

Работа с git

Лабораторная работа № 1

Шулуужук Айраана НПИбд-02-22

Содержание

1	Цель работы	6
2	Выполнение лабораторной работы	7
3	Выводы	35

Список иллюстраций

2.1	имя пользователя и его электронная почта	7
2.2	параметры установки окончаний строк и отображения unicode	7
2.3	создание рабочего каталога и необходимых файлов	7
2.4	просмотр состояние репозитория	8
2.5	файл hello.html	8
2.6	проверка состояние каталога	8
2.7	индексация изменений	9
2.8	файл hello.html	9
2.9	файл hello.html	9
2.10	проверка текущего статуса	10
2.11	произведение коммита проиндексированного изменения .	10
2.12	список произведенных изменений	11
2.13	варианты отображение лога	11
2.14	хэш для первого коммита	12
2.15	последняя версия в ветке master	12
2.16	создание тега первой версии	13
2.17	файл hello.html	13
2.18	переключение между двумя отмеченными версиями	14
2.19	отмена локальных изменений до индексации	14
2.20	отмена изменений в рабочем каталоге	14
2.21	изменение файла и проиндексирование изменений	15
2.22	сброс буферной зоны	15
2.23	файл hello.html	16
2.24	выполнение коммита с новыми изменениями, отменяющими предыдущие	16
2.25	брос коммитов к предшествующим коммиту Oops	17
2.26	удаление тега oops	17
2.27	файл hello.html	18
2.28	файл hello.html	18
2.29	изменение предыдущего коммита	18
2.30	создание каталога	19
2.31	коммит в новый каталог	19
2.32	файл index.html	19
2.33	выполнение коммита	19
2.34	index.html	20
2.35	каталог .git	20
2.36	файл конфигурации	21

2.37 ветки и теги	21
2.38 поиск последнего коммита	22
2.39 вывод дерева каталогов	22
2.40 создание новой ветки	23
2.41 файл hello.html	23
2.42 index.html	23
2.43 навигация по веткам	24
2.44 Переключение на ветку master	24
2.45 переход к ветке style	25
2.46 просмотр текущих веток	25
2.47 слияние master с веткой style	26
2.48 разрешение конфликта	26
2.49 сброс ветки style	27
2.50 сброс ветки style и проверка	28
2.51 слияние style в master	28
2.52 переход в рабочий каталог и создание клона репозитория .	29
2.53 просмотр клонированного репозитория	29
2.54 просмотр состояния репозитория	30
2.55 список удаленных веток	30
2.56 внесение изменения в оригинальный репозиторий hello . .	31
2.57 извлечение изменений	31
2.58 слияние извлеченных изменений	32
2.59 добавление локальной ветки, которая отслеживает удален- ную ветку	32
2.60 создание чистого репозитория	33
2.61 добавление удаленного репозитория и отправка изменений	33
2.62 извлечение общих изменений	34

Список таблиц

1 Цель работы

Научиться пользоваться с git и разобраться с его особенностями.

2 Выполнение лабораторной работы

Выполним следующие команды, чтобы git узнал наше имя и электронную почту (рис. 2.1)

```
[avshuluuzhuk@fedora ~]$ ^C
[avshuluuzhuk@fedora ~]$ git config --global user.name
avshuluuzhuk
[avshuluuzhuk@fedora ~]$ git config --global user.name "avshuluuzhuk"
[avshuluuzhuk@fedora ~]$ git config --global user.email
1132221890@pfur.ru
[avshuluuzhuk@fedora ~]$ git config --global user.email "1132221890@pfur.ru"
```

Рис. 2.1: имя пользователя и его электронная почта

Установим параметры установки окончаний строк и отображения unicode (рис. 2.2)

```
[avshuluuzhuk@fedora ~]$ git config --global core.autocrlf input
[avshuluuzhuk@fedora ~]$ git config --global core.safecrlf true
[avshuluuzhuk@fedora ~]$ git config --global core.quotePath off
[avshuluuzhuk@fedora ~]$ -
```

Рис. 2.2: параметры установки окончаний строк и отображения unicode

Начнем работу в пустом рабочем каталоге с создания пустого каталога с именем hello, затем войдем в него и создадим там файл с именем hello.html (рис. 2.3)

```
[avshuluuzhuk@fedora ~]$ mkdir hello
[avshuluuzhuk@fedora ~]$ cd hello
[avshuluuzhuk@fedora hello]$ touch hello.html
[avshuluuzhuk@fedora hello]$ echo "Hello, World!" > hello.html
[avshuluuzhuk@fedora hello]$
```

Рис. 2.3: создание рабочего каталога и необходимых файлов

Создадим репозиторий из этого каталога и добавим файл в него. Далее проверим состояние репозитория (рис. 2.4).

```
[avshuluuzhuk@fedora hello]$ git init
Инициализирован пустой репозиторий Git в /home/avshuluuzhuk/hello/.git/
[avshuluuzhuk@fedora hello]$ git add hello.html
[avshuluuzhuk@fedora hello]$ git commit -m "Initial Commit"
[master (корневой коммит) c61aa06] Initial Commit
1 file changed, 1 insertion(+)
create mode 100644 hello.html
[avshuluuzhuk@fedora hello]$ git status
Текущая ветка: master
нечего коммитить, нет изменений в рабочем каталоге
[avshuluuzhuk@fedora hello]$
```

Рис. 2.4: просмотр состояние репозитория

Добавим HTML-теги к нашему приветствию. Изменим содержимое файла hello.html (рис. 2.5). Проверим состояние рабочего каталога (рис. 2.6).

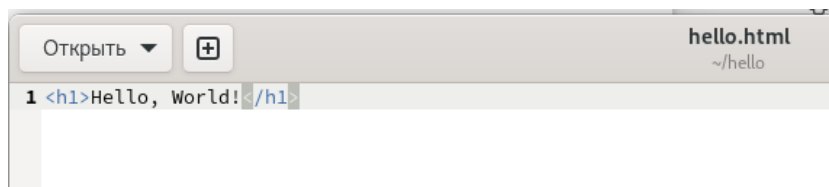
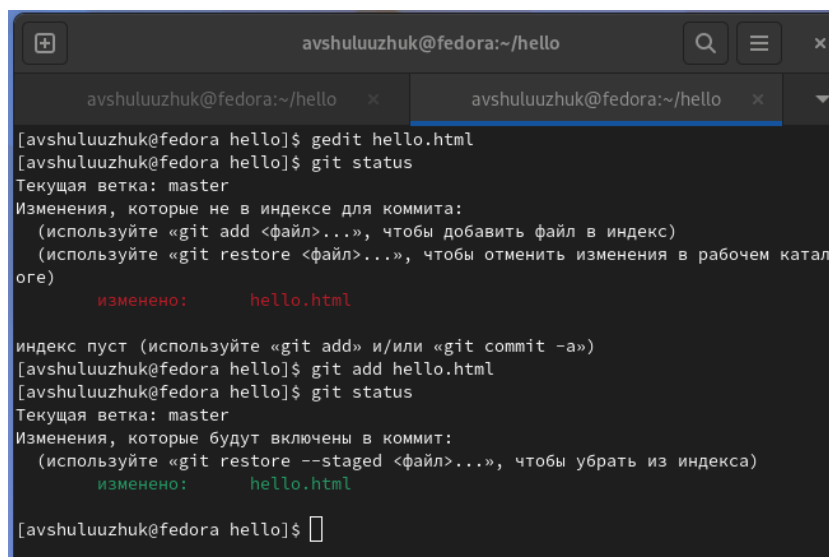


Рис. 2.5: файл hello.html



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

Рис. 2.6: проверка состояние каталога

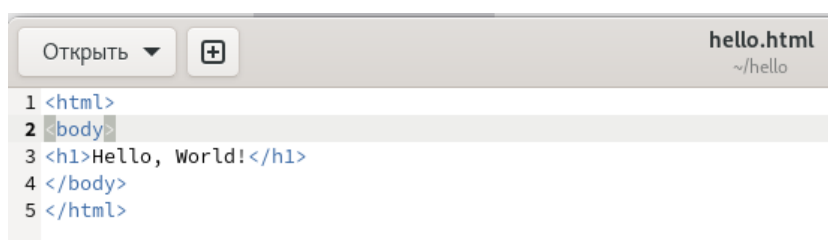
Теперь выполним команду `git`, чтобы проиндексировать изменения. Проверим его состояние. Далее сделаем коммит и проверим состояние (рис. 2.7).

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

[avshuluuzhuk@fedora hello]$ git commit
[master fea39e3] Added h1 tag
 1 file changed, 1 insertion(+), 1 deletion(-)
[avshuluuzhuk@fedora hello]$ git status
Текущая ветка: master
ничего коммитить, нет изменений в рабочем каталоге
[avshuluuzhuk@fedora hello]$
```

Рис. 2.7: индексация изменений

Изменим страницу «Hello, World», чтобы она содержала стандартные теги `html` и `body`. И добавим это изменение в индекс `git` (рис. 2.8).



```
1 <html>
2 <body>
3 <h1>Hello, World!</h1>
4 </body>
5 </html>
```

Рис. 2.8: файл `hello.html`

Теперь добавим заголовки HTML (секцию `head`) к странице «Hello, World» (рис. 2.9).



```
1 <html>
2 <head>
3 </head>
4 <body>
5 <h1>Hello, World!</h1>
6 </body>
7 </html>
```

Рис. 2.9: файл `hello.html`

Проверим текущий статус (рис. 2.10).

```

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

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

[avshuluuzhuk@fedora hello]$

```

Рис. 2.10: проверка текущего статуса

Произведем коммит проиндексированного изменения (значение по умолчанию), а затем еще раз проверим состояние. Состояние команды говорит о том, что `hello.html` имеет незафиксированные изменения, но уже не в буферной зоне. Теперь добавим второе изменение в индекс, а затем проверим состояние с помощью команды `git status`. Второе изменение было проиндексировано и готово к коммиту. Сделаем коммит второго изменения (рис. 2.11).

```

[avshuluuzhuk@fedora hello]$ git commit -m "Added standard HTML page tags"
[master c7e077c] Added standard HTML page tags
 1 file changed, 4 insertions(+)
[avshuluuzhuk@fedora 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»)
[avshuluuzhuk@fedora hello]$ git add .
[avshuluuzhuk@fedora hello]$ git status
Текущая ветка: master
Изменения, которые будут включены в коммит:
  (используйте «git restore --staged <файл>...», чтобы убрать из индекса)
    изменено:      hello.html

[avshuluuzhuk@fedora hello]$

```

Рис. 2.11: произведение коммита проиндексированного изменения

Получим список произведенных изменений (рис. 2.12).

