

РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ

Факультет физико-математических и естественных наук

Отчёт по лабораторной работе №1

Дисциплина: Операционные системы

Студент: Оразгелдиева Огулнур

Группа: НПИбд-02-20

Лабораторная работа №1

Цель работы: изучить идеологию и применение средств контроля версий git.

Теоретические сведения

Система контроля версий Git представляет собой набор программ командной строки.

Доступ к ним можно получить из терминала посредством ввода команды git с различными опциями.

Благодаря тому, что Git является распределённой системой контроля версий, резервную копию локального хранилища можно сделать простым копированием или архивацией.

Подготовка рабочего пространства

Сначала нужно создать учётную запись на git (см. рис. 1)

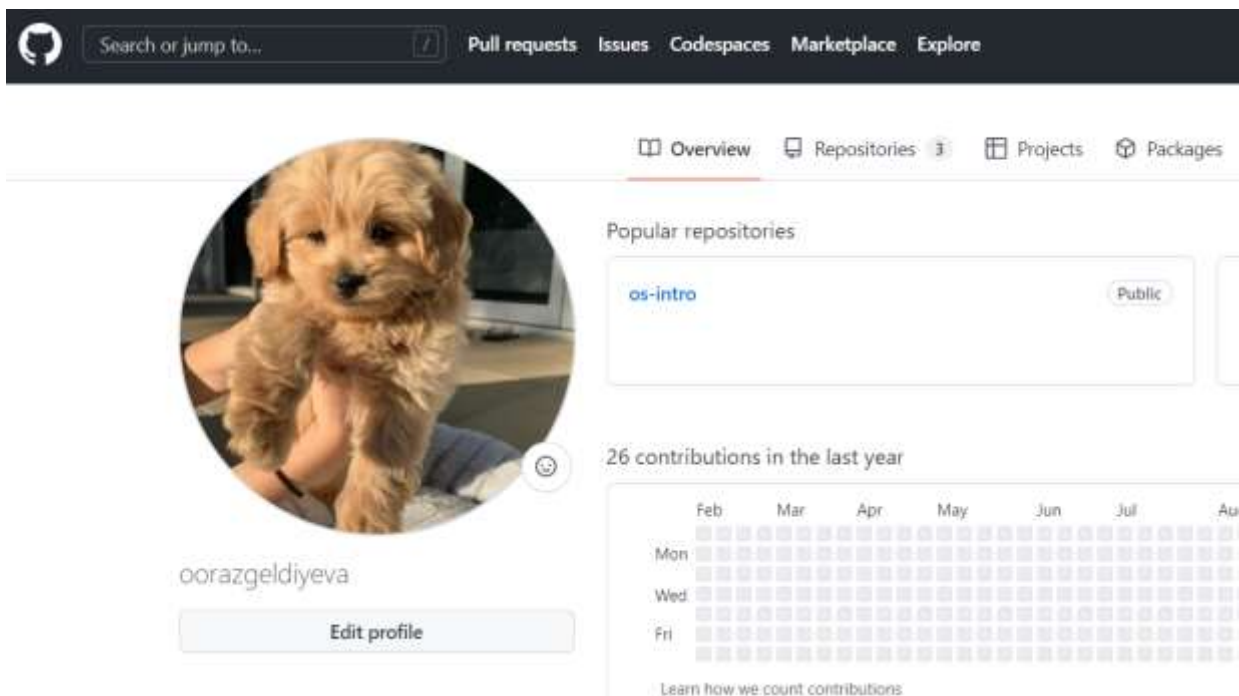


Рисунок 1. Аккаунт

Далее в рабочем каталоге создаём каталоги (см. рис. 2)

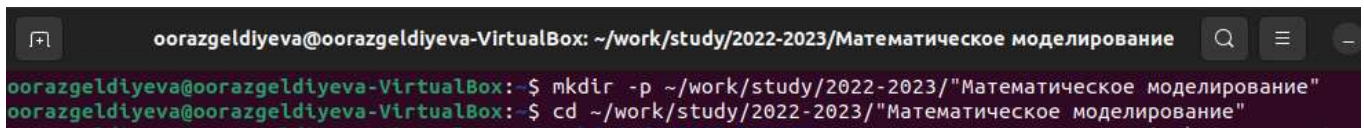


Рисунок 2. Создание каталогов

Перед началом работы нужно установить на нашей ОС git (см. рис. 3)

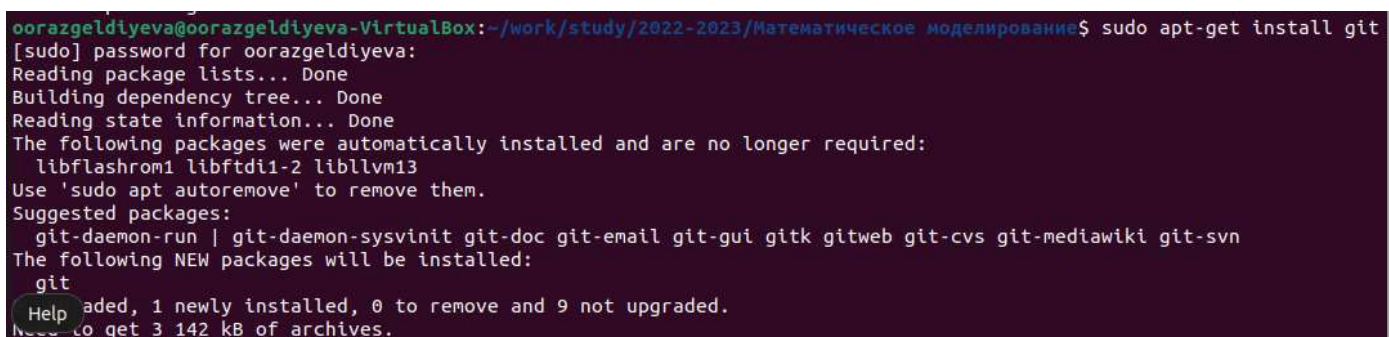


Рисунок 3. Установка git

Для последующей идентификации пользователя на сервере репозитория необходимо сгенерировать ssh-ключ (см. рис. 4)

```

To authenticate, please run 'git auth login'.
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/work/study/2022-2023/Математическое моделирование$ git config --global user
.name "oorazgeldiyeva"
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/work/study/2022-2023/Математическое моделирование$ git config --global user
.email "1032205431@pfur.ru"
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/work/study/2022-2023/Математическое моделирование$ ssh-keygen -C "oorazgeld
iyeva<1032205431@pfur.ru>"
Generating public/private rsa key pair.
Enter file in which to save the key (/home/oorazgeldiyeva/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/oorazgeldiyeva/.ssh/id_rsa
Your public key has been saved in /home/oorazgeldiyeva/.ssh/id_rsa.pub
The key fingerprint is:
XewrUMYqg72r0s95hPBb44j/5xQcLukapkIAKl0CxLA oorazgeldiyeva<1032205431@pfur.ru>
Trash
's randomart image is:
+---[RSA 3072]-----+
|B.
|++      = +
|E .      = + 0
|oo.      = 0
|.o .o S .
|.o=++ o .
|..O*=.. .
|ooo.o+. .
|..*++++.
+---[SHA256]-----+
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/work/study/2022-2023/Математическое моделирование$

```

Рисунок 4. Генерация ключа

Затем сгенерированный ключ можно вручную вставить настройках на git (см. рис. 5-6)

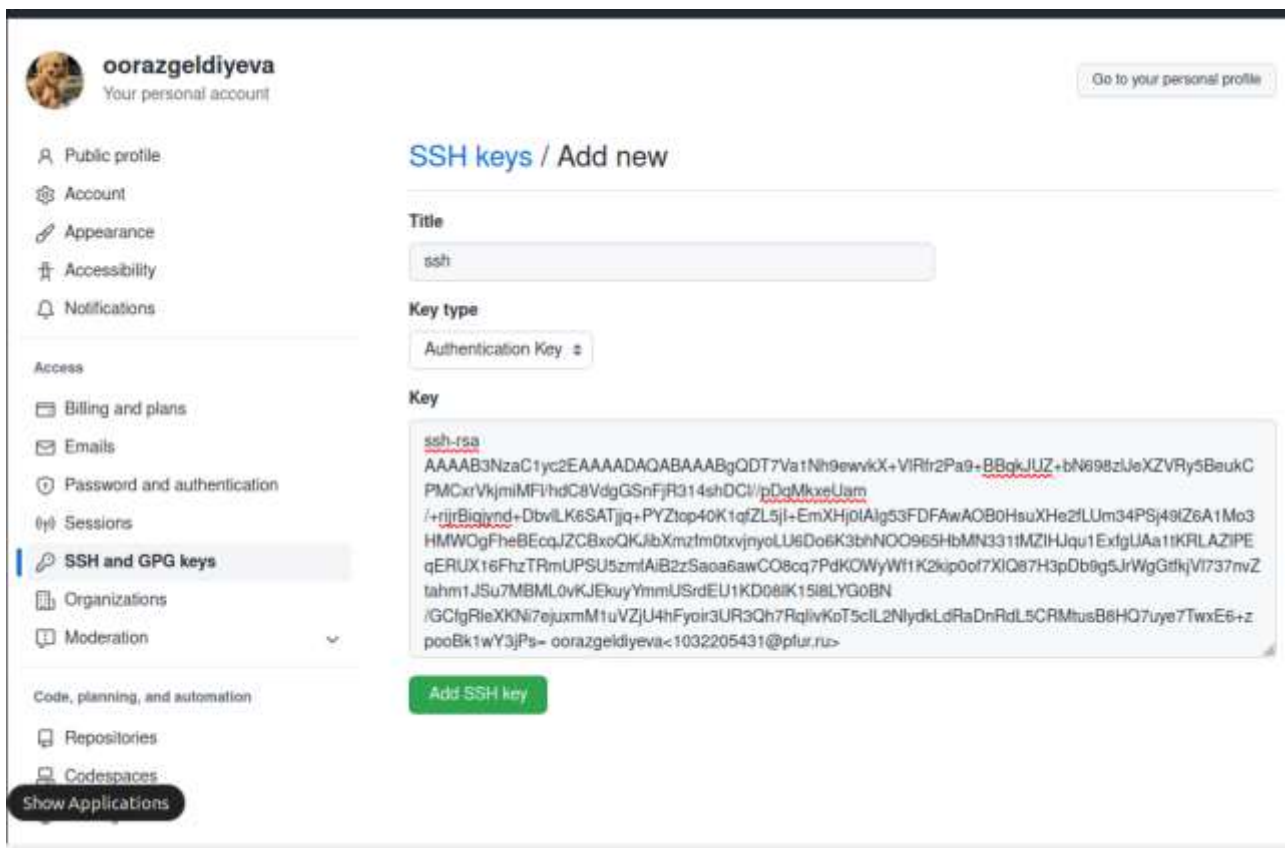



Рисунок 5. Сгенерированный ключ



oorazgeldiyeva
Your personal account

Public profile

Account

Appearance

Accessibility

Notifications

Access

Billing and plans


Emails

Password and authentication

SSH keys

This is a list of SSH keys associated with your account. Remove any keys that you no longer need.

Authentication Keys



ssh
SHA256:XewrUMyqg72r0s95hPBb44j/5xQc1ukapkiAKi0CxLA
Added on Feb 11, 2023
Never used — Read/write


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

Рисунок 6. Сгенерированный ключ

Теперь нужно создать репозиторий. Сделаем это вручную на основе шаблона (см. рис. 7-8)

Owner *

Repository name *


 oorazgeldiyeva


 /

study_2022-2023_mathmod ✓


Great repository names are short and memorable. Need inspiration? How about [ubiquitous-computing-machine](#)

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.

Create repository from template

Рисунок 7. Создание репозитория на основе шаблона

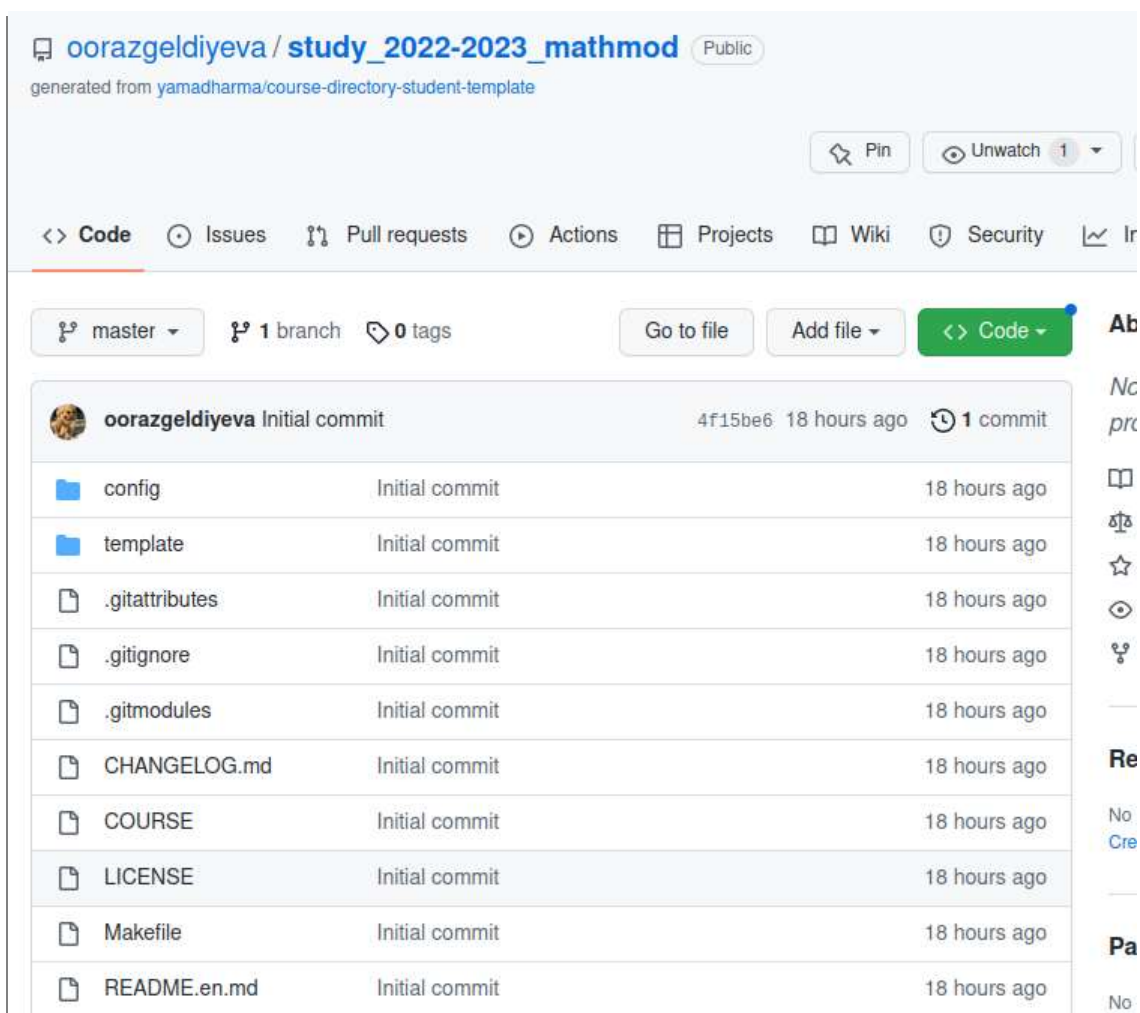


Рисунок 8. Репозиторий

Затем клонируем содержимое репозитория в каталог `matmod` (см. рис. 9)


```

bash: syntax error near unexpected token `/'
oorazgeldiyeva@oorazgeldiyeva-VirtualBox: ~/work/study/2022-2023/Математическое моделирование$ git clone --recursive gt
t@github.com:oorazgeldiyeva/study_2022-2023_mathmod.git matmod
Cloning into 'matmod'...
The authenticity of host 'github.com (140.82.121.4)' can't be established.
ED25519 key fingerprint is SHA256:+DiY3wvvV6TuJJhbpZIsF/zLDA0zPMSvHdkr4UvCOqU.
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: 27, done.
remote: Counting objects: 100% (27/27), done.
remote: Compressing objects: 100% (26/26), done.
remote: Total 27 (delta 1), reused 11 (delta 0), pack-reused 0
Receiving objects: 100% (27/27), 16.93 KiB | 8.47 MiB/s, done.
Resolving deltas: 100% (1/1), done.
Submodule 'template/presentation' (https://github.com/yamadharma/academic-presentation-markdown-template.git) register
ed for path 'template/presentation'
Submodule 'template/report' (https://github.com/yamadharma/academic-laboratory-report-template.git) registered for pat
h 'template/report'
Cloning into '/home/oorazgeldiyeva/work/study/2022-2023/Математическое моделирование/matmod/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
Receiving objects: 100% (82/82), 92.90 KiB | 590.00 KiB/s, done.
Resolving deltas: 100% (28/28), done.
Cloning into '/home/oorazgeldiyeva/work/study/2022-2023/Математическое моделирование/matmod/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
Receiving objects: 100% (101/101), 327.25 KiB | 1.86 MiB/s, done.
Resolving deltas: 100% (40/40), done.
Submodule path 'template/presentation': checked out 'b1be3808ee91f5809264cb755d316174540b753e'
Submodule path 'template/report': checked out '1d1b61dcac9c287a83917b82e3aef11a33b1e3b2'
oorazgeldiyeva@oorazgeldiyeva-VirtualBox: ~/work/study/2022-2023/Математическое моделирование$

```

Рисунок 9. Клонирование

В созданном каталоге смотрим доступные цели (см. рис. 10)

```

oorazgeldiyeva@oorazgeldiyeva-VirtualBox: ~/work/study/2022-2023/Математическое моделирование$ make help
make: *** No rule to make target 'help'. Stop.
oorazgeldiyeva@oorazgeldiyeva-VirtualBox: ~/work/study/2022-2023/Математическое моделирование$ make list
make: *** No rule to make target 'list'. Stop.
oorazgeldiyeva@oorazgeldiyeva-VirtualBox: ~/work/study/2022-2023/Математическое моделирование$ ls
matmod
oorazgeldiyeva@oorazgeldiyeva-VirtualBox: ~/work/study/2022-2023/Математическое моделирование$ cd matmod
oorazgeldiyeva@oorazgeldiyeva-VirtualBox: ~/work/study/2022-2023/Математическое моделирование/matmod$ ls
CHANGELOG.md  config  COURSE  LICENSE  Makefile  package.json  README.en.md  README.git-flow.md  README.md  template
oorazgeldiyeva@oorazgeldiyeva-VirtualBox: ~/work/study/2022-2023/Математическое моделирование/matmod$ make help
Usage:
  make <target>

Targets:
  list              List of courses
  prepare           Generate directories structure

oorazgeldiyeva@oorazgeldiyeva-VirtualBox: ~/work/study/2022-2023/Математическое моделирование/matmod$ make list
  arch-pc          Архитектура ЭВМ
  sciprog-intro     Введение в научное программирование
  infosec          Информационная безопасность
  mathsec          Математические основы защиты информации и информационной безопасности
  mathmod          Математическое моделирование
  sciprog          Научное программирование
  os-intro         Операционные системы
oorazgeldiyeva@oorazgeldiyeva-VirtualBox: ~/work/study/2022-2023/Математическое моделирование/matmod$

