

Лабораторная работа №1

Работа с git

Дворкина Ева Владимировна

Содержание

1	Цель работы	6
2	Задание	7
3	Теоретическое введение	8
4	Выполнение лабораторной работы	9
4.1	Подготовка	9
4.2	Создание проекта	9
4.3	Внесение изменений	10
4.4	История	13
4.5	Получение старых версий	14
4.6	Создание тегов версий	15
4.7	Отмена локальных изменений (до индексации)	17
4.8	Отмена проиндексированных изменений (перед коммитом)	18
4.9	Отмена коммитов	19
4.10	Удаление коммитов из ветки	20
4.11	Удаление тега oops	21
4.12	Изменение предыдущего коммита	22
4.13	Перемещение файлов	23
4.14	Подробнее о структуре	24
4.15	Git внутри: Каталог .git	25
4.16	Работа непосредственно с объектами git	25
4.17	Создание ветки	26
4.18	Навигация по веткам	28
4.19	Изменения в ветке master	29
4.20	Слияние	30
4.21	Создание конфликта	31
4.22	Разрешение конфликтов	32
4.23	Сброс ветки style	33
4.24	Сброс ветки master	35
4.25	Перебазирование	37
4.26	Слияние в ветку master	37
4.27	Клонирование репозитория	38
4.28	Что такое origin?	39
4.29	Удаленные ветки	40
4.30	Изменение оригинального репозитория	40

4.31 Слияние извлеченных изменений	41
4.32 Добавление ветки наблюдения	41
4.33 Создание чистого репозитория	42
4.34 Отправка и извлечение изменений	42
5 Выводы	44
Список литературы	45

Список иллюстраций

4.1	Настройка git	9
4.2	Создание репозитория	10
4.3	Добавление файла в репозиторий	10
4.4	Внесение изменений в содержимое репозитория	11
4.5	Изменение файла hello.html	11
4.6	Изменение файла hello.html	12
4.7	Внесение нескольких изменений в содержимое репозитория	12
4.8	Внесение нескольких изменений в содержимое репозитория	13
4.9	Просмотр истории	14
4.10	Просмотр разных версий репозитория	15
4.11	Создание тегов версий	16
4.12	Переключение по имени тега и просмотр доступных тегов	16
4.13	Внесение нежелательного комментария в hello.html (до индексации)	17
4.14	Отмена локальных изменений (до индексации)	17
4.15	Отмена проиндексированных изменений (перед коммитом)	18
4.16	Отмена коммитов	19
4.17	Удаление коммитов из ветки	20
4.18	Ничего никогда не теряется	21
4.19	Удаление тега oops	22
4.20	Изменение файла hello.html	22
4.21	Изменение предыдущего коммита	23
4.22	Создание структуры репозитория	23
4.23	Создание, индексация, коммит нового файла	24
4.24	Результат открытия index.html	24
4.25	Каталог .git	25
4.26	Работа непосредственно с объектами git	26
4.27	Самостоятельная работа непосредственно с объектами git	26
4.28	Создание ветки	27
4.29	Редактирование файла hwllo.html	27
4.30	Редактирование файла index.html	27
4.31	Создание ветки	28
4.32	Просмотр логов новой ветки	28
4.33	Сравнение изменений файла hello.html	29
4.34	Изменения в ветке master	29
4.35	Изменения в ветке master	30
4.36	Слияние веток	31
4.37	Создание конфликта	31

4.38 Создание конфликта	32
4.39 Конфликт	33
4.40 Разрешение конфликта	33
4.41 Коммит разрешения конфликта	33
4.42 Поиск коммита перед слиянием	34
4.43 Сброс ветки style	35
4.44 Поиск коммита перед конфликтом	36
4.45 Сброс ветки master	36
4.46 Перебазирование	37
4.47 Слияние style в master	38
4.48 Клонирование репозитория	38
4.49 Клонирование репозитория	39
4.50 Просмотр имени по умолчанию удаленного репозитория	39
4.51 Просмотр веток	40
4.52 Изменение оригинального репозитория	40
4.53 Извлечение изменений	41
4.54 Слияние извлеченных изменений	41
4.55 Добавление ветки наблюдения	42
4.56 Создание чистого репозитория	42
4.57 Отправка изменений	43
4.58 Извлечение изменений	43

1 Цель работы

Цель данной лабораторной работы - приобретение практических навыков работы с системой управления версиями Git.