```
avshuluuzhuk@fedora:~/hello
[avshuluuzhuk@fedora hello]$ git log
commit de2c7bbc296e82228e9d41f6e7976c95879c0470 (HEAD -> master)
Author: avshuluuzhuk <1132221890@pfur.ru>
Date: Tue Feb 11 16:36:57 2025 +0300

    Added HTML header

commit c7e077cd860e1a13b58dd6e7c163f4fff3411ef3
Author: avshuluuzhuk <1132221890@pfur.ru>
Date: Tue Feb 11 16:35:22 2025 +0300

    Added standard HTML page tags

commit fea39e31dd4c63e9253fc7062c2b241462e92bdb
Author: avshuluuzhuk <1132221890@pfur.ru>
Date: Tue Feb 11 16:27:35 2025 +0300

    Added h1 tag

commit c61aa0629f6def7b21799ee051a471f16e0c0c3c
Author: avshuluuzhuk <1132221890@pfur.ru>
Date: Tue Feb 11 16:21:06 2025 +0300

    Initial Commit
```

Рис. 2.12: список произведенных изменений

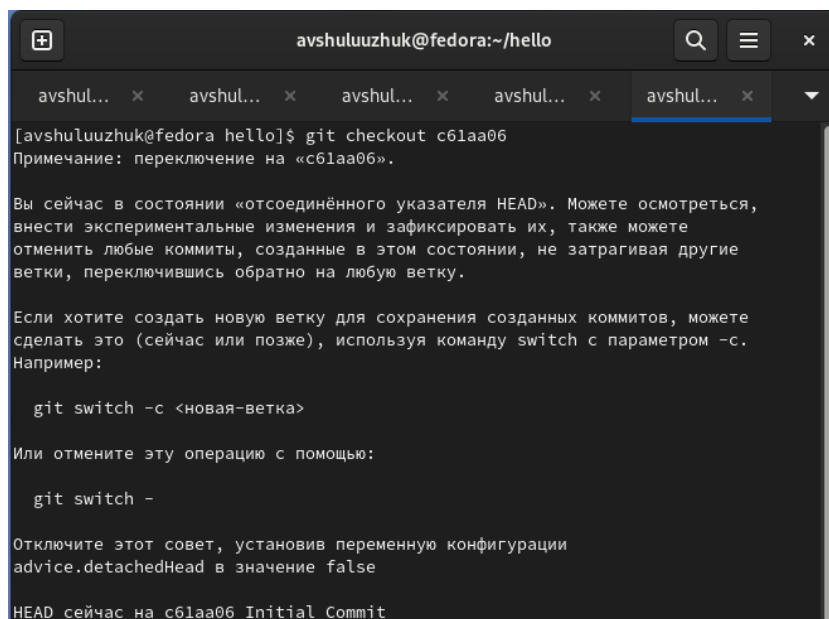
Варианты отображения лога (рис. 2.13).

```
avshuluuzhuk@fedora:~/hello
[avshuluuzhuk@fedora hello]$ git log --pretty=oneline --max-count=2
de2c7bbc296e82228e9d41f6e7976c95879c0470 (HEAD -> master) Added HTML header
c7e077cd860e1a13b58dd6e7c163f4fff3411ef3 Added standard HTML page tags
[avshuluuzhuk@fedora hello]$ git log --pretty=oneline --since='5 minutes ago'
de2c7bbc296e82228e9d41f6e7976c95879c0470 (HEAD -> master) Added HTML header
c7e077cd860e1a13b58dd6e7c163f4fff3411ef3 Added standard HTML page tags
[avshuluuzhuk@fedora hello]$ git log --pretty=oneline --until='5 minutes ago'
fea39e31dd4c63e9253fc7062c2b241462e92bdb Added h1 tag
c61aa0629f6def7b21799ee051a471f16e0c0c3c Initial Commit
[avshuluuzhuk@fedora hello]$ git log --pretty=oneline --author=<your name>
bash: синтаксическая ошибка рядом с неожиданным маркером «newline»
[avshuluuzhuk@fedora hello]$ git log --pretty=oneline --author=<avshuluuzhuk>
bash: синтаксическая ошибка рядом с неожиданным маркером «newline»
[avshuluuzhuk@fedora hello]$ git log --pretty=oneline --author=avshuluuzhuk
de2c7bbc296e82228e9d41f6e7976c95879c0470 (HEAD -> master) Added HTML header
c7e077cd860e1a13b58dd6e7c163f4fff3411ef3 Added standard HTML page tags
fea39e31dd4c63e9253fc7062c2b241462e92bdb Added h1 tag
c61aa0629f6def7b21799ee051a471f16e0c0c3c Initial Commit
[avshuluuzhuk@fedora hello]$ git log --pretty=oneline --all
de2c7bbc296e82228e9d41f6e7976c95879c0470 (HEAD -> master) Added HTML header
c7e077cd860e1a13b58dd6e7c163f4fff3411ef3 Added standard HTML page tags
fea39e31dd4c63e9253fc7062c2b241462e92bdb Added h1 tag
c61aa0629f6def7b21799ee051a471f16e0c0c3c Initial Commit
[avshuluuzhuk@fedora hello]$
```

Рис. 2.13: варианты отображение лога

Изучим данные лога и найдем хэш для первого коммита. Используем

этот хэш-код (достаточно первых 7 знаков). Затем проверим содержимое файла `hello.html` (рис. 2.14).



```
avshuluuzhuk@fedora:~/hello
[avshuluuzhuk@fedora hello]$ git checkout c61aa06
Примечание: переключение на «c61aa06».

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:

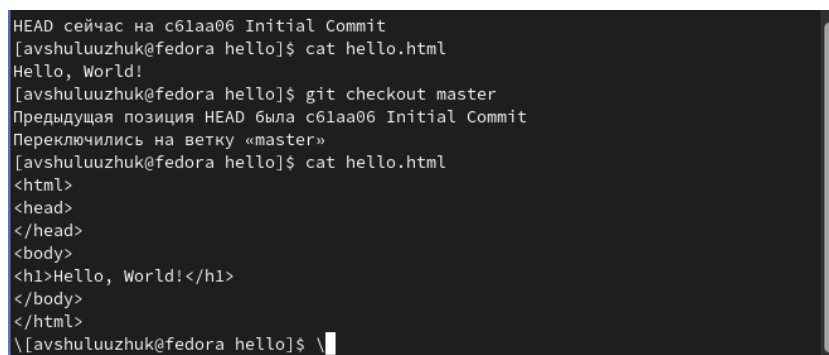
git switch -

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

HEAD сейчас на c61aa06 Initial Commit
```

Рис. 2.14: хэш для первого коммита

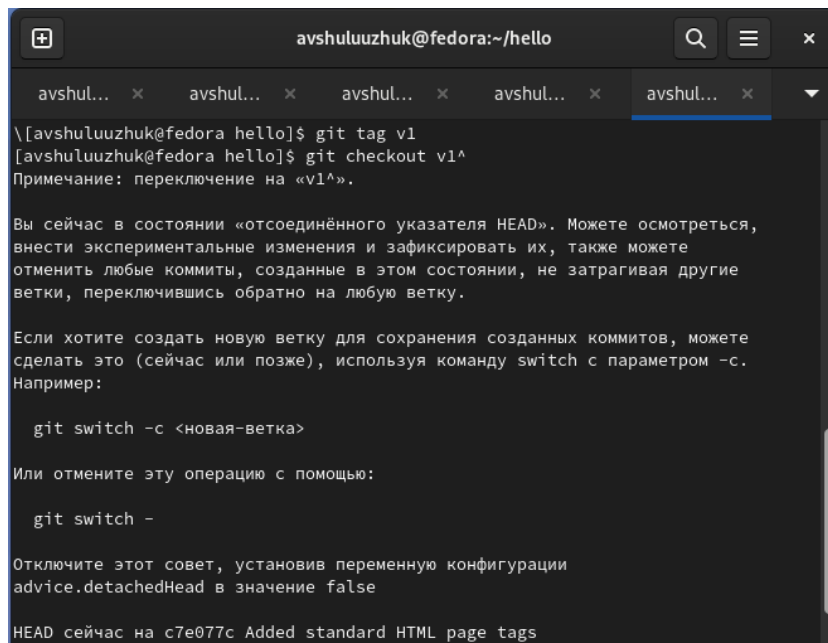
Вернемся к последней версии в ветке `master` (рис. 2.15).



```
HEAD сейчас на c61aa06 Initial Commit
[avshuluuzhuk@fedora hello]$ cat hello.html
Hello, World!
[avshuluuzhuk@fedora hello]$ git checkout master
Предыдущая позиция HEAD была c61aa06 Initial Commit
Переключились на ветку «master»
[avshuluuzhuk@fedora hello]$ cat hello.html
<html>
<head>
</head>
<body>
<h1>Hello, World!</h1>
</body>
</html>
[avshuluuzhuk@fedora hello]$
```

Рис. 2.15: последняя версия в ветке `master`

Назовем текущую версию страницы `hello` первой (`v1`). Создадим тег первой версии. Теперь текущая версия страницы называется `v1`. Создадим тег для версии, которая идет перед текущей версией и назовем его `v1-beta` (рис. 2.16).



```
avshuluuzhuk@fedora:~/hello
\[avshuluuzhuk@fedora hello]$ git tag v1
[avshuluuzhuk@fedora 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:

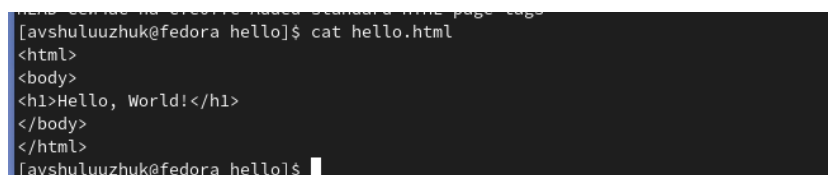
    git switch -

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

HEAD сейчас на c7e077c Added standard HTML page tags
```

Рис. 2.16: создание тега первой версии

Посмотрим файл hello.html (рис. 2.17).



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

Рис. 2.17: файл hello.html

Это версия с тегами

и

, но еще пока без

. Сделаем ее версией v1-beta. Попробуем попереключаться между двумя отмеченными версиями (рис. 2.18).

```
[avshuluuzhuk@fedora hello]$ git tag v1-beta
[avshuluuzhuk@fedora hello]$ git tag v1
fatal: метка «v1» уже существует
[avshuluuzhuk@fedora hello]$ git checkout v1
Предыдущая позиция HEAD была c7e077c Added standard HTML page tags
HEAD сейчас на de2c7bb Added HTML header
[avshuluuzhuk@fedora hello]$ git checkout v1-beta
Предыдущая позиция HEAD была de2c7bb Added HTML header
HEAD сейчас на c7e077c Added standard HTML page tags
[avshuluuzhuk@fedora hello]$
```

Рис. 2.18: переключение между двумя отмеченными версиями

Убедимся, что мы находимся на последнем коммите ветки master, прежде чем продолжить работу. Внесем изменение в файл hello.html в виде нежелательного комментария. Проверим состояние рабочего каталога. Мы видим, что файл hello.html был изменен, но еще не проиндексирован (рис. 2.19) (рис. 2.20).