```

Рисунок 10. Доступные цели

Удалим лишние файлы (см. рис. 11)

```

os-intro Операционные системы
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/work/study/2022-2023/Математическое моделирование/matmod$ ls
CHANGELOG.md  config  COURSE  LICENSE  Makefile  package.json  README.en.md  README.git-flow.md  README.md  template
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/work/study/2022-2023/Математическое моделирование/matmod$ rm package.json
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/work/study/2022-2023/Математическое моделирование/matmod$ ls
CHANGELOG.md  config  COURSE  LICENSE  Makefile  README.en.md  README.git-flow.md  README.md  template
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/work/study/2022-2023/Математическое моделирование/matmod$

```

Рисунок 11. Удаление файла

Создадим необходимые каталоги (см. рис. 12)

```

[make: *** [package.json: prepare] Error 1
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/work/study/2022-2023/Математическое моделирование/matmod$
URSE
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/work/study/2022-2023/Математическое моделирование/matmod$
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/work/study/2022-2023/Математическое моделирование/matmod$ ls
CHANGELOG.md  COURSE  LICENSE  prepare  project-group  README.git-flow.md  README.md
config  labs  Makefile  presentation  README.en.md  README.md
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/work/study/2022-2023/Математическое моделирование/matmod$

```

Рисунок 12. Создание каталога

Отправи файлы на сервер (см. рис. 13-14)

```

oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/work/study/2022-2023/Математическое моделирование/matmod$
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/work/study/2022-2023/Математическое моделирование/matmod$
eat(main): make course structure'
[master e6fe400] feat(main): make course structure
217 files changed, 59287 insertions(+), 14 deletions(-)
create mode 100644 labs/README.md
create mode 100644 labs/README.ru.md
Ubuntu Software 100644 labs/lab1/presentation/Makefile
create mode 100644 labs/lab1/presentation/image/kulyabov.jpg
create mode 100644 labs/lab1/presentation/presentation.md
create mode 100644 labs/lab1/report/Makefile
create mode 100644 labs/lab1/report/bib/cite.bib
create mode 100644 labs/lab1/report/image/placeimg_800_600_tech.jpg
create mode 100644 labs/lab1/report/pandoc/csl/gost-r-7-0-5-2008-numeric.csl
create mode 100755 labs/lab1/report/pandoc/filters/pandoc_eqnos.py
create mode 100755 labs/lab1/report/pandoc/filters/pandoc_fignos.py
create mode 100755 labs/lab1/report/pandoc/filters/pandoc_secnos.py
create mode 100755 labs/lab1/report/pandoc/filters/pandoc_tablenos.py
create mode 100644 labs/lab1/report/pandoc/filters/pandocxnos/init.py

```

Рисунок 13. Отправка файлов на сервер


```

create node 1000 / project-group/stage-1/report/report.md
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/work/study/2022-2023/Математическое
Enumerating objects: 40, done.
Counting objects: 100% (40/40), done.
Compressing objects: 100% (30/30), done.
Writing objects: 100% (38/38), 342.34 KiB | 2.52 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 github.com:oorazgeldiyeva/study_2022-2023_mathmod.git
 4f15be6..e6fe400 master -> master
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/work/study/2022-2023/Математическое

```

Рисунок 14. Отправка файлов на сервер

Проверили отправку файла (см. рис. 15)

oorazgeldiyeva / study_2022-2023_mathmod Public

generated from yamadharma/course-directory-student-template

<> Code Issues Pull requests Actions Projects Wiki Security Insights Settings

master 1 branch 0 tags

Go to file Add file

File/Folder	Commit Message	Commit Hash	Time
config	Initial commit	e6fe400	14 m
labs	feat(main): make course structure		
presentation	feat(main): make course structure		
project-group	feat(main): make course structure		
template	Initial commit		14 m
.gitattributes	Initial commit		14 m
.gitignore	Initial commit		14 m
.gitmodules	Initial commit		14 m
CHANGELOG.md	Initial commit		14 m
COURSE	feat(main): make course structure		
LICENSE	Initial commit		14 m
Makefile	Initial commit		14 m
README.en.md	Initial commit		14 m

Рисунок 15. Отправка файла

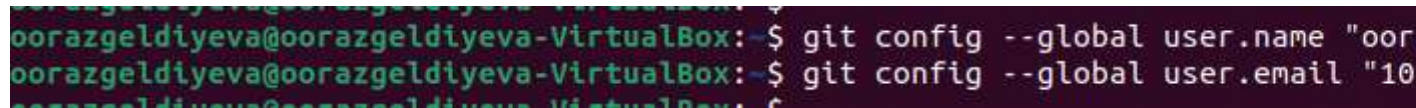
Ход работы:

1. Работа с git

1.1 Подготовка

Установка имени и электронной почты

Для начала нам необходимо осуществить установку. Выполним следующие команды, чтобы git узнал наше имя и электронную почту (см. рис. 16)



```
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~$ git config --global user.name "oorazgeldiyeva"
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~$ git config --global user.email "10oorazgeldiyeva@yandex.ru"
```

Рисунок 16. Имя и электронная почта

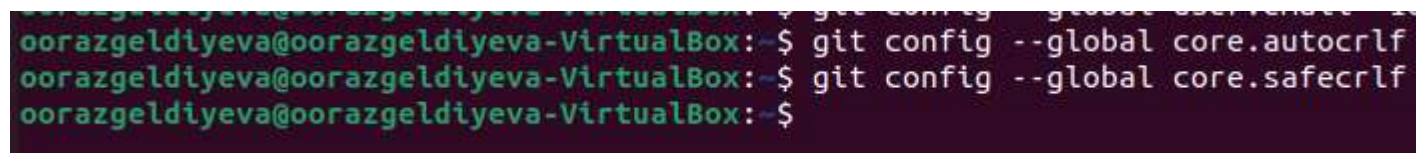
Параметры установки окончаний строк

Настройка `core.autocrlf` с параметрами `true` и `input` делает все переводы строк текстовых файлов в главном репозитории одинаковыми.

`core.autocrlf true` - git автоматически конвертирует CRLF->LF при коммите и обратно LF->CRLF при выгрузке кода из репозитория на файловую систему (используют в Windows). `core.autocrlf input` - конвертация CRLF в LF только при коммитах (используют в Mac/Linux).

Если `core.safecrlf` установлен в `true` или `warn`, git проверяет, если преобразование является обратимым для текущей настройки `core.autocrlf`.

`core.safecrlf true` - отвержение необратимого преобразования lf<->crlf. Полезно, когда специфические бинарники похожие на текстовые файлы. `core.safecrlf warn` - печать только предупреждение, но принимает необратимый переход.



```
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~$ git config --global core.autocrlf true
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~$ git config --global core.safecrlf warn
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~$
```

Рисунок 17. Установка окончания строк

Установка отображения unicode

По умолчанию, git будет печатать не-ASCII символов в именах файлов в виде

восьмеричных последовательностей \nnn. Что бы избежать нечитаемых строк,
установите соответствующий флаг (см. рис. 18)

```
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~$ git config --global core.autocrlf
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~$ git config --global core.safecrlf
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~$ git config --global core.quotepath
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~$
```

Рисунок 18. Установка отображения unicode

1.2 Создание проекта

Создайте страницу «Hello, World»

Начните работу в пустом рабочем каталоге с создания пустого каталога с именем

hello, затем войдите в него и создайте там файл с именем hello.html.

```
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~$ mkdir hello
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~$ cd hello
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ touch hello.html
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ echo "Hello, World!" > hello.html
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ cat hello.html
Hello, World!
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$
```

Рисунок 19. Создание файла

Создание репозитория

Чтобы создать git репозиторий из этого каталога, выполните команду git init.

```
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git init
hint: Using 'master' as the name for the initial branch. This default branch name
hint: is subject to change. To configure the initial branch name to use in all
hint: of your new repositories, which will suppress this warning, call:
hint:
hint:   git config --global init.defaultBranch <name>
hint:
hint: Names commonly chosen instead of 'master' are 'main', 'trunk' and
hint: 'development'. The just-created branch can be renamed via this command:
hint:
hint:   git branch -m <name>
Initialized empty Git repository in /home/oorazgeldiyeva/hello/.git/
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$
```

Рисунок 20. Создание репозитория

Добавление файла в репозиторий

Добавим файл в репозиторий (см. рис. 21)

```
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git add hello.html
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git commit -m "Initial Commi
[master (root-commit) f0b2344] Initial Commit
 1 file changed, 1 insertion(+)
 create mode 100644 hello.html
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$
```

Рисунок 21. Добавление файла

Проверка состояние репозитория

Используйте команду `git status`, чтобы проверить текущее состояние репозитория (см. рис. 22)

```
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git status
On branch master
nothing to commit, working tree clean
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$
```

Рисунок 22. Проверка состояния

Команда проверки состояния сообщит, что коммитить нечего. Это означает, что в репозитории хранится текущее состояние рабочего каталога, и нет никаких изменений, ожидающих записи.

1.3 Внесение изменений

Измените страницу «Hello, World»

Добавим кое-какие HTML-теги к нашему приветствию. Измените содержимое файла `hello.html` (см. рис. 23)

```
<h1>Hello, World!</h1>
```

Рисунок 23. Изменение файла

Проверьте состояние рабочего каталога (см. рис. 24)


```

oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ vim hello.html
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git status
On branch master
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   hello.html

no changes added to commit (use "git add" and/or "git commit -a")
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$

```

Рисунок 24. Проверка состояния

git знает, что файл hello.html был изменен, но при этом эти изменения еще не зафиксированы в репозитории.

Также обратите внимание на то, что сообщение о состоянии дает вам подсказку о том, что нужно делать дальше. Если вы хотите добавить эти изменения в репозиторий, используйте команду git add. В противном случае используйте команду git checkout для отмены изменений.

Индексация изменений

Теперь выполните команду git, чтобы проиндексировать изменения. Проверьте состояние (см. рис. 25)

```

oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git add hello.html
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   hello.html

oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$

```

Рисунок 25. Индексация изменений

Изменения файла hello.html были проиндексированы. Это означает, что git теперь знает об изменении, но изменение пока не записано в репозиторий. Следующий коммит будет включать в себя проиндексированные изменения.

Если вы решили, что не хотите коммитить изменения, команда состояния напомнит вам о том, что с помощью команды git reset можно снять индексацию этих изменений.

Отдельный шаг индексации в `git` позволяет вам продолжать вносить изменения

в рабочий каталог, а затем, в момент, когда вы захотите взаимодействовать с версионным контролем, `git` позволит записать изменения в малых коммитах, которые фиксируют то, что вы сделали.

Разделяя индексацию и коммит, вы имеете возможность с легкостью настроить, что идет в какой коммит.

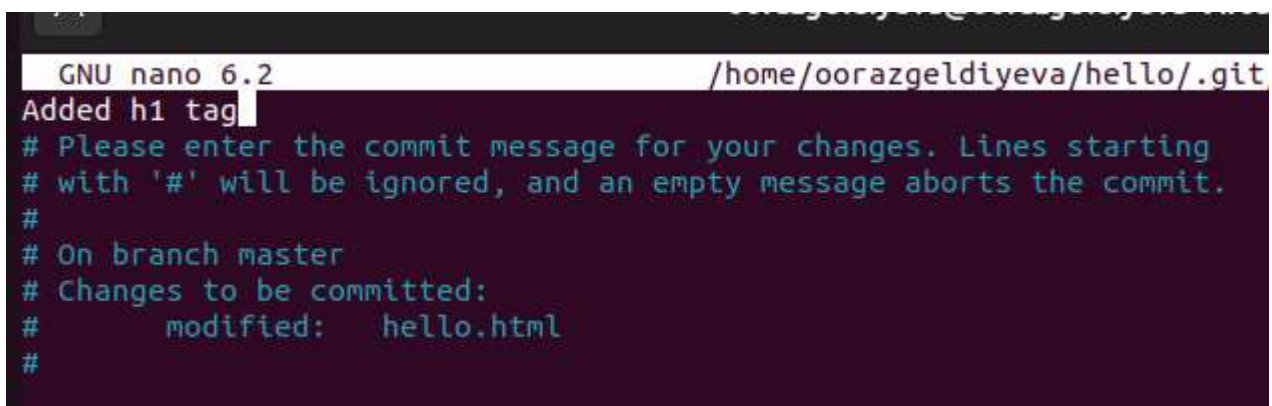
1. Коммит изменений

Когда вы ранее использовали `git commit` для коммита первоначальной версии файла `hello.html` в репозиторий, вы включили метку `-m`, которая делает комментарий в командной строке. Команда `commit` позволит вам интерактивно редактировать комментарии для коммита. Теперь давайте это проверим.

Если вы опустите метку `-m` из командной строки, `git` перенесет вас в редактор по вашему выбору. Редактор выбирается из следующего списка (в порядке приоритета):

- переменная среды `GIT_EDITOR`
- параметр конфигурации `core.editor`
- переменная среды `VISUAL`
- переменная среды `EDITOR`

Сделайте коммит и проверьте состояние (см. рис. 26-28)

A screenshot of a terminal window with a dark background. The title bar at the top shows "GNU nano 6.2" on the left and "/home/oorazgeldiyeva/hello/.git" on the right. The main content of the terminal is the output of the `git commit` command. It starts with "Added h1 tag" followed by a prompt: "# Please enter the commit message for your changes. Lines starting with '#' will be ignored, and an empty message aborts the commit." Below this, there are several lines of text: "#", "# On branch master", "# Changes to be committed:", and "# modified: hello.html". The cursor is positioned at the end of the last line.

```
GNU nano 6.2 /home/oorazgeldiyeva/hello/.git
Added h1 tag
# Please enter the commit message for your changes. Lines starting
# with '#' will be ignored, and an empty message aborts the commit.
#
# On branch master
# Changes to be committed:
#       modified:   hello.html
#
```

Рисунок 26. Коммит

```
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git commit
[master 026a401] Added h1 tag
 1 file changed, 3 insertions(+), 1 deletion(-)
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$
```

Рисунок 27. Коммит

```
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git status
On branch master
nothing to commit, working tree clean
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$
```

Рисунок 28. Проверка статуса

Откроется редактор.

В первой строке введите комментарий: «Added h1 tag». Сохраните файл и выйдите из редактора (для этого в редакторе по умолчанию (Vim) вам нужно нажать клавишу ESC, ввести :wq и нажать Enter).

Теперь еще раз проверим состояние.

Рабочий каталог чистый, можно продолжить работу.

Добавьте стандартные теги страницы

Измените страницу «Hello, World», чтобы она содержала стандартные теги (см. рис. 29)



```
oorazgeldiyeva@oorazgeldiyeva-VirtualBox: ~/hello
<html>
  <body>
    <h1>Hello, World!</h1>
  </body>
</html>
```

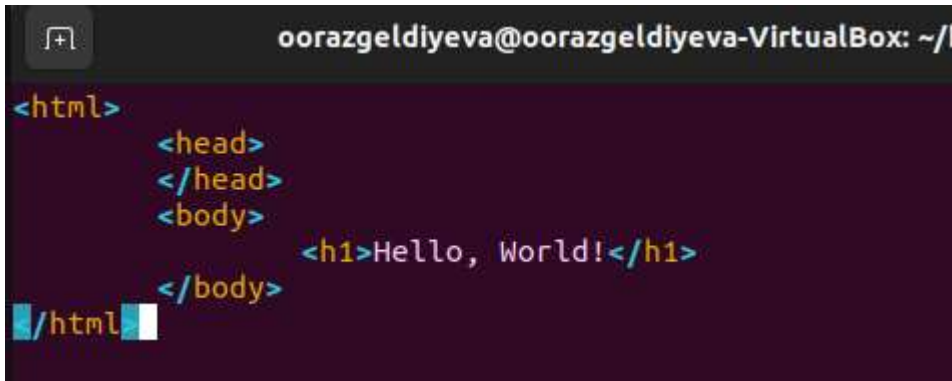
Рисунок 29. Изменение файла

Теперь добавьте это изменение в индекс git (см. рис. 30)

```
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ vim hello.html
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git add hello.html
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$
```


Рисунок 30. Добавление изменения

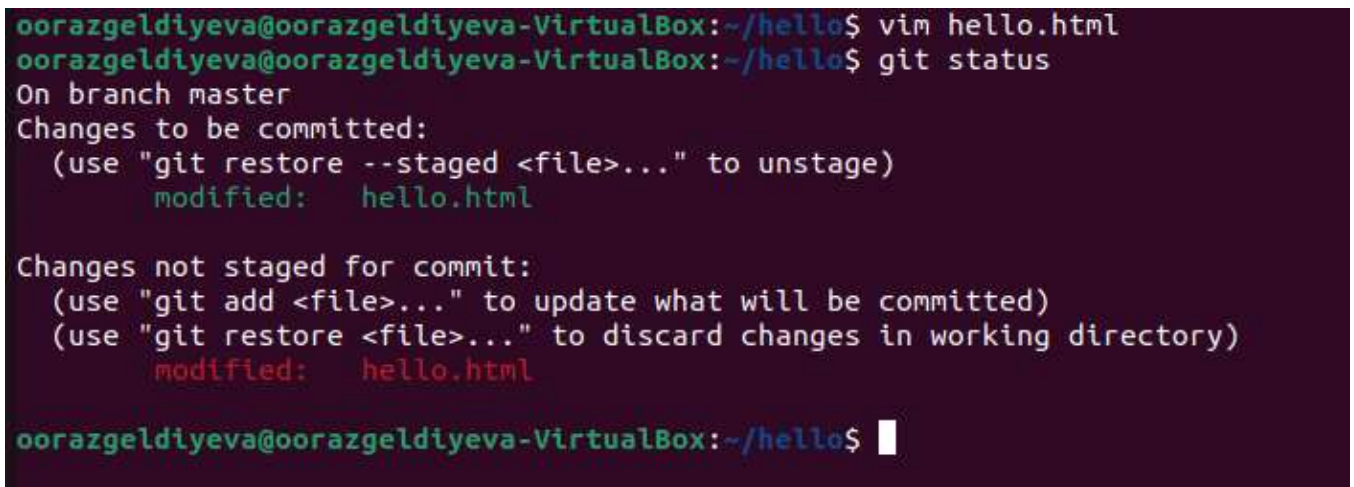
Теперь добавьте заголовки HTML к странице «Hello,World» (см. рис. 31)



```
oorazgeldiyeva@oorazgeldiyeva-VirtualBox: ~/t
<html>
  <head>
  </head>
  <body>
    <h1>Hello, World!</h1>
  </body>
</html>
```

Рисунок 31. Изменение файла

Проверьте текущий статус: git status (см. рис. 32)



```
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ vim hello.html
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    modified:   hello.html

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
    modified:   hello.html

oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$
```

Рисунок 32. Проверка статуса

Обратите внимание на то, что hello.html указан дважды в состоянии. Первое изменение (добавление стандартных тегов) проиндексировано и готово к коммиту. Второе изменение (добавление заголовков HTML) является непроиндексированным. Если бы вы делали коммит сейчас, заголовки не были бы сохранены в репозиторий.

Произведите коммит проиндексированного изменения (значение по умолчанию), а затем еще раз проверьте состояние (см. рис. 33)

```

oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   hello.html

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   hello.html

oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git commit -m "Added standart
HTML page tags"
[master 66ced28] Added standart HTML page tags
 1 file changed, 5 insertions(+), 1 deletion(-)
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git status
On branch master
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   hello.html

no changes added to commit (use "git add" and/or "git commit -a")
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$

```

Рисунок 33. Коммит

Состояние команды говорит о том, что hello.html имеет незафиксированные изменения, но уже не в буферной зоне.