2 Задание

1. Выполнить задания из лабораторной работы
2. Оформить отчет по лабораторной работе с помощью Markdown

3 Теоретическое введение

Git — распределённая система управления версиями. Проект был создан Линусом Торвальдсом для управления разработкой ядра Linux, первая версия выпущена 7 апреля 2005 года; координатор — Дзюн Хамано. [1].

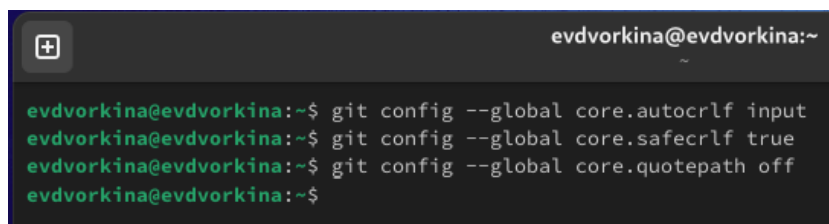
4 Выполнение лабораторной работы

Использованы материалы из [2].

Ранее я уже работаю с git, поэтому установка у меня уже осуществлена. Перейдем сразу к параметрам установки окончаний строк.

4.1 Подготовка

Сначала настроим `core.autocrlf` с параметрами `true` и `input`, чтобы сделать все переводы строк текстовых файлов в главном репозитории одинаковыми, а затем настроим отображение `unicode` (рис. 4.1).

A screenshot of a terminal window with a dark background. The window title is "evdvorkina@evdvorkina:~". The terminal shows four lines of commands and their outputs, all in green text. The commands are: "git config --global core.autocrlf input", "git config --global core.safecrlf true", "git config --global core.quotepath off", and a final prompt. The prompt "evdvorkina@evdvorkina:~\$" appears at the start of each line.

```
evdvorkina@evdvorkina:~$ git config --global core.autocrlf input
evdvorkina@evdvorkina:~$ git config --global core.safecrlf true
evdvorkina@evdvorkina:~$ git config --global core.quotepath off
evdvorkina@evdvorkina:~$
```

Рис. 4.1: Настройка git

4.2 Создание проекта

Создадим пустой каталог `hello`, а в нём файл с именем `hello.html`. В файл добавим строчку "Hello, World!". Затем создадим git репозиторий из этого каталога, выполнив команду `git init`. (рис. 4.2).

```

evdvorkina@evdvorkina:~$ mkdir hello
evdvorkina@evdvorkina:~$ cd hello/
evdvorkina@evdvorkina:~/hello$ touch hello.html
evdvorkina@evdvorkina:~/hello$ echo "Hello, World!" > hello.html
evdvorkina@evdvorkina:~/hello$ git init
Инициализирован пустой репозиторий Git в /home/evdvorkina/hello/.git/
evdvorkina@evdvorkina:~/hello$

```

Рис. 4.2: Создание репозитория

Добавим файл в репозиторий и проверим статус, мы увидим сообщение о том, что коммитить нечего (рис. 4.3).

```

Инициализирован пустой репозиторий Git в /home/evdvorkina/hello/.git/
evdvorkina@evdvorkina:~/hello$ git add hello.html
evdvorkina@evdvorkina:~/hello$ git commit -m "Initial Commit"
[master (корневой коммит) 482de9f] Initial Commit
1 file changed, 1 insertion(+)
create mode 100644 hello.html
evdvorkina@evdvorkina:~/hello$ git status
Текущая ветка: master
ничего коммитить, нет изменений в рабочем каталоге

```

Рис. 4.3: Добавление файла в репозиторий

4.3 Внесение изменений

Изменим содержимое файла hello.html на:

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

Проверив состояние рабочего каталога увидим, что git знает, что файл hello.html был изменен, но при этом эти изменения еще не зафиксированы в репозитории. Проиндексируем изменения и снова посмотрим статус, в нем указано, что изменения пока не записаны в репозиторий. И наконец закоммитим изменения, внеся их в репозиторий и снова посмотрим статус, который теперь показывает, что все изменения внесены в репозиторий (рис. 4.4).

```

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

evdvorkina@evdvorkina:~/hello$ git commit
[master e994932] Added h1 tag
1 file changed, 1 insertion(+), 1 deletion(-)
evdvorkina@evdvorkina:~/hello$ git status
Текущая ветка: master
ничего коммитить, нет изменений в рабочем каталоге