```
[avshuluuzhuk@fedora hello]$ git checkout master
Предыдущая позиция HEAD была c7e077c Added standard HTML page tags
Переключились на ветку «master»
[avshuluuzhuk@fedora hello]$ gedit hello.html
[avshuluuzhuk@fedora hello]$ gedit hello.html
[avshuluuzhuk@fedora 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»)
[avshuluuzhuk@fedora hello]$
```

Рис. 2.19: отмена локальных изменений до индексации

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

Рис. 2.20: отмена изменений в рабочем каталоге

Внесем изменение в файл hello.html в виде нежелательного коммента-

рия и проиндексируем это изменение. Проверим состояние нежелательного изменения (рис. 2.21).

```
[avshuluuzhuk@fedora hello]$ gedit hello.html
[avshuluuzhuk@fedora hello]$ git add hello.html
[avshuluuzhuk@fedora hello]$ git staus
git: «staus» не является командой git. Смотрите «git --help».

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

Рис. 2.21: изменение файла и проиндексирование изменений

Вывод состояния показывает нам именно то, что мы должны сделать для отмены индексации изменения. Переключимся на версию коммита и проверим состояние каталога. Видим, что рабочий каталога снова чист (рис. 2.22)

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

[avshuluuzhuk@fedora hello]$ git reset HEAD hello.html
Непроиндексированные изменения после сброса:
M   hello.html
[avshuluuzhuk@fedora hello]$ git checkout hello.html
Updated 1 path from the index
[avshuluuzhuk@fedora hello]$ git status
Текущая ветка: master
ничего коммитить, нет изменений в рабочем каталоге
[avshuluuzhuk@fedora hello]$
```

Рис. 2.22: сброс буферной зоны

Изменим файл `hello.html` и сделаем коммит (рис. 2.23)

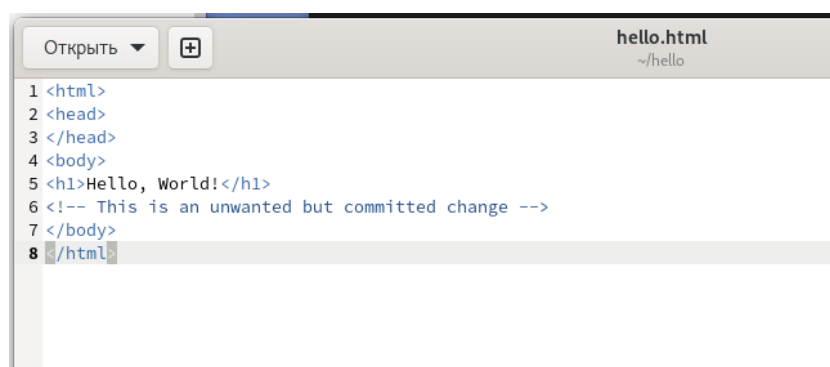


Рис. 2.23: файл hello.html

Чтобы отменить коммит, нам необходимо сделать коммит, который удаляет изменения, сохраненные нежелательным коммитом. Проверим лог (рис. 2.24)

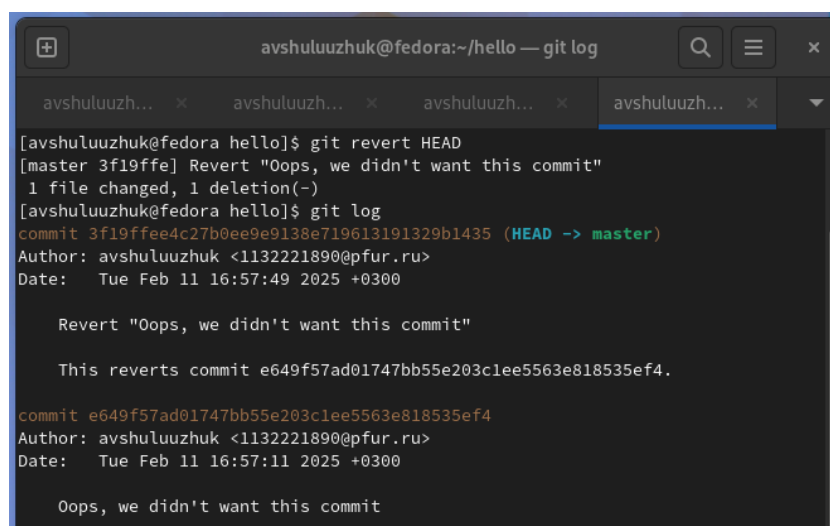
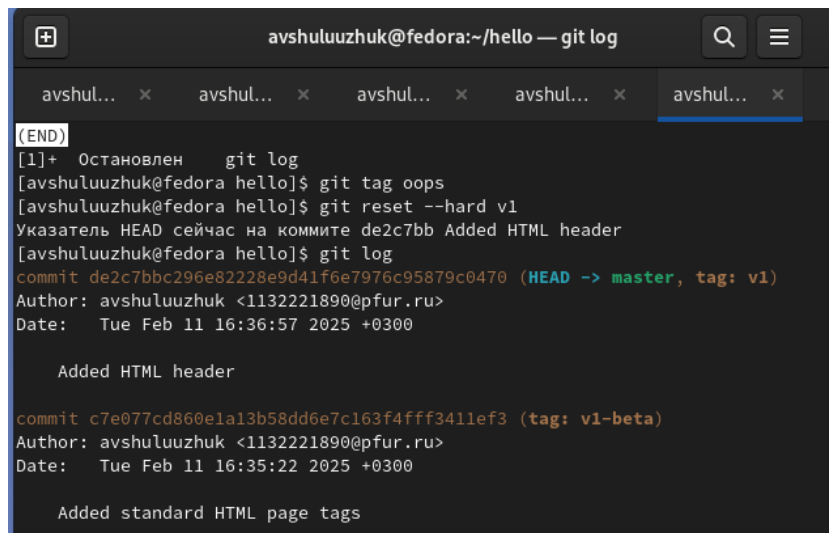


Рис. 2.24: выполнение коммита с новыми изменениями, отменяющими предыдущие

Сделаем быструю проверку нашей истории коммитов. Мы видим, что два последних коммита в этой ветке — «Oops» и «Revert Oops». Удалим их с помощью сброса. Для начала отметим эту ветку. Но прежде чем удалить коммиты, отметим последний коммит тегом, чтобы потом можно было его найти. Сбросим ветку до этой точки. Поскольку ветка имеет тег, мы можем использовать имя тега в команде сброса (рис. 2.25).



```
avshuluuzhuk@fedora:~/hello — git log
(END)
[1]+ Остановлен git log
[avshuluuzhuk@fedora hello]$ git tag oops
[avshuluuzhuk@fedora hello]$ git reset --hard v1
Указатель HEAD сейчас на коммите de2c7bb Added HTML header
[avshuluuzhuk@fedora hello]$ git log
commit de2c7bbc296e82228e9d41f6e7976c95879c0470 (HEAD -> master, tag: v1)
Author: avshuluuzhuk <1132221890@pfur.ru>
Date: Tue Feb 11 16:36:57 2025 +0300

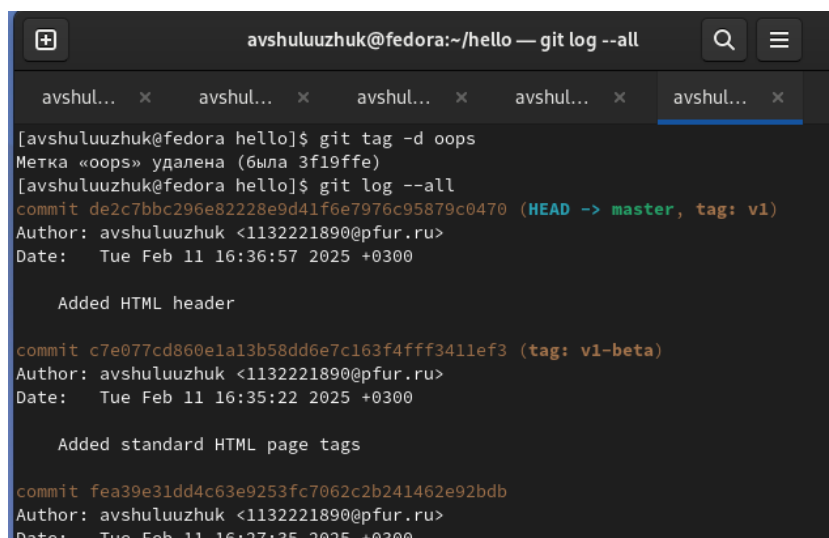
    Added HTML header

commit c7e077cd860e1a13b58dd6e7c163f4fff3411ef3 (tag: v1-beta)
Author: avshuluuzhuk <1132221890@pfur.ru>
Date: Tue Feb 11 16:35:22 2025 +0300

    Added standard HTML page tags
```

Рис. 2.25: брос коммитов к предшествующим коммиту Oops

Тег оорс свою функцию выполнил. Удалим его и коммиты, на которые он ссылался, сборщиком мусора. Тег «оорс» больше не будет отображаться в репозитории (рис. 2.26).



```
avshuluuzhuk@fedora:~/hello — git log --all
[avshuluuzhuk@fedora hello]$ git tag -d oops
Метка «oops» удалена (была 3f19ffe)
[avshuluuzhuk@fedora hello]$ git log --all
commit de2c7bbc296e82228e9d41f6e7976c95879c0470 (HEAD -> master, tag: v1)
Author: avshuluuzhuk <1132221890@pfur.ru>
Date: Tue Feb 11 16:36:57 2025 +0300

    Added HTML header

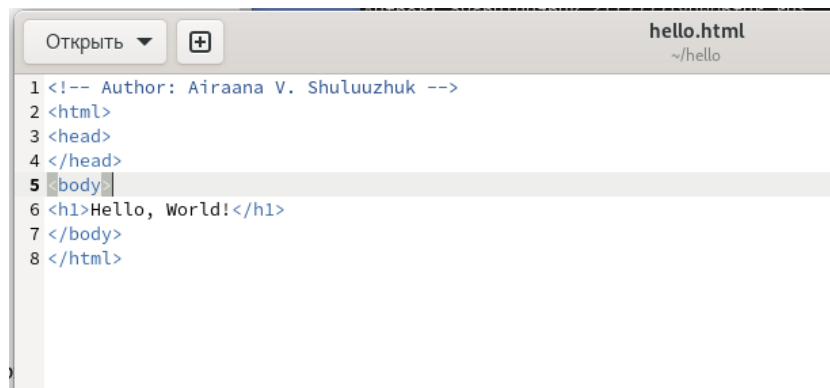
commit c7e077cd860e1a13b58dd6e7c163f4fff3411ef3 (tag: v1-beta)
Author: avshuluuzhuk <1132221890@pfur.ru>
Date: Tue Feb 11 16:35:22 2025 +0300

    Added standard HTML page tags

commit fea39e31dd4c63e9253fc7062c2b241462e92bdb
Author: avshuluuzhuk <1132221890@pfur.ru>
Date: Tue Feb 11 16:27:35 2025 +0300
```

Рис. 2.26: удаление тега оорс

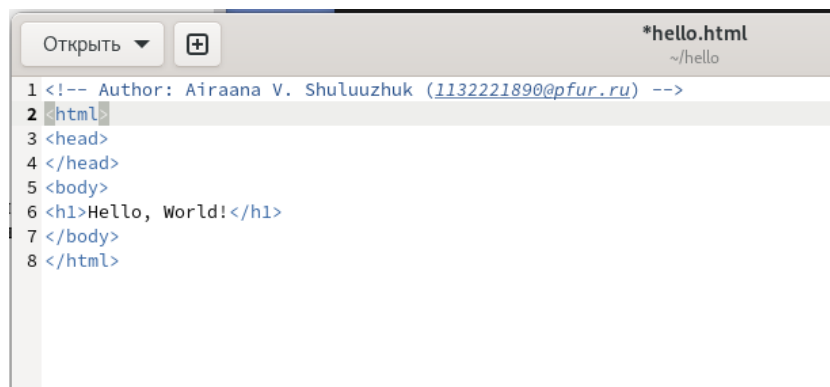
Изменим страницу, а затем сделаем коммит. Добавим в страницу комментарий автора (вставим свою фамилию) (рис. 2.27).