Теперь добавьте второе изменение в индекс, а затем проверьте состояние с помощью команды git status (см. рис. 34)

```

oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git add .
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ dit status
Command 'dit' not found, but there are 22 similar ones.
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   hello.html

oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$

```

Рисунок 34. Добавление изменений и проверка состояния

В качестве файла для добавления, мы использовали текущий каталог (.). Это

краткий и удобный путь для добавления всех изменений в файлы текущего каталога и его подкаталоги. Но поскольку он добавляет все, не лишним будет проверить состояние перед запуском add, просто чтобы

убедиться, что вы не добавили какой-то файл, который добавлять было не нужно.

Второе изменение было проиндексировано и готово к коммиту.

Сделайте коммит второго изменения (см. рис. 35)

```
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git commit -m "Added HTML header"
[master 4acd636] Added HTML header
1 file changed, 2 insertions(+)
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$
```

Рисунок 35. Коммит

История

Получим список произведенных изменений (см. рис. 36)

```
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git log
commit 4acd63657c36c6476d68dde507805c87564baccb (HEAD -> master)
Author: oorazgeldiyeva <1032205431@pfur.ru>
Date: Sat Feb 11 21:17:42 2023 +0300

    Added HTML header

commit 66ced280e42f295a8b1459db826163f1676a7bff
Author: oorazgeldiyeva <1032205431@pfur.ru>
Date: Sat Feb 11 21:15:54 2023 +0300

    Added standart HTML page tags

commit 026a40162b4ebc4d587fc58b9811c48016e61a6d
Author: oorazgeldiyeva <1032205431@pfur.ru>
Date: Sat Feb 11 21:08:35 2023 +0300

    Added h1 tag

commit f0b2344566edc3a063ced594714f243bda81c161
Author: oorazgeldiyeva <1032205431@pfur.ru>
Date: Sat Feb 11 20:53:43 2023 +0300

    Initial Commit
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$
```

Рисунок 36. История

Однострочный формат истории (см. рис. 37)


```

oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git log --pretty=oneline
4acd63657c36c6476d68dde507805c87564baccb (HEAD -> master) Added HTML header
66ced280e42f295a8b1459db826163f1676a7bfff Added standart HTML page tags
026a40162b4ebc4d587fc58b9811c48016e61a6d Added h1 tag
f0b2344566edc3a063ced594714f243bda81c161 Initial Commit
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$

```

Рисунок 37. История

Есть много вариантов отображения лога (см. рис. 38)

```

oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git log --pretty=oneline
4acd63657c36c6476d68dde507805c87564baccb (HEAD -> master) Added HTML header
66ced280e42f295a8b1459db826163f1676a7bfff Added standart HTML page tags
026a40162b4ebc4d587fc58b9811c48016e61a6d Added h1 tag
f0b2344566edc3a063ced594714f243bda81c161 Initial Commit
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git log --pretty=oneline --max-count=2
4acd63657c36c6476d68dde507805c87564baccb (HEAD -> master) Added HTML header
66ced280e42f295a8b1459db826163f1676a7bfff Added standart HTML page tags
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git log --pretty=oneline --since='5 minutes ago'
4acd63657c36c6476d68dde507805c87564baccb (HEAD -> master) Added HTML header
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git log --pretty=oneline --until='5 minutes ago'
66ced280e42f295a8b1459db826163f1676a7bfff Added standart HTML page tags
026a40162b4ebc4d587fc58b9811c48016e61a6d Added h1 tag
f0b2344566edc3a063ced594714f243bda81c161 Initial Commit
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git log --pretty=oneline --author=oorazgeldiyeva
4acd63657c36c6476d68dde507805c87564baccb (HEAD -> master) Added HTML header
66ced280e42f295a8b1459db826163f1676a7bfff Added standart HTML page tags
026a40162b4ebc4d587fc58b9811c48016e61a6d Added h1 tag
f0b2344566edc3a063ced594714f243bda81c161 Initial Commit
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git log --pretty=oneline --all
4acd63657c36c6476d68dde507805c87564baccb (HEAD -> master) Added HTML header
66ced280e42f295a8b1459db826163f1676a7bfff Added standart HTML page tags
026a40162b4ebc4d587fc58b9811c48016e61a6d Added h1 tag
f0b2344566edc3a063ced594714f243bda81c161 Initial Commit
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$

```

Рисунок 38. История

Справочная страница (см. рис. 39)

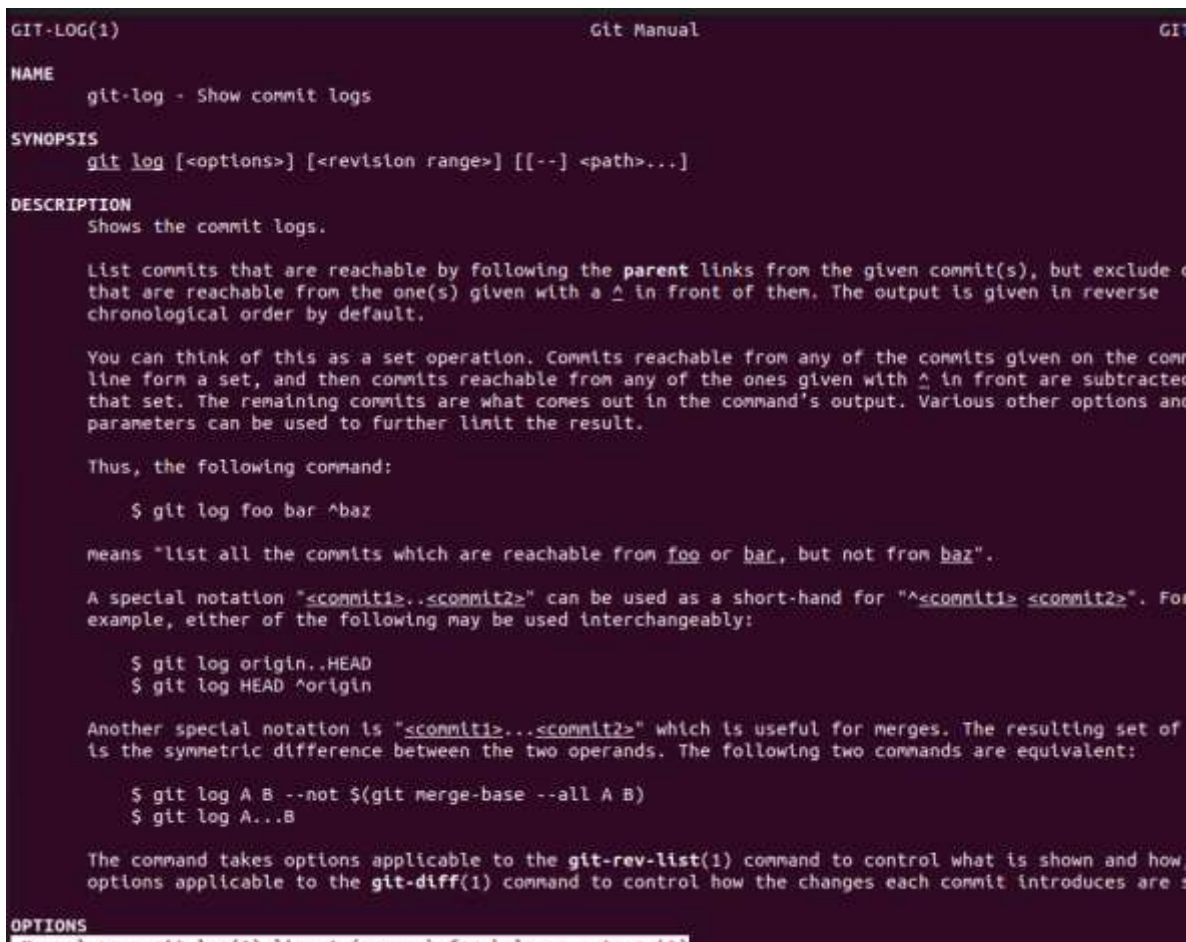


Рисунок 39. Справочная информация

Инструмент `gitk` полезен в изучении истории изменений.

Получение старых версий

Возвращаться назад в историю очень просто. Команда `checkout` скопирует любой снимок из репозитория в рабочий каталог.

Получите хэши предыдущих версий (см. рис. 38-39)

Изучите данные лога и найдите хэш для первого коммита. Он должен быть в последней строке данных. Используйте этот хэш-код (достаточно первых 7 знаков) в команде ниже. Затем проверьте содержимое файла `hello.html` (см. рис. 40)

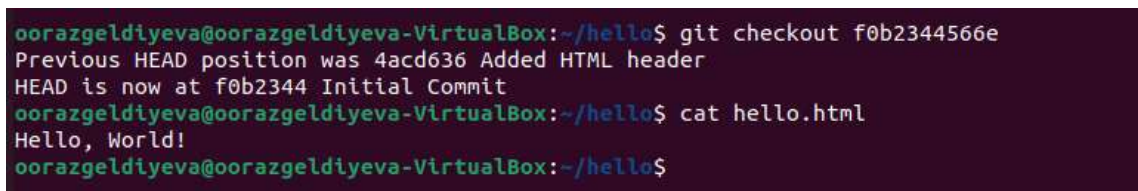


Рисунок 40. Содержимое файла

Вернитесь к последней версии в ветке master (см. рис. 41)

```
Hello, world!
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git checkout master
Previous HEAD position was f0b2344 Initial Commit
Switched to branch 'master'
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ cat hello.html
<html>
  <head>
  </head>
  <body>
    <h1>Hello, World!</h1>
  </body>
</html>

oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$
```

Рисунок 41. Переключение ветки

master — имя ветки по умолчанию. Переключая имена веток, вы попадаете на последнюю версию выбранной ветки.

Создание тегов версий

Давайте назовем текущую версию страницы hello первой (v1).

Создайте тег первой версии

Теперь текущая версия страницы называется v1.

Теги для предыдущих версий Давайте создадим тег для версии, которая идет перед текущей версией и назовем его v1-beta. В первую очередь нам надо переключиться на предыдущую версию. Вместо поиска до хэша, мы будем использовать

^, обозначающее «родитель v1». Вместо обозначения v1^ можно использовать v1~1. Это обозначение можно определить как «первую версию предшествующую

(см. рис. 42)


```
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git tag v1
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git checkout v1^
Note: switching to 'v1^'.

You are in 'detached HEAD' state. You can look around, make experimental
changes and commit them, and you can discard any commits you make in this
state without impacting any branches by switching back to a branch.

If you want to create a new branch to retain commits you create, you may
do so (now or later) by using -c with the switch command. Example:

    git switch -c <new-branch-name>

Or undo this operation with:

    git switch -

Turn off this advice by setting config variable advice.detachedHead to false

HEAD is now at 66ced28 Added standart HTML page tags
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ cat hello.html
<html>
    <body>
        <h1>Hello, World!</h1>
    </body>
</html>

oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$
```

Рисунок 42. Создание тега

Это версия стегами <html> и <body>,но еще пока без <head>.Давайте сделаем ее версией v1-beta (см. рис. 43):

```
git tag v1-beta
```

Переключение по имени тега

Теперь попробуйте попереключаться между двумя отмеченными версиями (см. рис. 44)

```

oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git tag v1-beta
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git checkout v1
Previous HEAD position was 66ced28 Added standart HTML page tags
HEAD is now at 4acd636 Added HTML header
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ cat hello.html
<html>
    <head>
    </head>
    <body>
        <h1>Hello, World!</h1>
    </body>
</html>

oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git checkout v1-beta
Previous HEAD position was 4acd636 Added HTML header
HEAD is now at 66ced28 Added standart HTML page tags
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ cat hello.html
<html>
    <body>
        <h1>Hello, World!</h1>
    </body>
</html>

oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$

```

Рисунок 44. Переключение между тегами

Просмотр тегов с помощью команды tag

Вы можете увидеть, какие теги доступны, используя команду git tag (см. рис. 45)

```

oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git tag
v1
v1-beta
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$

```

Рисунок 45. Доступные теги

Вы также можете посмотреть теги в логе (см. рис. 46)

```

v1-beta
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git log master --all
commit 4acd63657c36c6476d68dde507805c87564baccb (tag: v1, master)
Author: oorazgeldiyeva <1032205431@pfur.ru>
Date: Sat Feb 11 21:17:42 2023 +0300

    Added HTML header

commit 66ced280e42f295a8b1459db826163f1676a7bff (HEAD, tag: v1-beta)
Author: oorazgeldiyeva <1032205431@pfur.ru>
Date: Sat Feb 11 21:15:54 2023 +0300

    Added standart HTML page tags

commit 026a40162b4ebc4d587fc58b9811c48016e61a6d
Author: oorazgeldiyeva <1032205431@pfur.ru>
Date: Sat Feb 11 21:08:35 2023 +0300

    Added h1 tag

commit f0b2344566edc3a063ced594714f243bda81c161
Author: oorazgeldiyeva <1032205431@pfur.ru>
Date: Sat Feb 11 20:53:43 2023 +0300

    Initial Commit
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$

```

Рисунок 46. История

Вы можете видеть теги (v1 и v1-beta) в логе вместе с именем ветки (master). Кроме того HEAD показывает коммит, на который вы переключились (на данный момент это v1-beta).

Отмена локальных изменений (до индексации)

Переключитесь на ветку master

Убедитесь, что вы находитесь на последнем коммите ветки master, прежде чем продолжить работу (см. рис. 47)

```

oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git checkout master
Previous HEAD position was 66ced28 Added standart HTML page tags
Switched to branch 'master'
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$

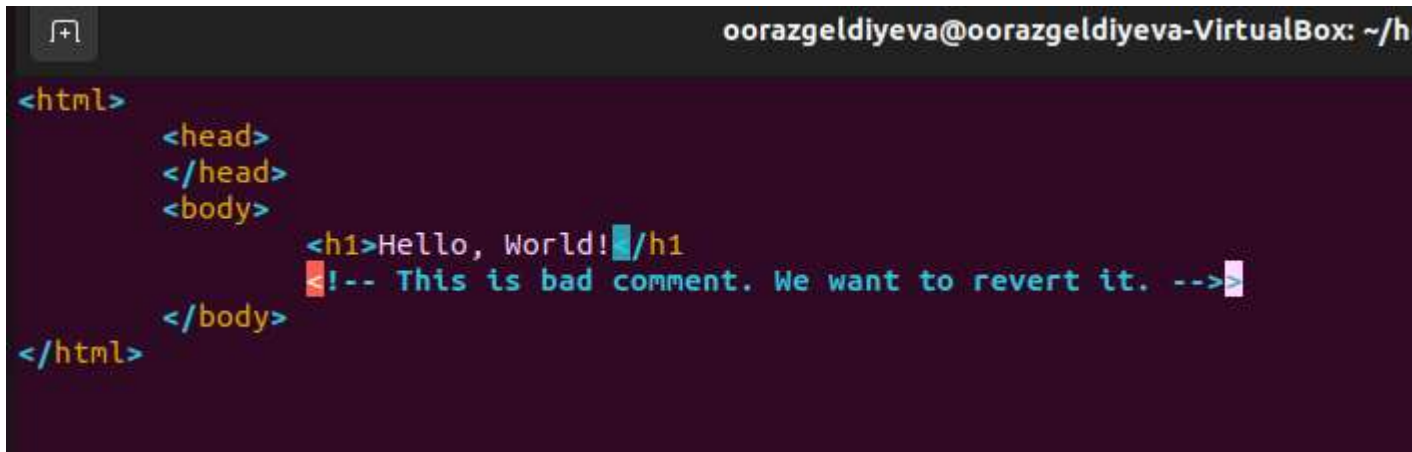
```

Рисунок 47. Переключение ветки

Измените hello.html

Иногда случается, что вы изменили файл в рабочем каталоге, и хотите отменить последние коммиты. С этим справится команда `git checkout`.

Внесите изменение в файл `hello.html` в виде нежелательного комментария (см. рис. 48)

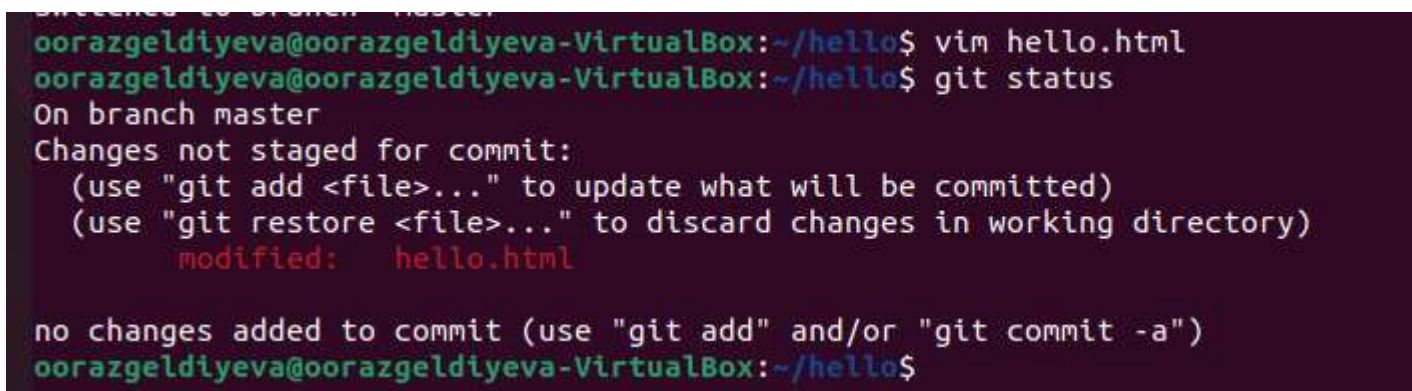


```
oorazgeldiyeva@oorazgeldiyeva-VirtualBox: ~/h
<html>
  <head>
  </head>
  <body>
    <h1>Hello, World!</h1>
    <!-- This is bad comment. We want to revert it. -->
  </body>
</html>
```

Рисунок 48. Изменение файла

Проверьте состояние

Сначала проверьте состояние рабочего каталога (см. рис. 49)



```
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ vim hello.html
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git status
On branch master
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   hello.html

no changes added to commit (use "git add" and/or "git commit -a")
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$
```

Рисунок 49. Проверка состояния

Мы видим, что файл `hello.html` был изменен, но еще не проиндексирован.

Отмена изменений в рабочем каталоге

Используйте команду `git checkout` для переключения версии файла `hello.html` в репозитории (см. рис. 50)

```

no changes added to commit (use "git add" and/or "git commit -a")
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git checkout hello.html
Updated 1 path from the index
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git status
On branch master
nothing to commit, working tree clean
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ cat hello.html
<html>
  <head>
  </head>
  <body>
    <h1>Hello, World!</h1>
  </body>
</html>

oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$

```

Рисунок 50. Переключение версии

Команда `git status` показывает нам, что не было произведено никаких изменений, не зафиксированных в рабочем каталоге.

Отмена проиндексированных изменений (перед коммитом)

Измените файл и проиндексируйте изменения

Внесите изменение в файл `hello.html` в виде нежелательного комментария (см. рис. 51)

```

oorazgeldiyeva@oorazgeldiyeva-VirtualBox: ~/he
<html>
  <head>
    <!-- This is an unwanted but staged comment -->
  </head>
  <body>
    <h1>Hello, World!</h1>
  </body>
</html>

```

Рисунок 51. Изменение файла

Проиндексируйте это изменение (см. рис. 52)

```

oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ vim hello.html
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git add hello.html
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   hello.html

oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ █

```

Рисунок 52. Индексация

Проверьте состояние

Проверьте состояние нежелательного изменения (см. рис. 52)

Состояние показывает, что изменение было проиндексировано и готово к ком- миту.

