta 14), reused 81 (delta 12), pack-reused 8
Получение объектов: 100% (53/53), 81.81 КБ/s | 296.00 КБ/с, готово.
Определение изменений: 100% (14/14), готово.
Клонирование в «/home/dmsmorchkov/work/study/2021-2022/Операционные системы/os-intro/template/report»...
remote: Enumerating objects: 78, done.
remote: Counting objects: 100% (78/78), done.
remote: Compressing objects: 100% (52/52), done.
remote: Total 78 (delta 31), reused 69 (delta 22), pack-reused 8
Получение объектов: 100% (78/78), 292.27 КБ/s | 3.21 МБ/с, готово.
Определение изменений: 100% (31/31), готово.
Submodule path «template/presentation»: checked out '3eabb7586f8a9aded2b586cd18186625b228b93'
Submodule path «template/report»: checked out 'd7b2ef88f8def3b9a496f8695277469a1a7842a'
dmsmorchkov@dmsmorchkov-VirtualBox:~/work/study/2021-2022/Операционные системы$
```

Figure 2.9: Загрузка шаблона

Подготовка репозитория и коммит изменений

```
dmsmorchkov@dmsmorchkov-VirtualBox:~/work/study/2021-2022/Операционные системы/os-intro$ git add .
dmsmorchkov@dmsmorchkov-VirtualBox:~/work/study/2021-2022/Операционные системы/os-intro$ git commit -am "repo"
[master cb53f8f] repo
16 files changed, 1589 insertions(+), 14 deletions(-)
create mode 100644 labs/lab[01..15]/presentation/Makefile
create mode 100644 labs/lab[01..15]/presentation/presentation.md
create mode 100644 labs/lab[01..15]/report/Makefile
create mode 100644 labs/lab[01..15]/report/bib/cite.bib
create mode 100644 labs/lab[01..15]/report/image/placeimg 800 600 tech.jpg
create mode 100644 labs/lab[01..15]/report/pandoc/csl/gost-r-7-0-5-2006-numeric.csl
create mode 100644 labs/lab[01..15]/report/report.md
delete mode 100644 package.json
create mode 100644 project-personal/stage[1..6]/presentation/Makefile
create mode 100644 project-personal/stage[1..6]/presentation/presentation.md
create mode 100644 project-personal/stage[1..6]/report/Makefile
create mode 100644 project-personal/stage[1..6]/report/bib/cite.bib
create mode 100644 project-personal/stage[1..6]/report/image/placeimg 800 600 tech.jpg
create mode 100644 project-personal/stage[1..6]/report/pandoc/csl/gost-r-7-0-5-2006-numeric.csl
create mode 100644 project-personal/stage[1..6]/report/report.md
create mode 100644 structure
dmsmorchkov@dmsmorchkov-VirtualBox:~/work/study/2021-2022/Операционные системы/os-intro$ git push
Перечисление объектов: 20, готово.
Подсчет объектов: 100% (20/20), готово.
При сжатии изменений используется до 4 потоков
Сжатие объектов: 100% (14/14), готово.
Запись объектов: 100% (19/19), 266.44 КБ/s | 2.05 МБ/с, готово.
Всего 19 (изменений 2), повторно использовано 0 (изменений 0), повторно использовано пакетов 0
remote: Resolving deltas: 100% (2/2), completed with 1 local object.
To github.com:dmsmorchkov/study_2021-2022_os-intro.git
da66529..cb53f8f master -> master
```

Figure 2.10: Первый коммит

3 Вывод

Мы приобрели практические навыки работы с сервисом github.

4 Контрольные вопросы

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

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 - пространство на накопителе где расположен репозиторий
- commit - сохранение состояния хранилища
- история - список изменений хранилища (коммитов)
- рабочая копия - локальная копия сетевого репозитория, в которой работает программист. Текущее состояние файлов проекта, основанное на версии, загруженной из хранилища (обычно на последней)

3. Что представляют собой и чем отличаются централизованные и децентрализованные VCS? Приведите примеры VCS каждого вида.

Централизованные системы контроля версий представляют собой приложения типа клиент-сервер, когда репозиторий проекта существует в единственном экземпляре и хранится на сервере. Доступ к нему осуществлялся через специальное клиентское приложение. В качестве примеров таких программных продуктов можно привести CVS, Subversion.

Распределенные системы контроля версий (Distributed Version Control System, DVCS) позволяют хранить репозиторий (его копию) у каждого разработчика, работающего с данной системой. При этом можно выделить центральный репозиторий (условно), в который будут отправляться изменения из локальных и, с ним же эти локальные репозитории будут синхронизироваться. При работе с такой системой, пользователи периодически синхронизируют свои локальные репозитории с центральным и работают непосредственно со своей локальной копией. После внесения достаточного количества изменений в локальную копию они (изменения) отправляются на сервер. При этом сервер, чаще всего, выбирается условно, т.к. в большинстве DVCS нет такого понятия как “выделенный сервер с центральным репозиторием”.

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

Один пользователь работает над проектом и по мере необходимости делает коммиты, сохраняя определенные этапы.

5. Опишите порядок работы с общим хранилищем VCS.

Несколько пользователей работают каждый над своей частью проекта. При этом каждый должен работать в своей ветки. При завершении работы ветка пользователя сливается с основной веткой проекта.

6. Каковы основные задачи, решаемые инструментальным средством git?

- Ведение истории версий проекта: журнал (log), метки (tags), ветвления (branches).

- Работа с изменениями: выявление (diff), слияние (patch, merge).
- Обеспечение совместной работы: получение версии с сервера, загрузка обновлений на сервер.

7. Назовите и дайте краткую характеристику командам git.

- git config - установка параметров
- git status - полный список изменений файлов, ожидающих коммита
- git add . - сделать все измененные файлы готовыми для коммита.
- git commit -m "[descriptive message]" - записать изменения с заданным сообщением.
- git branch - список всех локальных веток в текущей директории.
- git checkout [branch-name] - переключиться на указанную ветку и обновить рабочую директорию.
- git merge [branch] — соединить изменения в текущей ветке с изменениями из заданной.
- git push - запустить текущую ветку в удаленную ветку.
- git pull - загрузить историю и изменения удаленной ветки и произвести слияние с текущей веткой.

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

- git remote add [имя] [url] — добавляет удалённый репозиторий с заданным именем;
- git remote remove [имя] — удаляет удалённый репозиторий с заданным именем;
- git remote rename [старое имя] [новое имя] — переименовывает удалённый репозиторий;
- git remote set-url [имя] [url] — присваивает репозиторию с именем новый адрес;

- `git remote show [имя]` — показывает информацию о репозитории.

9. Что такое и зачем могут быть нужны ветви (branches)?

Ветвление — это возможность работать над разными версиями проекта: вместо одного списка с упорядоченными коммитами история будет расходиться в определённых точках. Каждая ветвь содержит легковесный указатель HEAD на последний коммит, что позволяет без лишних затрат создать много веток. Ветка по умолчанию называется `master`, но лучше назвать её в соответствии с разрабатываемой в ней функциональностью.

10. Как и зачем можно игнорировать некоторые файлы при `commit`?

Зачастую нам не нужно, чтобы Git отслеживал все файлы в репозитории, потому что в их число могут входить:

Список литературы

1. Лекция Системы контроля версий
2. GitHub для начинающих