```

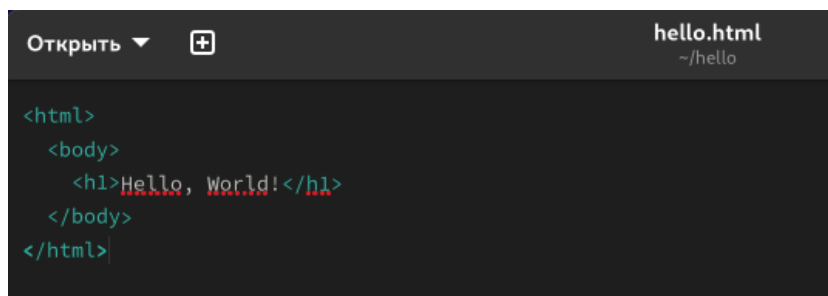
Рис. 4.4: Внесение изменений в содержимое репозитория

Изменим страницу «Hello, World», чтобы она содержала стандартные теги и (рис. 4.5).

```

<html>
  <body>
    <h1>Hello, World!</h1>
  </body>
</html>

```



```

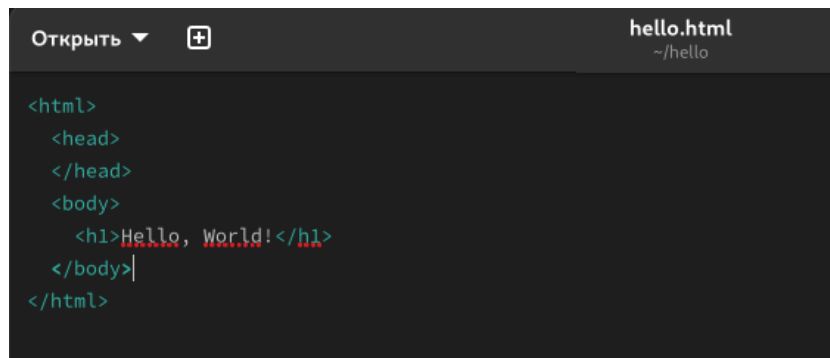
Открыть ▼ + hello.html
~/hello

<html>
  <body>
    <h1>Hello, World!</h1>
  </body>
</html>

```

Рис. 4.5: Изменение файла hello.html

Добавим это изменение в индекс git и добавим заголовки HTML (секцию) к странице «Hello, World». (рис. 4.6)

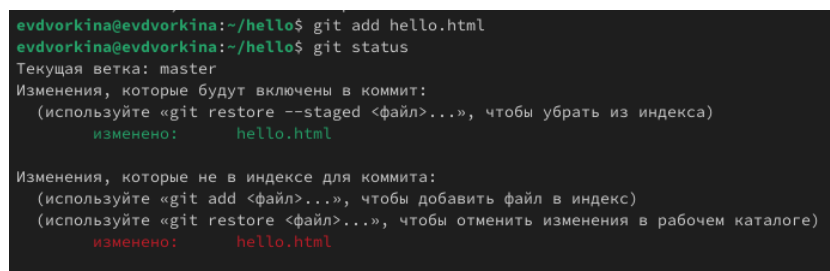


```
Открыть ▾ + hello.html
~/hello

<html>
  <head>
  </head>
  <body>
    <h1>Hello, World!</h1>
  </body>
</html>
```

Рис. 4.6: Изменение файла hello.html

Проверив текущий статус, увидим, что hello.html указан дважды в состоянии. Первое изменение (добавление стандартных тегов) проиндексировано и готово к коммиту. Второе изменение (добавление заголовков HTML) не проиндексировано. (рис. 4.7).



```
evdvorkina@evdvorkina:~/hello$ git add hello.html
evdvorkina@evdvorkina:~/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
```

Рис. 4.7: Внесение нескольких изменений в содержимое репозитория

Произведем коммит проиндексированного изменения. Проиндексируем оставшееся изменение, посмотрим статус и прокоммитим его тоже (рис. 4.8).

```

evdvorkina@evdvorkina:~/hello$ git commit -m "Addwd standatd HTML page tags"
[master acefaf6] Addwd standatd HTML page tags
1 file changed, 5 insertions(+), 1 deletion(-)
evdvorkina@evdvorkina:~/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»)
evdvorkina@evdvorkina:~/hello$ git add .
evdvorkina@evdvorkina:~/hello$ git status
Текущая ветка: master
Изменения, которые будут включены в коммит:
  (используйте «git restore --staged <файл>...», чтобы убрать из индекса)
    изменено:   hello.html

evdvorkina@evdvorkina:~/hello$ git commit -m "Added HTML header"
[master 7c6fbe6] Added HTML header
1 file changed, 2 insertions(+)
evdvorkina@evdvorkina:~/hello$

