

# **Математическое моделирование. Лабораторная Работа №1**

**Работа с git**

Барсегян Вардан Леонович

# Содержание

<b>1</b>	<b>Цель работы</b>	<b>7</b>
<b>2</b>	<b>Задание</b>	<b>8</b>
<b>3</b>	<b>Теоретическое введение</b>	<b>9</b>
<b>4</b>	<b>Выполнение лабораторной работы</b>	<b>10</b>
4.1	Подготовка . . . . .	10
4.2	Создание проекта и внесение изменений . . . . .	10
4.3	Индексация изменений . . . . .	11
4.4	История . . . . .	13
4.5	Отмена локальных изменений (до индексации) . . . . .	17
4.6	Отмена проиндексированных изменений (перед коммитом) . . . . .	18
4.7	Отмена коммитов . . . . .	19
4.8	Удаление коммитов из ветки . . . . .	20
4.9	Удаление тега oops . . . . .	21
4.10	Внесение изменений в коммиты . . . . .	22
4.11	Перемещение файлов . . . . .	23
4.12	Подробнее о структуре . . . . .	24
4.13	Git внутри: Каталог .git . . . . .	24
4.14	Работа непосредственно с объектами git . . . . .	25
4.15	Создание ветки . . . . .	27
4.16	Навигация по веткам . . . . .	27
4.17	Изменения в ветке master . . . . .	29
4.18	Слияние . . . . .	29
4.19	Создание конфликта . . . . .	30
4.20	Разрешение конфликтов . . . . .	31
4.21	Сброс ветки style . . . . .	31
4.22	Сброс ветки master . . . . .	32
4.23	Перебазирование . . . . .	33
4.24	Слияние в ветку master . . . . .	34
4.25	Клонирование репозитория и просмотр клонированного репозитория . . . . .	35
4.26	Что такое origin? Удаленные ветки . . . . .	36
4.27	Изменение оригинального репозитория . . . . .	37
4.28	Слияние извлеченных изменений . . . . .	38
4.29	Добавление ветки наблюдения . . . . .	39

4.30 Создание чистого репозитория, добавление удаленного репозитория	39
<b>5 Выводы</b>	<b>41</b>
<b>Список литературы</b>	<b>42</b>

# Список иллюстраций

4.1	Настройка git, инициализация репозитория . . . . .	10
4.2	Добавление файла в репозиторий и проверка состояния репозитория	11
4.3	hello.html . . . . .	11
4.4	Индексация и коммит изменений . . . . .	12
4.5	hello.html . . . . .	12
4.6	Обновление hello.html . . . . .	12
4.7	Проверка статуса и коммит . . . . .	13
4.8	Добавление второго изменения в индекс, проверка . . . . .	13
4.9	Список изменений . . . . .	14
4.10	man git-log . . . . .	14
4.11	Переход к первому коммиту . . . . .	15
4.12	Переход к последней версии и создание тега v1 . . . . .	16
4.13	Создание тэга предшествующей версии, просмотр . . . . .	16
4.14	hello.html . . . . .	17
4.15	Проверка состояния каталога до и после индексации изменений .	18
4.16	hello.html . . . . .	18
4.17	Отмена проиндексированных изменений перед коммитом . . . . .	19
4.18	hello.html . . . . .	19
4.19	Отмена коммита . . . . .	20
4.20	Проверка логов . . . . .	20
4.21	Сброс ветки и проверка логов . . . . .	21
4.22	Все коммиты . . . . .	21
4.23	Удаление тэга . . . . .	22
4.24	hello.html . . . . .	22
4.25	Создание коммита и его изменение . . . . .	23
4.26	Перемещение файла в папку lib . . . . .	23
4.27	index.html . . . . .	24
4.28	Просмотр каталога .git . . . . .	25
4.29	Просмотр последнего коммита . . . . .	25
4.30	Поиск по хэшу . . . . .	26
4.31	Исследование репозитория . . . . .	26
4.32	Обновление файлов, индексация и коммиты . . . . .	27
4.33	Проверка логов . . . . .	28
4.34	Проверка файла в разных ветках . . . . .	28
4.35	Создание README в ветке master, проверка дерева веток . . . . .	29
4.36	Слияние master с style . . . . .	30
4.37	Редактирование lib/hello.html в master . . . . .	30

4.38 Разрешение конфликта . . . . .	31
4.39 Логи изменений ветки style . . . . .	31
4.40 Сброс ветки style с помощью git reset . . . . .	32
4.41 Сброс ветки master . . . . .	33
4.42 Перебазирование ветки . . . . .	34
4.43 Слияние style с master . . . . .	35
4.44 Создание и просмотр клонированного репозитория . . . . .	36
4.45 git remote, просмотр веток . . . . .	37
4.46 Редактирование оригинального репозитория и извлечение изменений в клонированном . . . . .	38
4.47 Проверка файла до и после слияния изменений . . . . .	38
4.48 Добавление ветки наблюдения . . . . .	39
4.49 Создание чистого репозитория, добавление удаленного репозитория, извлечение общих изменений . . . . .	40

## **Список таблиц**

# 1 Цель работы

Знакомство с системой контроля версий git

## **2 Задание**



## 3 Теоретическое введение

Git (произносится «гит») — распределённая система управления версиями. Проект был создан Линусом Торвальдсом для управления разработкой ядра Linux, первая версия выпущена 7 апреля 2005 года; координатор — Дзюн Хамано.

Среди проектов, использующих Git, — ядро Linux, Swift, Android, Drupal, Cairo, GNU Core Utilities, Mesa, Wine, Chromium, Compiz Fusion, FlightGear, jQuery, PHP, NASM, MediaWiki, DokuWiki, Qt, ряд дистрибутивов Linux.

Программа является свободной и выпущена под лицензией GNU GPL версии 2. По умолчанию используется TCP-порт 9418.

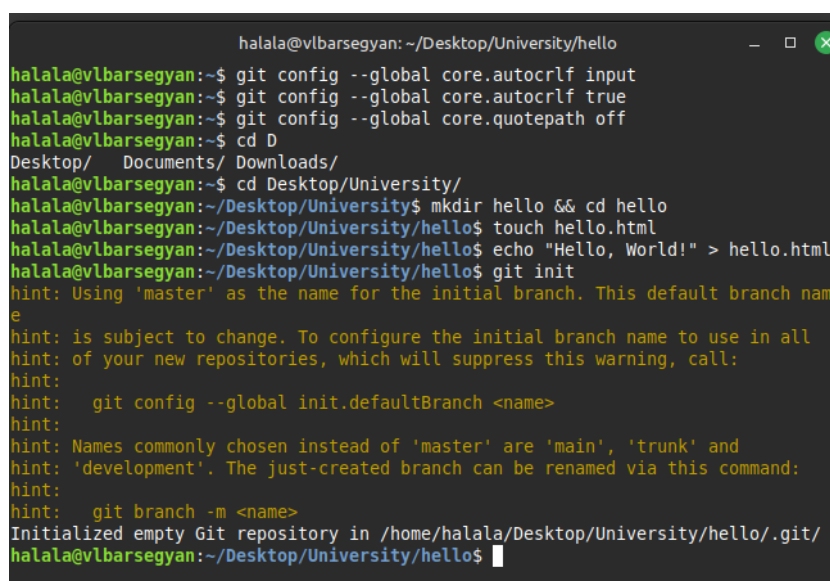
Разработка ядра Linux велась на проприетарной системе BitKeeper, которую автор — Ларри Маквой, сам разработчик Linux — предоставил проекту по бесплатной лицензии. Разработчики, высококлассные программисты, написали несколько утилит, и для одной Эндрю Триджелл произвёл реверс-инжиниринг формата передачи данных BitKeeper. В ответ Маквой обвинил разработчиков в нарушении соглашения и отозвал лицензию, и Торвальдс взялся за новую систему: ни одна из открытых систем не позволяла тысячам программистов кооперировать свои усилия (тот же конфликт привёл к написанию Mercurial). Идеология была проста: взять подход CVS и перевернуть с ног на голову, и заодно добавить надёжности.

Начальная разработка велась меньше чем неделю: 3 апреля 2005 года разработка началась, и уже 7 апреля код Git управлялся неготовой системой. 16 июня Linux был переведён на Git, а 25 июля Торвальдс отказался от обязанностей ведущего разработчика.[1].

## 4 Выполнение лабораторной работы

### 4.1 Подготовка

1. Настроим параметры установки окончаний строк и отображения unicode для устройства Linux. После, в папке *hello* создадим файл *hello.html* с содержимым, и инициализируем пустой git-репозиторий (рис. 4.1).



```
halala@vlbarsegyan: ~/Desktop/University/hello
halala@vlbarsegyan:~$ git config --global core.autocrlf input
halala@vlbarsegyan:~$ git config --global core.autocrlf true
halala@vlbarsegyan:~$ git config --global core.quotePath off
halala@vlbarsegyan:~$ cd D
Desktop/ Documents/ Downloads/
halala@vlbarsegyan:~$ cd Desktop/University/
halala@vlbarsegyan:~/Desktop/University$ mkdir hello && cd hello
halala@vlbarsegyan:~/Desktop/University/hello$ touch hello.html
halala@vlbarsegyan:~/Desktop/University/hello$ echo "Hello, World!" > hello.html
halala@vlbarsegyan:~/Desktop/University/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/halala/Desktop/University/hello/.git/
halala@vlbarsegyan:~/Desktop/University/hello$
```

Рис. 4.1: Настройка git, инициализация репозитория

### 4.2 Создание проекта и внесение изменений

2. Добавим файл в репозиторий.

*git add hello.html*

*git commit -m "Initial Commit"*

Проверим текущее состояние репозитория командой *git status* (рис. 4.2).

```
halala@vlbarsegyan: ~/Desktop/University/hello
halala@vlbarsegyan:~/Desktop/University/hello$ git add hello.html
warning: in the working copy of 'hello.html', LF will be replaced by CRLF the next time Git touches it
halala@vlbarsegyan:~/Desktop/University/hello$ git commit -m "Initial commit"
[master (root-commit) 6d52fa1] Initial commit
1 file changed, 1 insertion(+)
create mode 100644 hello.html
halala@vlbarsegyan:~/Desktop/University/hello$ git status
On branch master
nothing to commit, working tree clean
halala@vlbarsegyan:~/Desktop/University/hello$ echo "<h1>Hello, World!</h1>" > hello.html
bash: !: event not found
halala@vlbarsegyan:~/Desktop/University/hello$ nano hello.html
halala@vlbarsegyan:~/Desktop/University/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")
halala@vlbarsegyan:~/Desktop/University/hello$
```

Рис. 4.2: Добавление файла в репозиторий и проверка состояния репозитория

## 4.3 Индексация изменений

3. Текущее содержимое файла *hello.html* (рис. 4.3).

```
halala@vlbarsegyan: ~/Desktop/University/hello
/home/halala/Desktop/University/hello/.git/COMMIT_EDITMSG
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
#
```

Рис. 4.3: hello.html

4. Индексирую изменения командой *git add hello.html* и проверяю. Теперь изменения проиндексированы.