```
1 <!-- Author: Airaana V. Shuluuzhuk -->
2 <html>
3 <head>
4 </head>
5 <body>
6 <h1>Hello, World!</h1>
7 </body>
8 </html>
```

Рис. 2.27: файл hello.html

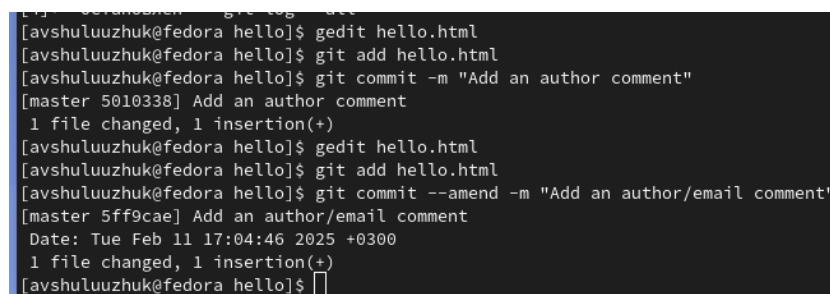
После совершения коммита мы понимаем, что любой хороший комментарий должен включать электронную почту автора. Обновите страницу hello, включив в нее email. (рис. 2.28).



```
1 <!-- Author: Airaana V. Shuluuzhuk (1132221890@pfur.ru) -->
2 <html>
3 <head>
4 </head>
5 <body>
6 <h1>Hello, World!</h1>
7 </body>
8 </html>
```

Рис. 2.28: файл hello.html

Изменим предыдущий коммит, включив в него адрес электронной почты (рис. 2.29).



```
[ayshuluuzhuk@fedora hello]$ gedit hello.html
[ayshuluuzhuk@fedora hello]$ git add hello.html
[ayshuluuzhuk@fedora hello]$ git commit -m "Add an author comment"
[master 5010338] Add an author comment
1 file changed, 1 insertion(+)
[ayshuluuzhuk@fedora hello]$ gedit hello.html
[ayshuluuzhuk@fedora hello]$ git add hello.html
[ayshuluuzhuk@fedora hello]$ git commit --amend -m "Add an author/email comment"
[master 5ff9cae] Add an author/email comment
Date: Tue Feb 11 17:04:46 2025 +0300
1 file changed, 1 insertion(+)
[ayshuluuzhuk@fedora hello]$
```

Рис. 2.29: изменение предыдущего коммита

Переместим файл `hello.html` в каталог `lib`. Создадим структуру нашего репозитория. Перенесем страницу в каталог `lib` (рис. 2.30).

```
[avshuluuzhuk@fedora hello]$ mkdir lib
[avshuluuzhuk@fedora hello]$ git mv hello.html lib
[avshuluuzhuk@fedora 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
[avshuluuzhuk@fedora hello]$
```

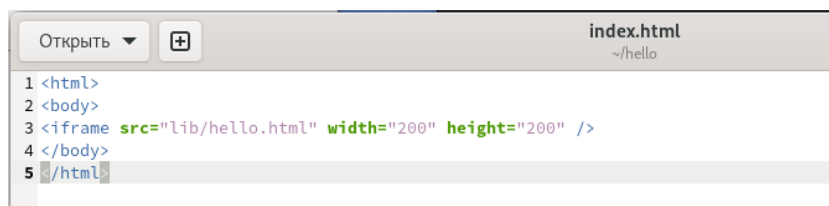
Рис. 2.30: создание каталога

Сделаем коммит этого перемещения (рис. 2.31).

```
[avshuluuzhuk@fedora hello]$ git commit -m "Moved hello.html to lib"
[master c058fec] Moved hello.html to lib
1 file changed, 0 insertions(+), 0 deletions(-)
rename hello.html => lib/hello.html (100%)
[avshuluuzhuk@fedora hello]$
```

Рис. 2.31: коммит в новый каталог

Добавим файл `index.html` в наш репозиторий и сделаем коммит (рис. 2.32) (рис. 2.33).



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

Рис. 2.32: файл `index.html`

```
rename hello.html -> lib/hello.html (100%)
[avshuluuzhuk@fedora hello]$ gedit index.html
[avshuluuzhuk@fedora hello]$ git add index.html
[avshuluuzhuk@fedora hello]$ git commit -m "Added index.html."
[master da0e4a1] Added index.html.
1 file changed, 5 insertions(+)
create mode 100644 index.html
[avshuluuzhuk@fedora hello]$
```

Рис. 2.33: выполнение коммита

Теперь при открытии `index.html`, мы видим кусок страницы `hello` в маленьком окошке (рис. 2.34).

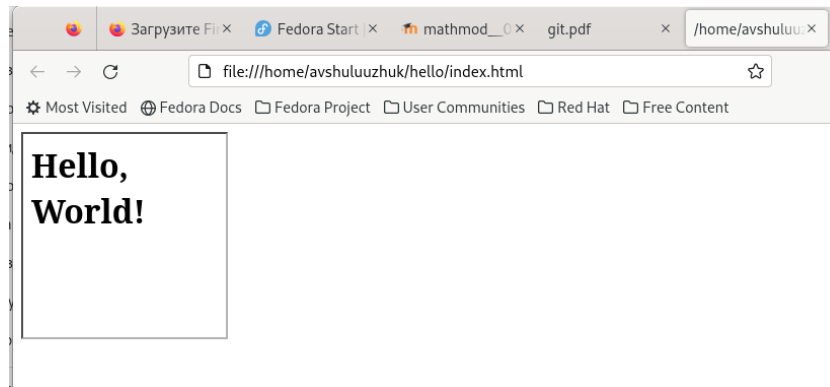


Рис. 2.34: index.html

Каталог `.git`. Это каталог, в котором хранится вся информация `git`. Посмотрим набор каталогов, имена которых состоят из 2 символов. Имена каталогов являются первыми двумя буквами хэша `sha1` объекта, хранящегося в `git` (рис. 2.35).

```
avshuluuzhuk@fedora:~/hello
[avshuluuzhuk@fedora hello]$ ls -C .git
branches      config        HEAD         index        logs         ORIG_HEAD    refs
COMMIT_EDITMSG description    hooks        info         objects      packed-refs
[avshuluuzhuk@fedora hello]$ ls -C .git/objects
00 0b 21 4f 52 62 86 8a c0 c7 cd da e6 f7 info
09 16 3f 50 5f 68 88 bd c6 cb d9 de f4 fe pack
[avshuluuzhuk@fedora hello]$ ls -C .git/objects/<dir>
bash: синтаксическая ошибка рядом с неожиданным маркером «newline»
[avshuluuzhuk@fedora hello]$ cat .git/config
[core]
    repositoryformatversion = 0
    filemode = true
    bare = false
    logallrefupdates = true
[avshuluuzhuk@fedora hello]$
```

Рис. 2.35: каталог `.git`

Посмотрим файл конфигурации. Это файл конфигурации, создающийся для каждого конкретного проекта. Записи в этом файле будут перезаписывать записи в файле `.gitconfig` главного каталога, по крайней мере в рамках этого проекта. (рис. 2.36).

```

[avshuluuzhuk@fedora hello]$ ls -C .git/objects/<dir>
bash: синтаксическая ошибка рядом с неожиданным маркером «newline»
[avshuluuzhuk@fedora hello]$ cat .git/config
[core]
    repositoryformatversion = 0
    filemode = true
    bare = false
    logallrefupdates = true
[avshuluuzhuk@fedora hello]$

```

Рис. 2.36: файл конфигурации

Просмотрим ветки и теги. Каждый файл соответствует тегу, ранее созданному с помощью команды `git tag`. Его содержание — это всего лишь хэш коммита, привязанный к тегу (рис. 2.37).

```

[avshuluuzhuk@fedora hello]$ ls .git/refs
heads  tags
[avshuluuzhuk@fedora hello]$ ls .git/refs/heads
master
[avshuluuzhuk@fedora hello]$ ls .git/refs/tags
v1  v1-beta
[avshuluuzhuk@fedora hello]$ cat .git/refs/tags/v1
de2c7bbc296e82228e9d41f6e7976c95879c0470
[avshuluuzhuk@fedora hello]$

```

Рис. 2.37: ветки и теги

Работа непосредственно с объектами `git`. Просмотрим последний коммит. Эта команда должна показать последний коммит в репозиторий и выведем последний коммит с помощью SHA1 хэша (рис. 2.38).

