

Отчёт по лабораторной работе №1

Система контроля версий Git

Дарья Эдуардовна Ибатулина

Содержание

1	Цель работы	6
2	Задание	7
3	Теоретическое введение	8
4	Выполнение лабораторной работы	10
5	Выводы	50
	Список литературы	51

Список иллюстраций

4.1	Настройка параметров установки окончаний строк	10
4.2	Установка флага	10
4.3	Создание директории и файла в ней	11
4.4	Создание Git-репозитория	11
4.5	Проверка состояния Git-репозитория	11
4.6	Создание фвйла и его заполнение	11
4.7	Проверка состояния репозитория	12
4.8	Добавление изменения в репозиторий	12
4.9	Коммит и редактирование коммита	13
4.10	Проверка состояния	13
4.11	Редактирование файла	13
4.12	Добавление изменений	13
4.13	Редактирование файла	14
4.14	Проверка состояния	14
4.15	Коммит	14
4.16	Проверка состояния	15
4.17	Добавление файлов, проверка состояния	15
4.18	Коммит второго изменения	15
4.19	Список изменений - лог	16
4.20	Формат истории - строка	16
4.21	Справочник по логу	17
4.22	Переключение на ветку мастер и проверка содержимого файла	18
4.23	Возврат к предыдущей версии и вывод содержимого файла	18
4.24	Переключение между тегами (1)	19
4.25	Переключение между тегами (2)	19
4.26	Переключение между тегами (3)	20
4.27	Шаг 1	20
4.28	Шаг 2	20
4.29	Шаг 3	21
4.30	Шаг 4	21
4.31	Шаг 5	21
4.32	Шаг 6	21
4.33	Шаг 7	22
4.34	Шаг 8	22
4.35	Шаг 9	22
4.36	Шаг 10	22
4.37	Шаг 11	23

4.38 Шаг 1	24
4.39 Шаг 2	25
4.40 Шаг 3	26
4.41 Шаг 1	26
4.42 Шаг 2	26
4.43 Шаг 3	27
4.44 Шаг 4	27
4.45 Шаг 5	27
4.46 Копирование	28
4.47 Отправка изменений	28
4.48 Создание и редактирование файла	28
4.49 Отправка изменений	28
4.50 Создание и редактирование файла	29
4.51 Шаг 1	29
4.52 Шаг 2	29
4.53 Шаг 3	29
4.54 Шаг 4	30
4.55 Шаг 5	30
4.56 Шаг 1	30
4.57 Шаг 2	31
4.58 Шаг 3	31
4.59 Шаг 4	31
4.60 Шаг 5	31
4.61 Шаг 6	32
4.62 Шаг 1	32
4.63 Шаг 2	32
4.64 Шаг 3	32
4.65 Шаг 4	33
4.66 Шаг 5	33
4.67 Шаг 1	33
4.68 Шаг 2	33
4.69 Шаг 3	34
4.70 Шаг 4	34
4.71 Шаг 5	35
4.72 Шаг 6	35
4.73 Шаг 7	35
4.74 Шаг 1	36
4.75 Шаг 2	36
4.76 Шаг 3	36
4.77 Шаг 4	37
4.78 Шаг 5	37
4.79 Шаг 6	37
4.80 Шаг 7	38
4.81 Шаг 8	38

4.82 Шаг 9	38
4.83 Шаг 10	39
4.84 Шаг 11	39
4.85 Шаг 12	40
4.86 Шаг 13	41
4.87 Шаг 14	42
4.88 Шаг 15	43
4.89 Шаг 16	43
4.90 Шаг 1	43
4.91 Шаг 2	43
4.92 Шаг 3	44
4.93 Шаг 4	44
4.94 git remote, origin	45
4.95 Просмотр всех веток (в том числе и удаленных)	45
4.96 Шаг 1	45
4.97 Шаг 2	46
4.98 Шаг 3	46
4.99 Шаг 4	46
4.100 Шаг 5	47
4.101 Шаг 6	47
4.102 Чистый репозиторий	47
4.103 Переход в каталог hello	48
4.104 Файл README.md	48
4.105 Коммит	48
4.106 Отправка изменений в общий репозиторий	48
4.107 Извлечение изменений из общего репозитория в клонированный	49

1 Цель работы

Познакомиться с новыми командами Git, повторить уже изученный материал.

2 Задание

- Настроить Git;
- Создать тестовый репозиторий и поработать с ним.

3 Теоретическое введение

Системы контроля версий (Version Control System, VCS) применяются при работе нескольких человек над одним проектом. Обычно основное дерево проекта хранится в локальном или удалённом репозитории, к которому настроен доступ для участников проекта. При внесении изменений в содержание проекта система контроля версий позволяет их фиксировать, совмещать изменения, произведённые разными участниками проекта, производить откат к любой более ранней версии проекта, если это требуется.

В классических системах контроля версий используется централизованная модель, предполагающая наличие единого репозитория для хранения файлов. Выполнение большинства функций по управлению версиями осуществляется специальным сервером. Участник проекта (пользователь) перед началом работы посредством определённых команд получает нужную ему версию файлов. После внесения изменений, пользователь размещает новую версию в хранилище. При этом предыдущие версии не удаляются из центрального хранилища и к ним можно вернуться в любой момент. Сервер может сохранять не полную версию изменённых файлов, а производить так называемую дельта-компрессию — сохранять только изменения между последовательными версиями, что позволяет уменьшить объём хранимых данных.

Системы контроля версий поддерживают возможность отслеживания и разрешения конфликтов, которые могут возникнуть при работе нескольких человек над одним файлом. Можно объединить (слить) изменения, сделанные разными участниками (автоматически или вручную), вручную выбрать нужную версию,

отменить изменения вовсе или заблокировать файлы для изменения. В зависимости от настроек блокировка не позволяет другим пользователям получить рабочую копию или препятствует изменению рабочей копии файла средствами файловой системы ОС, обеспечивая таким образом, привилегированный доступ только одному пользователю, работающему с файлом.

Системы контроля версий также могут обеспечивать дополнительные, более гибкие функциональные возможности. Например, они могут поддерживать работу с несколькими версиями одного файла, сохраняя общую историю изменений до точки ветвления версий и собственные истории изменений каждой ветви. Кроме того, обычно доступна информация о том, кто из участников, когда и какие изменения вносил. Обычно такого рода информация хранится в журнале изменений, доступ к которому можно ограничить.

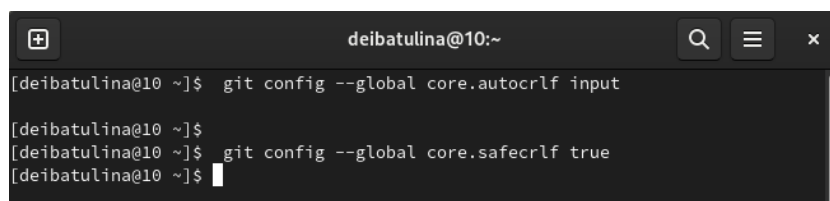
В отличие от классических, в распределённых системах контроля версий центральный репозиторий не является обязательным.

Среди классических VCS наиболее известны CVS, Subversion, а среди распределённых — Git, Bazaar, Mercurial. Принципы их работы схожи, отличаются они в основном синтаксисом используемых в работе команд.

4 Выполнение лабораторной работы

У меня уже установлен гит, так как я работала с ним на предыдущих курсах, поэтому пропускаю шаг с настройкой и перехожу сразу к заданию параметров.

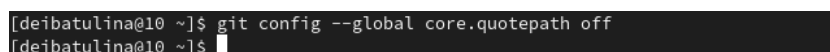
Настраиваю параметры установки окончаний строк. Если `core.safecrlf` установлен в `true` или `warn`, `git` проверяет, если преобразование является обратимым для текущей настройки `core.autocrlf`, `core.safecrlf true` - отвержение необратимого преобразования `lf<->crlf` (рис. [4.1]).

A screenshot of a terminal window titled 'deibatulina@10:~'. It shows the following commands and prompts:

```
[deibatulina@10 ~]$ git config --global core.autocrlf input
[deibatulina@10 ~]$
[deibatulina@10 ~]$ git config --global core.safecrlf true
[deibatulina@10 ~]$
```

Рис. 4.1: Настройка параметров установки окончаний строк

Чтобы избежать нечитаемых строк, установила соответствующий флаг (рис. [4.2]).

A screenshot of a terminal window showing the following commands and prompts:

```
[deibatulina@10 ~]$ git config --global core.quoteopath off
[deibatulina@10 ~]$
```

Рис. 4.2: Установка флага

Начну работу в домашнем каталоге с создания пустого каталога с именем `hello`, затем захожу в него и создала там файл с именем `hello.html` (рис. [4.3]).

```
[deibatulina@10 ~]$ mkdir hello
[deibatulina@10 ~]$ cd hello
[deibatulina@10 hello]$

[deibatulina@10 hello]$
[deibatulina@10 hello]$ touch hello.html
[deibatulina@10 hello]$ echo "Hello, world!" > hello.html
[deibatulina@10 hello]$
```

Рис. 4.3: Создание директории и файла в ней

Чтобы создать git репозиторий из этого каталога, выполнила команду `git init` (рис. [4.4]).

```
[deibatulina@10 hello]$ git init
подсказка: Using 'master' as the name for the initial branch. This default branch name
подсказка: is subject to change. To configure the initial branch name to use in all
подсказка: of your new repositories, which will suppress this warning, call:
подсказка:     git config --global init.defaultBranch <name>
подсказка:
подсказка: Names commonly chosen instead of 'master' are 'main', 'trunk' and
подсказка: 'development'. The just-created branch can be renamed via this command:
подсказка:     git branch -m <name>
Инициализирован пустой репозиторий Git в /home/deibatulina/hello/.git/
[deibatulina@10 hello]$ git add hello.html
[deibatulina@10 hello]$ git commit -m "Initial Commit"
[master (корневой коммит) ab59506] Initial Commit
1 file changed, 1 insertion(+)
create mode 100644 hello.html
[deibatulina@10 hello]$
```

Рис. 4.4: Создание Git-репозитория

Проверила текущее состояние репозитория (рис. [4.5]).

```
[deibatulina@10 hello]$ git status
Текущая ветка: master
ничего коммитить, нет изменений в рабочем каталоге
[deibatulina@10 hello]$
```

Рис. 4.5: Проверка состояния Git-репозитория

Создала файл и заполнила его (рис. [4.6]).