Коммичу изменения командой *git commit* и снова проверяю (рис. 4.4).

```
halala@vlbarsegyan: ~/Desktop/University/hello
halala@vlbarsegyan:~/Desktop/University/hello$ git add hello.html
warning: in the working copy of 'hello.html', LF will be replaced by CRLF the next time Git touches it
halala@vlbarsegyan:~/Desktop/University/hello$ git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    modified:   hello.html

halala@vlbarsegyan:~/Desktop/University/hello$ git commit
[master a3ef061] Added h1 tag
1 file changed, 1 insertion(+), 1 deletion(-)
halala@vlbarsegyan:~/Desktop/University/hello$ git status
On branch master
nothing to commit, working tree clean
halala@vlbarsegyan:~/Desktop/University/hello$
```

Рис. 4.4: Индексация и коммит изменений

5. Обновляю содержимое *hello.html* (рис. 4.5).

```
halala@vlbarsegyan: ~/Desktop/University/hello
GNU nano 7.2 hello.html
<html>
<body>
<h1>Hello, World!</h1>
</body>
</html>
```

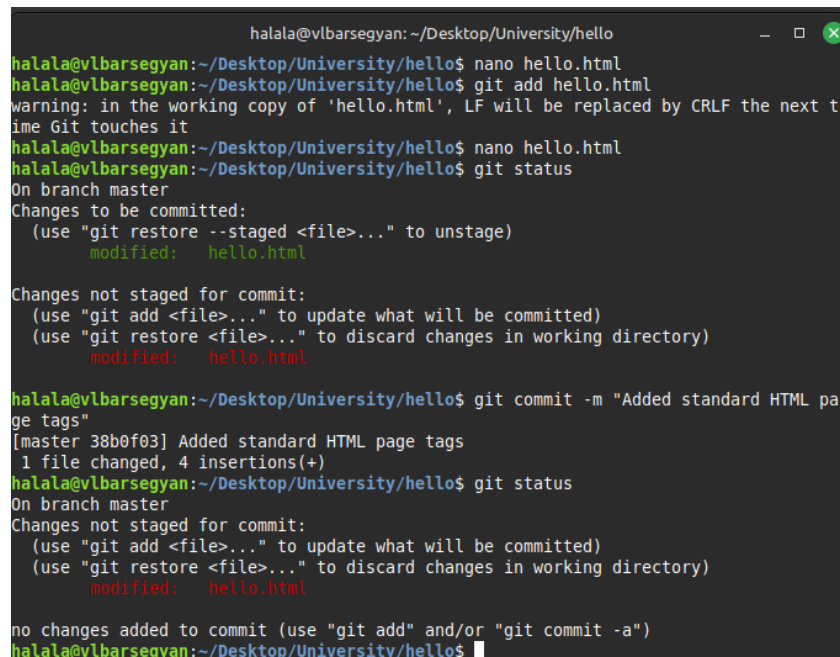
Рис. 4.5: hello.html

6. Добавляю это изменение в индекс командой *git add hello.html*. Снова обновляю *hello.html* (рис. 4.6).

```
halala@vlbarsegyan: ~/Desktop/University/hello
GNU nano 7.2 hello.html
<html>
<head>
</head>
<body>
<h1>Hello, World!</h1>
</body>
</html>
```

Рис. 4.6: Обновление hello.html

7. Проверяю текущие изменения. Добавляю последние изменения и коммичу.  
Снова проверяю статус (рис. 4.7).



```
halala@vlbarsegyan: ~/Desktop/University/hello
halala@vlbarsegyan:~/Desktop/University/hello$ nano hello.html
halala@vlbarsegyan:~/Desktop/University/hello$ git add hello.html
warning: in the working copy of 'hello.html', LF will be replaced by CRLF the next time Git touches it
halala@vlbarsegyan:~/Desktop/University/hello$ nano hello.html
halala@vlbarsegyan:~/Desktop/University/hello$ git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   hello.html

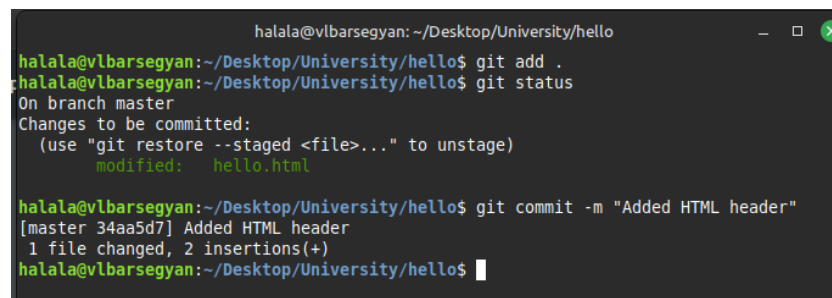
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   hello.html

halala@vlbarsegyan:~/Desktop/University/hello$ git commit -m "Added standard HTML page tags"
[master 38b0f03] Added standard HTML page tags
1 file changed, 4 insertions(+)
halala@vlbarsegyan:~/Desktop/University/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")
halala@vlbarsegyan:~/Desktop/University/hello$
```

Рис. 4.7: Проверка статуса и коммит

8. Добавляю второе изменение в индекс и проверяю (рис. 4.8).



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

halala@vlbarsegyan:~/Desktop/University/hello$ git commit -m "Added HTML header"
[master 34aa5d7] Added HTML header
1 file changed, 2 insertions(+)
halala@vlbarsegyan:~/Desktop/University/hello$
```

Рис. 4.8: Добавление второго изменения в индекс, проверка

## 4.4 История

9. Получим список изменений командой *git log*

Для однострочного формата введем *git log --pretty=oneline* (рис. 4.9).

```
halala@vlbarsegyan: ~/Desktop/University/hello
halala@vlbarsegyan:~/Desktop/University/hello$ git log
commit 34aa5d7b99d336527208a255ba802b226bbdfc2 (HEAD -> master)
Author: VARDamn <74242594+VARDamn@users.noreply.github.com>
Date: Sat Feb 22 19:09:33 2025 +0300

    Added HTML header

commit 38b0f0395belccd828be6d30265198e57b824644
Author: VARDamn <74242594+VARDamn@users.noreply.github.com>
Date: Sat Feb 22 19:08:53 2025 +0300

    Added standard HTML page tags

commit a3ef06196175da5961e58feba8c59016367c7123
Author: VARDamn <74242594+VARDamn@users.noreply.github.com>
Date: Sat Feb 22 19:06:57 2025 +0300

    Added h1 tag

commit 6d52fa18e52058cf7e63c9a1f3c538040c76e1c6
Author: VARDamn <74242594+VARDamn@users.noreply.github.com>
Date: Sat Feb 22 19:05:10 2025 +0300

    Initial commit
halala@vlbarsegyan:~/Desktop/University/hello$ git log --pretty=oneline
34aa5d7b99d336527208a255ba802b226bbdfc2 (HEAD -> master) Added HTML header
38b0f0395belccd828be6d30265198e57b824644 Added standard HTML page tags
a3ef06196175da5961e58feba8c59016367c7123 Added h1 tag
6d52fa18e52058cf7e63c9a1f3c538040c76e1c6 Initial commit
halala@vlbarsegyan:~/Desktop/University/hello$ git log --pretty=oneline --max-count=2
34aa5d7b99d336527208a255ba802b226bbdfc2 (HEAD -> master) Added HTML header
38b0f0395belccd828be6d30265198e57b824644 Added standard HTML page tags
halala@vlbarsegyan:~/Desktop/University/hello$
```

Рис. 4.9: Список изменений

10. Выведем справочную информацию командой *man git-log* (рис. 4.10).

```
halala@vlbarsegyan: ~/Desktop/University/hello
GIT-LOG(1)                               Git Manual                               GIT-LOG(1)
NAME
    git-log - Show commit logs

SYNOPSIS
    git log [<options>] [<revision-range>] [--] <path>...

DESCRIPTION
    Shows the commit logs.

    List commits that are reachable by following the parent links from the given
    commit(s), but exclude commits that are reachable from the one(s) given with
    a ^ in front of them. The output is given in reverse chronological order by
    default.

    You can think of this as a set operation. Commits reachable from any of the
    commits given on the command line form a set, and then commits reachable
    from any of the ones given with ^ in front are subtracted from that set. The
    remaining commits are what comes out in the command's output. Various other
    options and paths parameters can be used to further limit the result.

    Thus, the following command:

        $ git log foo bar ^baz

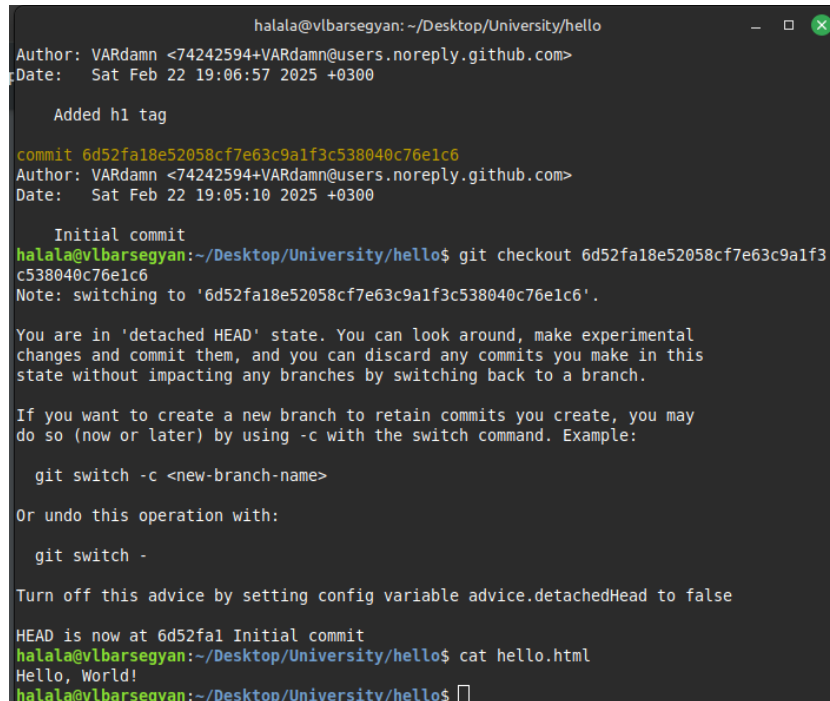
    means "list all the commits which are reachable from foo or bar, but not
    from baz".