Выполните сброс буферной зоны

К счастью, вывод состояния показывает нам именно то, что мы должны сделать для отмены индексации изменения (см. рис. 53)

```

oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git reset HEAD hello.html
Unstaged changes after reset:
M      hello.html
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ █

```

Рисунок 53. Отмена изменений

Команда `git reset` сбрасывает буферную зону к HEAD. Это очищает буферную зону от изменений, которые мы только что проиндексировали.

Команда `git reset` (по умолчанию) не изменяет рабочий каталог. Поэтому рабочий каталог все еще содержит нежелательный комментарий. Мы можем использовать команду `git checkout`, чтобы удалить нежелательные изменения в рабочем каталоге.

Переключитесь на версию коммита

Наш рабочий каталог опять чист (см. рис. 54)

```

oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git checkout hello.html
Updated 1 path from the index
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git status
On branch master
nothing to commit, working tree clean
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ █

```


Рисунок 54. Проверка статуса

Отмена коммитов

Отмена коммитов

Иногда вы понимаете, что новые коммиты являются неверными, и хотите их отменить. Есть несколько способов решения этого вопроса, здесь мы будем использовать самый безопасный.

Мы отменим коммит путем создания нового коммита, отменяющего нежелательные изменения.

Измените файл и сделайте коммит

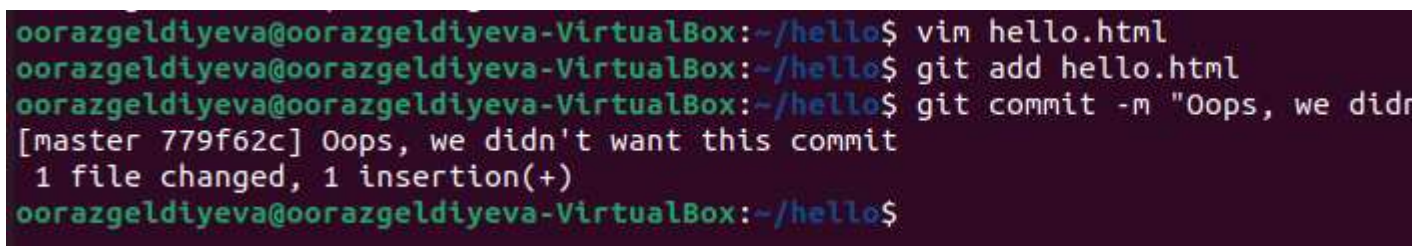
Измените файл hello.html на следующий (см. рис. 55)



```
oorazgeldiyeva@oorazgeldiyeva-VirtualBox: ~/hello
<html>
  <head>
  </head>
  <body>
    <h1>Hello, World!</h1>
    !-- This is unwanted but committed change --
  </body>
</html>
```

Рисунок 55. Изменение файла

Выполните (см. рис. 56)



```
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ vim hello.html
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git add hello.html
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git commit -m "Oops, we didn't want this commit"
[master 779f62c] Oops, we didn't want this commit
1 file changed, 1 insertion(+)
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$
```

Рисунок 56. Коммит

Сделайте коммит с новыми изменениями, отменяющими предыдущие

Чтобы отменить коммит, нам необходимо сделать коммит, который удаляет изменения, сохраненные нежелательным коммитом

Перейдите в редактор, где вы можете отредактировать коммит-сообщение по умолчанию или оставить все как есть. Сохраните и закройте файл.

Так как мы отменили самый последний произведенный коммит, мы смогли использовать HEAD в качестве аргумента для отмены. Мы можем отменить любой произвольный коммит в истории, указав его хэш-значение.

Проверьте лог

Проверка лога показывает нежелательные и отмененные коммиты в нашей репозитории (см. рис. 57)

```
1 file changed, 1 deletion(-)
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git log
commit c37fa988bb4f31f7013a742a79b0b9d846f4361d (HEAD -> master)
Author: oorazgeldiyeva <1032205431@pfur.ru>
Date: Sat Feb 11 21:45:07 2023 +0300

    Revert "Oops, we didn't want this commit"

    This reverts commit 779f62ceb6791b3fb314eab4a1e7625d12678366.

commit 779f62ceb6791b3fb314eab4a1e7625d12678366
Author: oorazgeldiyeva <1032205431@pfur.ru>
Date: Sat Feb 11 21:44:45 2023 +0300

    Oops, we didn't want this commit

commit 4acd63657c36c6476d68dde507805c87564baccb (tag: v1)
Author: oorazgeldiyeva <1032205431@pfur.ru>
Date: Sat Feb 11 21:17:42 2023 +0300

    Added HTML header

commit 66ced280e42f295a8b1459db826163f1676a7bff (tag: v1-beta)
Author: oorazgeldiyeva <1032205431@pfur.ru>
Date: Sat Feb 11 21:15:54 2023 +0300

    Added standart HTML page tags

commit 026a40162b4ebc4d587fc58b9811c48016e61a6d
Author: oorazgeldiyeva <1032205431@pfur.ru>
Date: Sat Feb 11 21:08:35 2023 +0300

    Added h1 tag

commit f0b2344566edc3a063ced594714f243bda81c161
Author: oorazgeldiyeva <1032205431@pfur.ru>
Date: Sat Feb 11 20:53:43 2023 +0300

    Initial Commit
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$
```

Рисунок 57. История

Эта техника будет работать с любым коммитом.

Удаление коммитов из ветки

git revert является мощной командой, которая позволяет отменить любые коммиты в репозитории. Однако, и оригинальный и «отмененный» коммиты видны в истории ветки (при использовании команды git log).

Часто мы делаем коммит, и сразу понимаем, что это была ошибка. Было бы неплохо иметь команду «возврата», которая позволила бы нам

сделать вид, что неправильного коммита никогда и не было. Команда «возврата» даже предотвратила бы появление нежелательного коммита в истории `git log`.

Команда `git reset`

При получении ссылки на коммит (т.е. хэш, ветка или имя тега), команда `git reset`:

- перепишет текущую ветку, чтобы она указывала на нужный коммит;
- опционально сбросит буферную зону для соответствия с указанным коммитом;
- опционально сбросит рабочий каталог для соответствия с указанным коммитом.

Проверьте нашу историю

Давайте сделаем быструю проверку нашей истории коммитов (см. рис. 58)


```

1 file changed, 1 deletion(-)
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git log
commit c37fa988bb4f31f7013a742a79b0b9d846f4361d (HEAD -> master)
Author: oorazgeldiyeva <1032205431@pfur.ru>
Date: Sat Feb 11 21:45:07 2023 +0300

    Revert "Oops, we didn't want this commit"

    This reverts commit 779f62ceb6791b3fb314eab4a1e7625d12678366.

commit 779f62ceb6791b3fb314eab4a1e7625d12678366
Author: oorazgeldiyeva <1032205431@pfur.ru>
Date: Sat Feb 11 21:44:45 2023 +0300

    Oops, we didn't want this commit

commit 4acd63657c36c6476d68dde507805c87564baccb (tag: v1)
Author: oorazgeldiyeva <1032205431@pfur.ru>
Date: Sat Feb 11 21:17:42 2023 +0300

    Added HTML header

commit 66ced280e42f295a8b1459db826163f1676a7bff (tag: v1-beta)
Author: oorazgeldiyeva <1032205431@pfur.ru>
Date: Sat Feb 11 21:15:54 2023 +0300

    Added standart HTML page tags

commit 026a40162b4ebc4d587fc58b9811c48016e61a6d
Author: oorazgeldiyeva <1032205431@pfur.ru>
Date: Sat Feb 11 21:08:35 2023 +0300

    Added h1 tag

commit f0b2344566edc3a063ced594714f243bda81c161
Author: oorazgeldiyeva <1032205431@pfur.ru>
Date: Sat Feb 11 20:53:43 2023 +0300

    Initial Commit
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ █

```

Рисунок 58. История

Мы видим, что два последних коммита в этой ветке — «Oops» и «Revert Oops». Давайте удалим их с помощью сброса.

Для начала отметьте эту ветку

Но прежде чем удалить коммиты, давайте отметим последний коммит тегом, чтобы потом можно было его найти (см. рис. 59)

Сброс коммитов к предшествующим коммитуOops

Глядя на историю лога, мы видим, что коммит с тегом «v1» является коммитом, предшествующим ошибочному коммиту. Давайте сбросим ветку до этой точки. Поскольку ветка имеет тег, мы можем использовать имя тега в команде сброса (если она не имеет тега, мы можем использовать хэш-значение) (см. рис. 59)

```
Initial Commit
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git tag oops
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git reset --hard v1
HEAD is now at 4acd636 Added HTML header
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git log
commit 4acd63657c36c6476d68dde507805c87564baccb (HEAD -> master, tag: v1)
Author: oorazgeldiyeva <1032205431@pfur.ru>
Date: Sat Feb 11 21:17:42 2023 +0300

    Added HTML header

commit 66ced280e42f295a8b1459db826163f1676a7bff (tag: v1-beta)
Author: oorazgeldiyeva <1032205431@pfur.ru>
Date: Sat Feb 11 21:15:54 2023 +0300

    Added standart HTML page tags

commit 026a40162b4ebc4d587fc58b9811c48016e61a6d
Author: oorazgeldiyeva <1032205431@pfur.ru>
Date: Sat Feb 11 21:08:35 2023 +0300

    Added h1 tag

commit f0b2344566edc3a063ced594714f243bda81c161
Author: oorazgeldiyeva <1032205431@pfur.ru>
Date: Sat Feb 11 20:53:43 2023 +0300

    Initial Commit
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$
```

Рисунок 59. Сброс коммита

Наша ветка master теперь указывает на коммит v1, а коммитов Oops и Revert Oops в ветке уже нет. Параметр --hard указывает, что рабочий каталог должен быть обновлен в соответствии с новым head ветки.

Ничего никогда не теряется

Что же случается с ошибочными коммитами? Оказывается, что коммиты все еще находятся в репозитории. На самом деле, мы все еще можем на них ссылаться. Помните, в начале этого урока мы создали для

отмененного коммита тег «oops». Давайте посмотрим на все коммиты (см. рис. 60)

```
Initial Commit
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git log --all
commit c37fa988bb4f31f7013a742a79b0b9d846f4361d (tag: oops)
Author: oorazgeldiyeva <1032205431@pfur.ru>
Date: Sat Feb 11 21:45:07 2023 +0300

    Revert "Oops, we didn't want this commit"

    This reverts commit 779f62ceb6791b3fb314eab4a1e7625d12678366.

commit 779f62ceb6791b3fb314eab4a1e7625d12678366
Author: oorazgeldiyeva <1032205431@pfur.ru>
Date: Sat Feb 11 21:44:45 2023 +0300

    Oops, we didn't want this commit

commit 4acd63657c36c6476d68dde507805c87564baccb (HEAD -> master, tag: v1)
Author: oorazgeldiyeva <1032205431@pfur.ru>
Date: Sat Feb 11 21:17:42 2023 +0300

    Added HTML header

commit 66ced280e42f295a8b1459db826163f1676a7bff (tag: v1-beta)
Author: oorazgeldiyeva <1032205431@pfur.ru>
Date: Sat Feb 11 21:15:54 2023 +0300

    Added standart HTML page tags

commit 026a40162b4ebc4d587fc58b9811c48016e61a6d
Author: oorazgeldiyeva <1032205431@pfur.ru>
Date: Sat Feb 11 21:08:35 2023 +0300

    Added h1 tag

commit f0b2344566edc3a063ced594714f243bda81c161
Author: oorazgeldiyeva <1032205431@pfur.ru>
Date: Sat Feb 11 20:53:43 2023 +0300

    Initial Commit
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$
```

Рисунок 60. История всех коммитов

Мы видим, что ошибочные коммиты не исчезли. Они все еще находятся в репозитории. Просто они отсутствуют в ветке master. Если бы мы не отметили их тегами, они по-прежнему находились бы в репозитории, но не было бы никакой возможности ссылаться на них, кроме как при

помощи их хэш имен. Коммиты, на которые нет ссылок, остаются в репозитории до тех пор, пока не будет запущен сборщик мусора.

Опасность сброса

Сброс в локальных ветках, как правило, безопасен. Последствия любой «аварии» как правило, можно восстановить простым сбросом с помощью нужного коммита. Однако, если ветка «расшарена» на удаленных репозиториях, сброс может сбить с толку других пользователей ветки.

Удаление тега оops

Удаление тега оops

Тег оops свою функцию выполнил. Давайте удалим его и коммиты, на которые он ссылался, сборщиком мусора (см. рис. 61)

```
Initial Commit
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git tag -d oops
Deleted tag 'oops' (was c37fa98)
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git log --all
commit 4acd63657c36c6476d68dde507805c87564baccb (HEAD -> master, tag: v1)
Author: oorazgeldiyeva <1032205431@pfur.ru>
Date: Sat Feb 11 21:17:42 2023 +0300

    Added HTML header

commit 66ced280e42f295a8b1459db826163f1676a7bff (tag: v1-beta)
Author: oorazgeldiyeva <1032205431@pfur.ru>
Date: Sat Feb 11 21:15:54 2023 +0300

    Added standart HTML page tags

commit 026a40162b4ebc4d587fc58b9811c48016e61a6d
Author: oorazgeldiyeva <1032205431@pfur.ru>
Date: Sat Feb 11 21:08:35 2023 +0300

    Added h1 tag

commit f0b2344566edc3a063ced594714f243bda81c161
Author: oorazgeldiyeva <1032205431@pfur.ru>
Date: Sat Feb 11 20:53:43 2023 +0300

    Initial Commit
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$
```

Рисунок 61. Удаление тега

Тег «oops» больше не будет отображаться в репозитории.

Внесение изменений в коммиты

Измените страницу,а затем сделайте коммит

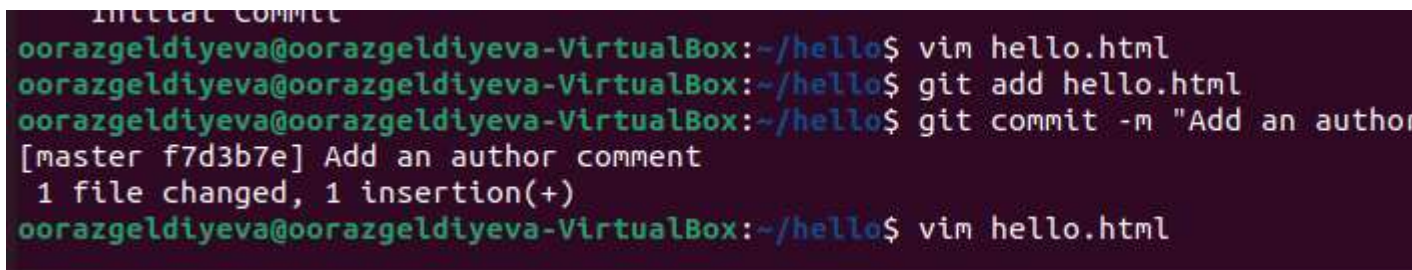
Добавьте в страницу комментарий автора (вставьте свою фамилию) (см. рис. 62)



```
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:
!-- Author: Ogulnur Orazgeldiyeva --
<html>
  <head>
  </head>
  <body>
    <h1>Hello, World!</h1>
  </body>
</html>
```

Рисунок 62. Изменение файла

Выполните (см. рис. 63)

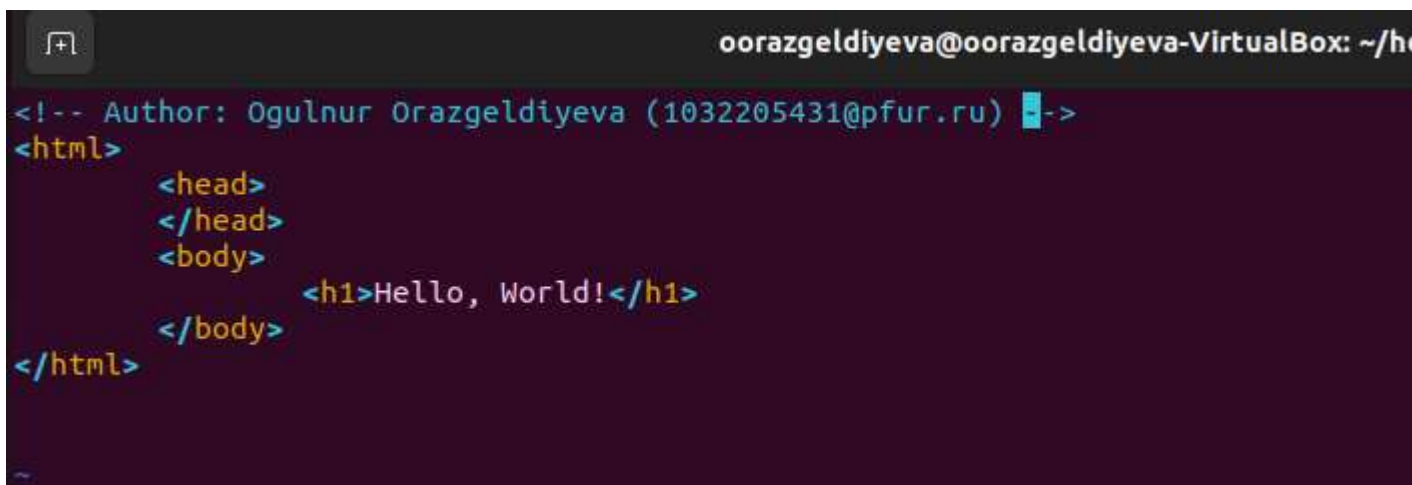


```
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ vim hello.html
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git add hello.html
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git commit -m "Add an author comment"
[master f7d3b7e] Add an author comment
1 file changed, 1 insertion(+)
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ vim hello.html
```

Рисунок 63. Коммит

Необходим email

После совершения коммита вы понимаете, что любой хороший комментарий должен включать электронную почту автора. Обновите страницу hello, включив в нее email (см. рис. 64)

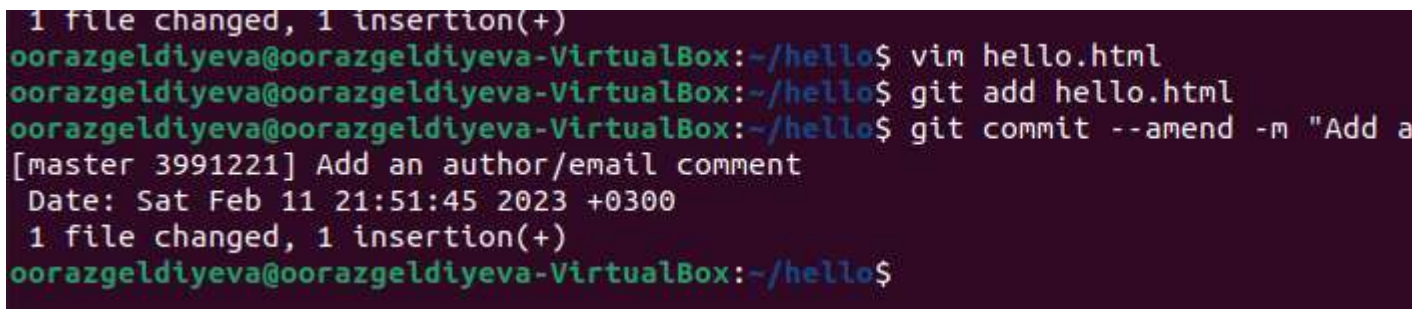


```
oorazgeldiyeva@oorazgeldiyeva-VirtualBox: ~/h
<!-- Author: Ogulnur Orazgeldiyeva (1032205431@pfur.ru) -->
<html>
  <head>
  </head>
  <body>
    <h1>Hello, World!</h1>
  </body>
</html>
```

Рисунок 64. Изменение файла

Измените предыдущий коммит

Мы действительно не хотим создавать отдельный коммит только ради электронной почты. Давайте изменим предыдущий коммит, включив в него адрес электронной почты (см. рис. 64)



```
1 file changed, 1 insertion(+)
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ vim hello.html
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git add hello.html
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git commit --amend -m "Add a
[master 3991221] Add an author/email comment
Date: Sat Feb 11 21:51:45 2023 +0300
1 file changed, 1 insertion(+)
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$
```

Рисунок 64. Коммит

Просмотр истории

Выполните (см. рис. 65)

```

1 file changed, 1 insertion(+)
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git log
commit 3991221920744d13bb952b3a4f13f26535116f16 (HEAD -> master)
Author: oorazgeldiyeva <1032205431@pfur.ru>
Date: Sat Feb 11 21:51:45 2023 +0300

    Add an author/email comment

commit 4acd63657c36c6476d68dde507805c87564baccb (tag: v1)
Author: oorazgeldiyeva <1032205431@pfur.ru>
Date: Sat Feb 11 21:17:42 2023 +0300

    Added HTML header

commit 66ced280e42f295a8b1459db826163f1676a7bff (tag: v1-beta)
Author: oorazgeldiyeva <1032205431@pfur.ru>
Date: Sat Feb 11 21:15:54 2023 +0300

    Added standart HTML page tags

commit 026a40162b4ebc4d587fc58b9811c48016e61a6d
Author: oorazgeldiyeva <1032205431@pfur.ru>
Date: Sat Feb 11 21:08:35 2023 +0300

    Added h1 tag

commit f0b2344566edc3a063ced594714f243bda81c161
Author: oorazgeldiyeva <1032205431@pfur.ru>
Date: Sat Feb 11 20:53:43 2023 +0300

    Initial Commit
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$

```

Рисунок 65. История

Мы можем увидеть, что оригинальный коммит «автор» заменен коммитом «ав-тор/email». Этого же эффекта можно достичь путем сброса последнего коммита в ветке, и повторного коммита новых изменений.