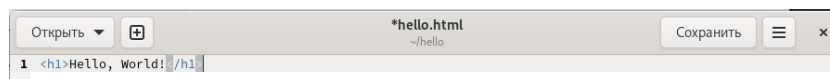


Рис. 4.6: Создание файла и его заполнение

Проверила состояние рабочего каталога. git знает, что файл hello.html был изменен, но при этом эти изменения еще не зафиксированы в репозитории (рис. [4.7]).

```
[deibatulina@10 hello]$ git status
Текущая ветка: master
Изменения, которые не в индексе для коммита:
  (используйте «git add <файл>...», чтобы добавить файл в индекс)
  (используйте «git restore <файл>...», чтобы отменить изменения в рабочем каталоге)
    изменено:      hello.html
индекс пуст (используйте «git add» и/или «git commit -a»)
[deibatulina@10 hello]$
```

Рис. 4.7: Проверка состояния репозитория

Добавила изменения в репозиторий (рис. [4.8]).

```
[deibatulina@10 hello]$ git add .
[deibatulina@10 hello]$ git status
Текущая ветка: master
Изменения, которые будут включены в коммит:
  (используйте «git restore --staged <файл>...», чтобы убрать из индекса)
    изменено:      hello.html
[deibatulina@10 hello]$
```

Рис. 4.8: Добавление изменения в репозиторий

Сделала коммит и проверю позже состояние. Откроется редактор. В первой строке ввела комментарий: «Added h1 tag» (рис. [4.9]).

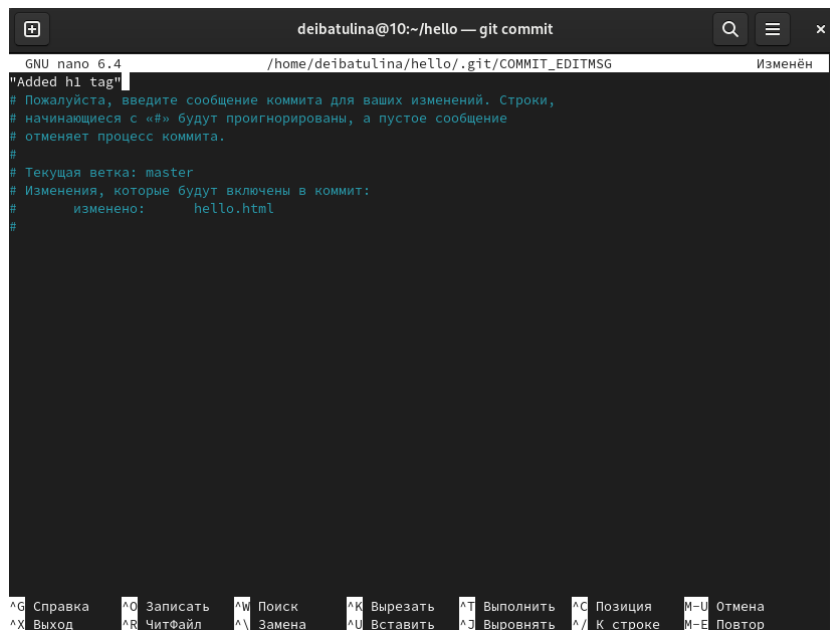


Рис. 4.9: Коммит и редактирование коммита

Проверила состояние. Теперь все изменения проиндексированы (рис. [4.10]).

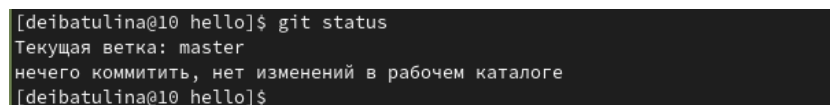


Рис. 4.10: Проверка состояния

Редактирую файл hello.html и добавляю изменение в гит (рис. [4.11], [4.12]).

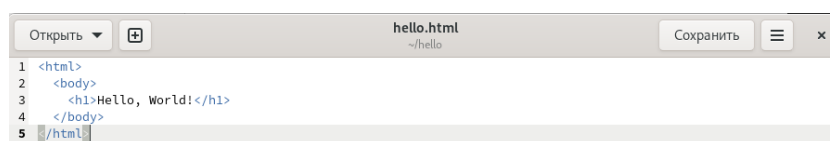


Рис. 4.11: Редактирование файла

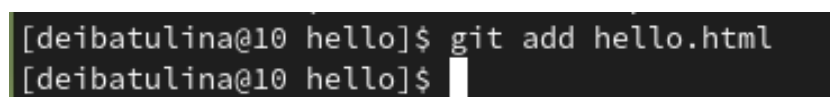


Рис. 4.12: Добавление изменений

Вновь редактирую файл `hello.html` и проверяю состояние репозитория (рис. [4.13], [4.14]).

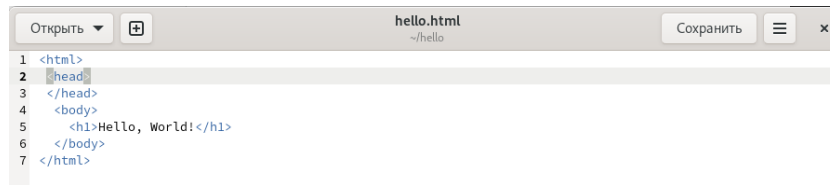


Рис. 4.13: Редактирование файла

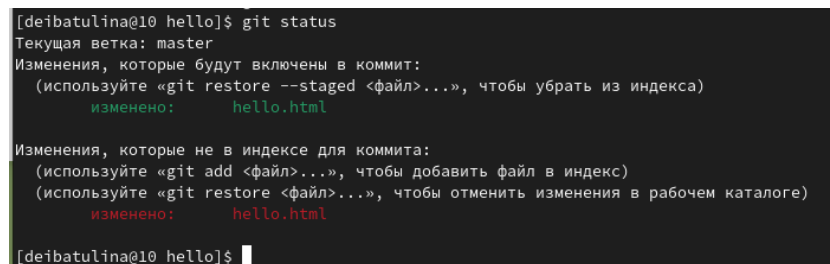


Рис. 4.14: Проверка состояния

`hello.html` указан дважды в состоянии. Первое изменение (добавление стандартных тегов) проиндексировано и готово к коммиту. Второе изменение (добавление заголовков HTML) является непроиндексированным.

Теперь добавляю второе изменение в индекс, а затем проверяю состояние (рис. [4.15], [4.16], [4.17]). Сделала коммит второго изменения (добавление заголовков) (рис. [4.18]).

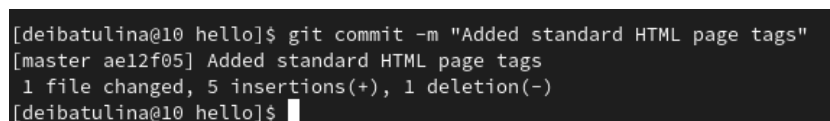


Рис. 4.15: Коммит

```
[deibatulina@10 hello]$ git status
Текущая ветка: master
Изменения, которые не в индексе для коммита:
  (используйте «git add <файл>...», чтобы добавить файл в индекс)
  (используйте «git restore <файл>...», чтобы отменить изменения в рабочем каталоге)
    изменено:      hello.html

индекс пуст (используйте «git add» и/или «git commit -a»)
[deibatulina@10 hello]$
```

Рис. 4.16: Проверка состояния

```
[deibatulina@10 hello]$ git add .
[deibatulina@10 hello]$ git status
Текущая ветка: master
Изменения, которые будут включены в коммит:
  (используйте «git restore --staged <файл>...», чтобы убрать из индекса)
    изменено:      hello.html

[deibatulina@10 hello]$
```

Рис. 4.17: Добавление файлов, проверка состояния

```
[deibatulina@10 hello]$ git commit -m "Added HTML header"
[master 1f73ffe] Added HTML header
 1 file changed, 2 insertions(+)
[deibatulina@10 hello]$
```

Рис. 4.18: Коммит второго изменения

Получим список произведенных изменений (рис. [4.19]).

```
[deibatulina@10 hello]$ git log
commit 1f73ffe708ac6e559a7627fba474ded3e517cb3f (HEAD -> master)
Author: Darya Ibatulina <fdarisha@yandex.ru>
Date: Sun Feb 16 15:11:04 2025 +0300

    Added HTML header

commit ae12f0518c1aa9107e4e835345174f3654a96ab0
Author: Darya Ibatulina <fdarisha@yandex.ru>
Date: Sun Feb 16 15:08:59 2025 +0300

    Added standard HTML page tags

commit eb9e461dc68c08fe014165c8b9bf35612dbc61fe
Author: Darya Ibatulina <fdarisha@yandex.ru>
Date: Sun Feb 16 15:02:02 2025 +0300

    "Added h1 tag"

commit ab5950644f692f714b320ed2a9d91a2f4e567118
Author: Darya Ibatulina <fdarisha@yandex.ru>
Date: Sun Feb 16 14:55:15 2025 +0300

    Initial Commit
[deibatulina@10 hello]$
```