    A special notation "<commit1>..<commit2>" can be used as a short-hand for
    "^<commit1> <commit2>". For example, either of the following may be used
    interchangeably:

    Manual page git-log(1) line 1 (press h for help or q to quit)
```

Рис. 4.10: man git-log

11. Ввожу команду *git checkout*, где хэш - хэш первого коммита. Также просматриваю содержимое файла - оно равняется содержимому первого коммита (рис. 4.11).

A terminal window titled 'halala@vlbarsegyan: ~/Desktop/University/hello'. It shows the output of a 'git checkout' command. The output includes: 'Added h1 tag', a commit hash '6d52fa18e52058cf7e63c9a1f3c538040c76e1c6', author information 'Author: VARDamn <74242594+VARDamn@users.noreply.github.com>', date 'Sat Feb 22 19:06:57 2025 +0300', 'Initial commit', the command 'halala@vlbarsegyan:~/Desktop/University/hello\$ git checkout 6d52fa18e52058cf7e63c9a1f3c538040c76e1c6', a note 'Note: switching to '6d52fa18e52058cf7e63c9a1f3c538040c76e1c6'.', instructions about the 'detached HEAD' state, a warning about creating new branches, the command 'git switch -c <new-branch-name>', the undo command 'git switch -', advice to turn off the detached head warning, and the final state 'HEAD is now at 6d52fal Initial commit'. At the bottom, it shows the command 'halala@vlbarsegyan:~/Desktop/University/hello\$ cat hello.html' and its output 'Hello, World!'.

```
halala@vlbarsegyan: ~/Desktop/University/hello
Author: VARDamn <74242594+VARDamn@users.noreply.github.com>
Date:   Sat Feb 22 19:06:57 2025 +0300

    Added h1 tag

commit 6d52fa18e52058cf7e63c9a1f3c538040c76e1c6
Author: VARDamn <74242594+VARDamn@users.noreply.github.com>
Date:   Sat Feb 22 19:05:10 2025 +0300

    Initial commit
halala@vlbarsegyan:~/Desktop/University/hello$ git checkout 6d52fa18e52058cf7e63c9a1f3c538040c76e1c6
Note: switching to '6d52fa18e52058cf7e63c9a1f3c538040c76e1c6'.

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 6d52fal Initial commit
halala@vlbarsegyan:~/Desktop/University/hello$ cat hello.html
Hello, World!
halala@vlbarsegyan:~/Desktop/University/hello$
```

Рис. 4.11: Переход к первому коммиту

12. Перехожу к последней версии ветки командой *git checkout master* и просматриваю содержимое файла - теперь оно актуальное. Создаю тег первой версии командой *git tag v1*. Перехожу к версии, предшествующей v1, командой *git checkout v1^*. (рис. 4.12).

```
halala@vlbarsegyan: ~/Desktop/University/hello
halala@vlbarsegyan:~/Desktop/University/hello$ git checkout master
Previous HEAD position was 6d52fa1 Initial commit
Switched to branch 'master'
halala@vlbarsegyan:~/Desktop/University/hello$ cat hello.html
<html>
<head>
</head>
<body>
<h1>Hello, World!</h1>
</body>
</html>
halala@vlbarsegyan:~/Desktop/University/hello$ git tag v1
halala@vlbarsegyan:~/Desktop/University/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 38b0f03 Added standard HTML page tags
halala@vlbarsegyan:~/Desktop/University/hello$
```

Рис. 4.12: Переход к последней версии и создание тега v1

13. Создаю для предшествующей версии тэг *v1-beta*. Перехожу к версии v1 и просматриваю все доступные тэги. Также просматриваю тэги в логе командой *git log master --all* (рис. 4.13).

```
halala@vlbarsegyan: ~/Desktop/University/hello
halala@vlbarsegyan:~/Desktop/University/hello$ cat hello.html
<html>
<body>
<h1>Hello, World!</h1>
</body>
</html>
halala@vlbarsegyan:~/Desktop/University/hello$ git tag v1-beta
halala@vlbarsegyan:~/Desktop/University/hello$ git checkout v1
Previous HEAD position was 38b0f03 Added standard HTML page tags
HEAD is now at 34aa5d7 Added HTML header
halala@vlbarsegyan:~/Desktop/University/hello$ git checkout v1-beta
Previous HEAD position was 34aa5d7 Added HTML header
HEAD is now at 38b0f03 Added standard HTML page tags
halala@vlbarsegyan:~/Desktop/University/hello$ git tag
v1
v1-beta
halala@vlbarsegyan:~/Desktop/University/hello$ git log master --all
commit 34aa5d7b99d336527208a255ba802b226bbdfdc2 (tag: v1, master)
Author: VARDamn <74242594+VARDamn@users.noreply.github.com>
Date: Sat Feb 22 19:09:33 2025 +0300

    Added HTML header

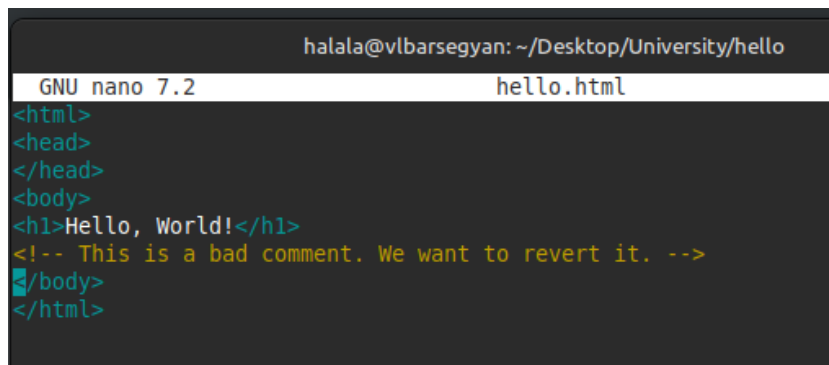
commit 38b0f0395be1ccd828be6d30265198e57b824644 (HEAD, tag: v1-beta)
```

Рис. 4.13: Создание тэга предшествующей версии, просмотр



## 4.5 Отмена локальных изменений (до индексации)

14. Обновляю содержимое hello.html (рис. 4.14).



```
halala@vlbarsegyan: ~/Desktop/University/hello
GNU nano 7.2 hello.html
<html>
<head>
</head>
<body>
<h1>Hello, World!</h1>
<!-- This is a bad comment. We want to revert it. -->
</body>
</html>
```

Рис. 4.14: hello.html

15. Проверяю состояние рабочего каталога (hello.html был изменен, но еще не проиндексирован).

Командой *git checkout* переключаю версии файла hello.html в репозитории

*git checkout hello.html*

*git status*

*cat hello.html*

Команда *git status* показывает нам, что не было произведено никаких изменений, не зафиксированных в рабочем каталоге. (рис. 4.15).

```
halala@vlbarsegyan: ~/Desktop/University/hello
Switched to branch 'master'
halala@vlbarsegyan:~/Desktop/University/hello$ nano hello.html
halala@vlbarsegyan:~/Desktop/University/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")
halala@vlbarsegyan:~/Desktop/University/hello$ git checkout hello.html
Updated 1 path from the index
halala@vlbarsegyan:~/Desktop/University/hello$ git status
On branch master
nothing to commit, working tree clean
halala@vlbarsegyan:~/Desktop/University/hello$ cat hello.html
<html>
<head>
</head>
<body>
<h1>Hello, World!</h1>
</body>
</html>
halala@vlbarsegyan:~/Desktop/University/hello$
```

Рис. 4.15: Проверка состояния каталога до и после индексации изменений

## 4.6 Отмена проиндексированных изменений (перед коммитом)

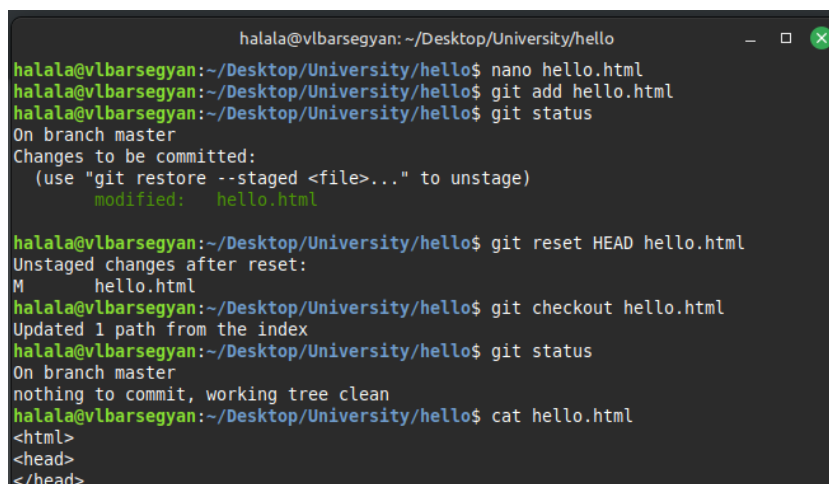
16. Изменяю содержимое hello.html (рис. 4.16).