```

Рис. 4.8: Внесение нескольких изменений в содержимое репозитория

4.4 История

Получим список произведенных изменений в стандартном виде `git log`, затем в однострочном `git log --pretty=oneline` и далее с указанием времени и количества (рис. 4.9).

```

git log --pretty=oneline --max-count=2
git log --pretty=oneline --since='5 minutes ago'
git log --pretty=oneline --until='5 minutes ago'

```

```

evdvorkina@evdvorkina:~/hello$ git commit -m "Added HTML header"
[master 7c6fbc6] Added HTML header
1 file changed, 2 insertions(+)
evdvorkina@evdvorkina:~/hello$ git log
commit 7c6fbc6e6364b2a0fd1ef94bc90f38433f48096 (HEAD -> master)
Author: evdvorkina <1132226447@pfur.ru>
Date:   Fri Feb 21 22:48:40 2025 +0300

    Added HTML header

commit acefaf69070826755b1b795af2947e5bc290428e
Author: evdvorkina <1132226447@pfur.ru>
Date:   Fri Feb 21 22:46:51 2025 +0300

    Addwd standatd HTML page tags

commit e99493247cb5e556e21151e3fbcf127c80fee91
Author: evdvorkina <1132226447@pfur.ru>
Date:   Fri Feb 21 22:40:52 2025 +0300

    Added h1 tag

commit 482de9fa98aff3d83f14194a97df3838a59e46ac
Author: evdvorkina <1132226447@pfur.ru>
Date:   Fri Feb 21 22:34:39 2025 +0300

    Initial Commit

evdvorkina@evdvorkina:~/hello$ git log --pretty=oneline
7c6fbc6e6364b2a0fd1ef94bc90f38433f48096 (HEAD -> master) Added HTML header
acefaf69070826755b1b795af2947e5bc290428e Addwd standatd HTML page tags
e99493247cb5e556e21151e3fbcf127c80fee91 Added h1 tag
482de9fa98aff3d83f14194a97df3838a59e46ac Initial Commit
evdvorkina@evdvorkina:~/hello$ git log --pretty=oneline --max-count=2
7c6fbc6e6364b2a0fd1ef94bc90f38433f48096 (HEAD -> master) Added HTML header
acefaf69070826755b1b795af2947e5bc290428e Addwd standatd HTML page tags
evdvorkina@evdvorkina:~/hello$ git log --pretty=oneline --since='5 minutes ago'
7c6fbc6e6364b2a0fd1ef94bc90f38433f48096 (HEAD -> master) Added HTML header
acefaf69070826755b1b795af2947e5bc290428e Addwd standatd HTML page tags
evdvorkina@evdvorkina:~/hello$ git log --pretty=oneline --until='5 minutes ago'
e99493247cb5e556e21151e3fbcf127c80fee91 Added h1 tag
482de9fa98aff3d83f14194a97df3838a59e46ac Initial Commit

```

Рис. 4.9: Просмотр истории

4.5 Получение старых версий

Изучим данные лога и найдем там хэш первого коммита, используя его, вернемся к первой версии репозитория и посмотрим файл `hello.html`, действительно, увидим первую версию. Затем вернемся к последней версии в ветке `master` и посмотрим на тот же файл (рис. 4.10).

```
commit 482de9fa98aff3d83f14194a97df3838a59e46ac
Author: evdvorkina <1132226447@pfur.ru>
Date:   Fri Feb 21 22:34:39 2025 +0300

Initial Commit
evdvorkina@evdvorkina:~/hello$ git checkout 482de9f
Примечание: переключение на «482de9f».

Вы сейчас в состоянии «отсоединенного 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 482de9f Initial Commit
evdvorkina@evdvorkina:~/hello$ cat hello.html
Hello, World!
evdvorkina@evdvorkina:~/hello$ git checkout master
Предыдущая позиция HEAD была 482de9f Initial Commit
Переключились на ветку «master»
evdvorkina@evdvorkina:~/hello$ cat hello.html
<html>
<head>
</head>
<body>
  <h1>Hello, World!</h1>
</body>
</html>
```