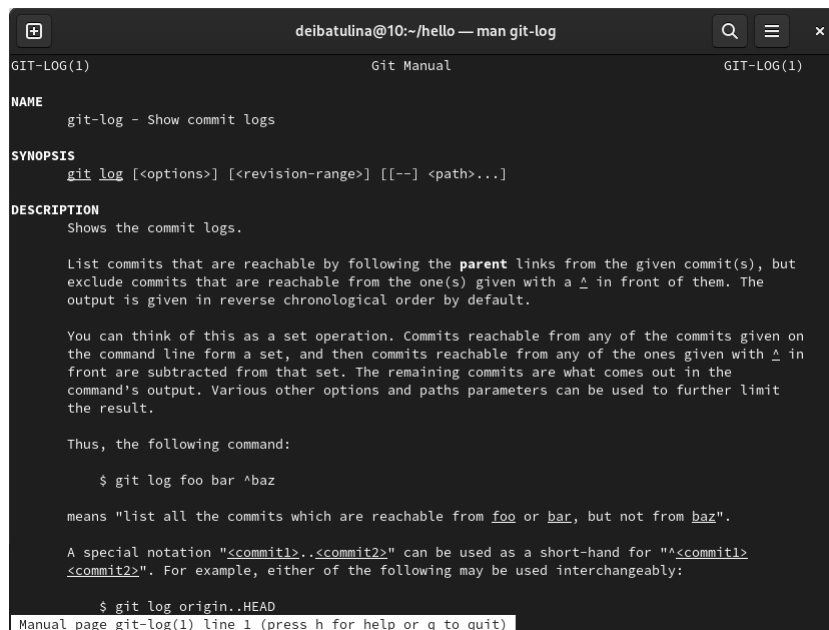
Рис. 4.19: Список изменений - лог

Однострочный формат истории (рис. [4.20]).

```
[deibatulina@10 hello]$ git log --pretty=oneline
1f73ffe708ac6e559a7627fba474ded3e517cb3f (HEAD -> master) Added HTML header
ae12f0518c1aa9107e4e835345174f3654a96ab0 Added standard HTML page tags
eb9e461dc68c08fe014165c8b9bf35612dbc61fe "Added h1 tag"
ab5950644f692f714b320ed2a9d91a2f4e567118 Initial Commit
[deibatulina@10 hello]$
```

Рис. 4.20: Формат истории - строка

Справочник по логу (рис. [4.21]).



```
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:

      $ git log origin..HEAD

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

Рис. 4.21: Справочник по логу

Получаю хэши предыдущих версий, изучаю данные лога и нахожу хэш для первого коммита. Он должен быть в последней строке данных. Использую этот хэш-код (достаточно первых 7 знаков) в команде ниже.

```
git checkout <hash>
```

Затем проверяю содержимое файла hello.html.

Возвращаюсь к последней версии в ветке master и вывожу содержимое файла (рис. [4.22], [4.23]).

```
deibatulina@10:~/hello
"Added h1 tag"
commit ab5950644f692f714b320ed2a9d91a2f4e567118
Author: Darya Ibatulina <fdarisha@yandex.ru>
Date: Sun Feb 16 14:55:15 2025 +0300

Initial Commit
[deibatulina@10 hello]$ git checkout ab59506
Примечание: переключение на «ab59506».

Вы сейчас в состоянии «отсоединённого указателя HEAD». Можете осмотреться,
внести экспериментальные изменения и зафиксировать их, также можете
отменить любые коммиты, созданные в этом состоянии, не затрагивая другие
ветки, переключившись обратно на любую ветку.

Если хотите создать новую ветку для сохранения созданных коммитов, можете
сделать это (сейчас или позже), используя команду switch с параметром -с.
Например:

    git switch -с <новая-ветка>

Или отмените эту операцию с помощью:

    git switch -

Отключите этот совет, установив переменную конфигурации
advice.detachedHead в значение false

HEAD сейчас на ab59506 Initial Commit
[deibatulina@10 hello]$ cat hello.html
Hello, world!
[deibatulina@10 hello]$
```

Рис. 4.22: Переключение на ветку мастер и проверка содержимого файла

```
[deibatulina@10 hello]$ git checkout master
Предыдущая позиция HEAD была ab59506 Initial Commit
Переключились на ветку «master»
[deibatulina@10 hello]$ cat hello.html
<html>
  <head>
  </head>
  <body>
    <h1>Hello, World!</h1>
  </body>
</html>
[deibatulina@10 hello]$
```

Рис. 4.23: Возврат к предыдущей версии и вывод содержимого файла

Создаю теги и переключаюсь между ними. Вывожу логи и могу просмотреть теги (рис. [4.24], [4.25], [4.26]).

```
[deibatulina@10 hello]$ git tag v1
[deibatulina@10 hello]$ git checkout v1^
Примечание: переключение на «v1^».
```

Вы сейчас в состоянии «отсоединённого указателя HEAD». Можете осмотреться, внести экспериментальные изменения и зафиксировать их, также можете отменить любые коммиты, созданные в этом состоянии, не затрагивая другие ветки, переключившись обратно на любую ветку.

Если хотите создать новую ветку для сохранения созданных коммитов, можете сделать это (сейчас или позже), используя команду switch с параметром -с. Например:

```
git switch -с <новая-ветка>
```

Или отмените эту операцию с помощью:

```
git switch -
```

Отключите этот совет, установив переменную конфигурации advice.detachedHead в значение false

HEAD сейчас на ae12f05 Added standard HTML page tags

```
[deibatulina@10 hello]$ cat hello.html
<html>
  <body>
    <h1>Hello, World!</h1>
  </body>
</html>
```

```
[deibatulina@10 hello]$ git tag v1-beta
[deibatulina@10 hello]$
```

Рис. 4.24: Переключение между тегами (1)

```
[deibatulina@10 hello]$ git checkout v1
Предыдущая позиция HEAD была ae12f05 Added standard HTML page tags
HEAD сейчас на 1f73ffe Added HTML header
[deibatulina@10 hello]$ git checkout v1-beta
error: pathspec 'v1-beta' did not match any file(s) known to git
[deibatulina@10 hello]$ git checkout v1-beta
Предыдущая позиция HEAD была 1f73ffe Added HTML header
HEAD сейчас на ae12f05 Added standard HTML page tags
[deibatulina@10 hello]$
```

Рис. 4.25: Переключение между тегами (2)

```

[deibatulina@10 hello]$ git tag
v1
v1-beta
[deibatulina@10 hello]$ git log master --all
commit 1f73ffe708ac6e559a7627fba474ded3e517cb3f (tag: v1, master)
Author: Darya Ibatulina <fdarisha@yandex.ru>
Date:   Sun Feb 16 15:11:04 2025 +0300

    Added HTML header

commit ae12f0518c1aa9107e4e835345174f3654a96ab0 (HEAD, tag: v1-beta)
Author: Darya Ibatulina <fdarisha@yandex.ru>
Date:   Sun Feb 16 15:08:59 2025 +0300

    Added standard HTML page tags

commit eb9e461dc68c08fe014165c8b9bf35612dbc61fe
Author: Darya Ibatulina <fdarisha@yandex.ru>
Date:   Sun Feb 16 15:02:02 2025 +0300

    "Added h1 tag"

commit ab5950644f692f714b320ed2a9d91a2f4e567118
Author: Darya Ibatulina <fdarisha@yandex.ru>
Date:   Sun Feb 16 14:55:15 2025 +0300

    Initial Commit
[deibatulina@10 hello]$

```

Рис. 4.26: Переключение между тегами (3)

Пробую отменить ошибочно добавленные коммиты и проверяю логи, используя команду `git revert`. Всё получается (рис. [4.27], [4.28], [4.29], [4.30], [4.31], [4.32], [4.33], [4.34], [4.35], [4.36], [4.37]).

```

[deibatulina@10 hello]$ git checkout master
Предыдущая позиция HEAD была ae12f05 Added standard HTML page tags
Переключились на ветку «master»
[deibatulina@10 hello]$

```

Рис. 4.27: Шаг 1



```

1 <html>
2 <head>
3 </head>
4 <body>
5   <h1>Hello, World!</h1>
6   <!-- This is a bad comment. We want to revert it. -->
7 </body>
8 </html>

```

Рис. 4.28: Шаг 2

```
[deibatulina@10 hello]$ git checkout hello.html
Updated 1 path from the index
[deibatulina@10 hello]$ git status
Текущая ветка: master
ничего коммитить, нет изменений в рабочем каталоге
[deibatulina@10 hello]$ cat hello.html
<html>
  <head>
  </head>
  <body>
    <h1>Hello, World!</h1>
  </body>
</html>
[deibatulina@10 hello]$
```

Рис. 4.29: Шаг 3



Рис. 4.30: Шаг 4

```
[deibatulina@10 hello]$ git add hello.html
[deibatulina@10 hello]$ git status
Текущая ветка: master
Изменения, которые будут включены в коммит:
  (используйте «git restore --staged <файл>...», чтобы убрать из индекса)
    изменено:   hello.html
[deibatulina@10 hello]$
```

Рис. 4.31: Шаг 5

```
[deibatulina@10 hello]$ git reset HEAD hello.html
Непроиндексированные изменения после сброса:
M       hello.html
[deibatulina@10 hello]$ git checkout
M       hello.html
[deibatulina@10 hello]$
```

Рис. 4.32: Шаг 6

```
[deibatulina@10 hello]$ git checkout hello.html
Updated 1 path from the index
[deibatulina@10 hello]$ git status
Текущая ветка: master
нечего коммитить, нет изменений в рабочем каталоге
[deibatulina@10 hello]$
```

Рис. 4.33: Шаг 7

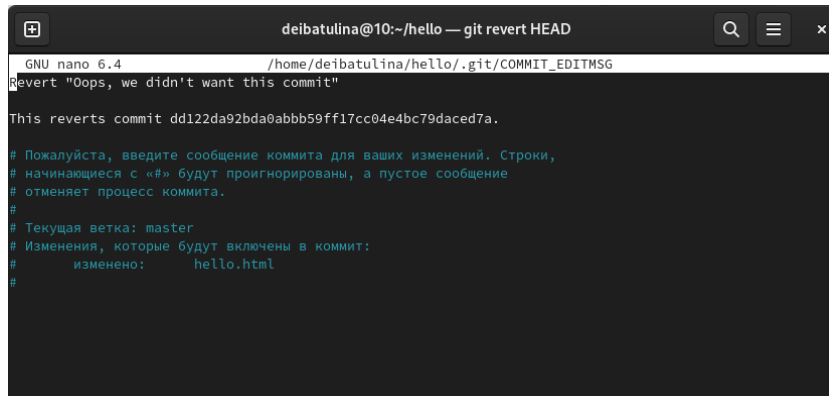


```
1 <html>
2 <head>
3 </head>
4 <body>
5 <h1>Hello, World!</h1>
6 <!-- This is an unwanted but committed change -->
7 </body>
8 </html>
```

Рис. 4.34: Шаг 8

```
[deibatulina@10 hello]$ git add hello.html
[deibatulina@10 hello]$ git commit -m "Oops, we didn't want this commit"
[master dd122da] Oops, we didn't want this commit
1 file changed, 6 insertions(+), 5 deletions(-)
[deibatulina@10 hello]$
```

Рис. 4.35: Шаг 9



```
deibatulina@10:~/hello — git revert HEAD
GNU nano 6.4 /home/deibatulina/hello/.git/COMMIT_EDITMSG
Revert "Oops, we didn't want this commit"

This reverts commit dd122da92bda0abbb59ff17cc04e4bc79daced7a.