```
halala@vlbarsegyan: ~/Desktop/University/hello
GNU nano 7.2 hello.html
<html>
<head>
<!-- This is an unwanted but staged comment -->
</head>
<body>
<h1>Hello, World!</h1>
</body>
</html>
```

Рис. 4.16: hello.html

17. Индексирую изменения командой *git add hello.html*. Проверяю состояние изменения, после чего сбрасываю буферную зону командой *git reset HEAD hello.html*. Переключаюсь на версию коммита командой *git checkout hello.html*

и снова проверяю статус. Проверяю содержимой файла - изменений теперь нет. (рис. 4.17).



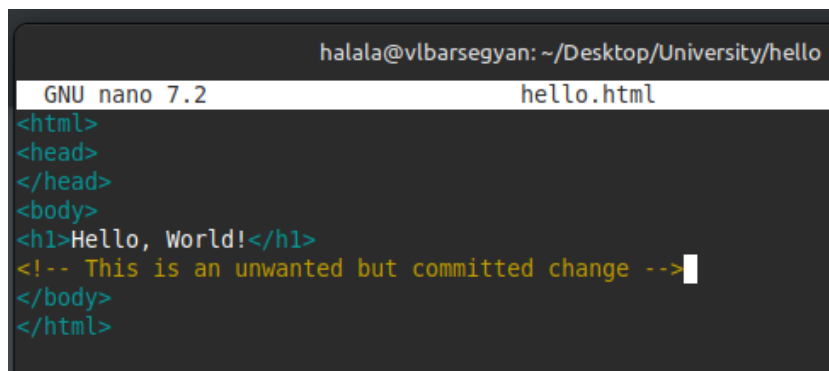
```
halala@vlbarsegyan: ~/Desktop/University/hello
halala@vlbarsegyan:~/Desktop/University/hello$ nano hello.html
halala@vlbarsegyan:~/Desktop/University/hello$ git add hello.html
halala@vlbarsegyan:~/Desktop/University/hello$ git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   hello.html

halala@vlbarsegyan:~/Desktop/University/hello$ git reset HEAD hello.html
Unstaged changes after reset:
M       hello.html
halala@vlbarsegyan:~/Desktop/University/hello$ git checkout hello.html
Updated 1 path from the index
halala@vlbarsegyan:~/Desktop/University/hello$ git status
On branch master
nothing to commit, working tree clean
halala@vlbarsegyan:~/Desktop/University/hello$ cat hello.html
<html>
<head>
</head>
```

Рис. 4.17: Отмена проиндексированных изменений перед коммитом

## 4.7 Отмена коммитов

18. Изменяю содержимое hello.html (рис. 4.18).



```
halala@vlbarsegyan: ~/Desktop/University/hello
GNU nano 7.2                                hello.html
<html>
<head>
</head>
<body>
<h1>Hello, World!</h1>
<!-- This is an unwanted but committed change -->
</body>
</html>
```

Рис. 4.18: hello.html

19. Индексирую изменения и коммичу их командой *git commit -m "Oops, we didn't want this commit"*. Отменяю коммит командой *git revert HEAD* - в результате получаю такой коммит (рис. 4.19).

```
halala@vlbarsegyan: ~/Desktop/University/hello
/home/halala/Desktop/University/hello/.git/COMMIT_EDITMSG
Revert "Oops, we didn't want this commit"

This reverts commit 223917838bb360f5752b3c50002206d05b43319c.

# 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
#
```

Рис. 4.19: Отмена коммита

20. Проверяю логи (рис. 4.20).

```
halala@vlbarsegyan: ~/Desktop/University/hello
halala@vlbarsegyan:~/Desktop/University/hello$ git add hello.html
halala@vlbarsegyan:~/Desktop/University/hello$ git commit -m "Oops, we didn't want this commit"
[master 2239178] Oops, we didn't want this commit
1 file changed, 1 insertion(+)
halala@vlbarsegyan:~/Desktop/University/hello$ git revert HEAD
[master 7a08da4] Revert "Oops, we didn't want this commit"
1 file changed, 1 deletion(-)
halala@vlbarsegyan:~/Desktop/University/hello$ git log
commit 7a08da416e5d00bfe692a4ac3edd4619ca8557c4 (HEAD -> master)
Author: VARDamn <74242594+VARDamn@users.noreply.github.com>
Date: Sat Feb 22 19:17:29 2025 +0300

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

    This reverts commit 223917838bb360f5752b3c50002206d05b43319c.

commit 223917838bb360f5752b3c50002206d05b43319c
Author: VARDamn <74242594+VARDamn@users.noreply.github.com>
Date: Sat Feb 22 19:17:24 2025 +0300

    Oops, we didn't want this commit
```

Рис. 4.20: Проверка логов

## 4.8 Удаление коммитов из ветки

21. Отмечаю последний коммит тэгом командой *git tag oops*. Сбрасываю ветку до точки v1 командой *git reset --hard v1*. Проверяю логи - информация о двух последних коммитах пропала (рис. 4.21).

```
halala@vlbarsegyan: ~/Desktop/University/hello
halala@vlbarsegyan:~/Desktop/University/hello$ git tag oops
halala@vlbarsegyan:~/Desktop/University/hello$ git reset --hard v1
HEAD is now at 34aa5d7 Added HTML header
halala@vlbarsegyan:~/Desktop/University/hello$ git log
commit 34aa5d7b99d336527208a255ba802b226bbfdcf2 (HEAD -> master, tag: v1)
Author: VARdamn <74242594+VARdamn@users.noreply.github.com>
Date: Sat Feb 22 19:09:33 2025 +0300

    Added HTML header

commit 38b0f0395be1ccd828be6d30265198e57b824644 (tag: v1-beta)
Author: VARdamn <74242594+VARdamn@users.noreply.github.com>
Date: Sat Feb 22 19:08:53 2025 +0300

    Added standard HTML page tags

commit a3ef06196175da5961e58feba8c59016367c7123
Author: VARdamn <74242594+VARdamn@users.noreply.github.com>
Date: Sat Feb 22 19:06:57 2025 +0300

    Added h1 tag
```

Рис. 4.21: Сброс ветки и проверка логов

22. Просматриваю все коммиты командой `git log --all` - здесь есть и последние два коммита (рис. 4.22).

```
halala@vlbarsegyan:~/Desktop/University/hello$ git log --all
commit 7a08da416e5d00bfe692a4ac3edd4619ca8557c4 (tag: oops)
Author: VARdamn <74242594+VARdamn@users.noreply.github.com>
Date: Sat Feb 22 19:17:29 2025 +0300

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

    This reverts commit 223917838bb360f5752b3c50002206d05b43319c.

commit 223917838bb360f5752b3c50002206d05b43319c
Author: VARdamn <74242594+VARdamn@users.noreply.github.com>
Date: Sat Feb 22 19:17:24 2025 +0300

    Oops, we didn't want this commit

commit 34aa5d7b99d336527208a255ba802b226bbfdcf2 (HEAD -> master, tag: v1)
Author: VARdamn <74242594+VARdamn@users.noreply.github.com>
Date: Sat Feb 22 19:09:33 2025 +0300

    Added HTML header
```

Рис. 4.22: Все коммиты

## 4.9 Удаление тега оорс

23. Удаляю тэг оорс командой `git tag -d oops` и проверяю (рис. 4.23).

```
halala@vlbarsegyan: ~/Desktop/University/hello
halala@vlbarsegyan:~/Desktop/University/hello$ git tag -d oops
Deleted tag 'oops' (was 7a08da4)
halala@vlbarsegyan:~/Desktop/University/hello$ git log --all
commit 34aa5d7b99d336527208a255ba802b226bbfdcf2 (HEAD -> master, tag: v1)
Author: VARdamn <74242594+VARdamn@users.noreply.github.com>
Date: Sat Feb 22 19:09:33 2025 +0300

    Added HTML header

commit 38b0f0395be1ccd828be6d30265198e57b824644 (tag: v1-beta)
Author: VARdamn <74242594+VARdamn@users.noreply.github.com>
Date: Sat Feb 22 19:08:53 2025 +0300

    Added standard HTML page tags

commit a3ef06196175da5961e58feba8c59016367c7123
Author: VARdamn <74242594+VARdamn@users.noreply.github.com>
Date: Sat Feb 22 19:06:53 2025 +0300
```

Рис. 4.23: Удаление тэга

## 4.10 Внесение изменений в коммиты

24. Изменяю содержимое hello.html (рис. 4.24).

```
halala@vlbarsegyan: ~/Desktop/University/hello
GNU nano 7.2 hello.html
<!-- Author: Dmitry S. Kulyabov -->
<html>
<head>
</head>
<body>
<h1>Hello, World!</h1>
</body>
</html>
```

Рис. 4.24: hello.html

25. Индексирую изменения и коммичу их. После, снова редактирую файл, добавляя почту, и коммичу изменения командой *git commit -amend -m "Add an author/email comment"*, т.е. изменяя последний коммит. Проверяю логи и вижу один новый коммит (рис. 4.25).

```
halala@vlbarsegyan: ~/Desktop/University/hello
halala@vlbarsegyan:~/Desktop/University/hello$ git add hello.html
halala@vlbarsegyan:~/Desktop/University/hello$ git commit -m "Add an author commit"
[master c5470bb] Add an author commit
1 file changed, 1 insertion(+)
halala@vlbarsegyan:~/Desktop/University/hello$ git add hello.html
halala@vlbarsegyan:~/Desktop/University/hello$ nano hello.html
halala@vlbarsegyan:~/Desktop/University/hello$ git add hello.html
halala@vlbarsegyan:~/Desktop/University/hello$ git commit --amend -m "Add an author/email comment"
[master b6f3165] Add an author/email comment
Date: Sat Feb 22 19:20:29 2025 +0300
1 file changed, 1 insertion(+)
halala@vlbarsegyan:~/Desktop/University/hello$ git log
commit b6f31650ebc5cc5eb5001acbdcc3fd50f6d782e0c (HEAD -> master)
Author: VARdamn <74242594+VARdamn@users.noreply.github.com>
Date: Sat Feb 22 19:20:29 2025 +0300

    Add an author/email comment

commit 34aa5d7b99d336527208a255ba802b226bbdfdc2 (tag: v1)
Author: VARdamn <74242594+VARdamn@users.noreply.github.com>
Date: Sat Feb 22 19:09:33 2025 +0300

    Added HTML header
```

Рис. 4.25: Создание коммита и его изменение

## 4.11 Перемещение файлов

26. Создаю папку `lib` и переношу страницу в каталог `lib` командой `git mv`. Проверяю изменения командой `git status`. Коммичу изменения (рис. 4.26).

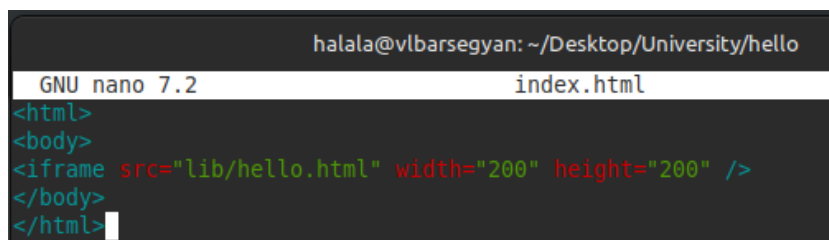
```
halala@vlbarsegyan: ~/Desktop/University/hello
halala@vlbarsegyan:~/Desktop/University/hello$ mkdir lib
halala@vlbarsegyan:~/Desktop/University/hello$ git mv hello.html lib
halala@vlbarsegyan:~/Desktop/University/hello$ git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        renamed:    hello.html -> lib/hello.html

halala@vlbarsegyan:~/Desktop/University/hello$ git commit -m "Moved hello.html to lib"
[master 4656c98] Moved hello.html to lib
1 file changed, 0 insertions(+), 0 deletions(-)
rename hello.html => lib/hello.html (100%)
halala@vlbarsegyan:~/Desktop/University/hello$
```

Рис. 4.26: Перемещение файла в папку `lib`

## 4.12 Подробнее о структуре

27. Создаю в корне файл *index.html* и заполняю его (рис. 4.27). Коммичу изменения



```
halala@vlbarsegyan: ~/Desktop/University/hello
GNU nano 7.2 index.html
<html>
<body>
<iframe src="lib/hello.html" width="200" height="200" />
</body>
</html>
```

Рис. 4.27: index.html

## 4.13 Git внутри: Каталог .git

28. Просматриваю каталог со всей информацией о git командой *ls -C .git*. Выполняю команду *ls -C .git/objects*. Также просматриваю один из каталогов с именем из 2 букв. Просматриваю файл конфигурации командой *cat .git/config*. Далее выполняю
- ls .git/refs*
  - ls .git/refs/heads*
  - ls .git/refs/tags*
  - cat .git/refs/tags/v1* (рис. 4.28).