Перемещение файлов

Переместите файл `hello.html` в каталог `lib`

Сейчас мы собираемся создать структуру нашего репозитория. Давайте перенесем страницу в каталог `lib` (см. рис. 66)


```

Initial Commit
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ mkdir lib
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git mv hello.html lib
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        renamed:    hello.html -> lib/hello.html

oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$

```

Рисунок 66. Перемещение файла

Перемещая файлы с помощью `git mv`, мы информируем `git` о 2 вещах:

- Что файл `hello.html` был удален.
- Что файл `lib/hello.html` был создан.
- Оба эти факта сразу же проиндексированы и готовы к коммиту. Команда `git status` сообщает, что файл был перемещен.

Второй способ перемещения файлов

Положительной чертой `git` является то, что вы можете забыть о версионном контроле до того момента, когда вы готовы приступить к коммиту кода. Что бы случилось, если бы мы использовали командную строку операционной системы для перемещения файлов вместо команды `git`?

Следующий набор команд идентичен нашим последним действиям. Работы здесь побольше, но результат тот же.

Мы могли бы выполнить:

```
mkdir lib
```

```
mv hello.html lib
```

```
git add lib/hello.html git rm hello.html
```

Коммит в новый каталог

Давайте сделаем коммит этого перемещения (см. рис. 67)

```

fatal: pathspec 'hello.html' did not match any files
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git commit -m "Moved hello.h
[master 2d4cd59] Moved hello.html to lib
 1 file changed, 0 insertions(+), 0 deletions(-)
  rename hello.html => lib/hello.html (100%)
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$

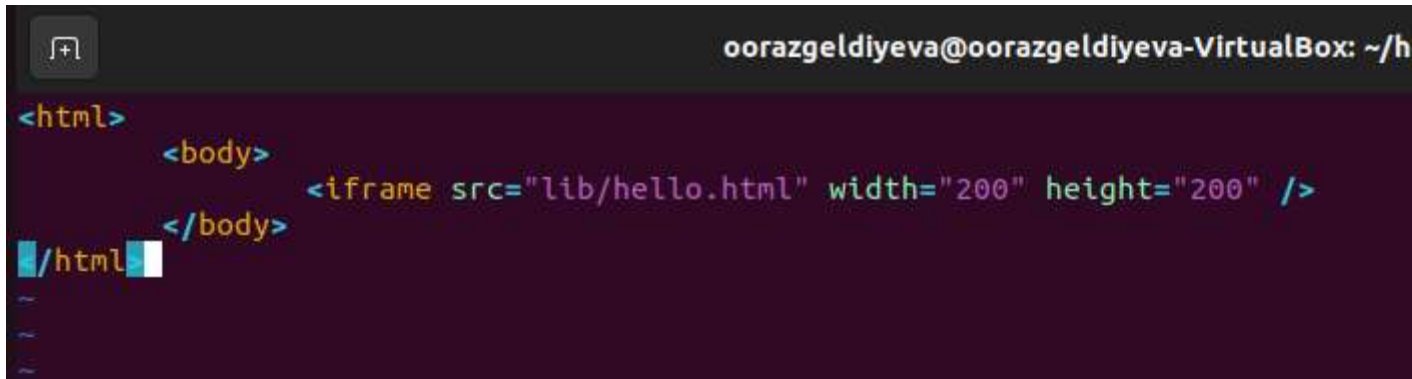
```

Рисунок 67. Коммит

Подробнее о структуре

Добавление index.html

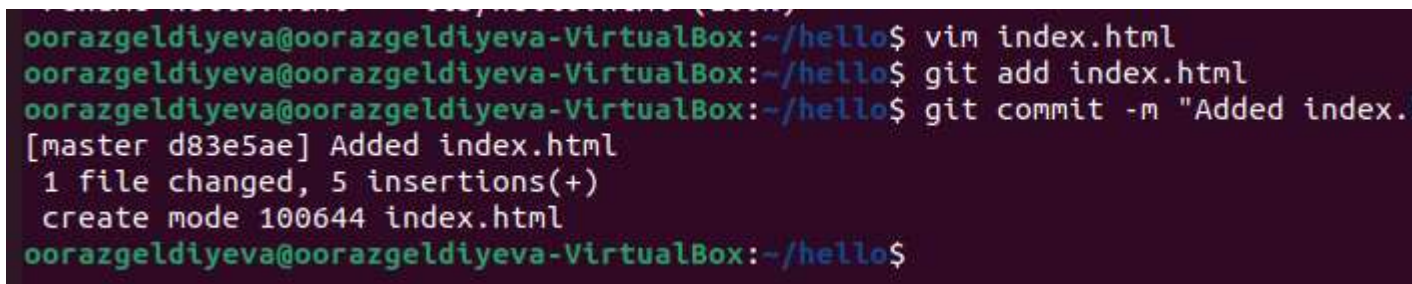
Добавим файл index.html в наш репозиторий (см. рис. 68)

A screenshot of a code editor window. The title bar shows the user 'oorazgeldiyeva' and the machine 'oorazgeldiyeva-VirtualBox'. The editor displays the following HTML code:

```
<html>
  <body>
    <iframe src="lib/hello.html" width="200" height="200" />
  </body>
</html>
```

Рисунок 68. Изменение файла

Добавьте файл и сделайте коммит (см. рис. 69)

A screenshot of a terminal window. The prompt is 'oorazgeldiyeva@oorazgeldiyeva-VirtualBox: ~/hello'. The user enters the following commands and receives the following output:

```
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ vim index.html
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git add index.html
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git commit -m "Added index."
[master d83e5ae] Added index.html
 1 file changed, 5 insertions(+)
 create mode 100644 index.html
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$
```

Рисунок 69. Коммит

Теперь при открытии index.html, вы должны увидеть кусок страницы hello в маленьком окошке (см. рис. 70)



Рисунок 70. Открытие файла

Git внутри: Каталог.git Каталог.git

Выполните (см. рис. 71)

```
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ ls -C .git
branches          COMMIT_EDITMSG.save  description  hooks  info  objects  packed-refs
COMMIT_EDITMSG    config              HEAD        index  logs  ORIG_HEAD refs
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$
```

Рисунок 71. Содержимое каталога

Это каталог, в котором хранится вся информация git.

База данных объектов

Выполните (см. рис. 72)

```
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ ls -C .git/objects
02  28  31  40  4a  55  5e  66  7a  90  c3  d8  e6  f0  info
20  2d  39  46  52  5d  62  77  8a  9b  cf  da  eb  f7  pack
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$
```

Рисунок 72. Содержимое каталога

Вы должны увидеть набор каталогов, имена которых состоят из 2 символов. Имена каталогов являются первыми двумя буквами хэша sha1 объекта, хранящегося в git.

Углубляемся в базу данных объектов

Выполните (см. рис. 73)

```

oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ ls -C .git/objects
02 28 31 40 4a 55 5e 66 7a 90 c3 d8 e6 f0 info
20 2d 39 46 52 5d 62 77 8a 9b cf da eb f7 pack
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ ls -C .git/objects/<dir>
bash: syntax error near unexpected token `newline'
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ ls -C .git/objects/28
153ca56d6ee0fee13fac1c8cb8263160046c46
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$

```

Рисунок 73. Содержимое каталога

Смотрим в один из каталогов с именем из 2 букв. Вы увидите файлы с именами из 38 символов. Это файлы, содержащие объекты, хранящиеся в git. Они сжаты и закодированы, поэтому просмотр их содержимого нам мало чем поможет.

Config File

Выполните (см. рис. 74)

```

oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ cat .git/config
[core]
    repositoryformatversion = 0
    filemode = true
    bare = false
    logallrefupdates = true
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$

```

Рисунок 74. Файл конфигурации

Это файл конфигурации, создающийся для каждого конкретного проекта. Записи в этом файле будут перезаписывать записи в файле .gitconfig вашего главного каталога, по крайней мере в рамках этого проекта.

Ветки и теги

Выполните (см. рис. 75)

```

logallrefupdates = true
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ cat .git/refs
cat: .git/refs: Is a directory
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ ls .git/refs
heads  tags
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ ls .git/refs/heads
master
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ ls .git/refs/tags
v1  v1-beta
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ cat .git/refs/tags/v1
4acd63657c36c6476d68dde507805c87564baccb
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$

```


Рисунок 75. Ветки и теги

Вы должны узнавать файлы в подкаталоге тегов. Каждый файл соответствует тегу, ранее созданному с помощью команды `git tag`. Его содержание—это всего лишь хэш коммита, привязанный к тегу.

Каталог `heads` практически аналогичен, но используется для веток, а не тегов. На данный момент у нас есть только одна ветка, так что все, что вы увидите в этом каталоге—это ветка `master`.

Файл HEAD

Выполните (см. рис. 76)

```
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ cat .git/HEAD
ref: refs/heads/master
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$
```

Рисунок 76. Файл HEAD

Файл `HEAD` содержит ссылку на текущую ветку, в данный момент это должна быть ветка `master`.

Работа непосредственно с объектами git

Поиск последнего коммита

Выполните (см. рис. 77)

```
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git log --max-count=1
commit d83e5aea4cf9b4b1551373255d3aefce54050a3e (HEAD -> master)
Author: oorazgeldiyeva <1032205431@pfur.ru>
Date: Sat Feb 11 21:59:40 2023 +0300

    Added index.html
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$
```

Рисунок 77. Коммит

Эта команда должна показать последний коммит в репозитории. SHA1 хэш в вашей системе, вероятно, отличается от моего, но вы увидите что-то наподобие этого.

Вывод последнего коммита с помощью SHA1 хэша

Выполните (см. рис. 78)

```

Added index.html
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git cat-file -t d83e5aea4cf9
commit
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git cat-file -p d83e5aea4cf9
tree 28153ca56d6ee0fee13fac1c8cb8263160046c46
parent 2d4cd5974a6c504cfa8d5ddc7c2162c909fa42fb
author oorazgeldiyeva <1032205431@pfur.ru> 1676141980 +0300
committer oorazgeldiyeva <1032205431@pfur.ru> 1676141980 +0300

Added index.html
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$

```

Рисунок 78. Коммит

Поиск дерева

Мы можем вывести дерево каталогов, ссылка на который идет в коммите. Это должно быть описание файлов (верхнего уровня) в нашем проекте (для конкретного коммита). Используйте SHA1 хэш из строки «дерева», из списка выше.

Выполните (см. рис. 79)

```

oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git cat-file -p d83e5aea4cf9
tree 28153ca56d6ee0fee13fac1c8cb8263160046c46
parent 2d4cd5974a6c504cfa8d5ddc7c2162c909fa42fb
author oorazgeldiyeva <1032205431@pfur.ru> 1676141980 +0300
committer oorazgeldiyeva <1032205431@pfur.ru> 1676141980 +0300

Added index.html
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git cat-file -p 28153ca56d6ee0fee
100644 blob cf5bc0ea78289c0be141b578ca407f1159bd25bc    index.html
040000 tree 5dc54da7794d81dc76b43ebd66f24ed91a0faa72    lib
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$

```

Рисунок 79. Поиск дерева

Вывод каталога lib

Выполните (см. рис. 80)

```

oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git cat-file -p 28153ca56d6ee0fee
100644 blob cf5bc0ea78289c0be141b578ca407f1159bd25bc    index.html
040000 tree 5dc54da7794d81dc76b43ebd66f24ed91a0faa72    lib
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git cat-file -p 5dc54da7794d81dc76b43e
100644 blob 551dd942106b81c86ce3b52fe711b7bf2106f30f    hello.html
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git cat-file -p 551dd942106b81
<!-- Author: Ogulnur Orazgeldiyeva (1032205431@pfur.ru) -->
<html>
  <head>
  </head>
  <body>
    <h1>Hello, World!</h1>
  </body>
</html>

oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$

```

Рисунок 80. Каталог lib

Исследуйте самостоятельно

Исследуйте git репозиторий вручную самостоятельно. Смотрите, удастся ли вам найти оригинальный файл hello.html с самого первого коммита вручную по ссылкам SHA1 хэша в последнем коммите.

Создание ветки

Пора сделать наш hello world более выразительным. Так как это может занять некоторое время, лучше переместить эти изменения в отдельную ветку, чтобы изолировать их от изменений в ветке master.

Создайте ветку

Давайте назовем нашу новую ветку «style» (см. рис. 81)

```
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git checkout -b style
Switched to a new branch 'style'
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git status
On branch style
nothing to commit, working tree clean
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$
```

Рисунок 81. Ветка

git checkout -b <имя_ветки> является шорткатом для git branch <имя_ветки> за которым идет git checkout <имя_ветки>.

Обратите внимание, что команда git status сообщает о том, что вы находитесь в ветке «style».

Добавьте файл стилей style.css

Файл lib/style.css (см. рис. 82)



Рисунок 82. Изменение файла

Выполните (см. рис. 83)


```

oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ touch lib/style.css
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ vim lib/style.css
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git add lib/style.css
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git commit -m "Added css st
[style 0a81c83] Added css stylesheet
1 file changed, 3 insertions(+)
create mode 100644 lib/style.css
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$

```

Рисунок 83. Добавление индексации и коммита

Измените основную страницу

Обновите файл hello.html, чтобы использовать стили style.css (см. рис. 84)

```

oorazgeldiyeva@oorazgeldiyeva-VirtualBox: ~/he

<!-- Author: Ogulnur Orazgeldiyeva (1032205431@pfur.ru) -->
<html>
  <head>
    <link type="text/css" rel="stylesheet" media="all" href="style.css" />
  </head>
  <body>
    <h1>Hello, World!</h1>
  </body>
</html>

```

Рисунок 84. Изменение файла

Выполните (см. рис. 85)

```

oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ vim lib/hello.html
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git add lib/hello.html
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git commit -m "Hello uses s
[style b3cb5c7] Hello uses style.css
1 file changed, 1 insertion(+)
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$

```

Рисунок 85. Коммит

Измените index.html

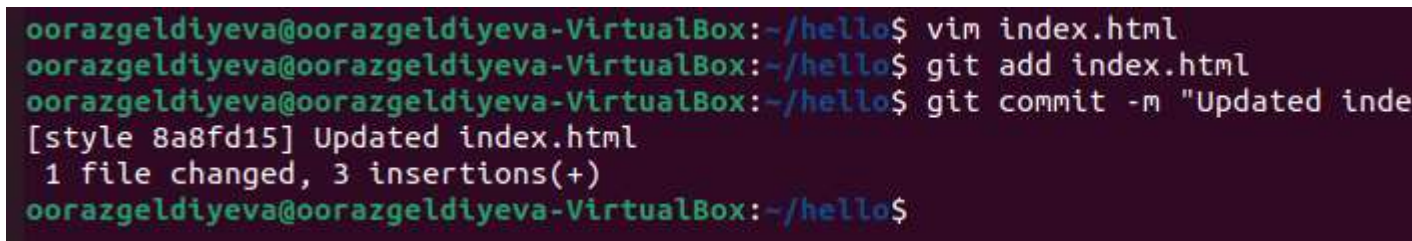
Обновите файл index.html, чтобы он тоже использовал style.css (см. рис. 86)



```
oorazgeldiyeva@oorazgeldiyeva-VirtualBox: ~/h
<html>
  <head>
    <link type="text/css" rel="stylesheet" media="all" href="lib/s
  </head>
  <body>
    <iframe src="lib/hello.html" width="200" height="200" />
  </body>
</html>
```

Рисунок 86. Изменение файла

Выполните (см. рис. 87)



```
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ vim index.html
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git add index.html
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git commit -m "Updated index.html"
[style 8a8fd15] Updated index.html
1 file changed, 3 insertions(+)
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$
```

Рисунок 87. Коммит

Навигация по веткам

Теперь в вашем проекте есть две ветки:

Выполните (см. рис. 88)

```
commit 8a8fd15a5ad8ad00904fe7471d9cbe465abdbaeb (HEAD -> style)
Author: oorazgeldiyeva <1032205431@pfur.ru>
Date: Sat Feb 11 22:19:40 2023 +0300
```

Updated index.html

```
commit b3cb5c78ecae97c82d1e064587a379c522ab5b85
Author: oorazgeldiyeva <1032205431@pfur.ru>
Date: Sat Feb 11 22:16:42 2023 +0300
```

Hello uses style.css

```
commit 0a81c8313d653cbe7cd59037f9e046ba9e93a413
Author: oorazgeldiyeva <1032205431@pfur.ru>
Date: Sat Feb 11 22:13:42 2023 +0300
```

Added css stylesheet

```
commit d83e5aea4cf9b4b1551373255d3aefce54050a3e (master)
Author: oorazgeldiyeva <1032205431@pfur.ru>
Date: Sat Feb 11 21:59:40 2023 +0300
```

Added index.html

```
commit 2d4cd5974a6c504cfa8d5ddc7c2162c909fa42fb
Author: oorazgeldiyeva <1032205431@pfur.ru>
Date: Sat Feb 11 21:57:22 2023 +0300
```

Moved hello.html to lib

```
commit 3991221920744d13bb952b3a4f13f26535116f16
Author: oorazgeldiyeva <1032205431@pfur.ru>
Date: Sat Feb 11 21:51:45 2023 +0300
```

Add an author/email comment

```
commit 4acd63657c36c6476d68dde507805c87564baccb (tag: v1)
Author: oorazgeldiyeva <1032205431@pfur.ru>
Date: Sat Feb 11 21:17:42 2023 +0300
```

Added HTML header

```
:|
```

Рисунок 88. История всех коммитов

Переключение на ветку master

Используйте команду `git checkout` для переключения между ветками (см. рис. 89)

```

oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git checkout master
Switched to branch 'master'
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ cat lib/hello.html
<!-- Author: Ogulnur Orazgeldiyeva (1032205431@pfur.ru) -->
<html>
    <head>
    </head>
    <body>
        <h1>Hello, World!</h1>
    </body>
</html>

oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$

```

Рисунок 89. Переключение между ветками

Сейчас мы находимся на ветке master. Это заметно по тому, что файл hello.html не использует стили style.css.