# Пожалуйста, введите сообщение коммита для ваших изменений. Строки,
# начинающиеся с «#» будут проигнорированы, а пустое сообщение
# отменяет процесс коммита.
#
# Текущая ветка: master
# Изменения, которые будут включены в коммит:
#       изменено:      hello.html
#
```

Рис. 4.36: Шаг 10

```

[deibatulina@i0 hello]$ git log
commit 85a9527312db39bdeb2eb0a44acff01f6e66df08 (HEAD -> master)
Author: Darya Ibatulina <fdarisha@yandex.ru>
Date: Sun Feb 16 15:36:21 2025 +0300

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

    This reverts commit dd122da92bda0abb59ff17cc04e4bc79daced7a.

commit dd122da92bda0abb59ff17cc04e4bc79daced7a
Author: Darya Ibatulina <fdarisha@yandex.ru>
Date: Sun Feb 16 15:35:52 2025 +0300

    Oops, we didn't want this commit

commit 1f73ffe708ac6e559a7627fba474ded3e517cb3f (tag: v1)
Author: Darya Ibatulina <fdarisha@yandex.ru>
Date: Sun Feb 16 15:11:04 2025 +0300

    Added HTML header

commit ae12f0518c1aa9107e4e835345174f3654a96ab0 (tag: v1-beta)
Author: Darya Ibatulina <fdarisha@yandex.ru>
Date: Sun Feb 16 15:08:59 2025 +0300

    Added standard HTML page tags

commit eb9e461dc68c08fe014165c8b9bf35612dbc61fe
Author: Darya Ibatulina <fdarisha@yandex.ru>
Date: Sun Feb 16 15:02:02 2025 +0300

    "Added h1 tag"
...skipping...
commit 85a9527312db39bdeb2eb0a44acff01f6e66df08 (HEAD -> master)
Author: Darya Ibatulina <fdarisha@yandex.ru>
Date: Sun Feb 16 15:36:21 2025 +0300

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

    This reverts commit dd122da92bda0abb59ff17cc04e4bc79daced7a.

```

Рис. 4.37: Шаг 11

Чтобы коммиты удалялись и из истории ветки, существует более мощная команда `git reset` (рис. [4.38], [4.39], [4.40]).

```

[deibatulina@10 hello]$ git tag oops
[deibatulina@10 hello]$ git reset --hard v1
Указатель HEAD сейчас на коммите 1f73ffe Added HTML header
[deibatulina@10 hello]$ git log
commit 1f73ffe708ac6e559a7627fba474ded3e517cb3f (HEAD -> master, tag: v1)
Author: Darya Ibatulina <fdarisha@yandex.ru>
Date: Sun Feb 16 15:11:04 2025 +0300

    Added HTML header

commit ae12f0518c1aa9107e4e835345174f3654a96ab0 (tag: v1-beta)
Author: Darya Ibatulina <fdarisha@yandex.ru>
Date: Sun Feb 16 15:08:59 2025 +0300

    Added standard HTML page tags

commit eb9e461dc68c08fe014165c8b9bf35612dbc61fe
Author: Darya Ibatulina <fdarisha@yandex.ru>
Date: Sun Feb 16 15:02:02 2025 +0300

    "Added h1 tag"

commit ab5950644f692f714b320ed2a9d91a2f4e567118
Author: Darya Ibatulina <fdarisha@yandex.ru>
Date: Sun Feb 16 14:55:15 2025 +0300

    Initial Commit
[deibatulina@10 hello]$

```

Рис. 4.38: Иллар 1


```

[deibatulina@10 hello]$ git log --all
commit 85a9527312db39bdeb2eb0a44acff01f6e66df08 (tag: oops)
Author: Darya Ibatulina <fdarisha@yandex.ru>
Date: Sun Feb 16 15:36:21 2025 +0300

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

    This reverts commit dd122da92bda0abbb59ff17cc04e4bc79daced7a.

commit dd122da92bda0abbb59ff17cc04e4bc79daced7a
Author: Darya Ibatulina <fdarisha@yandex.ru>
Date: Sun Feb 16 15:35:52 2025 +0300

    Oops, we didn't want this commit

commit 1f73ffe708ac6e559a7627fba474ded3e517cb3f (HEAD -> master, tag: v1)
Author: Darya Ibatulina <fdarisha@yandex.ru>
Date: Sun Feb 16 15:11:04 2025 +0300

    Added HTML header

commit ae12f0518c1aa9107e4e835345174f3654a96ab0 (tag: v1-beta)
Author: Darya Ibatulina <fdarisha@yandex.ru>
Date: Sun Feb 16 15:08:59 2025 +0300

    Added standard HTML page tags

commit eb9e461dc68c08fe014165c8b9bf35612dbc61fe
Author: Darya Ibatulina <fdarisha@yandex.ru>
Date: Sun Feb 16 15:02:02 2025 +0300

    "Added h1 tag"

commit ab5950644f692f714b320ed2a9d91a2f4e567118
Author: Darya Ibatulina <fdarisha@yandex.ru>
Date: Sun Feb 16 14:55:15 2025 +0300

    Initial Commit
[deibatulina@10 hello]$

```

Рис. 4.39: Шаг 2

```

[deibatulina@10 hello]$ git tag -d oops
Метка «oops» удалена (была 85a9527)
[deibatulina@10 hello]$ git log --all
commit 1f73ffe708ac6e559a7627fba474ded3e517cb3f (HEAD -> master, tag: v1)
Author: Darya Ibatulina <fdarisha@yandex.ru>
Date: Sun Feb 16 15:11:04 2025 +0300

    Added HTML header

commit ae12f0518c1aa9107e4e835345174f3654a96ab0 (tag: v1-beta)
Author: Darya Ibatulina <fdarisha@yandex.ru>
Date: Sun Feb 16 15:08:59 2025 +0300

    Added standard HTML page tags

commit eb9e461dc68c08fe014165c8b9bf35612dbc61fe
Author: Darya Ibatulina <fdarisha@yandex.ru>
Date: Sun Feb 16 15:02:02 2025 +0300

    "Added h1 tag"

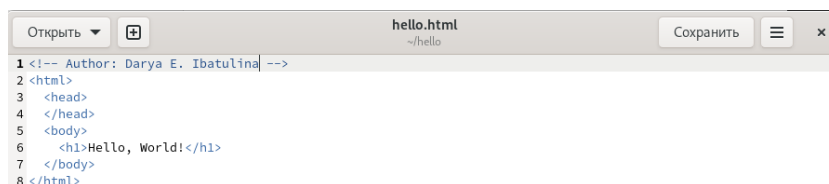
commit ab5950644f692f714b320ed2a9d91a2f4e567118
Author: Darya Ibatulina <fdarisha@yandex.ru>
Date: Sun Feb 16 14:55:15 2025 +0300

    Initial Commit
[deibatulina@10 hello]$

```

Рис. 4.40: Шаг 3

Добавление автора и электронной почты в файл приветствия (рис. [4.41], [4.42], [4.43], [4.44], [4.45]).



```

Открыть  hello.html  Сохранить  x
~/hello
1 <!-- Author: Darya E. Ibatulina -->
2 <html>
3 <head>
4 </head>
5 <body>
6 <h1>Hello, World!</h1>
7 </body>
8 </html>

```

Рис. 4.41: Шаг 1

```

[deibatulina@10 hello]$ git add hello.html
[deibatulina@10 hello]$ git commit -m "Add an author comment"
[master 4970e9a] Add an author comment
1 file changed, 6 insertions(+), 5 deletions(-)
[deibatulina@10 hello]$

```

Рис. 4.42: Шаг 2

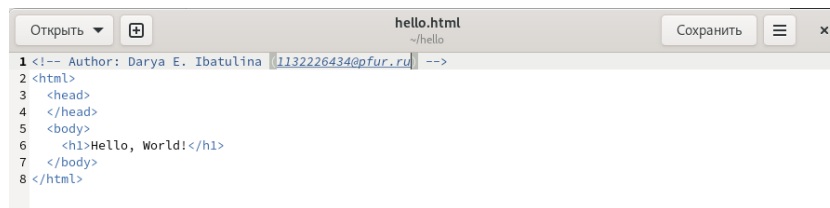


Рис. 4.43: Шаг 3

```
[deibatulina@10 hello]$ git add hello.html
[deibatulina@10 hello]$ git commit --amend -m "Add an author/email comment"
[master 1e4f07f] Add an author/email comment
Date: Sun Feb 16 15:47:31 2025 +0300
1 file changed, 6 insertions(+), 5 deletions(-)
[deibatulina@10 hello]$
```

Рис. 4.44: Шаг 4

```
[deibatulina@10 hello]$ git log
commit 1e4f07ff57574023c5819d1c4c7e240561cc63d9 (HEAD -> master)
Author: Darya Ibatulina <fdarisha@yandex.ru>
Date: Sun Feb 16 15:47:31 2025 +0300

    Add an author/email comment

commit 1f73ffe708ac6e559a7627fba474ded3e517cb3f (tag: v1)
Author: Darya Ibatulina <fdarisha@yandex.ru>
Date: Sun Feb 16 15:11:04 2025 +0300

    Added HTML header

commit ae12f0518c1aa9107e4e835345174f3654a96ab0 (tag: v1-beta)
Author: Darya Ibatulina <fdarisha@yandex.ru>
Date: Sun Feb 16 15:08:59 2025 +0300

    Added standard HTML page tags

commit eb9e461dc68c08fe014165c8b9bf35612dbc61fe
Author: Darya Ibatulina <fdarisha@yandex.ru>
Date: Sun Feb 16 15:02:02 2025 +0300

    "Added h1 tag"

commit ab5950644f692f714b320ed2a9d91a2f4e567118
Author: Darya Ibatulina <fdarisha@yandex.ru>
Date: Sun Feb 16 14:55:15 2025 +0300

    Initial Commit
[deibatulina@10 hello]$
```

Рис. 4.45: Шаг 5

Создаем новый каталог и копируем в него содержимое hello, отправляем в

репозиторий (рис. [4.46], [4.47]).