```
halala@vlbarsegyan: ~/Desktop/University/hello
halala@vlbarsegyan:~/Desktop/University/hello$ ls -C .git
branches      config      HEAD      index      logs      ORIG HEAD      refs
COMMIT_EDITMSG  description hooks      info      objects  packed-refs
halala@vlbarsegyan:~/Desktop/University/hello$ ls -C .git/objects/
00 10 2e 38 4b 52 62 6d 88 a3 b6 c3 c6 f4 info
09 22 34 46 4f 5e 68 7a 8a ac bd c5 cb f7 pack
halala@vlbarsegyan:~/Desktop/University/hello$ ls -C .git/objects/8a
b686eafeb1f44702738c8b0f24f2567c36da6d
halala@vlbarsegyan:~/Desktop/University/hello$ cat .git/config
[core]
    repositoryformatversion = 0
    filemode = true
    bare = false
    logallrefupdates = true
halala@vlbarsegyan:~/Desktop/University/hello$ ls .git/refs
heads tags
halala@vlbarsegyan:~/Desktop/University/hello$ ls .git/refs/he
ls: cannot access '.git/refs/he': No such file or directory
halala@vlbarsegyan:~/Desktop/University/hello$ ls .git/refs/heads/
master
halala@vlbarsegyan:~/Desktop/University/hello$ ls .git/refs/tags/
v1 v1-beta
halala@vlbarsegyan:~/Desktop/University/hello$ cat .git/refs/tags/v1
34aa5d7b99d336527208a255ba802b226bbdfdc2
halala@vlbarsegyan:~/Desktop/University/hello$
```

Рис. 4.28: Просмотр каталога .git

## 4.14 Работа непосредственно с объектами git

29. Выполняю команду `cat .git/HEAD` - содержится ссылка на ветку master. Про-  
сматриваю последний коммит командой `git log --max-count=1`. Также вывожу  
последний коммит с помощью хэша командой `git cat-file` (рис. 4.29).

```
halala@vlbarsegyan: ~/Desktop/University/hello
halala@vlbarsegyan:~/Desktop/University/hello$ cat .git/HEAD
ref: refs/heads/master
halala@vlbarsegyan:~/Desktop/University/hello$ git log --max-count=1
commit ac7af6f87c376835b2828381aa16199a729094a8 (HEAD -> master)
Author: VARDamn <74242594+VARDamn@users.noreply.github.com>
Date: Sat Feb 22 19:23:34 2025 +0300

    Added index.html.
halala@vlbarsegyan:~/Desktop/University/hello$ git cat-file -t ac7af6f87c376835b2
828381aa16199a729094a8
commit
halala@vlbarsegyan:~/Desktop/University/hello$ git cat-file -p ac7af6f87c376835b2
828381aa16199a729094a8
tree 4bc2c3bad04e01ec8776fd838421d0ec5a97c274
parent 4656c98420063f41e7ebb6e0375c2aedb2ddd18a
author VARDamn <74242594+VARDamn@users.noreply.github.com> 1740241414 +0300
committer VARDamn <74242594+VARDamn@users.noreply.github.com> 1740241414 +0300

Added index.html.
halala@vlbarsegyan:~/Desktop/University/hello$
```

Рис. 4.29: Просмотр последнего коммита

30. Произвожу поиск дерева командой `git cat-file`, указав хеш дерева. Аналогично вывожу каталог `lib` и файл `hello.html` (рис. 4.30).

```
halala@vlbarsegyan: ~/Desktop/University/hello
halala@vlbarsegyan:~/Desktop/University/hello$ git cat-file -p 4bc2c3bad04e01ec87
76fd838421d0ec5a97c274
100644 blob 68583eacd328506a31568f0df6f45ebb874238b5 index.html
040000 tree 105ece18ef1ac46bbaa848ab52836efcfde3e9db lib
halala@vlbarsegyan:~/Desktop/University/hello$ git cat-file -p 105ece18ef1ac46bba
a848ab52836efcfde3e9db
100644 blob c6b5be25f013151c362dec53ac3fb1a26c05f12a hello.html
halala@vlbarsegyan:~/Desktop/University/hello$ git cat-file -p c6b5be25f013151c36
2dec53ac3fb1a26c05f12a
<!-- Author: Dmitry S. Kulyabov (kulyabov-ds@rudn.ru) -->
<html>
<head>
</head>
<body>
<h1>Hello, World!</h1>
</body>
</html>
halala@vlbarsegyan:~/Desktop/University/hello$
```

Рис. 4.30: Поиск по хэшу

31. Исследую репозиторий самостоятельно: перехожу к последующему коммиту по хэшу родителя (рис. 4.31).

```
halala@vlbarsegyan: ~/Desktop/University/hello
Added HTML header
halala@vlbarsegyan:~/Desktop/University/hello$ git cat-file -p 38b0f0395be1ccd828be6d
30265198e57b824644
tree 885ab38edd6b66dee6559a73aeb58a92e6b75360
parent a3ef06196175da5961e58feba8c59016367c7123
author VARDamn <74242594+VARDamn@users.noreply.github.com> 1740240533 +0300
committer VARDamn <74242594+VARDamn@users.noreply.github.com> 1740240533 +0300

Added standard HTML page tags
halala@vlbarsegyan:~/Desktop/University/hello$ git cat-file -p a3ef06196175da5961e58f
eba8c59016367c7123
tree 629377ed8eb96afede9fd6cb7b48b1e4a1d3dc56
parent 6d52fa18e52058cf7e63c9a1f3c538040c76e1c6
author VARDamn <74242594+VARDamn@users.noreply.github.com> 1740240417 +0300
committer VARDamn <74242594+VARDamn@users.noreply.github.com> 1740240417 +0300

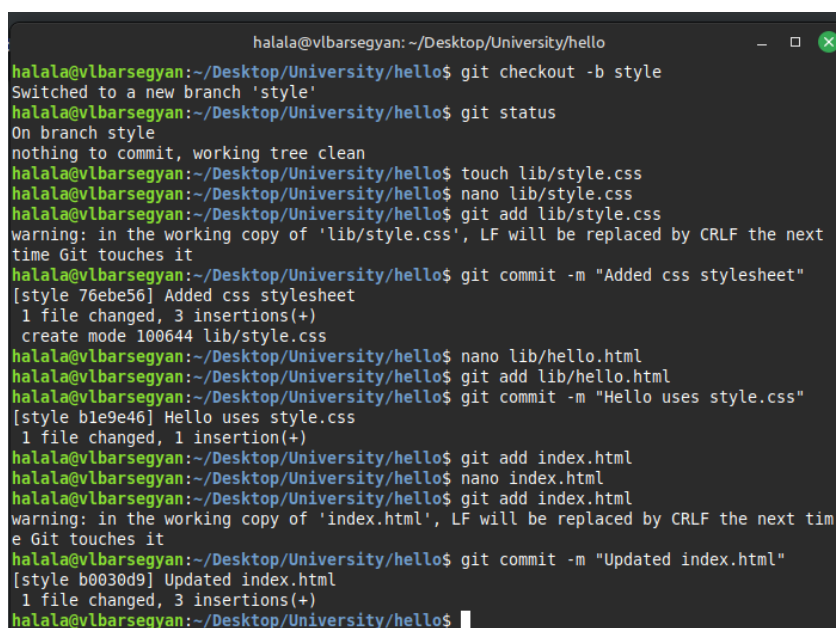
Added h1 tag
halala@vlbarsegyan:~/Desktop/University/hello$ git cat-file -p 6d52fa18e52058cf7e63c9
a1f3c538040c76e1c6
tree 523cd318fc3b74ad3852cd34a74720ad6f7dd422
author VARDamn <74242594+VARDamn@users.noreply.github.com> 1740240310 +0300
committer VARDamn <74242594+VARDamn@users.noreply.github.com> 1740240310 +0300

Initial commit
halala@vlbarsegyan:~/Desktop/University/hello$
```

Рис. 4.31: Исследование репозитория

## 4.15 Создание ветки

32. Создаю ветку командой `git checkout -b style` и проверяю ее статус. Создаю файл `style.css` в папке `lib`, заполняю его, индексирую и коммичу. Обновляю файл `lib/hello.html`, добавляя в него ссылку на `css`-файл, и также индексирую и коммичу изменения. Обновляю файл `index.html`, индексирую и коммичу изменения (рис. 4.32).