Рис. 4.10: Просмотр разных версий репозитория

4.6 Создание тегов версий

Назовем текущую версию страницы hello первой (v1). Создадим тег первой версии `git tag v1` и используем его для того, чтобы вернуться к предыдущей версии, которой также присвоим тег `v1-beta` (рис. 4.11).

```

evdvorkina@evdvorkina:~/hello$ git tag v1
evdvorkina@evdvorkina:~/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 acefaf6 Addwd standatd HTML page tags
evdvorkina@evdvorkina:~/hello$ cat hello.html
<html>
  <body>
    <h1>Hello, World!</h1>
  </body>
</html>
evdvorkina@evdvorkina:~/hello$ git tag v1-beta

```

Рис. 4.11: Создание тегов версий

Переключимся по тегам между двумя отмеченными версиями. Просмотрим все доступные теги (их два) и посмотрим теги в логе (рис. 4.12).

```

evdvorkina@evdvorkina:~/hello$ git checkout v1
Предыдущая позиция HEAD была acefaf6 Addwd standatd HTML page tags
HEAD сейчас на 7c6fbe6 Added HTML header
evdvorkina@evdvorkina:~/hello$ git checkout v1-beta
Предыдущая позиция HEAD была 7c6fbe6 Added HTML header
HEAD сейчас на acefaf6 Addwd standatd HTML page tags
evdvorkina@evdvorkina:~/hello$ git tag
v1
v1-beta
evdvorkina@evdvorkina:~/hello$ git log master --all
commit 7c6fbe6e06364b2a0fd1ef94bc90f38433f48096 (tag: v1, master)
Author: evdvorkina <1132226447@pfur.ru>
Date:   Fri Feb 21 22:48:40 2025 +0300

    Added HTML header

commit acefaf69070826755b1b795af2947e5bc290428e (HEAD, tag: v1-beta)
Author: evdvorkina <1132226447@pfur.ru>
Date:   Fri Feb 21 22:46:51 2025 +0300

    Addwd standatd HTML page tags

commit e99493247cb5e556e21151e3fbef127c80feea91
Author: evdvorkina <1132226447@pfur.ru>
Date:   Fri Feb 21 22:40:52 2025 +0300

    Added h1 tag

commit 482de9fa98aff3d83f14194a97df3838a59e46ac
Author: evdvorkina <1132226447@pfur.ru>
Date:   Fri Feb 21 22:34:39 2025 +0300

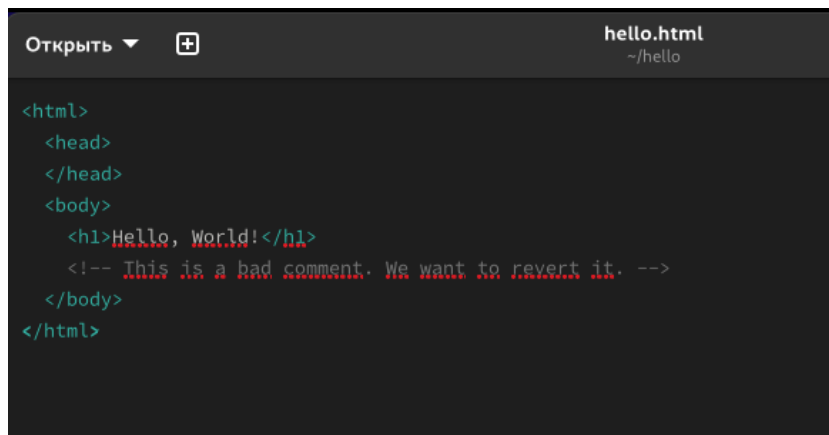
    Initial Commit

```

Рис. 4.12: Переключение по имени тега и просмотр доступных тегов

4.7 Отмена локальных изменений (до индексации)

Убедимся, что мы находимся на последнем коммите ветки master и внесем изменение в файл hello.html в виде нежелательного комментария (рис. 4.13).

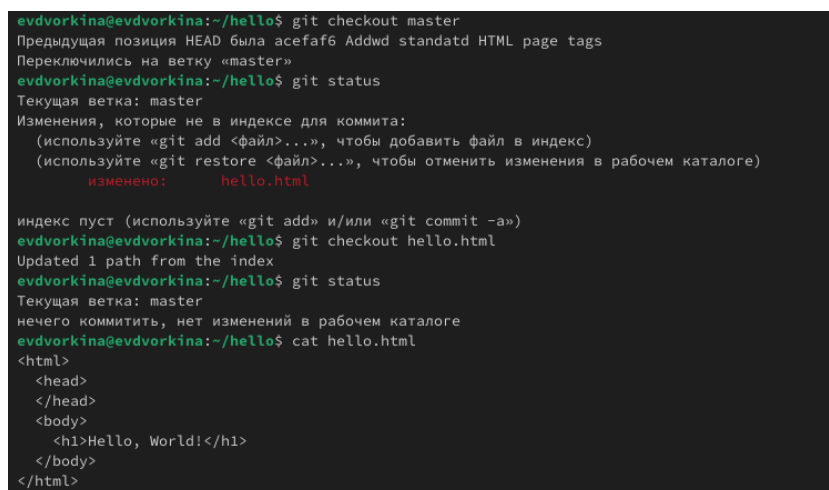


```
Открыть ▾ + hello.html
~/hello

<html>
  <head>
  </head>
  <body>
    <h1>Hello, World!</h1>
    <!-- This is a bad comment. We want to revert it. -->
  </body>
</html>
```