```
[deibatulina@10 hello]$ mkdir lib
[deibatulina@10 hello]$ git mv hello.html lib
[deibatulina@10 hello]$ git status
Текущая ветка: master
Изменения, которые будут включены в коммит:
  (используйте «git restore --staged <файл>...», чтобы убрать из индекса)
    переименовано: hello.html -> lib/hello.html
[deibatulina@10 hello]$
```

Рис. 4.46: Копирование

```
[deibatulina@10 hello]$ git commit -m "Moved hello.html to lib"
[master ca8acaf] Moved hello.html to lib
 1 file changed, 0 insertions(+), 0 deletions(-)
 rename hello.html => lib/hello.html (100%)
[deibatulina@10 hello]$
```

Рис. 4.47: Отправка изменений

Создаем новый файл, отправляем в репозиторий (рис. [4.48], [4.49]).



```
1 <html>
2   <body>
3     <iframe src="lib/hello.html" width="200" height="200" />
4   </body>
5 </html>
```

Рис. 4.48: Создание и редактирование файла

```
[deibatulina@10 hello]$ touch index.html
[deibatulina@10 hello]$ git add index.html
[deibatulina@10 hello]$ git commit -m "Added index.html"
[master 7e6fa11] Added index.html
 1 file changed, 5 insertions(+)
 create mode 100644 index.html
[deibatulina@10 hello]$
```

Рис. 4.49: Отправка изменений

Теперь при открытии index.html, я вижу кусок страницы hello в маленьком окошке (рис. [-4.50]).

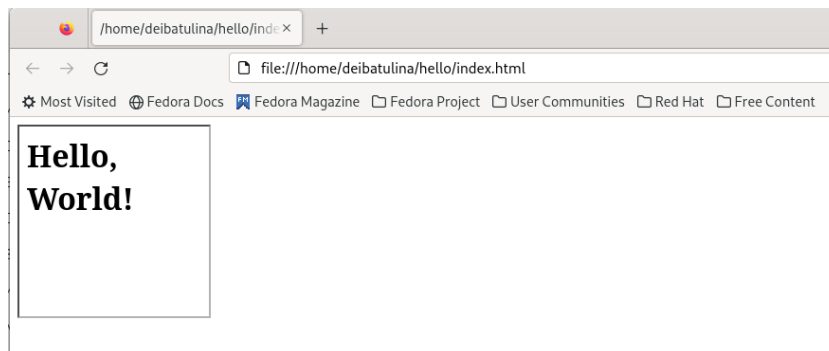


Рис. 4.50: Создание и редактирование файла

Углубляюсь в базу данных объектов, смотрю файлы конфига, ветки и теги, файлы HEAD (рис. [4.51], [4.52], [4.53], [4.54], [4.55]).

```
[deibatulina@10 hello]$ ls -C .git
branches      config        HEAD   index  logs    ORIG_HEAD
COMMIT_EDITMSG  description  hooks  info   objects packed-refs
[deibatulina@10 hello]$
```

Рис. 4.51: Шаг 1

```
[deibatulina@10 hello]$ ls -C .git/objects
01 1f 38 4b 70 7f 8e a4 ab ae ca d3 eb ff pack
1e 22 49 5f 7e 85 98 aa ac af cf dd f1 info
[deibatulina@10 hello]$
```

Рис. 4.52: Шаг 2

```
[deibatulina@10 hello]$ ls -C .git/objects/4b
c08b3766a6ef96f7f5a7be60b334ffb309cc5d
[deibatulina@10 hello]$
```

Рис. 4.53: Шаг 3

```
[deibatulina@10 hello]$ cat .git/config
[core]
    repositoryformatversion = 0
    filemode = true
    bare = false
    logallrefupdates = true
[deibatulina@10 hello]$
```

Рис. 4.54: Шаг 4

```
[deibatulina@10 hello]$ ls .git/refs
ls .git/refs/heads
ls .git/refs/tags
cat .git/refs/tags/v1
heads tags
master
v1 v1-beta
1f73ffe708ac6e559a7627fba474ded3e517cb3f
[deibatulina@10 hello]$
```

Рис. 4.55: Шаг 5

Работаю непосредственно с объектами git (рис. [4.56], [4.57], [4.58], [4.59], [4.60], [4.61]).

```
[deibatulina@10 hello]$ cat .git/HEAD
ref: refs/heads/master
[deibatulina@10 hello]$
```

Рис. 4.56: Шаг 1

```
[deibatulina@10 hello]$ git log --max-count=1
commit 7e6fa1dd9c7d323d31fdef8e74f92308eb6558f (HEAD -> master)
Author: Darya Ibatulina <fdarisha@yandex.ru>
Date: Sun Feb 16 16:02:11 2025 +0300

    Added index.html
[deibatulina@10 hello]$
```

Рис. 4.57: Шаг 2

```
[deibatulina@10 hello]$ git cat-file -t 7e6fa1
commit
[deibatulina@10 hello]$ git cat-file -p 7e6fa1
tree 4bc08b3766a6ef96f7f5a7be60b334ffb309cc5d
parent ca8acafe504d5519e02ad37c5f8a5697c798d2a7
author Darya Ibatulina <fdarisha@yandex.ru> 1739710931 +0300
committer Darya Ibatulina <fdarisha@yandex.ru> 1739710931 +0300

Added index.html
[deibatulina@10 hello]$
```

Рис. 4.58: Шаг 3

```
[deibatulina@10 hello]$ git cat-file -p 4bc08b3
100644 blob d3969cef6b037685ae71f943754de9c3cfaf28ec    index.html
040000 tree ff2de6beb630b954bc39d964f396636be420a641    lib
[deibatulina@10 hello]$
```

Рис. 4.59: Шаг 4

```
[deibatulina@10 hello]$ git cat-file -p ff2de6b
100644 blob f17e79d2eb598bb3e8da2fca49405375c1b22497    hello.html
[deibatulina@10 hello]$
```

Рис. 4.60: Шаг 5

```
[deibatulina@10 hello]$ git cat-file -p f17e79d
<!-- Author: Darya E. Ibatulina (1132226434@pfur.ru) -->
<html>
  <head>
  </head>
  <body>
    <h1>Hello, World!</h1>
  </body>
</html>
[deibatulina@10 hello]$
```

Рис. 4.61: Шаг 6

Добавляю файл стилей и новую ветку style (рис. [4.62], [4.63], [4.64], [4.65], [4.66]).

```
[deibatulina@10 hello]$ git checkout -b style
git status
Переключились на новую ветку «style»
Текущая ветка: style
нечего коммитить, нет изменений в рабочем каталоге
[deibatulina@10 hello]$
```

Рис. 4.62: Шаг 1



Рис. 4.63: Шаг 2

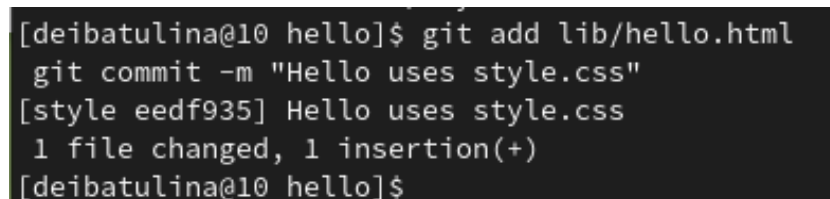
```
[deibatulina@10 hello]$ touch lib/style.css
[deibatulina@10 hello]$ git add lib/style.css
git commit -m "Added css stylesheet"
[style f9cc74b] Added css stylesheet
1 file changed, 3 insertions(+)
create mode 100644 lib/style.css
[deibatulina@10 hello]$
```

Рис. 4.64: Шаг 3



```
1 <!-- Author: Darya E. Ibatulina (1132226434@pfur.ru) -->
2 <html>
3 <head>
4 <link type="text/css" rel="stylesheet" media="all" href="style.css" />
5 </head>
6 <body>
7 <h1>Hello, World!</h1>
8 </body>
9 </html>
```

Рис. 4.65: Шаг 4



```
[deibatulina@10 hello]$ git add lib/hello.html
git commit -m "Hello uses style.css"
[style eedf935] Hello uses style.css
1 file changed, 1 insertion(+)
[deibatulina@10 hello]$
```

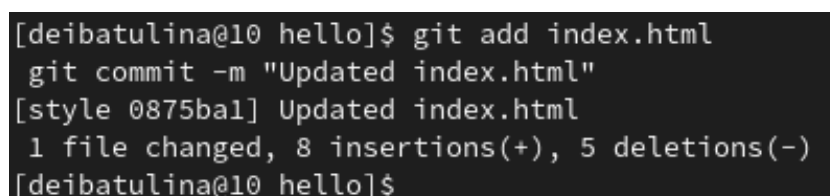
Рис. 4.66: Шаг 5

Изменяю файлы так, чтобы использовались стили style.css (рис. [4.67], [4.68], [4.69], [4.70], [4.71], [4.72], [4.73]).