```
[avshuluuzhuk@fedora hello]$ git log --max-count=1
commit da0e4a16f00ce08b182ae6d15c8dc99bb2de62fc (HEAD -> master)
Author: avshuluuzhuk <1132221890@pfur.ru>
Date: Tue Feb 11 17:12:37 2025 +0300

    Added index.html.
[avshuluuzhuk@fedora hello]$ git cat-file -t <hash>
bash: синтаксическая ошибка рядом с неожиданным маркером «newline»
[avshuluuzhuk@fedora hello]$ git cat-file -t ^C
[avshuluuzhuk@fedora hello]$ git cat-file -t da0e4a16f00ce08b182ae6d15c8dc99bb2d
e62fc
commit
[avshuluuzhuk@fedora hello]$ git cat-file -p da0e4a16f00ce08b182ae6d15c8dc99bb2d
e62fc
tree 16b3865f9360631693d4d6a9b3968a619b3bd5f4
parent c058fec1db60448daa82cb334e07746596c96eec
author avshuluuzhuk <1132221890@pfur.ru> 1739283157 +0300
committer avshuluuzhuk <1132221890@pfur.ru> 1739283157 +0300
gpgsig -----BEGIN PGP SIGNATURE-----

iQIzBAABCAAdFiEEWSqbbR1Wx5DL8peQ6a0LG4vYeTQFAmerWtUACgkQ6a0LG4vY
eTTiAw/9GreDLthjq7/jwZzqdzqtK/GfMz3m263/E4f6MRjFXV3VPsl0+nvSuXb+
fHx9n+42E0DCSmzJ830YN/dJAx0jUk3BSpICjnaVn7To3m0Mq3gVPzi/Ojv2vLKO
5sltQaWLWJ/BPAvnYLaG3iNC1+XM8Bug0BwoWH01CKRIbnk/0JcjdVKNuucv9g32
```

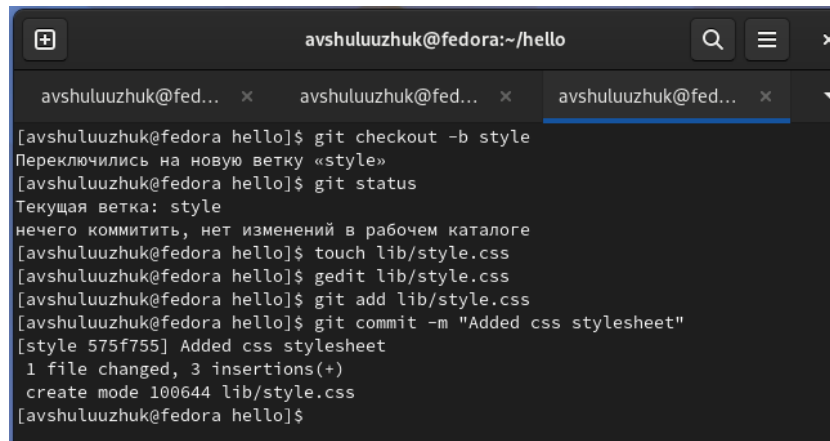
Рис. 2.38: поиск последнего коммита

Выведем дерево каталогов, ссылка на который идет в коммите. Это должно быть описание файлов (верхнего уровня) в нашем проекте (для конкретного коммита). Используем SHA1 хэш из строки «дерева», из списка выше. Выведем каталога `lib` и файл `hello.html` (рис. 2.39)

```
[avshuluuzhuk@fedora hello]$ git cat-file -p 16b3865f9360631693d4d6a9b3968a619b3
bd5f4
100644 blob 68583eacd328506a31568f0df6f45ebb874238b5    index.html
040000 tree 0b889424605fe47f7b4605d43fe0910503d476f2    lib
[avshuluuzhuk@fedora hello]$ git cat-file -p 0b889424605fe47f7b4605d43fe0910503d
476f2
100644 blob cdc0757cdecbb0d8b7ef726a50f1d05dd66c04129    hello.html
[avshuluuzhuk@fedora hello]$ git cat-file -p cdc0757cdecbb0d8b7ef726a50f1d05dd66c
04129
<!-- Author: Airaana V. Shuluuzhuk (1132221890@pfur.ru) -->
<html>
<head>
</head>
<body>
<h1>Hello, World!</h1>
</body>
</html>
[avshuluuzhuk@fedora hello]$
```

Рис. 2.39: вывод дерева каталогов

Создадим ветку. Назовем нашу новую ветку «`style`». Добавим изменения в файл стилей и выполним коммит (рис. 2.40).



```
avshuluuzhuk@fedora:~/hello
[avshuluuzhuk@fedora hello]$ git checkout -b style
Переключились на новую ветку «style»
[avshuluuzhuk@fedora hello]$ git status
Текущая ветка: style
ничего коммитить, нет изменений в рабочем каталоге
[avshuluuzhuk@fedora hello]$ touch lib/style.css
[avshuluuzhuk@fedora hello]$ gedit lib/style.css
[avshuluuzhuk@fedora hello]$ git add lib/style.css
[avshuluuzhuk@fedora hello]$ git commit -m "Added css stylesheet"
[style 575f755] Added css stylesheet
 1 file changed, 3 insertions(+)
 create mode 100644 lib/style.css
[avshuluuzhuk@fedora hello]$
```

Рис. 2.40: создание новой ветки

Обновим файл `hello.html`, чтобы использовать стили `style.css` и выполним коммит (рис. 2.41).



```
*hello.html
~/hello/lib

1 <!-- Author: Airaana V. Shuluuzhuk (1132221890@pfur.ru) -->
2 <html>
3 <head>
4 <link type="text/css" rel="stylesheet"
5 media="all" href="style.css" />
6 </head>
7 <body>
8 <h1>Hello, World!</h1>
9 </body>
10 </html>
```

Рис. 2.41: файл `hello.html`

Обновим файл `index.html`, чтобы он тоже использовал `style.css` и выполним коммит (рис. 2.42)

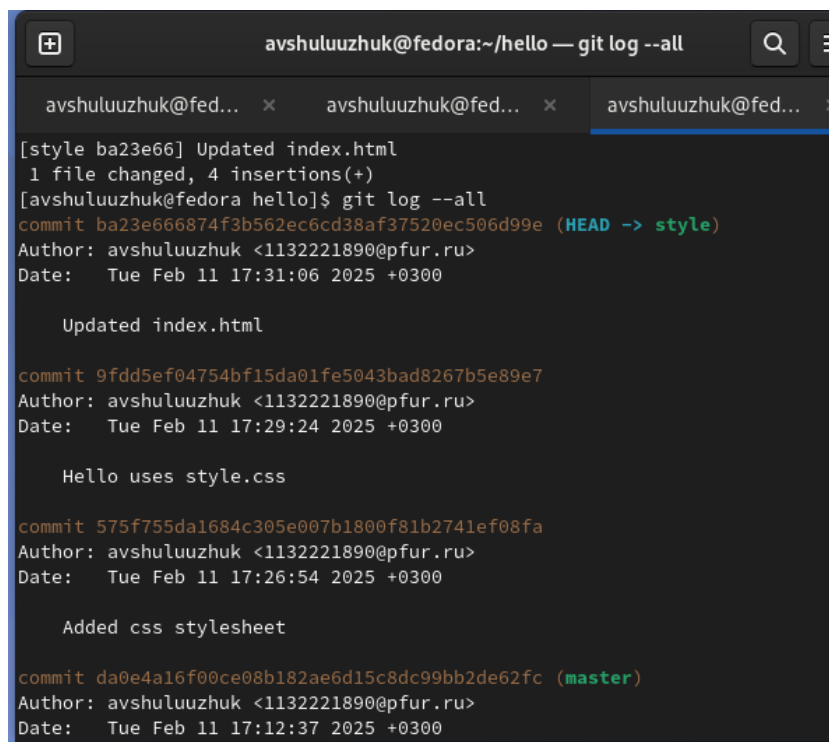


```
index.html
~/hello

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

Рис. 2.42: `index.html`

Теперь в нашем проекте есть две ветки (рис. 2.43)



```
avshuluuzhuk@fedora:~/hello — git log --all
[style ba23e66] Updated index.html
1 file changed, 4 insertions(+)
[avshuluuzhuk@fedora hello]$ git log --all
commit ba23e66874f3b562ec6cd38af37520ec506d99e (HEAD -> style)
Author: avshuluuzhuk <1132221890@pfur.ru>
Date: Tue Feb 11 17:31:06 2025 +0300

    Updated index.html

commit 9fdd5ef04754bf15da01fe5043bad8267b5e89e7
Author: avshuluuzhuk <1132221890@pfur.ru>
Date: Tue Feb 11 17:29:24 2025 +0300

    Hello uses style.css

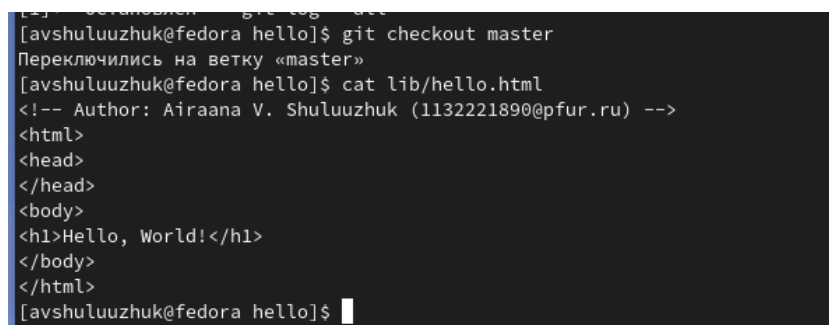
commit 575f755da1684c305e007b1800f81b2741ef08fa
Author: avshuluuzhuk <1132221890@pfur.ru>
Date: Tue Feb 11 17:26:54 2025 +0300

    Added css stylesheet

commit da0e4a16f00ce08b182ae6d15c8dc99bb2de62fc (master)
Author: avshuluuzhuk <1132221890@pfur.ru>
Date: Tue Feb 11 17:12:37 2025 +0300
```

Рис. 2.43: навигация по веткам

Используем команду `git checkout` для переключения между ветками. Мы находимся на ветке `master`. Это заметно по тому, что файл `hello.html` не использует стили `style.css` (рис. 2.44)



```
[avshuluuzhuk@fedora hello]$ git checkout master
Переключились на ветку «master»
[avshuluuzhuk@fedora hello]$ cat lib/hello.html
<!-- Author: Airaana V. Shuluuzhuk (1132221890@pfur.ru) -->
<html>
<head>
</head>
<body>
<h1>Hello, World!</h1>
</body>
</html>
[avshuluuzhuk@fedora hello]$
```

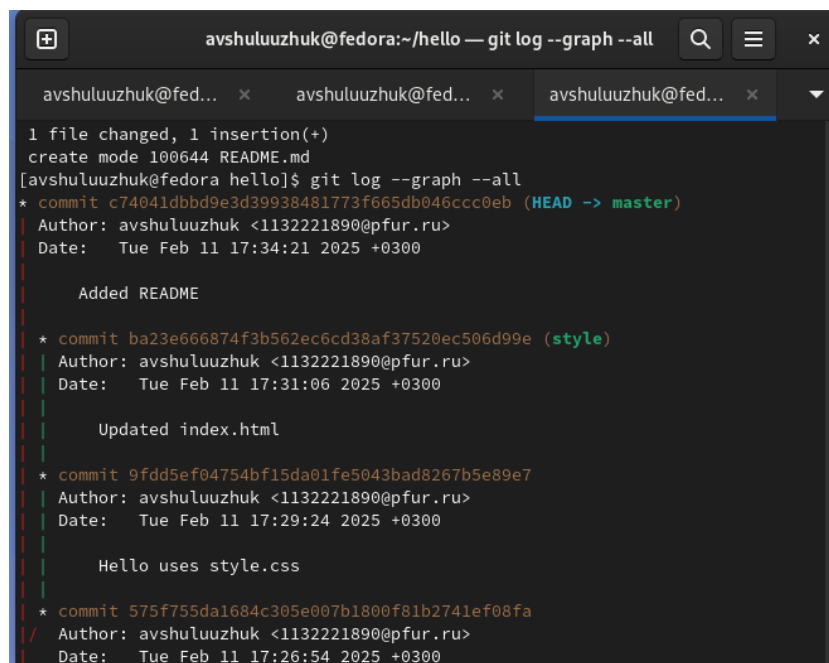
Рис. 2.44: Переключение на ветку master

Вернемся к ветке `style`. Содержимое `lib/hello.html` подтверждает, что мы вернулись на ветку `style` (рис. 2.45).


```
[avshuluuzhuk@fedora hello]$ git checkout style
Переключились на ветку «style»
[avshuluuzhuk@fedora hello]$ cat lib/hello.html
<!-- Author: Airaana V. Shuluuzhuk (1132221890@pfur.ru) -->
<html>
<head>
<link type="text/css" rel="stylesheet"
media="all" href="style.css" />
</head>
<body>
<h1>Hello, World!</h1>
</body>
</html>
[avshuluuzhuk@fedora hello]$
```