Рис. 4.13: Внесение нежелательного комментария в hello.html (до индексации)

Затем проверим статус, увидим, что изменения ещё не проиндексированы. Используем команду git checkout для переключения версии файла hello.html в репозитории, видим, что нежелательный комментарий отсутствует в файле (рис. 4.14).



```
evdvorkina@evdvorkina:~/hello$ git checkout master
Предыдущая позиция HEAD была acefaf6 Addwd standatd HTML page tags
Переключились на ветку «master»
evdvorkina@evdvorkina:~/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»)
evdvorkina@evdvorkina:~/hello$ git checkout hello.html
Updated 1 path from the index
evdvorkina@evdvorkina:~/hello$ git status
Текущая ветка: master
нечего коммитить, нет изменений в рабочем каталоге
evdvorkina@evdvorkina:~/hello$ cat hello.html
<html>
<head>
</head>
<body>
  <h1>Hello, World!</h1>
</body>
</html>
```

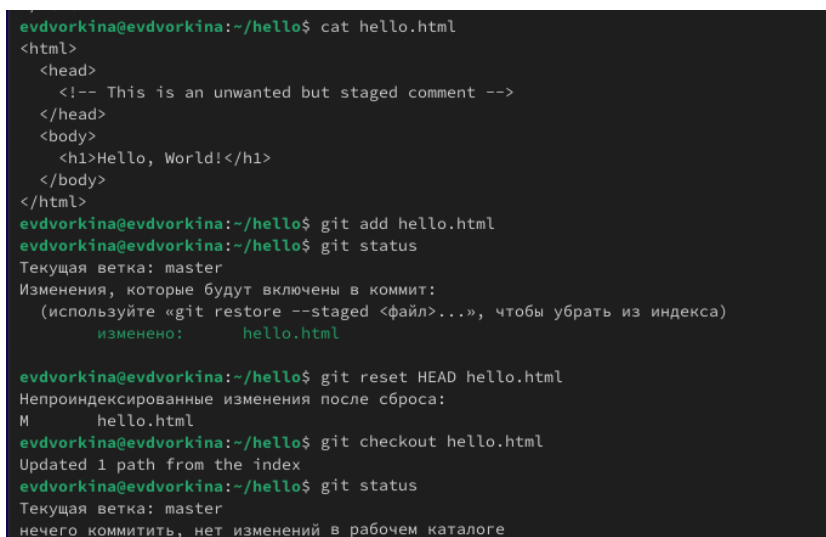
Рис. 4.14: Отмена локальных изменений (до индексации)

4.8 Отмена проиндексированных изменений (перед коммитом)

Внесем изменение в файл hello.html в виде нежелательного комментария

```
<html>
<head>
  <!-- This is an unwanted but staged comment -->
</head>
<body>
  <h1>Hello, World!</h1>
</body>
</html>
```

Проиндексируем это изменение и проверим состояние. Состояние показывает, что изменение было проиндексировано и готово к коммиту. Используем команду `git reset`, чтобы сбросить буферную зону к HEAD. Это очищает буферную зону от изменений, которые мы только что проиндексировали. Переключимся на последнюю версию коммита, посмотрев статус увидим, что наш каталог опять чист (рис. 4.15).



```
evdvorkina@evdvorkina:~/hello$ cat hello.html
<html>
<head>
  <!-- This is an unwanted but staged comment -->
</head>
<body>
  <h1>Hello, World!</h1>
</body>
</html>
evdvorkina@evdvorkina:~/hello$ git add hello.html
evdvorkina@evdvorkina:~/hello$ git status
Текущая ветка: master
Изменения, которые будут включены в коммит:
(используйте «git restore --staged <файл>...», чтобы убрать из индекса)
    изменено:   hello.html

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

Рис. 4.15: Отмена проиндексированных изменений (перед коммитом)

4.9 Отмена коммитов

Изменим файл `hello.html` на следующий.