```
1 <html>
2 <head>
3 <link type="text/css" rel="stylesheet" media="all" href="lib/style.css" />
4 </head>
5 <body>
6 <iframe src="lib/hello.html" width="200" height="200" />
7 </body>
8 </html>
```

Рис. 4.67: Шаг 1



```
[deibatulina@10 hello]$ git add index.html
git commit -m "Updated index.html"
[style 0875ba1] Updated index.html
1 file changed, 8 insertions(+), 5 deletions(-)
[deibatulina@10 hello]$
```

Рис. 4.68: Шаг 2

```
[deibatulina@10 hello]$ git log --all
commit 0875balaeb94f276d1c09f46b3f525709c22e041 (HEAD -> style)
Author: Darya Ibatulina <fdarisha@yandex.ru>
Date: Sun Feb 16 16:21:07 2025 +0300

    Updated index.html

commit eedf935407d2514c1e3f7d05b28fb2c02d7088ad
Author: Darya Ibatulina <fdarisha@yandex.ru>
Date: Sun Feb 16 16:19:06 2025 +0300

    Hello uses style.css

commit f9cc74b9dbe7729e036e6a0fee517f8a6676416f
Author: Darya Ibatulina <fdarisha@yandex.ru>
Date: Sun Feb 16 16:17:15 2025 +0300

    Added css stylesheet

commit 7e6fa1dd9c7d323d31fdef8e74f92308eb6558f (master)
Author: Darya Ibatulina <fdarisha@yandex.ru>
Date: Sun Feb 16 16:02:11 2025 +0300

    Added index.html

commit ca8acafe504d5519e02ad37c5f8a5697c798d2a7
Author: Darya Ibatulina <fdarisha@yandex.ru>
Date: Sun Feb 16 15:59:38 2025 +0300
```

Рис. 4.69: Шаг 3

```
[deibatulina@10 hello]$ git checkout master
cat lib/hello.html
Переключились на ветку «master»
<!-- Author: Darya E. Ibatulina (1132226434@pfur.ru) -->
<html>
  <head>
  </head>
  <body>
    <h1>Hello, World!</h1>
  </body>
</html>
[deibatulina@10 hello]$
```

Рис. 4.70: Шаг 4

```
[deibatulina@10 hello]$ git checkout style
cat lib/hello.html
Переключились на ветку «style»
<!-- Author: Darya E. Ibatulina (1132226434@pfur.ru) -->
<html>
  <head>
    <link type="text/css" rel="stylesheet" media="all" href="style.css" />
  </head>
  <body>
    <h1>Hello, World!</h1>
  </body>
</html>
[deibatulina@10 hello]$
```

Рис. 4.71: Шаг 5

```
[deibatulina@10 hello]$ git checkout master
Переключились на ветку «master»
[deibatulina@10 hello]$ echo "This is the Hello World example from the git tutorial." > README.md
[deibatulina@10 hello]$ git add README.md
[deibatulina@10 hello]$ git commit -m "Added README"
[master 0d39c8e] Added README
1 file changed, 1 insertion(+)
create mode 100644 README.md
[deibatulina@10 hello]$
```

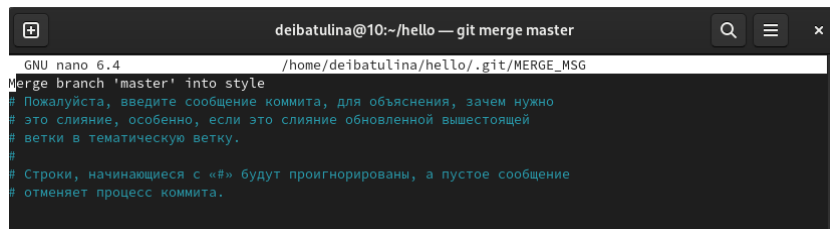
Рис. 4.72: Шаг 6

```
[deibatulina@10 hello]$ git log --graph --all
* commit 0d39c8e503fc4f5d7078b6cac72d07371c39f210 (HEAD -> master)
| Author: Darya Ibatulina <fdarisha@yandex.ru>
| Date: Sun Feb 16 16:24:21 2025 +0300
|
| Added README
|
+---* commit 0875balaeb94f276d1c09f46b3f525709c22e041 (style)
| Author: Darya Ibatulina <fdarisha@yandex.ru>
| Date: Sun Feb 16 16:21:07 2025 +0300
|
| Updated index.html
|
+---* commit eedf935407d2514c1e3f7d05b28fb2c02d7088ad
| Author: Darya Ibatulina <fdarisha@yandex.ru>
| Date: Sun Feb 16 16:19:06 2025 +0300
|
| Hello uses style.css
|
+---* commit f9cc74b9dbe7729e036e6a0fee517f8a6676416f
| Author: Darya Ibatulina <fdarisha@yandex.ru>
| Date: Sun Feb 16 16:17:15 2025 +0300
|
| Added css stylesheet
```

Рис. 4.73: Шаг 7

Выполняю слияние веток. Появляются конфликты, которые необходимо разрешить. Успешно справляюсь с данной задачей (рис. [4.74], [4.75], [4.76], [4.77],

[4.78], [4.79], [4.80], [4.81], [4.82], [4.83], [4.84], [4.85], [4.86], [4.87], [4.88], [4.89]).



The screenshot shows a terminal window with the title bar "deibatulina@10:~/hello — git merge master". The nano editor is open, editing a file named "/home/deibatulina/hello/.git/MERGE_MSG". The text in the editor is as follows:

```
GNU nano 6.4 /home/deibatulina/hello/.git/MERGE_MSG
Merge branch 'master' into style
# Пожалуйста, введите сообщение коммита, для объяснения, зачем нужно
# это слияние, особенно, если это слияние обновленной вышестоящей
# ветки в тематическую ветку.
#
# Строки, начинающиеся с «#» будут проигнорированы, а пустое сообщение
# отменяет процесс коммита.
```

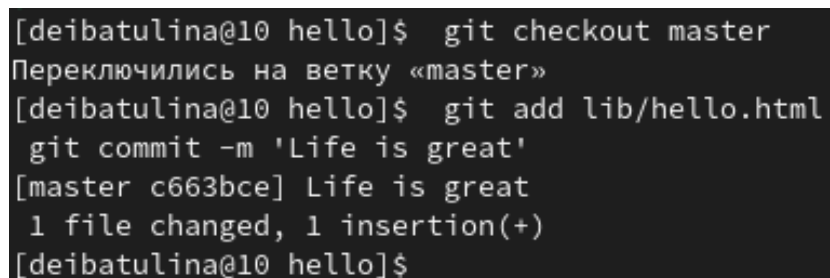
Рис. 4.74: Шаг 1



The screenshot shows a text editor window titled "hello.html" with the path "~/hello/lib". The editor contains the following HTML code:

```
1 <!-- Author: Darya E. Ibatulina (1132226434@pfur.ru) -->
2 <html>
3   <head>
4     <!-- no style -->
5   </head>
6   <body>
7     <h1>Hello, World!</h1>
8   </body>
9 </html>
```

Рис. 4.75: Шаг 2



The screenshot shows a terminal window with the following commands and output:

```
[deibatulina@10 hello]$ git checkout master
Переключились на ветку «master»
[deibatulina@10 hello]$ git add lib/hello.html
git commit -m 'Life is great'
[master c663bce] Life is great
1 file changed, 1 insertion(+)
[deibatulina@10 hello]$
```

Рис. 4.76: Шаг 3

```
[deibatulina@10 hello]$ git log --graph --all
* commit c663bce8daba6f975a07ffd4d90e27693807482f (HEAD -> master)
| Author: Darya Ibatulina <fdarisha@yandex.ru>
| Date: Sun Feb 16 16:29:00 2025 +0300
|
| Life is great
|
* commit fbdb81d985d431f44ad97b480790f978e7a20a81 (style)
| \ Merge: 0875ba1 0d39c8e
| / Author: Darya Ibatulina <fdarisha@yandex.ru>
| / Date: Sun Feb 16 16:26:24 2025 +0300
|
| Merge branch 'master' into style
|
* commit 0d39c8e503fc4f5d7078b6cac72d07371c39f210
| Author: Darya Ibatulina <fdarisha@yandex.ru>
| Date: Sun Feb 16 16:24:21 2025 +0300
|
| Added README
```

Рис. 4.77: Шаг 4

```
[deibatulina@10 hello]$ git checkout style
git merge master
Переключились на ветку «style»
Автослияние lib/hello.html
КОНФЛИКТ (содержимое): Конфликт слияния в lib/hello.html
Сбой автоматического слияния; исправьте конфликты, затем зафиксируйте результат.
[deibatulina@10 hello]$
```

Рис. 4.78: Шаг 5

```
Открыть  hello.html  Сохранить  x
~/hello/lib
1 <!-- Author: Darya E. Ibatulina (1132226434@pfur.ru) -->
2 <html>
3 <head>
4 <==== HEAD
5 <link type="text/css" rel="stylesheet" media="all" href="style.css" />
6 =====
7 <!-- no style -->
8 >>>>>> master
9 </head>
10 <body>
11 <h1>Hello, World!</h1>
12 </body>
13 </html>
```

Рис. 4.79: Шаг 6



Рис. 4.80: Шаг 7

```

[deibatulina@10 hello]$ git add lib/hello.html
git commit -m "Merged master fixed conflict."
[style c290dal] Merged master fixed conflict.
[deibatulina@10 hello]$

```

Рис. 4.81: Шаг 8

```

[deibatulina@10 hello]$ git checkout style
git log --graph
Уже на «style»
* commit c290dal652af70766c3c446172fbf75bc9b5e9a3 (HEAD -> style)
| Merge: fbdb81d c663bce
| Author: Darya Ibatulina <fdarisha@yandex.ru>
| Date: Sun Feb 16 16:32:43 2025 +0300
|
| Merged master fixed conflict.
|
* commit c663bce8daba6f975a07ffd4d90e27693807482f (master)
| Author: Darya Ibatulina <fdarisha@yandex.ru>
| Date: Sun Feb 16 16:29:00 2025 +0300
|
| Life is great
|
* commit fbdb81d985d431f44ad97b480790f978e7a20a81
| Merge: 0875ba1 0d39c8e
| Author: Darya Ibatulina <fdarisha@yandex.ru>
| Date: Sun Feb 16 16:26:24 2025 +0300
|
| Merge branch 'master' into style
|
* commit 0d39c8e503fc4f5d7078b6cac72d07371c39f210
| Author: Darya Ibatulina <fdarisha@yandex.ru>
| Date: Sun Feb 16 16:24:21 2025 +0300
|
| Added README
|
* commit 0875ba1aeb94f276d1c09f46b3f525709c22e041
| Author: Darya Ibatulina <fdarisha@yandex.ru>
| Date: Sun Feb 16 16:21:07 2025 +0300
|
| Updated index.html
...skipping...

```

Рис. 4.82: Шаг 9