Вернемся к ветке style

Выполните (см. рис. 90)

git checkout style cat lib/hello.html

```

oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git checkout style
Switched to branch 'style'
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ cat lib/hello.html
<!-- Author: Ogulnur Orazgeldiyeva (1032205431@pfur.ru) -->
<html>
    <head>
    <link type="text/css" rel="stylesheet" media="all" href="style.css" />
    </head>
    <body>
        <h1>Hello, World!</h1>
    </body>
</html>

oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$

```

Рисунок 90. Переключение между ветками

Содержимое lib/hello.html подтверждает, что мы вернулись на ветку style.

Изменения в ветке master

Пока вы меняли ветку style, кто-то решил обновить ветку master. Они добавили файл README.md.

Создайте файл README в ветке master

Создайте файл README.md (см. рис. 91)

```

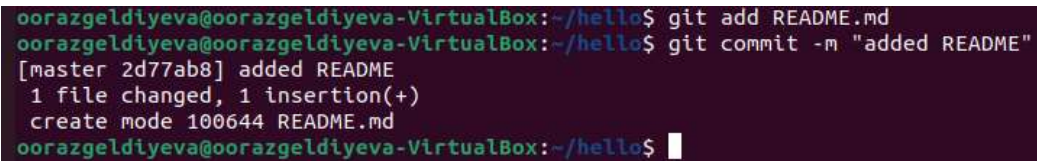
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git checkout master
Already on 'master'
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ echo "This is the Hello World exanple from the git tutorial." > README.md
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$

```


Рисунок 91. Создание файла README

Сделайте коммит изменений README.md в ветку master .

Выполните (см. рис. 92)

A screenshot of a terminal window with a dark purple background and green text. The prompt is 'oorazgeldiyeva@oorazgeldiyeva-VirtualBox: ~/hello\$'. The first command is 'git add README.md'. The second command is 'git commit -m "added README"'. The output shows the commit hash '[master 2d77ab8]', the file added, and the file mode 'create mode 100644 README.md'. The prompt returns to 'oorazgeldiyeva@oorazgeldiyeva-VirtualBox: ~/hello\$' with a cursor.

```
oorazgeldiyeva@oorazgeldiyeva-VirtualBox: ~/hello$ git add README.md
oorazgeldiyeva@oorazgeldiyeva-VirtualBox: ~/hello$ git commit -m "added README"
[master 2d77ab8] added README
1 file changed, 1 insertion(+)
create mode 100644 README.md
oorazgeldiyeva@oorazgeldiyeva-VirtualBox: ~/hello$
```

Рисунок 92. Коммит

Просмотр отличающихся веток

Посмотрите текущие ветки

Теперь у нас в репозитории есть две отличающиеся ветки. Используйте следующую лог-команду для просмотра веток и их отличий.

Выполните (см. рис. 93)

```

commit 2d77ab8f1ac4da13cb663b634df24e5811b85597 (HEAD -> master)
Author: oorazgeldiyeva <1032205431@pfur.ru>
Date: Sat Feb 11 22:24:48 2023 +0300

    added README

* commit 8a8fd15a5ad8ad00904fe7471d9cbe465abdbaeb (style)
| Author: oorazgeldiyeva <1032205431@pfur.ru>
| Date: Sat Feb 11 22:19:40 2023 +0300
|
|     Updated index.html
|
* commit b3cb5c78ecae97c82d1e064587a379c522ab5b85
| Author: oorazgeldiyeva <1032205431@pfur.ru>
| Date: Sat Feb 11 22:16:42 2023 +0300
|
|     Hello uses style.css
|
* commit 0a81c8313d653cbe7cd59037f9e046ba9e93a413
| Author: oorazgeldiyeva <1032205431@pfur.ru>
| Date: Sat Feb 11 22:13:42 2023 +0300
|
|     Added css stylesheet
|
commit d83e5aea4cf9b4b1551373255d3aefce54050a3e
Author: oorazgeldiyeva <1032205431@pfur.ru>
Date: Sat Feb 11 21:59:40 2023 +0300

    Added index.html

commit 2d4cd5974a6c504cfa8d5ddc7c2162c909fa42fb
Author: oorazgeldiyeva <1032205431@pfur.ru>
Date: Sat Feb 11 21:57:22 2023 +0300

    Moved hello.html to lib

commit 3991221920744d13bb952b3a4f13f26535116f16
Author: oorazgeldiyeva <1032205431@pfur.ru>
Date: Sat Feb 11 21:51:45 2023 +0300

    Add an author/email comment

```

Рисунок 93. Просмотр веток

Добавление опции `--graph` в `git log` вызывает построение дерева коммитов с помощью простых ASCII символов. Мы видим обе ветки (`style` и `master`), и то, что ветка `master` является текущей HEAD. Общим предшественником обеих веток является коммит «Added index.html».

Опция `--all` гарантированно означает, что мы видим все ветки. По умолчанию показывается только текущая ветка.

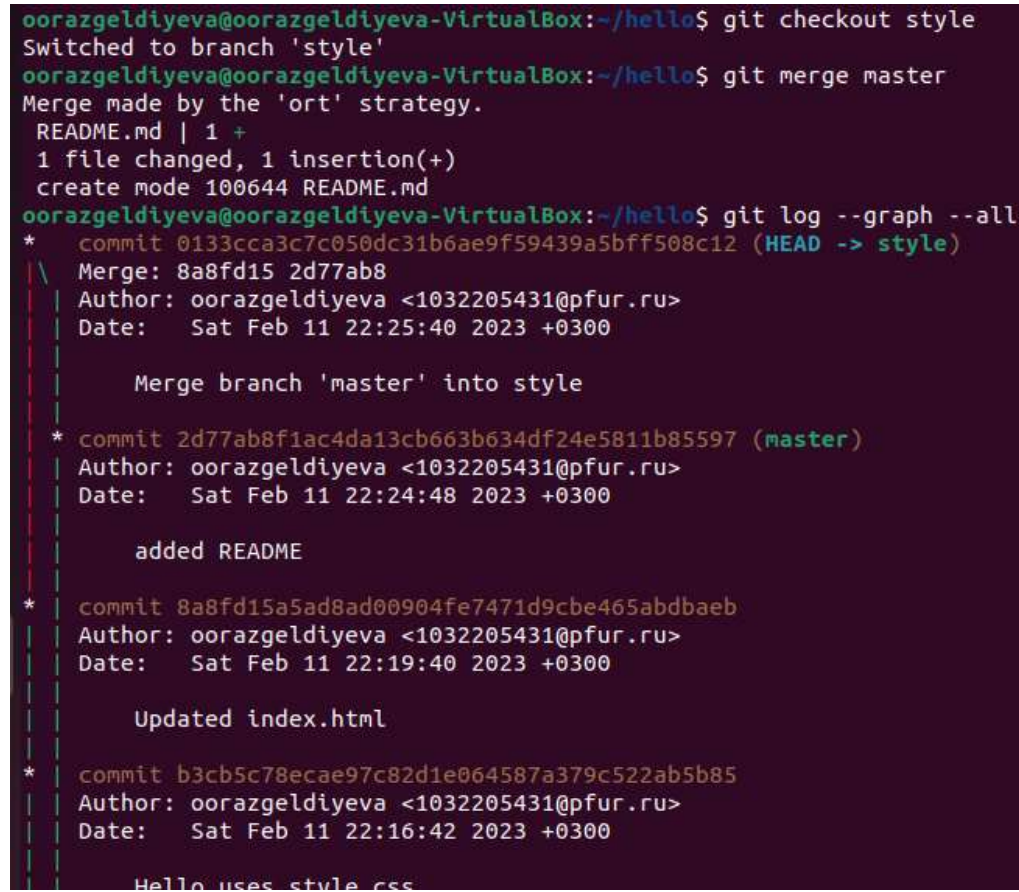
Слияние

Слияние веток

Слияние переносит изменения из двух веток в одну. Давайте вернемся к ветке style и сольем master с style.

Выполните (см. рис. 94)

git checkout style git merge master git log --graph --all



```
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git checkout style
Switched to branch 'style'
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git merge master
Merge made by the 'ort' strategy.
 README.md | 1 +
 1 file changed, 1 insertion(+)
 create mode 100644 README.md
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git log --graph --all
*   commit 0133cca3c7c050dc31b6ae9f59439a5bff508c12 (HEAD -> style)
|  \ Merge: 8a8fd15 2d77ab8
|   Author: oorazgeldiyeva <1032205431@pfur.ru>
|   Date:   Sat Feb 11 22:25:40 2023 +0300
|
|       Merge branch 'master' into style
|
| *   commit 2d77ab8f1ac4da13cb663b634df24e5811b85597 (master)
| |  \ Author: oorazgeldiyeva <1032205431@pfur.ru>
| |   Date:   Sat Feb 11 22:24:48 2023 +0300
| |
| |       added README
| |
| *   commit 8a8fd15a5ad8ad00904fe7471d9cbe465abdbaeb
| |  \ Author: oorazgeldiyeva <1032205431@pfur.ru>
| |   Date:   Sat Feb 11 22:19:40 2023 +0300
| |
| |       Updated index.html
| |
| *   commit b3cb5c78ecae97c82d1e064587a379c522ab5b85
| |  \ Author: oorazgeldiyeva <1032205431@pfur.ru>
| |   Date:   Sat Feb 11 22:16:42 2023 +0300
| |
| |       Hello uses style.css
```

Рисунок 94. Просмотр веток

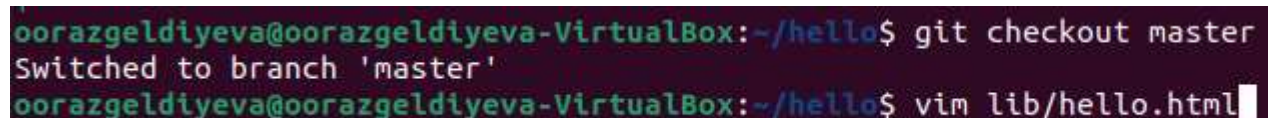
Путем периодического слияния ветки master с веткой style вы можете переносить из master любые изменения и поддерживать совместимость изменений style с изменениями в основной ветке.

Но что если изменения в ветке master конфликтуют с изменениями в style?

Создание конфликта

Вернитесь в master и создайте конфликт

Вернитесь в ветку master и внесите следующие изменения (см. рис. 95)



```
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git checkout master
Switched to branch 'master'
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ vim lib/hello.html
```

Рисунок 95. Переключение между ветками

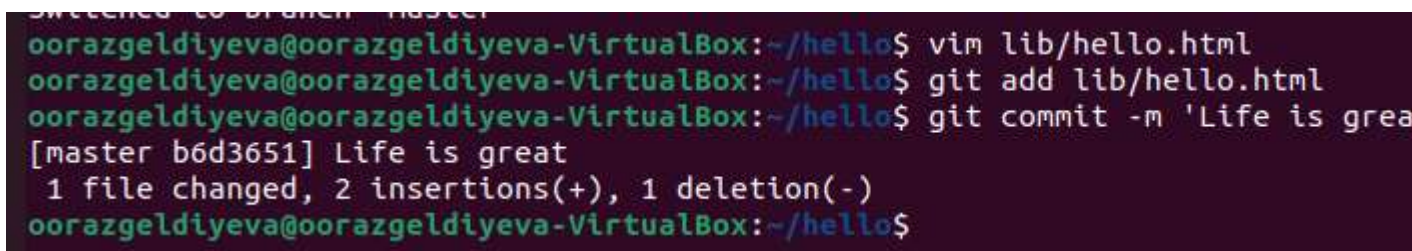
Файл lib/hello.html (см. рис. 96)



```
oorazgeldiyeva@oorazgeldiyeva-VirtualBox: ~/hello
<!-- Author: Ogulnur Orazgeldiyeva (1032205431@pfur.ru) -->
<html>
  <head>
    <!-- no style>
  </head>
  <body>
    <h1>Hello, World! Life is great!</h1>
  </body>
</html>
```

Рисунок 96. Изменение файла

Выполните (см. рис. 97)



```
Switched to branch 'master'
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ vim lib/hello.html
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git add lib/hello.html
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git commit -m 'Life is great'
[master b6d3651] Life is great
 1 file changed, 2 insertions(+), 1 deletion(-)
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$
```

Рисунок 97. Коммит

Просмотр веток

Выполните (см. рис. 98)


```

oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git log --graph --all
* commit b6d3651040e1f30b5ea3d144c9fbc32630f3705d (HEAD -> master)
  Author: oorazgeldiyeva <1032205431@pfur.ru>
  Date: Sat Feb 11 22:29:01 2023 +0300

    Life is great

*   commit 0133cca3c7c050dc31b6ae9f59439a5bffa508c12 (style)
  |   Merge: 8a8fd15 2d77ab8
  |   Author: oorazgeldiyeva <1032205431@pfur.ru>
  |   Date: Sat Feb 11 22:25:40 2023 +0300
  |
  |   Merge branch 'master' into style
  |
  * commit 2d77ab8f1ac4da13cb663b634df24e5811b85597
    Author: oorazgeldiyeva <1032205431@pfur.ru>
    Date: Sat Feb 11 22:24:48 2023 +0300

      added README

  * commit 8a8fd15a5ad8ad00904fe7471d9cbe465abdbaeb
    Author: oorazgeldiyeva <1032205431@pfur.ru>
    Date: Sat Feb 11 22:19:40 2023 +0300

      Updated index.html

  * commit b3cb5c78ecae97c82d1e064587a379c522ab5b85
    Author: oorazgeldiyeva <1032205431@pfur.ru>
    Date: Sat Feb 11 22:16:42 2023 +0300

      Hello uses style.css

  * commit 0a81c8313d653cbe7cd59037f9e046ba9e93a413
    Author: oorazgeldiyeva <1032205431@pfur.ru>
    Date: Sat Feb 11 22:13:42 2023 +0300

      Added css stylesheet

```

Рисунок 98. Просмотр веток

После коммита «Added README» ветка master была объединена с веткой style, но в настоящее время в master есть дополнительный коммит, который не был слит с style.

Последнее изменение в master конфликтует с некоторыми изменениями в style. На следующем шаге мы решим этот конфликт.

Разрешение конфликтов

Слияние master с веткой style

Теперь вернемся к ветке style и попытаемся объединить ее с новой веткой master.

Выполните (см. рис. 99)

```
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git checkout style
Switched to branch 'style'
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git merge master
Auto-merging lib/hello.html
CONFLICT (content): Merge conflict in lib/hello.html
Automatic merge failed; fix conflicts and then commit the result.
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$
```

Рисунок 99. Слияние веток

Если вы откроете lib/hello.html,вы увидите (см. рис. 100)

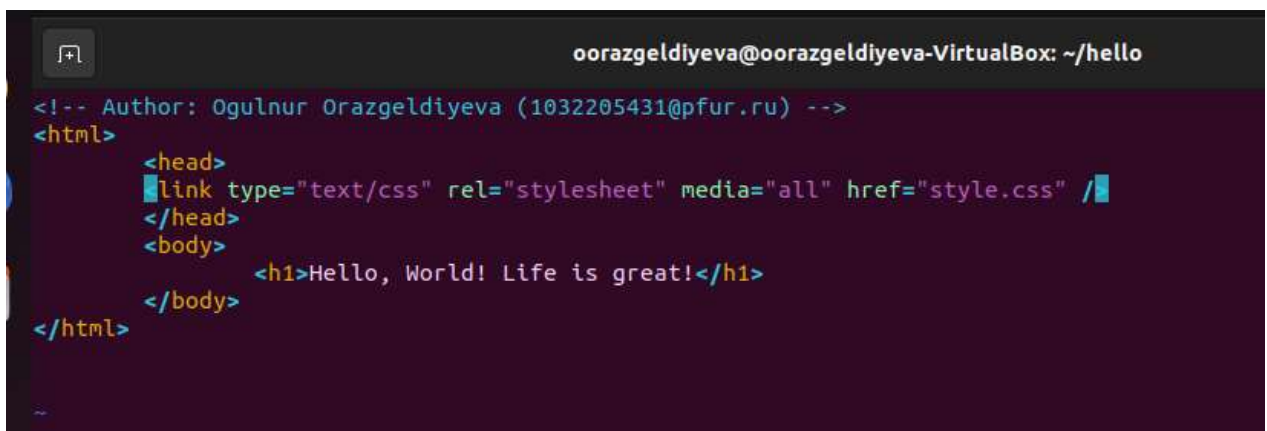
```
oorazgeldiyeva@oorazgeldiyeva-VirtualBox: ~/
<!-- Author: Ogulnur Orazgeldiyeva (1032205431@pfur.ru) -->
<html>
  <head>
<<<<<< HEAD
    <link type="text/css" rel="stylesheet" media="all" href="style.css" /
=====
    <!-- no style
>>>>>> master
    </head>
    <body>
        <h1>Hello, World! Life is great!</h1>
    </body>
  </html>
```

Рисунок 100. Содержимое файла

Первый раздел—версия текущей ветки (style).Второй раздел—версия ветки master.

Решение конфликта

Вам необходимо вручную разрешить конфликт. Внесите изменения в lib/hello.html для достижения следующего результата (см. рис. 101)



```
oorazgeldiyeva@oorazgeldiyeva-VirtualBox: ~/hello
<!-- Author: Ogulnur Orazgeldiyeva (1032205431@pfur.ru) -->
<html>
  <head>
    <link type="text/css" rel="stylesheet" media="all" href="style.css" />
  </head>
  <body>
    <h1>Hello, World! Life is great!</h1>
  </body>
</html>
```

Рисунок 101. Изменение файла

Сделайте коммит решения конфликта

Выполните (см. рис. 102)



```
oorazgeldiyeva@oorazgeldiyeva-VirtualBox: ~/hello$ vln lib/hello.html
oorazgeldiyeva@oorazgeldiyeva-VirtualBox: ~/hello$ git add lib/hello.html
oorazgeldiyeva@oorazgeldiyeva-VirtualBox: ~/hello$ git commit -m "Merged master fixed conflict."
[style cb823f2] Merged master fixed conflict.
oorazgeldiyeva@oorazgeldiyeva-VirtualBox: ~/hello$
```

Рисунок 102. Коммит

Перебазирование как альтернатива слиянию

Рассмотрим различия между слиянием и перебазированием. Для того, чтобы это сделать, нам нужно вернуться в репозиторий в момент до первого слияния, а затем повторить те же действия, но с использованием перебазирования вместо слияния.