```
halala@vlbarsegyan: ~/Desktop/University/hello
halala@vlbarsegyan:~/Desktop/University/hello$ git checkout -b style
Switched to a new branch 'style'
halala@vlbarsegyan:~/Desktop/University/hello$ git status
On branch style
nothing to commit, working tree clean
halala@vlbarsegyan:~/Desktop/University/hello$ touch lib/style.css
halala@vlbarsegyan:~/Desktop/University/hello$ nano lib/style.css
halala@vlbarsegyan:~/Desktop/University/hello$ git add lib/style.css
warning: in the working copy of 'lib/style.css', LF will be replaced by CRLF the next
time Git touches it
halala@vlbarsegyan:~/Desktop/University/hello$ git commit -m "Added css stylesheet"
[style 76be56] Added css stylesheet
1 file changed, 3 insertions(+)
create mode 100644 lib/style.css
halala@vlbarsegyan:~/Desktop/University/hello$ nano lib/hello.html
halala@vlbarsegyan:~/Desktop/University/hello$ git add lib/hello.html
halala@vlbarsegyan:~/Desktop/University/hello$ git commit -m "Hello uses style.css"
[style ble9e46] Hello uses style.css
1 file changed, 1 insertion(+)
halala@vlbarsegyan:~/Desktop/University/hello$ git add index.html
halala@vlbarsegyan:~/Desktop/University/hello$ nano index.html
halala@vlbarsegyan:~/Desktop/University/hello$ git add index.html
warning: in the working copy of 'index.html', LF will be replaced by CRLF the next tim
e Git touches it
halala@vlbarsegyan:~/Desktop/University/hello$ git commit -m "Updated index.html"
[style b0030d9] Updated index.html
1 file changed, 3 insertions(+)
halala@vlbarsegyan:~/Desktop/University/hello$
```

Рис. 4.32: Обновление файлов, индексация и коммиты

## 4.16 Навигация по веткам

33. Проверяю логи последних коммитов, вижу все созданные коммиты, они в ветке `style` (рис. 4.33).

```
halala@vlbarsegyan: ~/Desktop/University/hello
style.css updated index.html
1 file changed, 3 insertions(+)
halala@vlbarsegyan:~/Desktop/University/hello$ git log --all
commit b0030d9bf9746d51f2c735d690ae44c11ec193af (HEAD -> style)
Author: VARDamn <74242594+VARDamn@users.noreply.github.com>
Date: Sat Feb 22 19:33:55 2025 +0300

    Updated index.html

commit b1e9e46db5d62d5697e1341daa914f398f75d396
Author: VARDamn <74242594+VARDamn@users.noreply.github.com>
Date: Sat Feb 22 19:33:20 2025 +0300

    Hello uses style.css

commit 76ebe56f243f0bb6fceb2a1e0a1efb4919eb052d
Author: VARDamn <74242594+VARDamn@users.noreply.github.com>
Date: Sat Feb 22 19:32:44 2025 +0300

    Added css stylesheet

commit ac7af6f87c376835b2828381aa16199a729094a8 (master)
Author: VARDamn <74242594+VARDamn@users.noreply.github.com>
Date: Sat Feb 22 19:23:34 2025 +0300
```

Рис. 4.33: Проверка логов

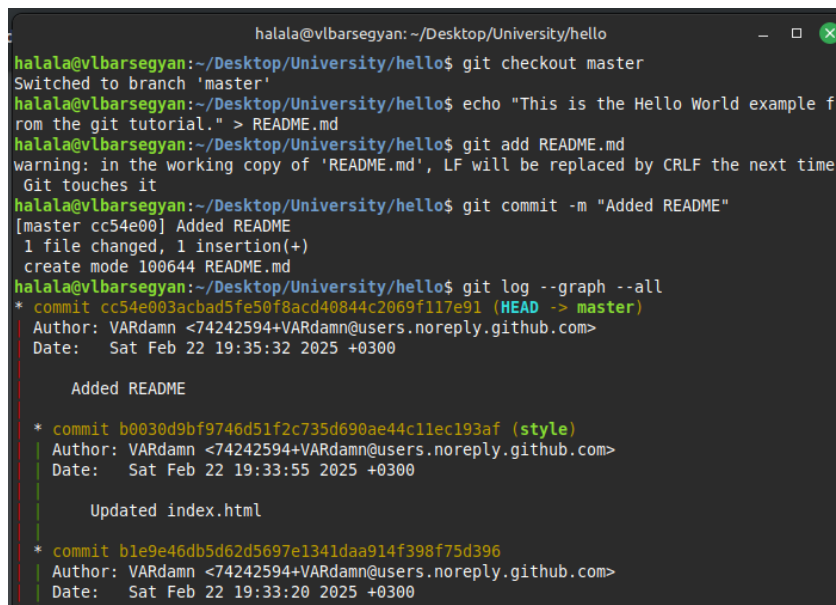
34. Перехожу в ветку master и просматриваю содержимое файла lib/hello.html - в нем нет последних изменений. Просматриваю тот же файл в ветке style - изменения есть (рис. 4.34).

```
halala@vlbarsegyan: ~/Desktop/University/hello
halala@vlbarsegyan:~/Desktop/University/hello$ git checkout master
Switched to branch 'master'
halala@vlbarsegyan:~/Desktop/University/hello$ cat lib/hello.html
<!-- Author: Dmitry S. Kulyabov (kulyabov-ds@rudn.ru) -->
<html>
<head>
</head>
<body>
<h1>Hello, World!</h1>
</body>
</html>
halala@vlbarsegyan:~/Desktop/University/hello$ git checkout style
Switched to branch 'style'
halala@vlbarsegyan:~/Desktop/University/hello$ cat lib/hello.html
<!-- Author: Dmitry S. Kulyabov (kulyabov-ds@rudn.ru) -->
<html>
<head>
<link type="text/css" rel="stylesheet" media="all" href="style.css" />
</head>
<body>
<h1>Hello, World!</h1>
</body>
</html>
halala@vlbarsegyan:~/Desktop/University/hello$
```

Рис. 4.34: Проверка файла в разных ветках

## 4.17 Изменения в ветке master

35. Перехожу в ветку master и создаю файл README.md командой *echo "This is the Hello World example from the git tutorial." > README.md*. Коммичу изменения и просматриваю логи, включая различия между ветками. (рис. 4.35).



```
halala@vlbarsegyan: ~/Desktop/University/hello
halala@vlbarsegyan:~/Desktop/University/hello$ git checkout master
Switched to branch 'master'
halala@vlbarsegyan:~/Desktop/University/hello$ echo "This is the Hello World example f
rom the git tutorial." > README.md
halala@vlbarsegyan:~/Desktop/University/hello$ git add README.md
warning: in the working copy of 'README.md', LF will be replaced by CRLF the next time
Git touches it
halala@vlbarsegyan:~/Desktop/University/hello$ git commit -m "Added README"
[master cc54e00] Added README
1 file changed, 1 insertion(+)
create mode 100644 README.md
halala@vlbarsegyan:~/Desktop/University/hello$ git log --graph --all
* commit cc54e003acbad5fe50f8acd40844c2069f117e91 (HEAD -> master)
  Author: VARdamn <74242594+VARdamn@users.noreply.github.com>
  Date:   Sat Feb 22 19:35:32 2025 +0300
      Added README
* commit b0030d9bf9746d51f2c735d690ae44c11ec193af (style)
  Author: VARdamn <74242594+VARdamn@users.noreply.github.com>
  Date:   Sat Feb 22 19:33:55 2025 +0300
      Updated index.html
* commit b1e9e46db5d62d5697e1341daa914f398f75d396
  Author: VARdamn <74242594+VARdamn@users.noreply.github.com>
  Date:   Sat Feb 22 19:33:20 2025 +0300
```

Рис. 4.35: Создание README в ветке master, проверка дерева веток

## 4.18 Слияние

36. Перехожу в ветку style командой *git checkout style*. Сливаю master с style командой *git merge master*. Вывожу логи с деревом всех веток (рис. 4.36).

```
halala@vlbarsegyan: ~/Desktop/University/hello
halala@vlbarsegyan:~/Desktop/University/hello$ git checkout style
Switched to branch 'style'
halala@vlbarsegyan:~/Desktop/University/hello$ git merge master
Merge made by the 'ort' strategy.
 README.md | 1 +
 1 file changed, 1 insertion(+)
 create mode 100644 README.md
halala@vlbarsegyan:~/Desktop/University/hello$ git log --graph --all
*   commit 1a7dcf074afa6e16375af238a5309df25bee408f (HEAD -> style)
  \ Merge: b0030d9 cc54e00
   Author: VARdamn <74242594+VARdamn@users.noreply.github.com>
   Date: Sat Feb 22 19:36:05 2025 +0300

   Merge branch 'master' into style
*   commit cc54e003acbad5fe50f8acd40844c2069f117e91 (master)
  \ Author: VARdamn <74242594+VARdamn@users.noreply.github.com>
   Date: Sat Feb 22 19:35:32 2025 +0300

   Added README
*   commit b0030d9bf9746d51f2c735d690ae44c11ec193af
  \ Author: VARdamn <74242594+VARdamn@users.noreply.github.com>
   Date: Sat Feb 22 19:33:55 2025 +0300

   Updated index.html
*   commit b1e9e46db5d62d5697e1341daa914f398f75d396
```

Рис. 4.36: Слияние master с style

## 4.19 Создание конфликта

37. Возвращаюсь в ветку master и редактирую файл lib/hello.html. Коммичу изменения, просматриваю логи (рис. 4.37).

```
halala@vlbarsegyan: ~/Desktop/University/hello
halala@vlbarsegyan:~/Desktop/University/hello$ git checkout master
Switched to branch 'master'
halala@vlbarsegyan:~/Desktop/University/hello$ nano lib/hello.html
halala@vlbarsegyan:~/Desktop/University/hello$ git add lib/hello.html
halala@vlbarsegyan:~/Desktop/University/hello$ git commit -m 'Life is great'
[master 10b297c] Life is great
 1 file changed, 3 insertions(+), 2 deletions(-)
halala@vlbarsegyan:~/Desktop/University/hello$ git log --graph --all
*   commit 10b297c3d797727f98810bc6b97056134b1a7475 (HEAD -> master)
  \ Author: VARdamn <74242594+VARdamn@users.noreply.github.com>
   Date: Sat Feb 22 19:36:59 2025 +0300

   Life is great
*   commit 1a7dcf074afa6e16375af238a5309df25bee408f (style)
  \ Merge: b0030d9 cc54e00
   Author: VARdamn <74242594+VARdamn@users.noreply.github.com>
   Date: Sat Feb 22 19:36:05 2025 +0300

   Merge branch 'master' into style
*   commit cc54e003acbad5fe50f8acd40844c2069f117e91
  \ Author: VARdamn <74242594+VARdamn@users.noreply.github.com>
   Date: Sat Feb 22 19:35:32 2025 +0300

   Added README
*   commit b0030d9bf9746d51f2c735d690ae44c11ec193af
```

Рис. 4.37: Редактирование lib/hello.html в master

## 4.20 Разрешение конфликтов

38. Перехожу в ветку `style` и пытаюсь слить `master` с `style`. В файле `lib/hello.html` возникли конфликты. Разрешаю конфликты и коммичу изменения (рис. 4.38).