```
[deibatulina@10 hello]$ git reset --hard 0875ba1
Указатель HEAD сейчас на коммите 0875ba1 Updated index.html
[deibatulina@10 hello]$
```

Рис. 4.83: Шаг 10

```
[deibatulina@10 hello]$ git log --graph --all
* commit c663bce8daba6f975a07ffd4d90e27693807482f (master)
  Author: Darya Ibatulina <fdarisha@yandex.ru>
  Date: Sun Feb 16 16:29:00 2025 +0300

    Life is great

* commit 0d39c8e503fc4f5d7078b6cac72d07371c39f210
  Author: Darya Ibatulina <fdarisha@yandex.ru>
  Date: Sun Feb 16 16:24:21 2025 +0300

    Added README

* commit 0875balaeb94f276d1c09f46b3f525709c22e041 (HEAD -> style)
  Author: Darya Ibatulina <fdarisha@yandex.ru>
  Date: Sun Feb 16 16:21:07 2025 +0300

    Updated index.html

* commit eedf935407d2514c1e3f7d05b28fb2c02d7088ad
  Author: Darya Ibatulina <fdarisha@yandex.ru>
  Date: Sun Feb 16 16:19:06 2025 +0300

    Hello uses style.css

* commit f9cc74b9dbe7729e036e6a0fee517f8a6676416f
  Author: Darya Ibatulina <fdarisha@yandex.ru>
  Date: Sun Feb 16 16:17:15 2025 +0300

    Added css stylesheet

* commit 7e6fa11dd9c7d323d31fdef8e74f92308eb6558f
  Author: Darya Ibatulina <fdarisha@yandex.ru>
  Date: Sun Feb 16 16:02:11 2025 +0300

    Added index.html
```

Рис. 4.84: Шаг 11

```
[deibatulina@10 hello]$ git checkout master
git log --graph
Переключились на ветку «master»
* commit c663bce8daba6f975a07ffd4d90e27693807482f (HEAD -> master)
| Author: Darya Ibatulina <fdarisha@yandex.ru>
| Date: Sun Feb 16 16:29:00 2025 +0300
|
| Life is great
|
* commit 0d39c8e503fc4f5d7078b6cac72d07371c39f210
| Author: Darya Ibatulina <fdarisha@yandex.ru>
| Date: Sun Feb 16 16:24:21 2025 +0300
|
| Added README
|
* commit 7e6fa11dd9c7d323d31fdef8e74f92308eb6558f
| Author: Darya Ibatulina <fdarisha@yandex.ru>
| Date: Sun Feb 16 16:02:11 2025 +0300
|
| Added index.html
|
* commit ca8acafe504d5519e02ad37c5f8a5697c798d2a7
| Author: Darya Ibatulina <fdarisha@yandex.ru>
| Date: Sun Feb 16 15:59:38 2025 +0300
|
| Moved hello.html to lib
|
* commit 1e4f07ff57574023c5819d1c4c7e240561cc63d9
| Author: Darya Ibatulina <fdarisha@yandex.ru>
| Date: Sun Feb 16 15:47:31 2025 +0300
|
| Add an author/email comment
```

Рис. 4.85: Шаг 12


```

[deibatulina@10 hello]$ git reset --hard 0d39c8e
git log --graph --all
Указатель HEAD сейчас на коммите 0d39c8e Added README
* commit 0d39c8e503fc4f5d7078b6cac72d07371c39f210 (HEAD -> master)
| Author: Darya Ibatulina <fdarisha@yandex.ru>
| Date: Sun Feb 16 16:24:21 2025 +0300
|
| Added README
|
* commit 0875balaeb94f276d1c09f46b3f525709c22e041 (style)
| Author: Darya Ibatulina <fdarisha@yandex.ru>
| Date: Sun Feb 16 16:21:07 2025 +0300
|
| Updated index.html
|
* commit eedf935407d2514c1e3f7d05b28fb2c02d7088ad
| Author: Darya Ibatulina <fdarisha@yandex.ru>
| Date: Sun Feb 16 16:19:06 2025 +0300
|
| Hello uses style.css
|
* commit f9cc74b9dbe7729e036e6a0fee517f8a6676416f
| / Author: Darya Ibatulina <fdarisha@yandex.ru>
| Date: Sun Feb 16 16:17:15 2025 +0300
|
| Added css stylesheet
|
* commit 7e6fa11dd9c7d323d31fdef8e74f92308eb6558f
| Author: Darya Ibatulina <fdarisha@yandex.ru>
| Date: Sun Feb 16 16:02:11 2025 +0300
|
| Added index.html
|
* commit ca8acafe504d5519e02ad37c5f8a5697c798d2a7
| Author: Darya Ibatulina <fdarisha@yandex.ru>
| Date: Sun Feb 16 15:59:38 2025 +0300
|
| Moved hello.html to lib

```

Рис. 4.86: Шаг 13

```

[deibatulina@10 hello]$ git checkout style
git rebase master
git log --graph
Переключились на ветку «style»
Успешно перемещён и обновлён refs/heads/style.
* commit f4b5a61745eceb40214ac2197cdb9ba7dee08b2d (HEAD -> style)
| Author: Darya Ibatulina <fdarisha@yandex.ru>
| Date: Sun Feb 16 16:21:07 2025 +0300
|
| Updated index.html
|
* commit 610e035e0ae25d195afb401f15fc1f4c2c9a48e4
| Author: Darya Ibatulina <fdarisha@yandex.ru>
| Date: Sun Feb 16 16:19:06 2025 +0300
|
| Hello uses style.css
|
* commit flcb644759a740ca17c5ab49d68ddc58a545fc8b
| Author: Darya Ibatulina <fdarisha@yandex.ru>
| Date: Sun Feb 16 16:17:15 2025 +0300
|
| Added css stylesheet
|
* commit 0d39c8e503fc4f5d7078b6cac72d07371c39f210 (master)
| Author: Darya Ibatulina <fdarisha@yandex.ru>
| Date: Sun Feb 16 16:24:21 2025 +0300
|
| Added README
|
* commit 7e6fa11dd9c7d323d31fdef8e74f92308eb6558f
| Author: Darya Ibatulina <fdarisha@yandex.ru>
| Date: Sun Feb 16 16:02:11 2025 +0300
|
| Added index.html
|
* commit ca8acafe504d5519e02ad37c5f8a5697c798d2a7
| Author: Darya Ibatulina <fdarisha@yandex.ru>
| Date: Sun Feb 16 15:59:38 2025 +0300
|
| Moved hello.html to lib

```

Рис. 4.87: Шаг 14

```

[deibatulina@10 hello]$ git checkout master
git merge style
Переключились на ветку «master»
Обновление 0d39c8e..f4b5a61
Fast-forward
 index.html      | 13 ++++++-----
 lib/hello.html  |  1 +
 lib/style.css   |  3 +++
 3 files changed, 12 insertions(+), 5 deletions(-)
 create mode 100644 lib/style.css
[deibatulina@10 hello]$

```

Рис. 4.88: Шаг 15

```

[deibatulina@10 hello]$ git log
commit f4b5a61745eceb40214ac2197cdb9ba7dee08b2d (HEAD -> master, style)
Author: Darya Ibatulina <fdarisha@yandex.ru>
Date:   Sun Feb 16 16:21:07 2025 +0300

    Updated index.html

```

Рис. 4.89: Шаг 16

Клонируем один репозиторий в другой созданный (рис. [4.90], [4.91], [4.92], [4.93]).

```

[deibatulina@10 hello]$ cd ..
[deibatulina@10 ~]$ pwd
/home/deibatulina
[deibatulina@10 ~]$ ls
backup      pandoc-2.19.2-linux-amd64.tar.gz  Видео      Музыка
bin         pandoc-crossref                   'Внешний курс инфобез'  Общедоступные
hello      pandoc-crossref.1                 Документы   'Рабочий стол'
lab07.sh~  pandoc-crossref-Linux.tar.xz      Загрузки    Шаблоны
pandoc-2.19.2  work                               Изображения
[deibatulina@10 ~]$

```

Рис. 4.90: Шаг 1

```

[deibatulina@10 ~]$ git clone hello cloned_hello
ls
Клонирование в «cloned_hello»...
готово.
backup      pandoc-2.19.2                   work                               Изображения
bin         pandoc-2.19.2-linux-amd64.tar.gz Видео                             Музыка
cloned_hello pandoc-crossref                 'Внешний курс инфобез'  Общедоступные
hello      pandoc-crossref.1                 Документы   'Рабочий стол'
lab07.sh~  pandoc-crossref-Linux.tar.xz      Загрузки    Шаблоны
[deibatulina@10 ~]$

```

Рис. 4.91: Шаг 2

```
[deibatulina@10 ~]$ cd cloned_hello
ls
index.html  lib  README.md
[deibatulina@10 cloned_hello]$
```

Рис. 4.92: Шаг 3

```
[deibatulina@10 cloned_hello]$ git log --all
commit f4b5a61745eceb40214ac2197cdb9ba7dee08b2d (HEAD -> master, origin/style, origin/master, origin/H
EAD)
Author: Darya Ibatulina <fdarisha@yandex.ru>
Date: Sun Feb 16 16:21:07 2025 +0300

    Updated index.html

commit 610e035e0ae25d195afb401f15fc1f4c2c9a48e4
Author: Darya Ibatulina <fdarisha@yandex.ru>
Date: Sun Feb 16 16:19:06 2025 +0300

    Hello uses style.css

commit f1cb644759a740ca17c5ab49d68ddc58a545fc8b
Author: Darya Ibatulina <fdarisha@yandex.ru>
Date: Sun Feb 16 16:17:15 2025 +0300

    Added css stylesheet

commit 0d39c8e503fc4f5d7078b6cac72d07371c39f210
Author: Darya Ibatulina <fdarisha@yandex.ru>
Date: Sun Feb 16 16:24:21 2025 +0300

    Added README

commit 7e6fa1dd9c7d323d31fdef8e74f92308eb6558f
Author: Darya Ibatulina <fdarisha@yandex.ru>
Date: Sun Feb 16 16:02:11 2025 +0300

    Added index.html
```

Рис. 4.93: Шаг 4

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. 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) рис. [4.94]).