Мы будем использовать команду reset для возврата веток к предыдущему состоянию.

Сброс ветки style

Сброс ветки style

Вернемся на ветке style к точке перед тем, как мы слили ее с веткой master. Мы можем сбросить ветку к любому коммиту. По сути, это изменение указателя ветки на любую точку дерева коммитов.

В этом случае мы хотим вернуться к ветке style в точку перед слиянием с master. Нам необходимо найти последний коммит перед слиянием.

Выполните (см. рис. 103)

```

oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git checkout style
Already on 'style'
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git log --graph
*   commit c8823f201a7820a2082ef49a5dd52811caa1a840 (HEAD -> style)
|\
| Merge: 0133cca b6d3651
| Author: oorazgeldiyeva <1032205431@pfur.ru>
| Date:   Sat Feb 11 22:32:48 2023 +0300
|
|     Merged master fixed conflict.
|
*   commit b6d3651040e1f30b5ea3d144c9fbc32630f3705d (master)
|   Author: oorazgeldiyeva <1032205431@pfur.ru>
|   Date:   Sat Feb 11 22:29:01 2023 +0300
|
|       Life is great
|
*   commit 0133cca3c7c050dc31b6ae9f59439a5bff508c12
|\
| Merge: 8a8fd15 2d77ab8
| Author: oorazgeldiyeva <1032205431@pfur.ru>
| Date:   Sat Feb 11 22:25:40 2023 +0300
|
|     Merge branch 'master' into style
|
*   commit 2d77ab8f1ac4da13cb663b634df24e5811b85597
|   Author: oorazgeldiyeva <1032205431@pfur.ru>
|   Date:   Sat Feb 11 22:24:48 2023 +0300
|
|     added README
|
*   commit 8a8fd15a5ad8ad00904fe7471d9cbe465abdbaeb
|   Author: oorazgeldiyeva <1032205431@pfur.ru>
|   Date:   Sat Feb 11 22:19:40 2023 +0300
|
|     Updated index.html
|
*   commit b3cb5c78ecae97c82d1e064587a379c522ab5b85
|   Author: oorazgeldiyeva <1032205431@pfur.ru>
|   Date:   Sat Feb 11 22:16:42 2023 +0300
|
|     Hello uses style.css
|
*   commit 0a81c8313d653cbe7cd59037f9e046ba9e93a413
|   Author: oorazgeldiyeva <1032205431@pfur.ru>
|   Date:   Sat Feb 11 22:15:42 2023 +0300
|
|     Hello uses style.css

```

Рисунок 103. Просмотр веток

Мы видим, что коммит «Updated index.html» был последним на ветке style перед слиянием. Давайте сбросим ветку style к этому коммиту.

Выполните (см. рис. 104)


```
*   commit 8a8fd15a5ad8ad00904fe7471d9cbe465abdbaeb
    Author: oorazgeldiyeva <1032205431@pfur.ru>
    Date:   Sat Feb 11 22:19:40 2023 +0300

        Updated index.html

*   commit b3cb5c78ecae97c82d1e064587a379c522ab5b85
    Author: oorazgeldiyeva <1032205431@pfur.ru>
    Date:   Sat Feb 11 22:16:42 2023 +0300

        Hello uses style.css

*   commit 0a81c8313d653cbe7cd59037f9e046ba9e93a413
    Author: oorazgeldiyeva <1032205431@pfur.ru>
    Date:   Sat Feb 11 22:13:42 2023 +0300

oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git reset --hard 8a8fd15a5ad8ad00
HEAD is now at 8a8fd15 Updated index.html
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$
```

Рисунок 104. Сброс

Проверьте ветку.

Поищите лог ветки style. У нас в истории больше нет коммитов слияний.

Выполните (см. рис. 105)

```

oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git log --graph --all
* commit b6d3651040e1f30b5ea3d144c9fbc32630f3705d (master)
  Author: oorazgeldiyeva <1032205431@pfur.ru>
  Date: Sat Feb 11 22:29:01 2023 +0300

    Life is great

* commit 2d77ab8f1ac4da13cb663b634df24e5811b85597
  Author: oorazgeldiyeva <1032205431@pfur.ru>
  Date: Sat Feb 11 22:24:48 2023 +0300

    added README

* commit 8a8fd15a5ad8ad00904fe7471d9cbe465abdbaeb (HEAD -> style)
  Author: oorazgeldiyeva <1032205431@pfur.ru>
  Date: Sat Feb 11 22:19:40 2023 +0300

    Updated index.html

* commit b3cb5c78ecae97c82d1e064587a379c522ab5b85
  Author: oorazgeldiyeva <1032205431@pfur.ru>
  Date: Sat Feb 11 22:16:42 2023 +0300

    Hello uses style.css

* commit 0a81c8313d653cbe7cd59037f9e046ba9e93a413
  Author: oorazgeldiyeva <1032205431@pfur.ru>
  Date: Sat Feb 11 22:13:42 2023 +0300

    Added css stylesheet

* commit d83e5aea4cf9b4b1551373255d3aefce54050a3e
  Author: oorazgeldiyeva <1032205431@pfur.ru>
  Date: Sat Feb 11 21:59:40 2023 +0300

    Added index.html

* commit 2d4cd5974a6c504cfa8d5ddc7c2162c909fa42fb
  Author: oorazgeldiyeva <1032205431@pfur.ru>
  Date: Sat Feb 11 21:59:40 2023 +0300

    Added index.html

```

Рисунок 106. Просмотр веток

Сброс ветки master

Сброс ветки master

Добавив интерактивный режим в ветку master, мы внесли изменения, конфликтующие с изменениями в ветке style. Давайте вернемся в ветку master в точку перед внесением конфликтующих изменений. Это позволяет нам продемонстрировать работу команды git rebase, не беспокоясь о конфликтах.

Выполните (см. рис. 107)

```
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git checkout master
Switched to branch 'master'
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git log --graph
* commit b6d3651040e1f30b5ea3d144c9fbc32630f3705d (HEAD -> master)
| Author: oorazgeldiyeva <1032205431@pfur.ru>
| Date: Sat Feb 11 22:29:01 2023 +0300
|
| Life is great
|
* commit 2d77ab8f1ac4da13cb663b634df24e5811b85597
| Author: oorazgeldiyeva <1032205431@pfur.ru>
| Date: Sat Feb 11 22:24:48 2023 +0300
|
| added README
|
* commit d83e5aea4cf9b4b1551373255d3aefce54050a3e
| Author: oorazgeldiyeva <1032205431@pfur.ru>
| Date: Sat Feb 11 21:59:40 2023 +0300
|
| Added index.html
|
* commit 2d4cd5974a6c504cfa8d5ddc7c2162c909fa42fb
| Author: oorazgeldiyeva <1032205431@pfur.ru>
| Date: Sat Feb 11 21:57:22 2023 +0300
|
| Moved hello.html to lib
|
* commit 3991221920744d13bb952b3a4f13f26535116f16
| Author: oorazgeldiyeva <1032205431@pfur.ru>
| Date: Sat Feb 11 21:51:45 2023 +0300
|
| Add an author/email comment
|
* commit 4acd63657c36c6476d68dde507805c87564baccb (tag: v1)
| Author: oorazgeldiyeva <1032205431@pfur.ru>
| Date: Sat Feb 11 21:17:42 2023 +0300
|
| Added HTML header
|
* commit 66ced280e42f295a8b1459db826163f1676a7bfff (tag: v1-beta)
| Author: oorazgeldiyeva <1032205431@pfur.ru>
| Date: Sat Feb 11 21:15:54 2023 +0300
```

Рисунок 107. Просмотр веток

Коммит «Added README» идет непосредственно перед коммитом конфлик- тующего интерактивного режима. Мы сбросим ветку master к коммиту «Added README».

Выполните (см. рис. 108)

```

oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git reset --hard 2d77ab8f1ac4d
HEAD is now at 2d77ab8 added README
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git log --graph --all
* commit 2d77ab8f1ac4da13cb663b634df24e5811b85597 (HEAD -> master)
  Author: oorazgeldiyeva <1032205431@pfur.ru>
  Date: Sat Feb 11 22:24:48 2023 +0300

    added README

* commit 8a8fd15a5ad8ad00904fe7471d9cbe465abdbaeb (style)
  Author: oorazgeldiyeva <1032205431@pfur.ru>
  Date: Sat Feb 11 22:19:40 2023 +0300

    Updated index.html

* commit b3cb5c78ecae97c82d1e064587a379c522ab5b85
  Author: oorazgeldiyeva <1032205431@pfur.ru>
  Date: Sat Feb 11 22:16:42 2023 +0300

    Hello uses style.css

* commit 0a81c8313d653cbe7cd59037f9e046ba9e93a413
  Author: oorazgeldiyeva <1032205431@pfur.ru>
  Date: Sat Feb 11 22:13:42 2023 +0300

    Added css stylesheet

* commit d83e5aea4cf9b4b1551373255d3aefce54050a3e
  Author: oorazgeldiyeva <1032205431@pfur.ru>
  Date: Sat Feb 11 21:59:40 2023 +0300

    Added index.html

* commit 2d4cd5974a6c504cfa8d5ddc7c2162c909fa42fb
  Author: oorazgeldiyeva <1032205431@pfur.ru>
  Date: Sat Feb 11 21:57:22 2023 +0300

    Moved hello.html to lib

* commit 3991221920744d13bb952b3a4f13f26535116f16
  Author: oorazgeldiyeva <1032205431@pfur.ru>
  Date: Sat Feb 11 21:51:45 2023 +0300

```

Рисунок 108. Сброс и просмотр веток

Просмотрите лог. Он должен выглядеть, как будто репозиторий был перемотан назад во времени к точке до какого-либо слияния.

Перебазирование

Используем команду rebase вместо команды merge. Мы вернулись в точку до первого слияния и хотим перенести изменения из ветки master в нашу ветку style. На этот раз для переноса изменений из ветки master мы будем использовать команду git rebase вместо слияния.

Выполните (см. рис. 109)


```
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git checkout style
Switched to branch 'style'
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git rebase master
Successfully rebased and updated refs/heads/style.
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git log --graph
* commit bdd7905b0af72117525bb6826deb6e2c911e7f7a (HEAD -> style)
  Author: oorazgeldiyeva <1032205431@pfur.ru>
  Date: Sat Feb 11 22:19:40 2023 +0300

    Updated index.html

* commit c71019e3b6df65ac921b7bf4f4954fca26d8befc
  Author: oorazgeldiyeva <1032205431@pfur.ru>
  Date: Sat Feb 11 22:16:42 2023 +0300

    Hello uses style.css

* commit e55f2368043dd49f9fe66f254ba690f94e1c7f04
  Author: oorazgeldiyeva <1032205431@pfur.ru>
  Date: Sat Feb 11 22:13:42 2023 +0300

    Added css stylesheet

* commit 2d77ab8f1ac4da13cb663b634df24e5811b85597 (master)
  Author: oorazgeldiyeva <1032205431@pfur.ru>
  Date: Sat Feb 11 22:24:48 2023 +0300

    added README

* commit d83e5aea4cf9b4b1551373255d3aefce54050a3e
  Author: oorazgeldiyeva <1032205431@pfur.ru>
  Date: Sat Feb 11 21:59:40 2023 +0300

    Added index.html

* commit 2d4cd5974a6c504cfa8d5ddc7c2162c909fa42fb
  Author: oorazgeldiyeva <1032205431@pfur.ru>
  Date: Sat Feb 11 21:57:22 2023 +0300
```

Рисунок 109. Перебазирование

Слияние VS перебазирование

Конечный результат перебазирования очень похож на результат слияния. Ветка style в настоящее время содержит все свои изменения, а также все изменения ветки master. Однако, дерево коммитов значительно отличается. Дерево коммитов ветки style было переписано таким образом, что ветка master является частью истории коммитов. Это делает цепь коммитов линейной и гораздо более читабельной.

Не используйте перебазирование:

- если ветка является публичной и расшаренной, поскольку переписывание общих веток будет мешать работе других членов команды;
- когда важна точная история коммитов ветки, так как команда rebase переписывает историю коммитов;

Учитывая приведенные выше рекомендации, рекомендуется использовать git rebase для кратковременных, локальных веток, а слияние для веток в публичном репозитории.

Слияние в ветку master

Мы поддерживали соответствие ветки style с веткой master (с помощью rebase), теперь давайте сольем изменения style в ветку master.

Слияние style в master

Выполните (см. рис. 110)

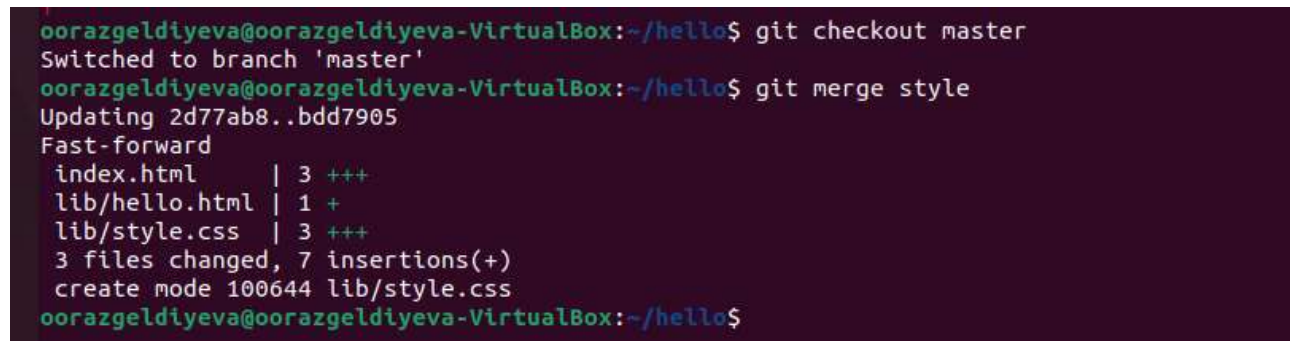
A screenshot of a terminal window with a dark background and light-colored text. The prompt is 'oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello\$'. The first command is 'git checkout master', which outputs 'Switched to branch 'master''. The second command is 'git merge style', which outputs 'Updating 2d77ab8..bdd7905', 'Fast-forward', a diff summary for 'index.html' (3 additions), 'lib/hello.html' (1 addition), and 'lib/style.css' (3 additions), and '3 files changed, 7 insertions(+)' followed by 'create mode 100644 lib/style.css'. The prompt returns to 'oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello\$'.

Рисунок 110. Слияние веток

Поскольку последний коммит ветки master прямо предшествует последнему коммиту ветки style, git может выполнить ускоренное слияние-перемотку. При быстрой перемотке вперед git просто передвигает указатель вперед, таким образом указывая на тот же коммит, что и ветка style.

При быстрой перемотке конфликтов быть не может.

Просмотрите логи

Выполните (см. рис. 111)

```

oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git log
commit bdd7905b0af72117525bb6826deb6e2c911e7f7a (HEAD -> master, style)
Author: oorazgeldiyeva <1032205431@pfur.ru>
Date: Sat Feb 11 22:19:40 2023 +0300

    Updated index.html

commit c71019e3b6df65ac921b7bf4f4954fca26d8befc
Author: oorazgeldiyeva <1032205431@pfur.ru>
Date: Sat Feb 11 22:16:42 2023 +0300

    Hello uses style.css

commit e55f2368043dd49f9fe66f254ba690f94e1c7f04
Author: oorazgeldiyeva <1032205431@pfur.ru>
Date: Sat Feb 11 22:13:42 2023 +0300

    Added css stylesheet

commit 2d77ab8f1ac4da13cb663b634df24e5811b85597
Author: oorazgeldiyeva <1032205431@pfur.ru>
Date: Sat Feb 11 22:24:48 2023 +0300

    added README

commit d83e5aea4cf9b4b1551373255d3aefce54050a3e
Author: oorazgeldiyeva <1032205431@pfur.ru>
Date: Sat Feb 11 21:59:40 2023 +0300

    Added index.html

commit 2d4cd5974a6c504cfa8d5ddc7c2162c909fa42fb
Author: oorazgeldiyeva <1032205431@pfur.ru>
Date: Sat Feb 11 21:57:22 2023 +0300

    Moved hello.html to lib

commit 3991221920744d13bb952b3a4f13f26535116f16
Author: oorazgeldiyeva <1032205431@pfur.ru>
Date: Sat Feb 11 21:51:45 2023 +0300

    Add an author/email comment

```

Рисунок 111. Просмотр логов

Теперь ветки style и master идентичны.

Клонирование репозитория

Перейдите в рабочий каталог

Перейдите в рабочий каталог и сделайте клон вашего репозитория hello (см. рис. 112)

```

oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ cd ..
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~$ pwd
/home/oorazgeldiyeva
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~$ ls
Desktop Documents Downloads hello Music Pictures Public snap Templates Videos work
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~$

```

Рисунок 112. Переход в каталог

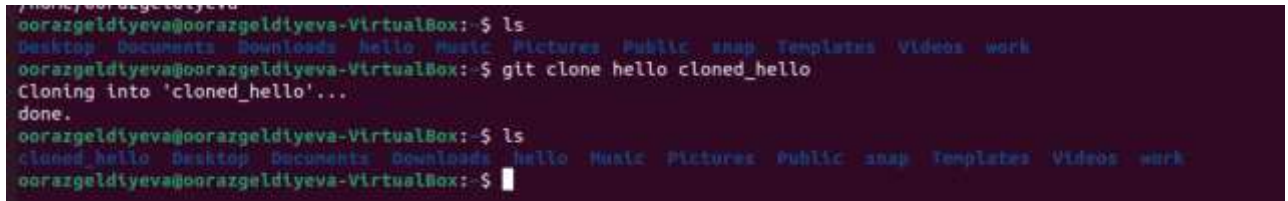
Сейчас мы находимся в рабочем каталоге.

В этот момент вы должны находиться в «рабочем» каталоге. Здесь должен быть единственный репозиторий под названием «hello».

Создайте клон репозитория hello

Создадим клон репозитория.

Выполните (см. рис. 113)

A terminal window with a dark background and light-colored text. The prompt is 'oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~\$'. The first command is 'ls', showing a list of directories: Desktop, Documents, Downloads, hello, Music, Pictures, Public, snap, Templates, Videos, work. The second command is 'git clone hello cloned_hello'. The output shows 'Cloning into 'cloned_hello'...' and 'done.'. The third command is 'ls', showing the updated directory list which now includes 'cloned_hello' at the beginning.

```
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~$ ls
Desktop Documents Downloads hello Music Pictures Public snap Templates Videos work
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~$ git clone hello cloned_hello
Cloning into 'cloned_hello'...
done.
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~$ ls
cloned_hello Desktop Documents Downloads hello Music Pictures Public snap Templates Videos work
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~$
```

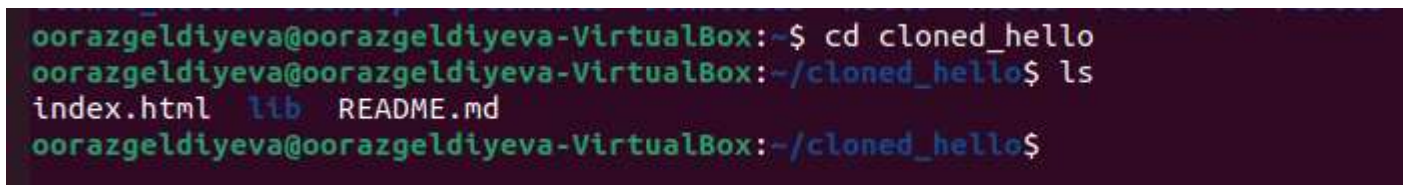
Рисунок 113. Создание клона

В вашем рабочем каталоге теперь должно быть два репозитория: оригинальный репозиторий «hello» и клонированный репозиторий «cloned_hello»

Просмотр клонированного репозитория

Давайте взглянем на клонированный репозиторий.