```
<html>
  <head>
</head>
  <body>
    <h1>Hello, World!</h1>
    <!-- This is an unwanted but committed change -->
  </body>
</html>
```

Проиндексируем изменения файла и прокоммитим их. Чтобы отменить коммит, нам необходимо сделать коммит, который удаляет изменения, сохраненные нежелательным коммитом. Перейдем в редактор, где изменим нежелательный коммит. Проверим лог. Проверка лога показывает нежелательные и отмененные коммиты в нашем репозитории (рис. 4.16).

```
evdvorkina@evdvorkina:~/hello$ cat hello.html
<html>
  <head>
</head>
  <body>
    <h1>Hello, World!</h1>
    <!-- This is an unwanted but committed change -->
  </body>
</html>
evdvorkina@evdvorkina:~/hello$ git add hello.html
evdvorkina@evdvorkina:~/hello$ git commit -m "Oops, we didn't want this commit"
[master b47150d] Oops, we didn't want this commit
 1 file changed, 1 insertion(+)
evdvorkina@evdvorkina:~/hello$ git revert HEAD
[master 6396643] Revert "Oops, we didn't want this commit"
 1 file changed, 1 deletion(-)
evdvorkina@evdvorkina:~/hello$ git log
commit 639664324c052145c766cb8121af2fc73a205aae (HEAD -> master)
Author: evdvorkina <1132226447@pfur.ru>
Date:   Fri Feb 21 23:09:39 2025 +0300

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

    This reverts commit b47150d952dc71d8ce0f1c6d8114e845fb8a104b.

commit b47150d952dc71d8ce0f1c6d8114e845fb8a104b
Author: evdvorkina <1132226447@pfur.ru>
Date:   Fri Feb 21 23:09:02 2025 +0300

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

Рис. 4.16: Отмена коммитов

4.10 Удаление коммитов из ветки

Удалим последние два коммита с помощью сброса, сначала отметим последний коммит тегом, чтобы его можно было потом найти. Используем команду `git reset --hard v1`, чтобы вернуться к версии до этих коммитов. (рис. 4.17).

```
Initial Commit
evdvorkina@evdvorkina:~/hello$ git tag oops
evdvorkina@evdvorkina:~/hello$ git reset --hard v1
Указатель HEAD сейчас на коммите 7c6fbe6 Added HTML header
evdvorkina@evdvorkina:~/hello$ git log
commit 7c6fbe6e06364b2a0fd1ef94bc98f38433f48096 (HEAD -> master, tag: v1)
Author: evdvorkina <1132226447@pfur.ru>
Date:   Fri Feb 21 22:48:40 2025 +0300

    Added HTML header

commit acefaf69070826755b1b795af2947e5bc290428e (tag: v1-beta)
Author: evdvorkina <1132226447@pfur.ru>
Date:   Fri Feb 21 22:46:51 2025 +0300

    Addwd standatd HTML page tags

commit e99493247cb5e556e21151e3fbef127c80feea91
Author: evdvorkina <1132226447@pfur.ru>
Date:   Fri Feb 21 22:40:52 2025 +0300

    Added h1 tag

commit 482de9fa98aff3d83f14194a97df3838a59e46ac
Author: evdvorkina <1132226447@pfur.ru>
Date:   Fri Feb 21 22:34:39 2025 +0300

    Initial Commit
```

Рис. 4.17: Удаление коммитов из ветки

Теперь в логе их нет, но если посмотреть логи с опцией `--all` можно их увидеть, однако метка HEAD находится на нужной нам версии (рис. 4.18).

```
Initial Commit
evdvorkina@evdvorkina:~/hello$ git log --all
commit 639664324c052145c766cb8121af2fc73a205aae (tag: oops)
Author: evdvorkina <1132226447@pfur.ru>
Date:   Fri Feb 21 23:09:39 2025 +0300

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

    This reverts commit b47150d952dc71d8ce0f1c6d8114e845fb8a104b.

commit b47150d952dc71d8ce0f1c6d8114e845fb8a104b
Author: evdvorkina <1132226447@pfur.ru>
Date:   Fri Feb 21 23:09:02 2025 +0300

    Oops, we didn't want this commit

commit 7c6f6e06364b2a0fd1ef94bc90f38433f48096 (HEAD -> master, tag: v1)
Author: evdvorkina <1132226447@pfur.ru>
Date:   Fri Feb 21 22:48:40 2025 +0300

    Added HTML header

commit acefaf69070826755b1b795af2947e5bc290428e (tag: v1-beta)
Author: evdvorkina <1132226447@pfur.ru>
Date:   Fri Feb 21 22:46:51 2025 +0300

    Addwd standatd HTML page tags

commit e99493247cb5e556e21151e3fbef127c80feea91
Author: evdvorkina <1132226447@pfur.ru>
Date:   Fri Feb 21 22:40:52 2025 +0300

    Added h1 tag

commit 482de9fa98aff3d83f14194a97df3838a59e46ac
Author: evdvorkina <1132226447@pfur.ru>
Date:   Fri Feb 21 22:34:39 2025 +0300

    Initial Commit
```