Рис. 2.45: переход к ветке style

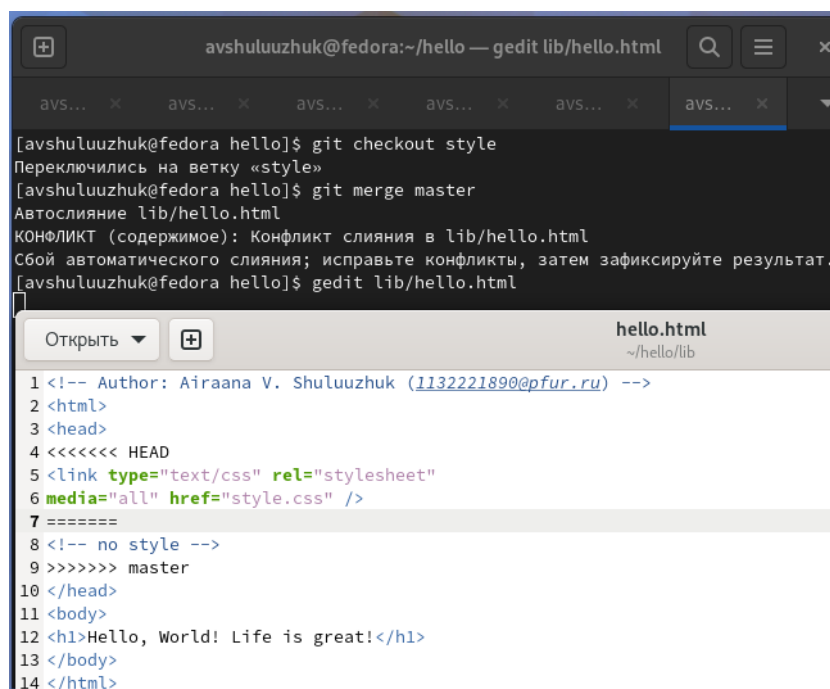
Теперь у нас в репозитории есть две отличающиеся ветки. Используем следующую лог-команду для просмотра веток и их отличий (рис. 2.46).



```
avshuluuzhuk@fedora:~/hello — git log --graph --all
1 file changed, 1 insertion(+)
create mode 100644 README.md
[avshuluuzhuk@fedora hello]$ git log --graph --all
* commit c74041dbbd9e3d39938481773f665db046ccc0eb (HEAD -> master)
  Author: avshuluuzhuk <1132221890@pfur.ru>
  Date: Tue Feb 11 17:34:21 2025 +0300
  |
  | Added README
  |
  * commit ba23e666874f3b562ec6cd38af37520ec506d99e (style)
  | Author: avshuluuzhuk <1132221890@pfur.ru>
  | Date: Tue Feb 11 17:31:06 2025 +0300
  | |
  | | Updated index.html
  | |
  * commit 9fdd5ef04754bf15da01fe5043bad8267b5e89e7
  | Author: avshuluuzhuk <1132221890@pfur.ru>
  | Date: Tue Feb 11 17:29:24 2025 +0300
  | |
  | | Hello uses style.css
  | |
  * commit 575f755da1684c305e007b1800f81b2741ef08fa
  | Author: avshuluuzhuk <1132221890@pfur.ru>
  | Date: Tue Feb 11 17:26:54 2025 +0300
```

Рис. 2.46: просмотр текущих веток

Теперь вернемся к ветке style и попытаемся объединить ее с новой веткой master. Если мы откроем lib/hello.html, мы увидим (рис. 2.47).



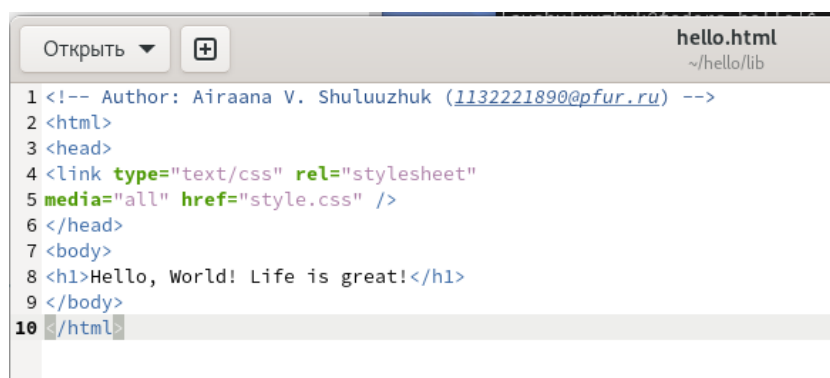
```
[avshuluuzhuk@fedora hello]$ git checkout style
Переключились на ветку «style»
[avshuluuzhuk@fedora hello]$ git merge master
Автослияние lib/hello.html
КОНФЛИКТ (содержимое): Конфликт слияния в lib/hello.html
Сбой автоматического слияния; исправьте конфликты, затем зафиксируйте результат.
[avshuluuzhuk@fedora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

Открыть ▾ + hello.html
~/hello/lib

```
1 <!-- Author: Airaana V. Shuluuzhuk (1132221890@pfur.ru) -->
2 <html>
3 <head>
4 <<<<<<< HEAD
5 <link type="text/css" rel="stylesheet"
6 media="all" href="style.css" />
7 =====
8 <!-- no style -->
9 >>>>>>> master
10 </head>
11 <body>
12 <h1>Hello, World! Life is great!</h1>
13 </body>
14 </html>
```

Рис. 2.47: слияние master с веткой style

Необходимо вручную разрешить конфликт. Внесем изменения в lib/hello.html для достижения следующего результата и сделаем коммит решения коммита (рис. 2.48).

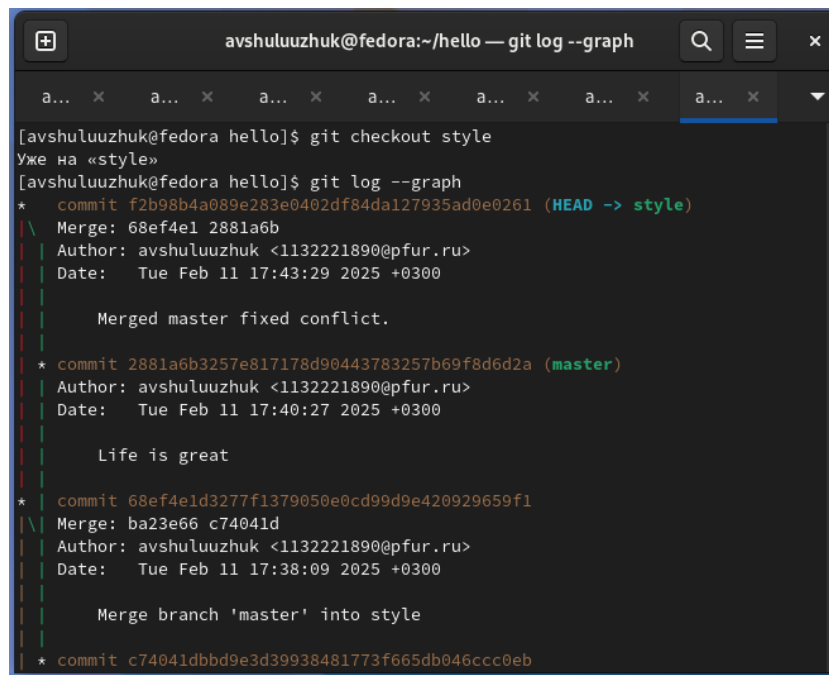


Открыть ▾ + hello.html
~/hello/lib

```
1 <!-- Author: Airaana V. Shuluuzhuk (1132221890@pfur.ru) -->
2 <html>
3 <head>
4 <link type="text/css" rel="stylesheet"
5 media="all" href="style.css" />
6 </head>
7 <body>
8 <h1>Hello, World! Life is great!</h1>
9 </body>
10 </html>
```

Рис. 2.48: разрешение конфликта

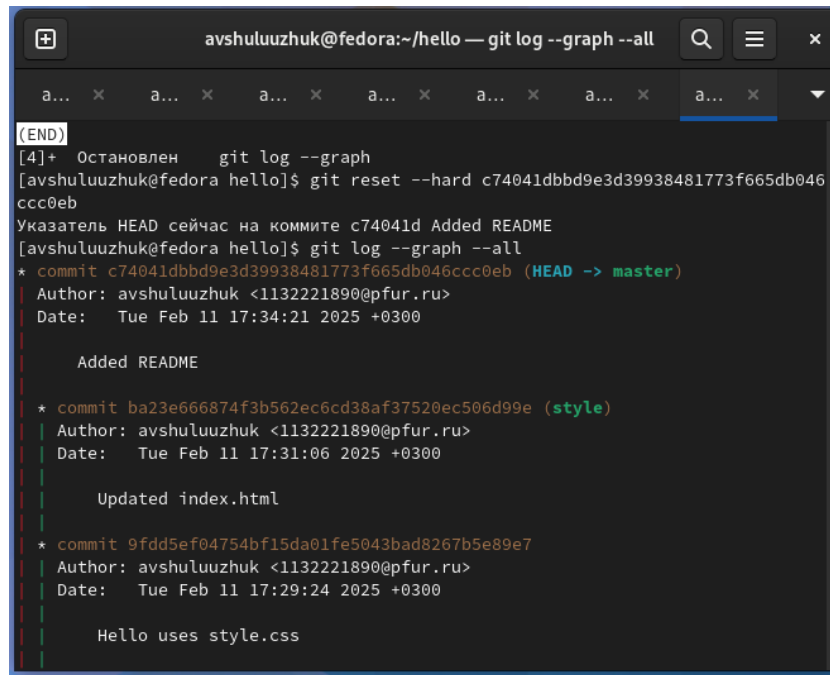
Для сброса ветки style нам необходимо найти последний коммит перед слиянием (рис. 2.49).

A terminal window titled 'avshuluuzhuk@fedora:~/hello — git log --graph'. The terminal shows the following commands and output:

```
[avshuluuzhuk@fedora hello]$ git checkout style
Уже на «style»
[avshuluuzhuk@fedora hello]$ git log --graph
*   commit f2b98b4a089e283e0402df84da127935ad0e0261 (HEAD -> style)
| \
|  Merge: 68ef4e1 2881a6b
|  Author: avshuluuzhuk <1132221890@pfur.ru>
|  Date:   Tue Feb 11 17:43:29 2025 +0300
|
|      Merged master fixed conflict.
|
| *   commit 2881a6b3257e817178d90443783257b69f8d6d2a (master)
| | \
| |  Merge: ba23e66 c74041d
| |  Author: avshuluuzhuk <1132221890@pfur.ru>
| |  Date:   Tue Feb 11 17:40:27 2025 +0300
| |
| |      Life is great
| |
| *   commit 68ef4e1d3277f1379050e0cd99d9e420929659f1
| | \
| |  Merge: ba23e66 c74041d
| |  Author: avshuluuzhuk <1132221890@pfur.ru>
| |  Date:   Tue Feb 11 17:38:09 2025 +0300
| |
| |      Merge branch 'master' into style
| |
| *   commit c74041dbbd9e3d39938481773f665db046ccc0eb
```

Рис. 2.49: сброс ветки style

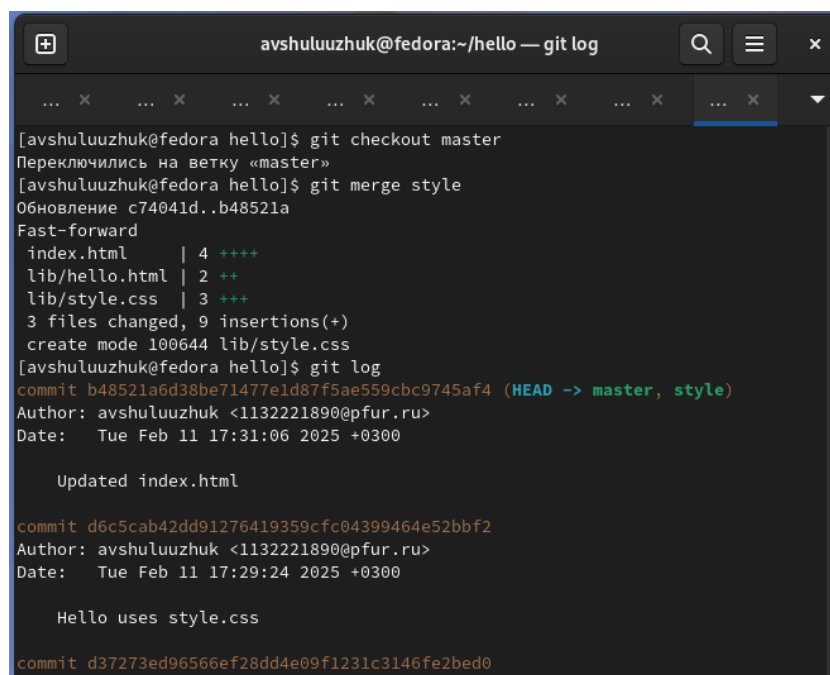
Мы видим, что коммит «Updated index.html» был последним на ветке style перед слиянием. Сбросим ветку style к этому коммиту. Проверим ветку. Поищем лог ветки style. У нас в истории больше нет коммитов слияний (рис. 2.50).



```
avshuluuzhuk@fedora:~/hello — git log --graph --all
(END)
[4]+ Остановлен  git log --graph
[avshuluuzhuk@fedora hello]$ git reset --hard c74041dbbd9e3d39938481773f665db046ccc0eb
Указатель HEAD сейчас на коммите c74041d Added README
[avshuluuzhuk@fedora hello]$ git log --graph --all
* commit c74041dbbd9e3d39938481773f665db046ccc0eb (HEAD -> master)
  | Author: avshuluuzhuk <1132221890@pfur.ru>
  | Date: Tue Feb 11 17:34:21 2025 +0300
  |
  |     Added README
  |
  * commit ba23e666874f3b562ec6cd38af37520ec506d99e (style)
  | Author: avshuluuzhuk <1132221890@pfur.ru>
  | Date: Tue Feb 11 17:31:06 2025 +0300
  |
  |     Updated index.html
  |
  * commit 9fdd5ef04754bf15da01fe5043bad8267b5e89e7
  | Author: avshuluuzhuk <1132221890@pfur.ru>
  | Date: Tue Feb 11 17:29:24 2025 +0300
  |
  |     Hello uses style.css
```

Рис. 2.50: сброс ветки style и проверка

Просмотрим лог историй. Теперь ветки style и master идентичны (рис. 2.51).



```
avshuluuzhuk@fedora:~/hello — git log
[avshuluuzhuk@fedora hello]$ git checkout master
Переключились на ветку «master»
[avshuluuzhuk@fedora hello]$ git merge style
Обновление c74041d..b48521a
Fast-forward
 index.html      | 4 ++++
 lib/hello.html  | 2 ++
 lib/style.css   | 3 +++
 3 files changed, 9 insertions(+)
 create mode 100644 lib/style.css
[avshuluuzhuk@fedora hello]$ git log
commit b48521a6d38be71477e1d87f5ae559cbc9745af4 (HEAD -> master, style)
Author: avshuluuzhuk <1132221890@pfur.ru>
Date: Tue Feb 11 17:31:06 2025 +0300

    Updated index.html

commit d6c5cab42dd91276419359cfc04399464e52bbf2
Author: avshuluuzhuk <1132221890@pfur.ru>
Date: Tue Feb 11 17:29:24 2025 +0300

    Hello uses style.css

commit d37273ed96566ef28dd4e09f1231c3146fe2bed0
```

Рис. 2.51: слияние style в master

Перейдем в рабочий каталог и сделаем клон репозитория hello. Сейчас мы находимся в рабочем каталоге. В этот момент вы должны находиться в «рабочем» каталоге. Здесь должен быть единственный репозиторий под названием «hello». Создадим клон репозитория. В вашем рабочем каталоге теперь должно быть два репозитория: оригинальный репозиторий «hello» и клонированный репозиторий «cloned_hello» (рис. 2.52)

```
[avshuluuzhuk@fedora hello]$ cd ..
[avshuluuzhuk@fedora ~]$ pwd
/home/avshuluuzhuk
[avshuluuzhuk@fedora ~]$ ls
bin          lab10_3.sh~  work        Изображения  Шаблоны
hello        lab10_4.sh~  Видео       Музыка
lab10_1.sh~  os-intro     Документы   Общедоступные
lab10_2.sh~  snap        Загрузки    'Рабочий стол'
[avshuluuzhuk@fedora ~]$ git clone hello cloned_hello
Клонирование в «cloned_hello»...
готово.
[avshuluuzhuk@fedora ~]$ ls
bin          lab10_2.sh~  snap        Загрузки    'Рабочий стол'
cloned_hello lab10_3.sh~  work        Изображения  Шаблоны
hello        lab10_4.sh~  Видео       Музыка
lab10_1.sh~  os-intro     Документы   Общедоступные
[avshuluuzhuk@fedora ~]$
```

Рис. 2.52: переход в рабочий каталог и создание клона репозитория

Взглянем на клонированный репозиторий. Увидим список всех файлов на верхнем уровне оригинального репозитория README.md, index.html и lib. Просмотрим историю репозитория (рис. 2.53).

```
[avshuluuzhuk@fedora ~]$ cd cloned_hello
[avshuluuzhuk@fedora cloned_hello]$ ls
index.html  lib  README.md
[avshuluuzhuk@fedora cloned_hello]$ git log --all
commit b48521a6d38be71477e1d87f5ae559cbc9745af4 (HEAD -> master, origin/style, o
origin/master, origin/HEAD)
Author: avshuluuzhuk <1132221890@pfur.ru>
Date: Tue Feb 11 17:31:06 2025 +0300

    Updated index.html

commit d6c5cab42dd91276419359cfc04399464e52bbf2
Author: avshuluuzhuk <1132221890@pfur.ru>
Date: Tue Feb 11 17:29:24 2025 +0300

    Hello uses style.css

commit d37273ed96566ef28dd4e09f1231c3146fe2bed0
Author: avshuluuzhuk <1132221890@pfur.ru>
Date: Tue Feb 11 17:26:54 2025 +0300

    Added css stylesheet
```

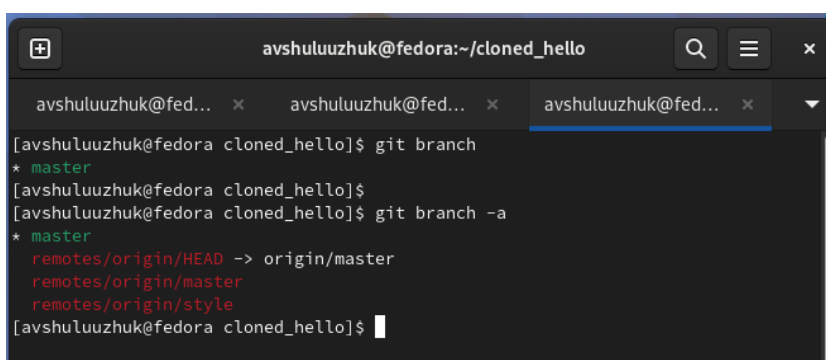
Рис. 2.53: просмотр клонированного репозитория

Выполним `git remote`. Мы видим, что клонированный репозиторий знает об имени по умолчанию удаленного репозитория. Посмотрим, можем ли мы получить более подробную информацию об имени по умолчанию (рис. 2.54).

```
[avshuluuzhuk@fedora cloned_hello]$ git remote
origin
[avshuluuzhuk@fedora cloned_hello]$ git remote show origin
* внешний репозиторий origin
  URL для извлечения: /home/avshuluuzhuk/hello
  URL для отправки: /home/avshuluuzhuk/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)
[avshuluuzhuk@fedora cloned_hello]$
```

Рис. 2.54: просмотр состояния репозитория

Посмотрим на ветки, доступные в нашем клонированном репозитории. Как мы видим, в списке только ветка `master`. Команда `git branch` выводит только список локальных веток по умолчанию. Для того, чтобы увидеть все ветки, попробуйте следующую команду: `git branch -a`. Git выводит все коммиты в оригинальный репозиторий, но ветки в удаленном репозитории не рассматриваются как локальные (рис. 2.55).