Выполните (см. рис. 114)

A terminal window with a dark background and light-colored text. The prompt is 'oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~\$'. The first command is 'cd cloned_hello'. The second command is 'ls', showing the contents of the directory: index.html, lib, README.md.

```
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~$ cd cloned_hello
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/cloned_hello$ ls
index.html  lib  README.md
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/cloned_hello$
```

Рисунок 114. Переход к каталог

Вы увидите список всех файлов на верхнем уровне оригинального репозитория README.md, index.html и lib.

Просмотрите историю репозитория

Выполните (см. рис. 115)


```
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~$ cd cloned_hello
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/cloned_hello$ ls
index.html  README.md
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/cloned_hello$ git log --all
commit bdd7905b0af72117525bb6826deb0e2c911e7f7a (HEAD -> master, origin/style, origin/master, origin/HEAD)
Author: oorazgeldiyeva <1032205431@pfur.ru>
Date: Sat Feb 11 22:19:40 2023 +0300

    Updated index.html

commit c71019e3b6df63ac921b7bf4f4954fca26d8befc
Author: oorazgeldiyeva <1032205431@pfur.ru>
Date: Sat Feb 11 22:16:42 2023 +0300

    Hello uses style.css

commit e55f2368043dd49f9fe66f254ba690f94e1c7f04
Author: oorazgeldiyeva <1032205431@pfur.ru>
Date: Sat Feb 11 22:13:42 2023 +0300

    Added css stylesheet

commit 2d77ab8f1ac4da13cb663b634df24e5811b85597
Author: oorazgeldiyeva <1032205431@pfur.ru>
Date: Sat Feb 11 22:24:48 2023 +0300

    added README

commit d83e5aea4cf9b4b1551373255d3aefce54050a3e
Author: oorazgeldiyeva <1032205431@pfur.ru>
Date: Sat Feb 11 21:59:40 2023 +0300

    Added index.html

commit 2d4cd5974a6c504cfa8d5ddc7c2162c909fa42fb
Author: oorazgeldiyeva <1032205431@pfur.ru>
Date: Sat Feb 11 21:57:22 2023 +0300
```

Рисунок 115. Просмотр истории репозитория

Вы увидите список всех коммитов в новом репозитории, и он должен (более или менее) совпадать с историей коммитов в оригинальном репозитории. Единственная разница должна быть в названиях веток.

Удаленные ветки

Вы увидите ветку master (HEAD) в списке истории. Вы также увидите ветки со странными именами (origin/master, origin/style и origin/HEAD).

Что такое origin?

Выполните (см. рис. 116)

```
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/cloned_hello$ git remote
origin
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/cloned_hello$
```

Рисунок 116. Удаленная ветка

Мы видим, что клонированный репозиторий знает об имени по умолчанию удаленного репозитория. Давайте посмотрим, можем ли мы получить более подробную информацию об имени по умолчанию:

Выполните (см. рис. 117)

```

oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/cloned_hello$ git remote
origin
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/cloned_hello$ git remote show origin
* remote origin
  Fetch URL: /home/oorazgeldiyeva/hello
  Push URL: /home/oorazgeldiyeva/hello
  HEAD branch: master
  Remote branches:
    master tracked
    style tracked
  Local branch configured for 'git pull':
    master merges with remote master
  Local ref configured for 'git push':
    master pushes to master (up to date)
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/cloned_hello$

```

Рисунок 117. Удаленная ветка

Удаленные репозитории обычно размещаются на отдельной машине, возможно, централизованном сервере. Однако, как мы видим здесь, они могут с тем же успехом указывать на репозиторий на той же машине. Нет ничего особенного в имени «origin», однако существует традиция использовать «origin» в качестве имени первичного централизованного репозитория (если таковой имеется).

Удаленные ветки

Давайте посмотрим на ветки, доступные в нашем клонированном репозитории.

Выполните (см. рис. 118)

```

oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/cloned_hello$ git branch
* master
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/cloned_hello$

```

Рисунок 118. Ветки

Как мы видим, в списке только ветка master. Где ветка style? Команда git branch выводит только список локальных веток по умолчанию.

Список удаленных веток

Для того, чтобы увидеть все ветки, попробуйте следующую команду (см. рис. 119)

```

oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/cloned_hello$ git branch -a
* master
  remotes/origin/HEAD -> origin/master
  remotes/origin/master
  remotes/origin/style
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/cloned_hello$

```

Рисунок 119. Удаленные ветки

Git выводит все коммиты в оригинальный репозиторий, но ветки в удаленном репозитории не рассматриваются как локальные. Если мы хотим собственную ветку style, мы должны сами ее создать. Через минуту вы увидите, как это делается.

Изменение оригинального репозитория

Внесите некоторые изменения в оригинальный репозиторий, чтобы затем попытаться извлечь и слить изменения из удаленной ветки в текущую

Внесите изменения в оригинальный репозиторий hello

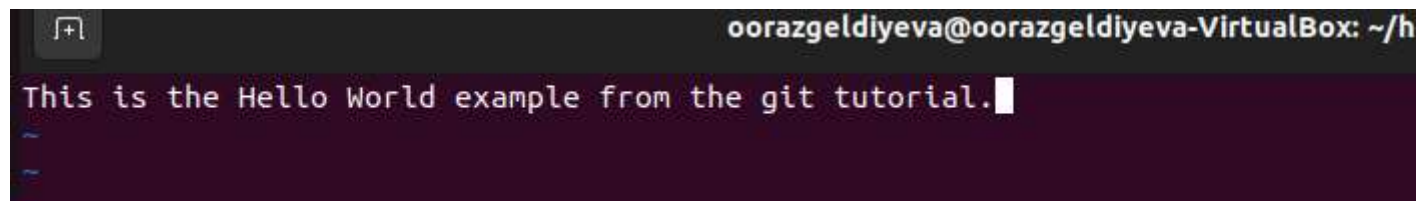
Выполните: (см. рис. 120)

```
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/cloned_hello$ cd ../hello
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ ls
index.html  lib  README.md
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$
```

Рисунок 120. Переход в каталог

Примечание: Сейчас мы находимся в репозитории hello. Внесите следующие изменения в файл README.md (см. рис. 121)

This is the Hello World example from the git tutorial.



```
oorazgeldiyeva@oorazgeldiyeva-VirtualBox: ~/h
This is the Hello World example from the git tutorial.
~
~
~
```

Рисунок 121. Изменение файла

Теперь добавьте это изменение и сделайте коммит

Выполните (см. рис. 122)

```
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ vim README.md
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git add README.md
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git commit -m "Changed README in original repo"
On branch master
nothing to commit, working tree clean
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$
```

Рисунок 122. Коммит

Теперь в оригинальном репозитории есть более поздние изменения, которых нет в клонированной версии. Далее мы извлечем и сольем эти изменения в клонированный репозиторий.

Извлечение изменений

Научиться извлекать изменения из удаленного репозитория.

Выполните (см. рис. 123)

```
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ cd ../cloned_hello
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/cloned_hello$ git fetch
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/cloned_hello$ git log --all
commit bdd7985b0af72117525bb6826deb0e2c911e7f7a (HEAD -> master, origin/style, origin/master, origin/HEAD)
Author: oorazgeldiyeva <1032205431@pfur.ru>
Date: Sat Feb 11 22:19:40 2023 +0300

    Updated index.html

commit c71019e3b6df05ac921b7bf4f4954fca26d8befc
Author: oorazgeldiyeva <1032205431@pfur.ru>
Date: Sat Feb 11 22:16:42 2023 +0300

    Hello uses style.css

commit e55f2368043dd49f9fe06f254ba690f94e1c7f04
Author: oorazgeldiyeva <1032205431@pfur.ru>
Date: Sat Feb 11 22:13:42 2023 +0300

    Added css stylesheet

commit 2d77ab8f1ac4da13cb663b634df24e5811b85597
Author: oorazgeldiyeva <1032205431@pfur.ru>
Date: Sat Feb 11 22:24:48 2023 +0300

    added README

commit d83e5aea4cf9b4b1551373255d3aefce54050a3e
Author: oorazgeldiyeva <1032205431@pfur.ru>
Date: Sat Feb 11 21:59:40 2023 +0300

    Added index.html

commit 2d4cd5974a6c504cfa8d5ddc7c2162c909fa42fb
Author: oorazgeldiyeva <1032205431@pfur.ru>
Date: Sat Feb 11 21:57:22 2023 +0300
```

Рисунок 123. Изменения

Сейчас мы находимся в репозитории cloned_hello.

На данный момент в репозитории есть все коммиты из оригинального репозитория, но они не интегрированы в локальные ветки клонированного репозитория.

В истории выше найдите коммит «Changed README in original repo». Обратите внимание, что коммит включает в себя коммиты «origin/master» и «origin/HEAD».

Теперь давайте посмотрим на коммит «Updated index.html». Вы увидите, что локальная ветка master указывает на этот коммит, а не на новый коммит, который мы только что извлекли.

Выводом является то, что команда git fetch будет извлекать новые коммиты из удаленного репозитория, но не будет сливать их с вашими наработками в локальных ветках.

Проверьте README.md

Мы можем продемонстрировать, что клонированный файл README.md не изменился.

Выполните (см. рис. 124)

```
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/cloned_hello$ cat README.md
This is the Hello World example from the git tutorial.
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/cloned_hello$
```

Рисунок 124. Содержимое файла

Слияние извлеченных изменений

Слейте извлеченные изменения в локальную ветку master

Выполните (см. рис. 125)

```
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/cloned_hello$ cat README.md
This is the Hello World example from the git tutorial.
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/cloned_hello$ git merge origin/master
Auto-merging README.md
CONFLICT (content): Merge conflict in README.md
Automatic merge failed; fix conflicts and then commit the result.
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/cloned_hello$
```

Рисунок 125. Слияние изменений

Еще раз проверьте файл README.md

Сейчас мы должны увидеть изменения.

Выполните (см. рис. 126)

```
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/cloned_hello$ cat README.md
<<<<<<< HEAD
This is the Hello World example from the git tutorial.
=====
This isthe Hello World example from the git tutorial.
>>>>>>> origin/master
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/cloned_hello$
```

Рисунок 126. Изменения

Хотя команда git fetch не сливает изменения, мы можем вручную слить изменения из удаленного репозитория.

Теперь давайте рассмотрим объединение fetch и merge в одну команду. Выполнение git pull эквивалентно двум следующим шагам git fetch и git merge origin/master

Добавление ветки наблюдения

Ветки, которые начинаются с `remotes/origin` являются ветками оригинального репозитория. Обратите внимание, что у вас больше нет ветки под названием `style`, но система контроля версий знает, что в оригинальном репозитории ветка `style` была.

Добавьте локальную ветку, которая отслеживает удаленную ветку

Выполните (см. рис. 127-128)

```
Already up to date.
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/cloned_hello$ git branch --track style origin/style
Branch 'style' set up to track remote branch 'style' from 'origin'.
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/cloned_hello$ git branch -a
* master
  style
remotes/origin/HEAD -> origin/master
remotes/origin/master
remotes/origin/style
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/cloned_hello$ git log --max-count=2
commit 5952c5bec1b372867136c5a7da0408ce0e6b0226 (HEAD -> master)
Merge: b11085e d667f56
Author: oorazgeldiyeva <1032205431@pfur.ru>
Date: Sat Feb 11 22:53:30 2023 +0300

    Changed README in original repo

commit d667f5638f1ff60c07cd442abee4c0fca00eda55 (origin/master, origin/HEAD)
Author: oorazgeldiyeva <1032205431@pfur.ru>
Date: Sat Feb 11 22:49:01 2023 +0300

    Changed in original repo
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/cloned_hello$
```

Рисунок 127. Добавление ветки

```
commit bdd7905b0af72117525bb6826deb6e2c911e7f7a (origin/style, style)
Author: oorazgeldiyeva <1032205431@pfur.ru>
Date: Sat Feb 11 22:19:40 2023 +0300

    Updated index.html
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/cloned_hello$
```

Рисунок 128. Добавление ветки

Теперь мы можем видеть ветку `style` в списке веток и логе.

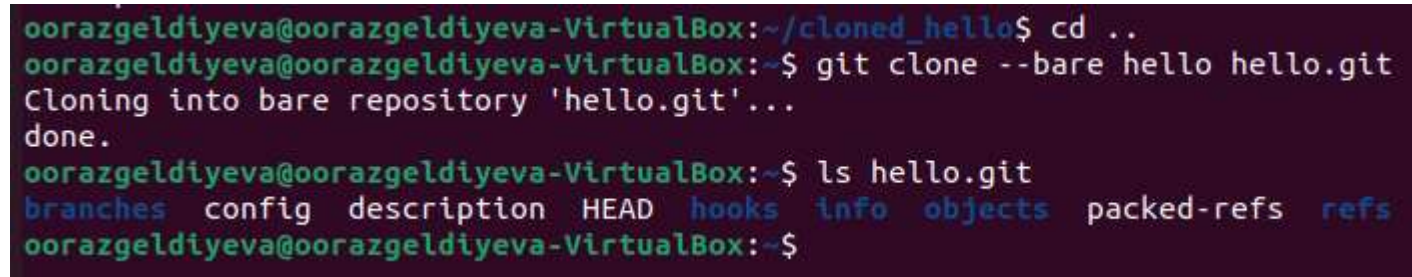
Чистые репозитории

Чистые репозитории (без рабочих каталогов) обычно используются для расшаривания. Обычный `git`-репозиторий подразумевает, что вы будете использовать его как рабочую директорию, поэтому вместе с файлами проекта в актуальной версии, `git` хранит все служебные, «чисто-репозиторийские» файлы в поддиректории `.git`. В удаленных репозиториях нет смысла хранить рабочие файлы на диске (как это делается в рабочих копиях), а все что им действительно нужно—это

дель- ты изменений и другие бинарные данные репозитория. Вот это и есть «чистый репозиторий».

Создайте чистый репозиторий

Выполните (см. рис. 129)



```
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/cloned_hello$ cd ..
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~$ git clone --bare hello hello.git
Cloning into bare repository 'hello.git'...
done.
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~$ ls hello.git
branches  config  description  HEAD  hooks  info  objects  packed-refs  refs
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~$
```

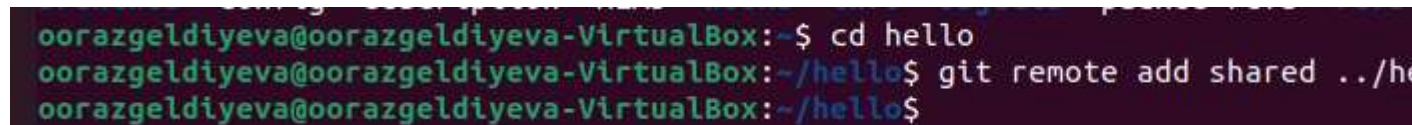
Рисунок 129. Создание репозитория

Сейчас мы находимся в рабочем каталоге

Как правило, репозитории, оканчивающиеся на .git являются чистыми репозиториями. Мы видим, что в репозитории hello.git нет рабочего каталога. По сути, это есть не что иное, как каталог.git нечистого репозитория.

Добавление удаленного репозитория

Давайте добавим репозиторий hello.git к нашему оригинальному репозиторию (см. рис. 130)

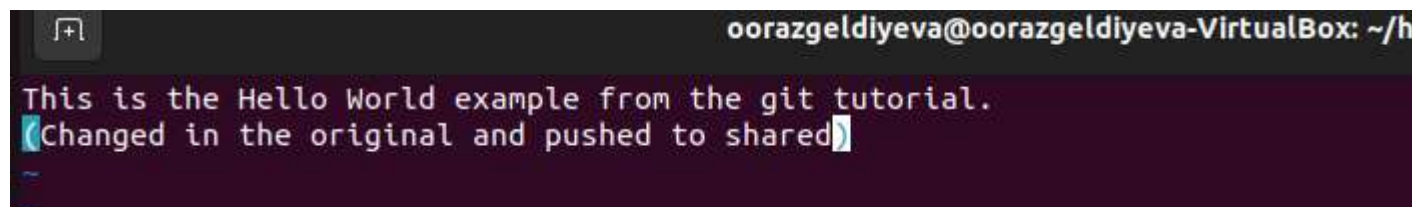


```
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~$ cd hello
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git remote add shared ../hello.git
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$
```

Рисунок 130. Добавление репозитория

Отправка изменений

Так как чистые репозитории, как правило, расшариваются на каком-нибудь сетевом сервере, нам необходимо отправить наши изменения в другие репозитории. Начнем с создания изменения для отправки. Отредактируйте файл README.md и сделайте коммит (см. рис. 131)



```
oorazgeldiyeva@oorazgeldiyeva-VirtualBox: ~/h
This is the Hello World example from the git tutorial.
[Changed in the original and pushed to shared]
```

Рисунок 131. Изменение файла

Выполните (см. рис. 132)

```
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git checkout master
M      README.md
Already on 'master'
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git add README
fatal: pathspec 'README' did not match any files
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git add README.md
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$ git commit -m "Added shared comment to readme"
[master 1efb253] Added shared comment to readme
 1 file changed, 2 insertions(+), 1 deletion(-)
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/hello$
```

Рисунок 132. Коммит

Теперь отправьте изменения в общий репозиторий. Выполните (см. рис. 133)

```
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/cloned_hello$ git remote add shared ../hello.git
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/cloned_hello$ git branch --track shared master
Branch 'shared' set up to track local branch 'master'.
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/cloned_hello$ git pull shared master
remote: Enumerating objects: 5, done.
remote: Counting objects: 100% (5/5), done.
remote: Compressing objects: 100% (3/3), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (3/3), 385 bytes | 385.00 KiB/s, done.
From ../hello
 * branch                master       -> FETCH_HEAD
 * [new branch]          master       -> shared/master
hint: You have divergent branches and need to specify how to reconcile them.
hint: You can do so by running one of the following commands sometime before
hint: your next pull:
hint:
hint:   git config pull.rebase false  # merge (the default strategy)
hint:   git config pull.rebase true   # rebase
hint:   git config pull.ff only       # fast-forward only
hint:
hint: You can replace "git config" with "git config --global" to set a default
hint: preference for all repositories. You can also pass --rebase, --no-rebase,
hint: or --ff-only on the command line to override the configured default per
hint: invocation.
fatal: Need to specify how to reconcile divergent branches.
oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/cloned_hello$ cat README.md
  This is the Hello World example from the git tutorial.

oorazgeldiyeva@oorazgeldiyeva-VirtualBox:~/cloned_hello$
```

Рисунок 133. Изменения

Общим называется репозиторий, получающий отправленные нами изменения.

38. Извлечение общих изменений

Научиться извлекать изменения из общего репозитория. Быстро переключитесь в клонированный репозиторий и извлеките изменения, только что отправленные в общий репозиторий

Вывод: в ходе лабораторной работы изучили применение средств контроля версий, научились работать с коммитами, тэгами, научились перемещаться между веткам.