```
halala@vlbarsegyan:~/Desktop/University/hello$ git checkout style
Switched to branch 'style'
halala@vlbarsegyan:~/Desktop/University/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.
halala@vlbarsegyan:~/Desktop/University/hello$ nano lib/hello.html
halala@vlbarsegyan:~/Desktop/University/hello$ git add lib/hello.html
halala@vlbarsegyan:~/Desktop/University/hello$ git commit -m "Merged master fixed conf
lict."
[style 16f4bd2] Merged master fixed conflict.
halala@vlbarsegyan:~/Desktop/University/hello$
```

Рис. 4.38: Разрешение конфликта

## 4.21 Сброс ветки `style`

39. Перехожу в ветку `style` и просматриваю логи изменений (рис. 4.39).

```
halala@vlbarsegyan:~/Desktop/University/hello
halala@vlbarsegyan:~/Desktop/University/hello$ git checkout style
Already on 'style'
halala@vlbarsegyan:~/Desktop/University/hello$ git log --graph
fatal: unrecognized argument: --graph
halala@vlbarsegyan:~/Desktop/University/hello$ git log --graph
* commit 16f4bd2fdf9c7fe861720925a6b909184376aaba (HEAD -> style)
|\
| Merge: 1a7dcf0 10b297c
| Author: VARdamn <74242594+VARdamn@users.noreply.github.com>
| Date: Sat Feb 22 19:40:46 2025 +0300
|
| Merged master fixed conflict.
|
* commit 10b297c3d797727f98810bc6b97056134b1a7475 (master)
| Author: VARdamn <74242594+VARdamn@users.noreply.github.com>
| Date: Sat Feb 22 19:36:59 2025 +0300
|
| Life is great
|
* commit 1a7dcf074afa6e16375af238a5309df25bee408f
|\
| Merge: b0030d9 cc54e00
| Author: VARdamn <74242594+VARdamn@users.noreply.github.com>
```

Рис. 4.39: Логи изменений ветки `style`

40. Сбрасываю ветку к коммиту «Updated index.html» командой `git reset --hard` с использованием хэша коммита. Снова проверяю лог ветки `style` (рис. 4.40).

```
halala@vlbarsegyan: ~/Desktop/University/hello
halala@vlbarsegyan:~/Desktop/University/hello$ git reset --hard b0030d9bf9746d51f2c73d690ae44c11ec193af
HEAD is now at b0030d9 Updated index.html
halala@vlbarsegyan:~/Desktop/University/hello$ git log --graph --all
* commit 10b297c3d797727f98810bc6b97056134b1a7475 (master)
  Author: VARdamn <74242594+VARdamn@users.noreply.github.com>
  Date: Sat Feb 22 19:36:59 2025 +0300

    Life is great

* commit cc54e003acbad5fe50f8acd40844c2069f117e91
  Author: VARdamn <74242594+VARdamn@users.noreply.github.com>
  Date: Sat Feb 22 19:35:32 2025 +0300

    Added README

* commit b0030d9bf9746d51f2c735d690ae44c11ec193af (HEAD -> style)
  Author: VARdamn <74242594+VARdamn@users.noreply.github.com>
  Date: Sat Feb 22 19:33:55 2025 +0300

    Updated index.html

* commit b1e9e46db5d62d5697e1341daa914f398f75d396
  Author: VARdamn <74242594+VARdamn@users.noreply.github.com>
  Date: Sat Feb 22 19:33:20 2025 +0300

    Hello uses style.css
```

Рис. 4.40: Сброс ветки style с помощью git reset

## 4.22 Сброс ветки master

41. Перехожу в ветку master и аналогичным образом сбрасываю ее к коммиту «Added README». Просматриваю лог ветки master (рис. 4.41).



```
halala@vlbarsegyan: ~/Desktop/University/hello
* commit cc54e003acbad5fe50f8acd40844c2069f117e91
Author: VARdamn <74242594+VARdamn@users.noreply.github.com>
Date: Sat Feb 22 19:35:32 2025 +0300

    Added README

* commit ac7af6f87c376835b2828381aa16199a729094a8
Author: VARdamn <74242594+VARdamn@users.noreply.github.com>
Date: Sat Feb 22 19:23:34 2025 +0300

    Added index.html.

* commit 4656c98420063f41e7ebb6e0375c2aedb2ddd18a
Author: VARdamn <74242594+VARdamn@users.noreply.github.com>
Date: Sat Feb 22 19:22:44 2025 +0300

    Moved hello.html to lib

* commit b6f31650ebc5cc5eb5001acbd3fd50f6d782e0c
Author: VARdamn <74242594+VARdamn@users.noreply.github.com>
Date: Sat Feb 22 19:20:29 2025 +0300
halala@vlbarsegyan:~/Desktop/University/hello$ git reset --hard cc54e003acbad5fe50f8acd40844c2069f117e91
HEAD is now at cc54e00 Added README
halala@vlbarsegyan:~/Desktop/University/hello$ git log --graph -all
error: switch 'l' expects a numerical value
halala@vlbarsegyan:~/Desktop/University/hello$ git log --graph --all
* commit cc54e003acbad5fe50f8acd40844c2069f117e91 (HEAD -> master)
Author: VARdamn <74242594+VARdamn@users.noreply.github.com>
Date: Sat Feb 22 19:35:32 2025 +0300

    Added README
```

Рис. 4.41: Сброс ветки master

## 4.23 Перебазирование

42. Перехожу в ветку style и использую вместо слияния команду *git rebase master*.

Просматриваю лог ветки. (рис. 4.42).

```
halala@vlbarsegyan: ~/Desktop/University/hello
halala@vlbarsegyan:~/Desktop/University/hello$ git checkout style
Switched to branch 'style'
halala@vlbarsegyan:~/Desktop/University/hello$ git rebase master
Successfully rebased and updated refs/heads/style.
halala@vlbarsegyan:~/Desktop/University/hello$ git log --graph
* commit 6e965428b89f4a7293122e82a6a43e3f024fa2ab (HEAD -> style)
  Author: VARDamn <74242594+VARDamn@users.noreply.github.com>
  Date: Sat Feb 22 19:33:55 2025 +0300

    Updated index.html

* commit 065205de484428b98faccdd474eb1c27cc9d9192
  Author: VARDamn <74242594+VARDamn@users.noreply.github.com>
  Date: Sat Feb 22 19:33:20 2025 +0300

    Hello uses style.css

* commit 85ea8708102279486ac63e722ece2c5db712bca9
  Author: VARDamn <74242594+VARDamn@users.noreply.github.com>
  Date: Sat Feb 22 19:32:44 2025 +0300

    Added css stylesheet

* commit cc54e003acbad5fe50f8acd40844c2069f117e91 (master)
  Author: VARDamn <74242594+VARDamn@users.noreply.github.com>
  Date: Sat Feb 22 19:35:32 2025 +0300

    Added README

* commit ac7af6f87c376835b2828381aa16199a729094a8
  Author: VARDamn <74242594+VARDamn@users.noreply.github.com>
  Date: Sat Feb 22 19:23:34 2025 +0300
```

Рис. 4.42: Перебазирование ветки

## 4.24 Слияние в ветку master

43. Перехожу в ветку master и сливаю style с master. Просматриваю лог - теперь ветки style и master идентичны. (рис. 4.43).

```

halala@vlbarsegyan: ~/Desktop/University/hello
halala@vlbarsegyan:~/Desktop/University/hello$ git checkout master
Switched to branch 'master'
halala@vlbarsegyan:~/Desktop/University/hello$ git merge style
Updating cc54e00..6e96542
Fast-forward
 index.html | 3 +++
 lib/hello.html | 1 +
 lib/style.css | 3 +++
 3 files changed, 7 insertions(+)
 create mode 100644 lib/style.css
halala@vlbarsegyan:~/Desktop/University/hello$ git log
commit 6e965428b89f4a7293122e82a6a43e3f024fa2ab (HEAD -> master, style)
Author: VARDamn <74242594+VARDamn@users.noreply.github.com>
Date: Sat Feb 22 19:33:55 2025 +0300

    Updated index.html

commit 065205de484428b98faccdd474eb1c27cc9d9192
Author: VARDamn <74242594+VARDamn@users.noreply.github.com>
Date: Sat Feb 22 19:33:20 2025 +0300

    Hello uses style.css

commit 85ea8708102279486ac63e722ece2c5db712bca9
Author: VARDamn <74242594+VARDamn@users.noreply.github.com>
Date: Sat Feb 22 19:32:44 2025 +0300

    Added css stylesheet

commit cc54e003acbad5fe50f8acd40844c2069f117e91
Author: VARDamn <74242594+VARDamn@users.noreply.github.com>
Date: Sat Feb 22 19:35:32 2025 +0300

```

Рис. 4.43: Слияние style с master

## 4.25 Клонирование репозитория и просмотр клонированного репозитория

44. Перехожу в родительский каталог. Создаю клон репозитория hello в папке cloned\_hello командой *git clone hello cloned\_hello*. Перехожу в cloned\_hello и просматриваю папки и файлы. Вывожу список всех коммитов командой *git log -all* (рис. 4.44).

```
halala@vlbarsegyan: ~/Desktop/University/cloned_hello
halala@vlbarsegyan:~/Desktop/University/hello$ cd ..
halala@vlbarsegyan:~/Desktop/University$ pwd
/home/halala/Desktop/University
halala@vlbarsegyan:~/Desktop/University$ ls
hello  sem6  sem6
halala@vlbarsegyan:~/Desktop/University$ git clone hello cloned_hello
Cloning into 'cloned_hello'...
done.
halala@vlbarsegyan:~/Desktop/University$ ls
cloned_hello  hello  sem6  sem6
halala@vlbarsegyan:~/Desktop/University$ cd cloned_hello/ && ls
index.html  lib  README.md
halala@vlbarsegyan:~/Desktop/University/cloned_hello$ git log --all
commit 6e965428b89f4a7293122e82a6a43e3f024fa2ab (HEAD -> master, origin/master, origin/HEAD)
Author: VARdamn <74242594+VARdamn@users.noreply.github.com>
Date: Sat Feb 22 19:33:55 2025 +0300

    Updated index.html

commit 065205de484428b98faccdd474eb1c27cc9d9192
Author: VARdamn <74242594+VARdamn@users.noreply.github.com>
Date: Sat Feb 22 19:33:20 2025 +0300

    Hello uses style.css

commit 85ea8708102279486ac63e722ece2c5db712bca9
Author: VARdamn <74242594+VARdamn@users.noreply.github.com>
Date: Sat Feb 22 19:32:44 2025 +0300

    Added css stylesheet
```

Рис. 4.44: Создание и просмотр клонированного репозитория

## 4.26 Что такое origin? Удаленные ветки

45. Выполняю команды *git remote* и *git remote show origin* для более подробной информации. Просматриваю ветки, доступные в клонированном репозитории, командой *git branch*. Просматриваю все ветки командой *git branch -a* (рис. 4.45).