```
avshuluuzhuk@fedora:~/cloned_hello
[avshuluuzhuk@fedora cloned_hello]$ git branch
* master
[avshuluuzhuk@fedora cloned_hello]$
[avshuluuzhuk@fedora cloned_hello]$ git branch -a
* master
  remotes/origin/HEAD -> origin/master
  remotes/origin/master
  remotes/origin/style
[avshuluuzhuk@fedora cloned_hello]$
```

Рис. 2.55: список удаленных веток

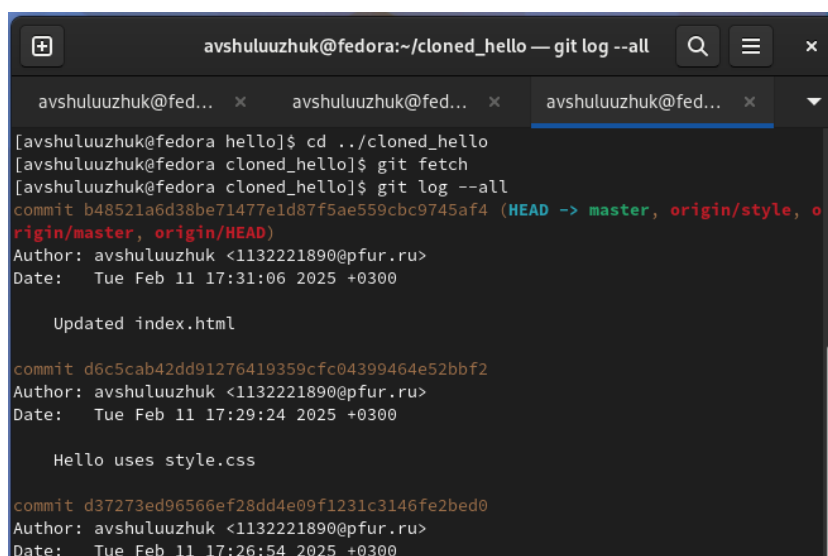
Внесем изменения в оригинальный репозиторий `hello`. Сейчас мы находимся в репозитории `hello`. Внесем следующие изменения в файл

README.md: This is the Hello World example from the git tutorial. Теперь добавим это изменение и сделаем коммит. Теперь в оригинальном репозитории есть более поздние изменения, которых нет в клонированной версии. Далее мы извлечем и сольем эти изменения в клонированный репозиторий (рис. 2.56).

```
[avshuluuzhuk@fedora cloned_hello]$ cd ../hello
bash: cd: ../hello: Нет такого файла или каталога
[avshuluuzhuk@fedora cloned_hello]$ cd ../hello
[avshuluuzhuk@fedora hello]$ gedit README.md
[avshuluuzhuk@fedora hello]$ git add README
fatal: спецификатор пути «README» не соответствует ни одному файлу
[avshuluuzhuk@fedora hello]$ git commit -m "Changed README in original repo"
Текущая ветка: master
нечего коммитить, нет изменений в рабочем каталоге
[avshuluuzhuk@fedora hello]$
```

Рис. 2.56: внесение изменения в оригинальный репозиторий hello

Научимся извлекать изменения из удаленного репозитория. Сейчас мы находимся в репозитории cloned_hello (рис. 2.57).

The image shows a terminal window titled 'avshuluuzhuk@fedora:~/cloned_hello — git log --all'. It displays the output of the 'git log --all' command, showing three commits. The first commit is the HEAD, followed by a commit from 'origin/style', and then a commit from 'origin/master'. The terminal text is as follows:

```
avshuluuzhuk@fed... x avshuluuzhuk@fed... x avshuluuzhuk@fed... x
[avshuluuzhuk@fedora hello]$ cd ../cloned_hello
[avshuluuzhuk@fedora cloned_hello]$ git fetch
[avshuluuzhuk@fedora cloned_hello]$ git log --all
commit b48521a6d38be71477e1d87f5ae559cbc9745af4 (HEAD -> master, origin/style, o
origin/master, origin/HEAD)
Author: avshuluuzhuk <1132221890@pfur.ru>
Date: Tue Feb 11 17:31:06 2025 +0300

    Updated index.html

commit d6c5cab42dd91276419359cfc04399464e52bbf2
Author: avshuluuzhuk <1132221890@pfur.ru>
Date: Tue Feb 11 17:29:24 2025 +0300

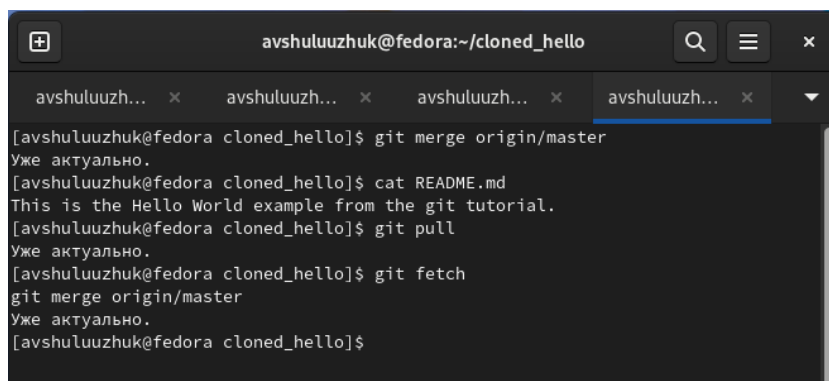
    Hello uses style.css

commit d37273ed96566ef28dd4e09f1231c3146fe2bed0
Author: avshuluuzhuk <1132221890@pfur.ru>
Date: Tue Feb 11 17:26:54 2025 +0300
```

Рис. 2.57: извлечение изменений

Сольем извлеченные изменения в локальную ветку master. Еще раз проверим файл README.md. Мы должны увидеть изменения. Команда git fetch не сливает изменения, мы можем вручную слить изменения из уда-

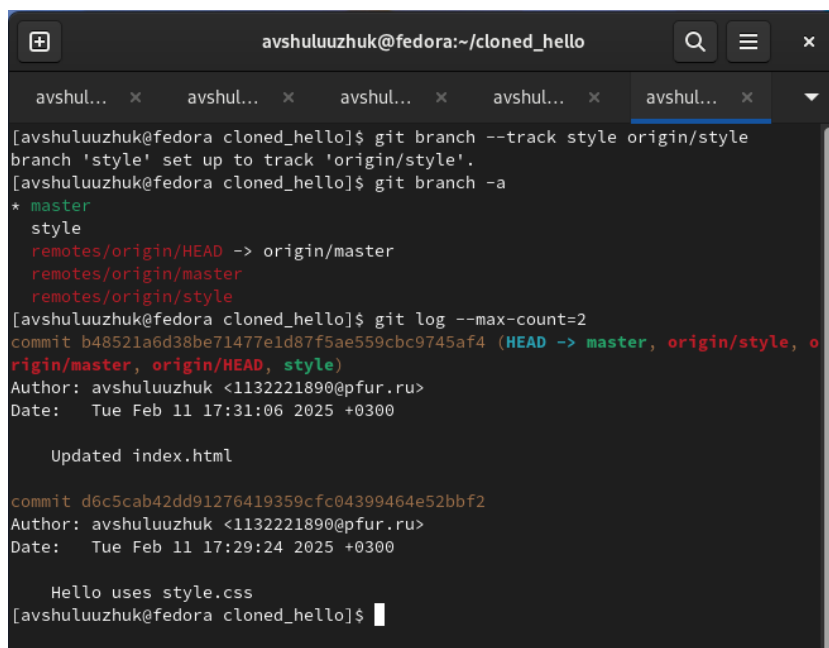
ленного репозитория. Теперь рассмотрим объединение fetch и merge в одну команду (рис. 2.58).



```
avshuluuzhuk@fedora:~/cloned_hello
[avshuluuzhuk@fedora cloned_hello]$ git merge origin/master
Уже актуально.
[avshuluuzhuk@fedora cloned_hello]$ cat README.md
This is the Hello World example from the git tutorial.
[avshuluuzhuk@fedora cloned_hello]$ git pull
Уже актуально.
[avshuluuzhuk@fedora cloned_hello]$ git fetch
git merge origin/master
Уже актуально.
[avshuluuzhuk@fedora cloned_hello]$
```

Рис. 2.58: слияние извлеченных изменений

Добавим локальную ветку, которая отслеживает удаленную ветку. Теперь можем видеть ветку style в списке веток и логе (рис. 2.59)



```
avshuluuzhuk@fedora:~/cloned_hello
[avshuluuzhuk@fedora cloned_hello]$ git branch --track style origin/style
branch 'style' set up to track 'origin/style'.
[avshuluuzhuk@fedora cloned_hello]$ git branch -a
* master
  style
  remotes/origin/HEAD -> origin/master
  remotes/origin/master
  remotes/origin/style
[avshuluuzhuk@fedora cloned_hello]$ git log --max-count=2
commit b48521a6d38be71477e1d87f5ae559cbc9745af4 (HEAD -> master, origin/style, origin/master, origin/HEAD, style)
Author: avshuluuzhuk <1132221890@pfur.ru>
Date: Tue Feb 11 17:31:06 2025 +0300

    Updated index.html

commit d6c5cab42dd91276419359cfc04399464e52bbf2
Author: avshuluuzhuk <1132221890@pfur.ru>
Date: Tue Feb 11 17:29:24 2025 +0300

    Hello uses style.css
[avshuluuzhuk@fedora cloned_hello]$
```

Рис. 2.59: добавление локальной ветки, которая отслеживает удаленную ветку

Создадим чистый репозиторий. Сейчас мы находимся в рабочем каталоге (рис. 2.60).


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

Рис. 2.60: создание чистого репозитория

Добавим репозиторий `hello.git` к нашему оригинальному репозиторию. 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. Отредактируем файл `README.md` и сделаем коммит. Теперь отправим изменения в общий репозиторий. Общим называется репозиторий, получающий отправленные нами изменения (рис. 2.61).

```
[avshuluuzhuk@fedora hello]$ git remote add shared ../hello.git
[avshuluuzhuk@fedora hello]$ gedit README.md
[avshuluuzhuk@fedora hello]$ git checkout master
M      README.md
Уже на «master»
[avshuluuzhuk@fedora hello]$ git add README
fatal: спецификатор пути «README» не соответствует ни одному файлу
[avshuluuzhuk@fedora hello]$ git commit -m "Added shared comment to readme"
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:      README.md

индекс пуст (используйте «git add» и/или «git commit -a»)
[avshuluuzhuk@fedora hello]$ git push shared master
Everything up-to-date
[avshuluuzhuk@fedora hello]$
```

Рис. 2.61: добавление удаленного репозитория и отправка изменений

Научимся извлекать изменения из общего репозитория. Быстро переключимся в клонированный репозиторий и извлечем изменения, только что отправленные в общий репозиторий. Сейчас мы находимся в репозитории `cloned_hello` (рис. 2.62).

```
[avshuluuzhuk@fedora hello]$ cd ../cloned_hello
[avshuluuzhuk@fedora cloned_hello]$ git remote add shared ../hello.git
[avshuluuzhuk@fedora cloned_hello]$ git branch --track shared master
branch 'shared' set up to track 'master'.
[avshuluuzhuk@fedora cloned_hello]$ git pull shared master
Из ../hello
* branch          master      -> FETCH_HEAD
* [новая ветка]   master      -> shared/master
Уже актуально.
[avshuluuzhuk@fedora cloned_hello]$ cat README.md
This is the Hello World example from the git tutorial.
[avshuluuzhuk@fedora cloned_hello]$
```

Рис. 2.62: извлечение общих изменений

3 Выводы

В результате выполнения лабораторной работы научились пользоваться с git и разобрались с его особенностями.