```
[deibatulina@10 cloned_hello]$ git remote
origin
[deibatulina@10 cloned_hello]$ git remote show origin
* внешний репозиторий origin
  URL для извлечения: /home/deibatulina/hello
  URL для отправки: /home/deibatulina/hello
  HEAD ветка: master
  Внешние ветки:
    master отслеживается
    style отслеживается
  Локальная ветка, настроенная для «git pull»:
    master будет слита с внешней веткой master
  Локальная ссылка, настроенная для «git push»:
    master будет отправлена в master (уже актуальна)
[deibatulina@10 cloned_hello]$
```

Рис. 4.94: git remote, origin

Посмотрим все ветки (в том числе и удаленные) рис. [4.95]).

```
[deibatulina@10 cloned_hello]$ git branch
* master
[deibatulina@10 cloned_hello]$ git branch -a
* master
  remotes/origin/HEAD -> origin/master
  remotes/origin/master
  remotes/origin/style
[deibatulina@10 cloned_hello]$
```

Рис. 4.95: Просмотр всех веток (в том числе и удаленных)

Клонируем один репозиторий в другой созданный (рис. [4.96], [4.97], [4.98], [4.99], [4.100], [4.101]).

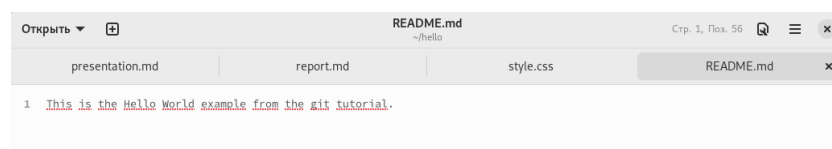


Рис. 4.96: Шаг 1

```
[deibatulina@10 hello]$ git add README.md
[deibatulina@10 hello]$ git commit -m "Changed README in original repo"
[master 0982024] Changed README in original repo
1 file changed, 1 insertion(+), 1 deletion(-)
[deibatulina@10 hello]$
```

Рис. 4.97: Шаг 2

```
[deibatulina@10 hello]$ git add README.md
[deibatulina@10 hello]$ git commit -m "Changed README in original repo"
[master 0982024] Changed README in original repo
1 file changed, 1 insertion(+), 1 deletion(-)
[deibatulina@10 hello]$ cd ../cloned_hello
[deibatulina@10 cloned_hello]$ git fetch
[deibatulina@10 cloned_hello]$ git log --all
remote: Перечисление объектов: 5, готово.
remote: Подсчет объектов: 100% (5/5), готово.
remote: Сжатие объектов: 100% (3/3), готово.
remote: Всего 3 (изменений 0), повторно использовано 0 (изменений 0), повторно использовано пакетов 0
Распаковка объектов: 100% (3/3), 360 байтов | 360.00 КиБ/с, готово.
Из /home/deibatulina/hello
f4b5a61..0982024 master -> origin/master
commit 09820241514c917f2fcd4acd4475f20b3dc45c7f (origin/master, origin/HEAD)
Author: Darya Ibatulina <fdarisha@yandex.ru>
Date: Sun Feb 16 16:51:42 2025 +0300

    Changed README in original repo

commit f4b5a61745eceb40214ac2197cdb9ba7dee08b2d (HEAD -> master, origin/style)
Author: Darya Ibatulina <fdarisha@yandex.ru>
Date: Sun Feb 16 16:21:07 2025 +0300

    Updated index.html

commit 610e035e0ae25d195afb401f15fc1f4c2c9a48e4
Author: Darya Ibatulina <fdarisha@yandex.ru>
Date: Sun Feb 16 16:19:06 2025 +0300

    Hello uses style.css

commit f1cb644759a740ca17c5ab49d68ddc58a545fc8b
Author: Darya Ibatulina <fdarisha@yandex.ru>
Date: Sun Feb 16 16:17:15 2025 +0300

    Added css stylesheet
```

Рис. 4.98: Шаг 3

```
[deibatulina@10 cloned_hello]$ cat README.md
This is the Hello World example from the git tutorial.
[deibatulina@10 cloned_hello]$
```

Рис. 4.99: Шаг 4

```
[deibatulina@10 cloned_hello]$ cat README.md
This is the Hello World example from the git tutorial.
[deibatulina@10 cloned_hello]$ git merge origin/master
Обновление f4b5a61..0982024
Fast-forward
 README.md | 2 +-
 1 file changed, 1 insertion(+), 1 deletion(-)
[deibatulina@10 cloned_hello]$ cat README.md
This is the Hello World example from the git tutorial.
[deibatulina@10 cloned_hello]$ git pull
Уже актуально.
[deibatulina@10 cloned_hello]$
```

Рис. 4.100: Шаг 5

```
[deibatulina@10 cloned_hello]$ git branch --track style origin/style
branch 'style' set up to track 'origin/style'.
[deibatulina@10 cloned_hello]$ git branch -a
* master
  style
remotes/origin/HEAD -> origin/master
remotes/origin/master
remotes/origin/style
[deibatulina@10 cloned_hello]$ git log --max-count=2
commit 09820241514c917f2fcd4acd4475f20b3dc45c7f (HEAD -> master, origin/master, origin/HEAD)
Author: Darya Ibatulina <fdarisha@yandex.ru>
Date: Sun Feb 16 16:51:42 2025 +0300

    Changed README in original repo

commit f4b5a61745eceb40214ac2197cdb9ba7dee08b2d (origin/style, style)
Author: Darya Ibatulina <fdarisha@yandex.ru>
Date: Sun Feb 16 16:21:07 2025 +0300

    Updated index.html
[deibatulina@10 cloned_hello]$
```

Рис. 4.101: Шаг 6

Создадим чистый репозиторий. Сейчас мы находимся в рабочем (домашнем) 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 (рис. [4.102]).

```
[deibatulina@10 cloned_hello]$ cd ..
git clone --bare hello hello.git
ls hello.git
Клонирование в голый репозиторий «hello.git»...
готово.
branches config description HEAD hooks info objects packed-refs refs
[deibatulina@10 ~]$
```

Рис. 4.102: Чистый репозиторий

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. Отредактирую файл README.md и сделаю коммит (рис. [4.103], [4.104], [4.105]).

```
[deibatulina@10 ~]$ cd hello
git remote add shared ../hello.git
[deibatulina@10 hello]$
```

Рис. 4.103: Переход в каталог hello

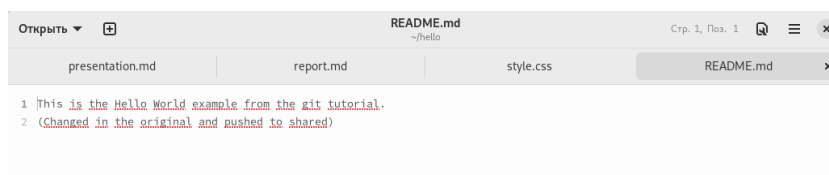


Рис. 4.104: Файл README.md

```
[deibatulina@10 hello]$ git add README.md
[deibatulina@10 hello]$ git commit -m "Added shared comment to readme"
[master 476163a] Added shared comment to readme
1 file changed, 2 insertions(+), 1 deletion(-)
[deibatulina@10 hello]$
```

Рис. 4.105: Коммит

Отправляю изменения в общий репозиторий (рис. [4.106]).

```
[deibatulina@10 hello]$ git push shared master
Перечисление объектов: 5, готово.
Подсчет объектов: 100% (5/5), готово.
При сжатии изменений используется до 4 потоков
Сжатие объектов: 100% (3/3), готово.
Запись объектов: 100% (3/3), 410 байтов | 27.00 КиБ/с, готово.
Всего 3 (изменений 0), повторно использовано 0 (изменений 0), повторно использовано пакетов 0
To ../hello.git
0982024..476163a master -> master
[deibatulina@10 hello]$
```

Рис. 4.106: Отправка изменений в общий репозиторий

Переключаемся в клонированный репозиторий и извлекаем изменения из общего репозитория (рис. [4.107]).


```

[deibatulina@10 hello]$ cd ../cloned_hello
[deibatulina@10 cloned_hello]$ git remote add shared ../hello.git
git branch --track shared master
git pull shared master
cat README.md
branch 'shared' set up to track 'master'.
remote: Перечисление объектов: 5, готово.
remote: Подсчет объектов: 100% (5/5), готово.
remote: Сжатие объектов: 100% (3/3), готово.
remote: Всего 3 (изменений 0), повторно использовано 0 (изменений 0), повторно использовано пакетов 0
Распаковка объектов: 100% (3/3), 390 байтов | 390.00 КиБ/с, готово.
Из ../hello
* branch          master      -> FETCH_HEAD
* [новая ветка]   master      -> shared/master
Обновление 0982024..476163a
Fast-forward
 README.md | 3 ++-
 1 file changed, 2 insertions(+), 1 deletion(-)
This is the Hello World example from the git tutorial.
(Changed in the original and pushed to shared)
[deibatulina@10 cloned_hello]$

```

Рис. 4.107: Извлечение изменений из общего репозитория в клонированный

5 Выводы

В результате проделывания лабораторной работы я разобралась с тем, как работает гит и его команды. Научилась отменять коммиты не только из репозитория, но и из истории веток. Вспомнила основные команды гита, уже изученные мною, и применила полученные знания на практике, создав тестовый репозиторий.

Список литературы

1. Кулябов Д.С., Королькова А.В. Руководство к лабораторной работе №1. Математическое моделирование. - 2025. — 31 с.