```
halala@vlbarsegyan: ~/Desktop/University/cloned_hello
halala@vlbarsegyan:~/Desktop/University/cloned_hello$ git remote
origin
halala@vlbarsegyan:~/Desktop/University/cloned_hello$ git remote show origin
* remote origin
Fetch URL: /home/halala/Desktop/University/hello
Push URL: /home/halala/Desktop/University/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)
halala@vlbarsegyan:~/Desktop/University/cloned_hello$ git branch
* master
halala@vlbarsegyan:~/Desktop/University/cloned_hello$ git branch -a
* master
remotes/origin/HEAD -> origin/master
remotes/origin/master
remotes/origin/style
halala@vlbarsegyan:~/Desktop/University/cloned_hello$
```

Рис. 4.45: git remote, просмотр веток

## 4.27 Изменение оригинального репозитория

46. Редактирую README.md в репозитории hello и коммичу изменения. В клонированном репозитории выполняю *git fetch* для извлечения изменений, проверяю лог - вижу последний сделанный коммит (рис. 4.46).

```
halala@vlbarsegyan: ~/Desktop/University/cloned_hello
halala@vlbarsegyan:~/Desktop/University/cloned_hello$ cd ../hello/
halala@vlbarsegyan:~/Desktop/University/hello$ nano README.md
halala@vlbarsegyan:~/Desktop/University/hello$ git add README.md
halala@vlbarsegyan:~/Desktop/University/hello$ git commit -m "Changed README in original repo"
[master c403197] Changed README in original repo
1 file changed, 1 insertion(+), 1 deletion(-)
halala@vlbarsegyan:~/Desktop/University/hello$ cd ../cloned_hello/
halala@vlbarsegyan:~/Desktop/University/cloned_hello$ git fetch
remote: Enumerating objects: 5, done.
remote: Counting objects: 100% (5/5), done.
remote: Compressing objects: 100% (3/3), done.
remote: Total 3 (delta 1), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (3/3), 346 bytes | 346.00 KiB/s, done.
From /home/halala/Desktop/University/hello
6e96542..c403197 master -> origin/master
halala@vlbarsegyan:~/Desktop/University/cloned_hello$ git log --all
commit c403197b80d00b11fde639b9a1912a2bc6b6ffc0 (origin/master, origin/HEAD)
Author: VARDamn <74242594+VARDamn@users.noreply.github.com>
Date: Sat Feb 22 19:52:50 2025 +0300

    Changed README in original repo

commit 6e965428b89f4a7293122e82a6a43e3f024fa2ab (HEAD -> master, origin/style)
Author: VARDamn <74242594+VARDamn@users.noreply.github.com>
Date: Sat Feb 22 19:33:55 2025 +0300

    Updated index.html

commit 065205de484428b98faccdd474eb1c27cc9d9192
Author: VARDamn <74242594+VARDamn@users.noreply.github.com>
Date: Sat Feb 22 19:33:20 2025 +0300
```

Рис. 4.46: Редактирование оригинального репозитория и извлечение изменений в клонированном

## 4.28 Слияние извлеченных изменений

47. Проверяю клонированный файл README.md - он не изменился. Выполняю слияние извлеченных изменений командой *git merge origin/master*. Снова проверяю содержимое файла - теперь оно обновилось. (рис. 4.47).

```
halala@vlbarsegyan:~/Desktop/University/cloned_hello$ cat README.md
This is the Hello World example from the git tutorial.
halala@vlbarsegyan:~/Desktop/University/cloned_hello$ git merge origin/master
Updating 6e96542..c403197
Fast-forward
 README.md | 2 +-
1 file changed, 1 insertion(+), 1 deletion(-)
halala@vlbarsegyan:~/Desktop/University/cloned_hello$ cat README.md
This is the Hello World example from the git tutorial.
halala@vlbarsegyan:~/Desktop/University/cloned_hello$ git pull
Already up to date.
```

Рис. 4.47: Проверка файла до и после слияния изменений

## 4.29 Добавление ветки наблюдения

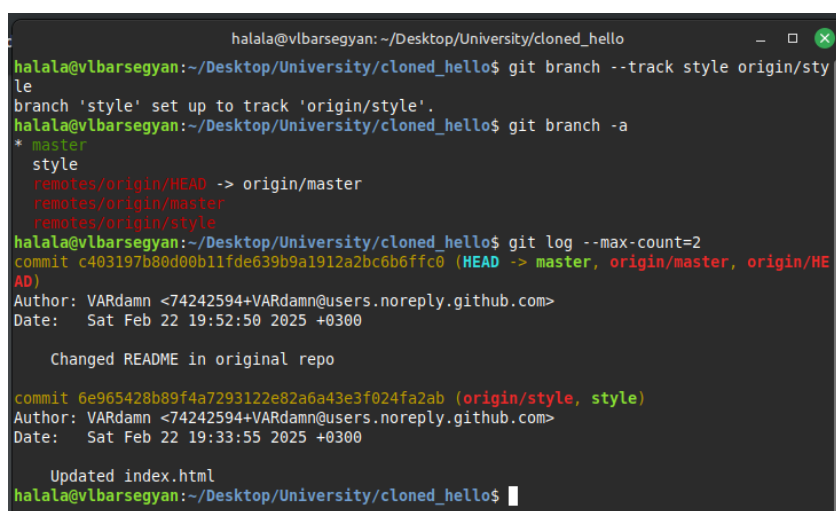
48. Выполняю:

```
git branch --track style origin/style
```

```
git branch -a
```

```
git log --max-count=2
```

Теперь можно видеть ветку style в списке веток и логе (рис. 4.48).



```
halala@vlbarsegyan: ~/Desktop/University/cloned_hello
halala@vlbarsegyan:~/Desktop/University/cloned_hello$ git branch --track style origin/style
branch 'style' set up to track 'origin/style'.
halala@vlbarsegyan:~/Desktop/University/cloned_hello$ git branch -a
* master
  style
  remotes/origin/HEAD -> origin/master
  remotes/origin/master
  remotes/origin/style
halala@vlbarsegyan:~/Desktop/University/cloned_hello$ git log --max-count=2
commit c403197b80d00b11fde639b9a1912a2bc6b6fffc0 (HEAD -> master, origin/master, origin/HEAD)
Author: VARdamn <74242594+VARdamn@users.noreply.github.com>
Date: Sat Feb 22 19:52:50 2025 +0300

    Changed README in original repo

commit 6e965428b89f4a7293122e82a6a43e3f024fa2ab (origin/style, style)
Author: VARdamn <74242594+VARdamn@users.noreply.github.com>
Date: Sat Feb 22 19:33:55 2025 +0300

    Updated index.html
halala@vlbarsegyan:~/Desktop/University/cloned_hello$
```

Рис. 4.48: Добавление ветки наблюдения

## 4.30 Создание чистого репозитория, добавление удаленного репозитория

49. Создаю чистый репозиторий командой *git clone --bare hello hello.git* и перехожу в него. Добавляю удаленный репозиторий командой *git remote add shared ../hello.git*. Редактирую файл README.md и отправляю изменения в общий репозиторий. В клонированном репозитории добавляю командой *git remote add shared ../hello.git*, и извлекаю сделанные изменения командой *git pull shared master*, проверяю файл (рис. 4.49).

```
halala@vlabsegyan: ~/Desktop/University/cloned_hello
halala@vlabsegyan:~/Desktop/University/cloned_hello$ cd ..
halala@vlabsegyan:~/Desktop/University$ git clone --bare hello hello.git
Cloning into bare repository 'hello.git'...
done.
halala@vlabsegyan:~/Desktop/University$ ls hello.git
branches  config  description  HEAD  hooks  info  objects  packed-refs  refs
halala@vlabsegyan:~/Desktop/University$ cd hello
halala@vlabsegyan:~/Desktop/University/hello$ git remote add shared ../hello.git
halala@vlabsegyan:~/Desktop/University/hello$ nano README.md
halala@vlabsegyan:~/Desktop/University/hello$ git checkout master
M      README.md
Already on 'master'
halala@vlabsegyan:~/Desktop/University/hello$ git add README.md
halala@vlabsegyan:~/Desktop/University/hello$ git commit -m "Added shared comment to readme"
[master f19acd8] Added shared comment to readme
1 file changed, 2 insertions(+), 1 deletion(-)
halala@vlabsegyan:~/Desktop/University/hello$ git push shared master
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 12 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 420 bytes | 420.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To ../hello.git
c403197..f19acd8  master -> master
halala@vlabsegyan:~/Desktop/University/hello$ cd ../cloned_hello/
halala@vlabsegyan:~/Desktop/University/cloned_hello$ git remote add shared ../hello.git
halala@vlabsegyan:~/Desktop/University/cloned_hello$ git branch --track shared master
branch 'shared' set up to track 'master'.
halala@vlabsegyan:~/Desktop/University/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), 400 bytes | 133.00 KiB/s, done.
From ../hello
* branch      master      -> FETCH HEAD
* [new branch] master      -> shared/master
Updating c403197..f19acd8
Fast-forward
 README.md | 3 ++
1 file changed, 2 insertions(+), 1 deletion(-)
halala@vlabsegyan:~/Desktop/University/cloned_hello$ cat README.md
This is the Hello World example from the git tutorial.
(Changed in the original and pushed to shared)
halala@vlabsegyan:~/Desktop/University/cloned_hello$
```

Рис. 4.49: Создание чистого репозитория, добавление удаленного репозитория, извлечение общих изменений



## 5 Выводы

Я настроил git, научился с ним работать, узнал про различные ветки, слияние, перебазирование, чистые репозитории.

## Список литературы

1. Git [Электронный ресурс]. 2025. URL: <https://ru.wikipedia.org/wiki/Git>.