Рис. 4.18: Ничего никогда не теряется

4.11 Удаление тега oops

Удалим тег oops и коммиты, на которые он ссылался, сборщиком мусора. Теперь этот тег не отображается в репозитории (рис. 4.19).

```
evdvorkina@evdvorkina:~/hello$ git tag -d oops
Метка «oops» удалена (была 6396643)
evdvorkina@evdvorkina:~/hello$ git log --all
commit 7c6fbe6e06364b2a0fd1ef94bc90f38433f48096 (HEAD -> master, tag: v1)
Author: evdvorkina <1132226447@pfur.ru>
Date:   Fri Feb 21 22:48:40 2025 +0300

    Added HTML header

commit acefaf69070826755b1b795af2947e5bc290428e (tag: v1-beta)
Author: evdvorkina <1132226447@pfur.ru>
Date:   Fri Feb 21 22:46:51 2025 +0300

    Addwd standatd HTML page tags

commit e99493247cb5e556e21151e3fbef127c80feea91
Author: evdvorkina <1132226447@pfur.ru>
Date:   Fri Feb 21 22:40:52 2025 +0300

    Added h1 tag

commit 482de9fa98aff3d83f14194a97df3838a59e46ac
Author: evdvorkina <1132226447@pfur.ru>
Date:   Fri Feb 21 22:34:39 2025 +0300

    Initial Commit
```

Рис. 4.19: Удаление тега oops

4.12 Изменение предыдущего коммита

Добавим в страницу комментарий автора (рис. 4.20).

```
Открыть ▾ + hello.html
~/hello

<!-- Author: Eva V. Dvorkina -->
<html>
  <head>
  </head>
  <body>
    <h1>Hello, World!</h1>
  </body>
</html>
```

Рис. 4.20: Изменение файла hello.html

Затем добавим изменения в репозиторий. Теперь мы хотим добавить в комментарий автора почту, обновим страницу hello, включив в неё почту. Чтобы у нас остался один коммит, а не два, изменим последний с помощью опции `--amend`, теперь в логах отображается последняя версия коммита (рис. 4.21).

```

evdvorkina@evdvorkina:~/hello$ git add hello.html
evdvorkina@evdvorkina:~/hello$ git commit -m "Add an author comment"
[master fb2bb04] Add an author comment
1 file changed, 1 insertion(+)
evdvorkina@evdvorkina:~/hello$ cat hello.html
<!-- Author: Eva V. Dvorkina -->
<html>
<head>
</head>
<body>
<h1>Hello, World!</h1>
</body>
</html>
evdvorkina@evdvorkina:~/hello$ cat hello.html
<!-- Author: Eva V. Dvorkina (1132226447@pfur.ru) -->
<html>
<head>
</head>
<body>
<h1>Hello, World!</h1>
</body>
</html>
evdvorkina@evdvorkina:~/hello$ git add hello.html
evdvorkina@evdvorkina:~/hello$ git commit --amend -m "Add an author/email comment"
[master 5df24b3] Add an author/email comment
Date: Fri Feb 21 23:16:42 2025 +0300
1 file changed, 1 insertion(+)
evdvorkina@evdvorkina:~/hello$ git log
commit 5df24b3c99b26530caac26b692ecb838d4a20faf (HEAD -> master)
Author: evdvorkina <1132226447@pfur.ru>
Date: Fri Feb 21 23:16:42 2025 +0300

    Add an author/email comment

commit 7c6fbe6e06364b2a0fd1ef94bc90f38433f48096 (tag: v1)
Author: evdvorkina <1132226447@pfur.ru>
Date: Fri Feb 21 23:16:42 2025 +0300

    Add an author comment

```

Рис. 4.21: Изменение предыдущего коммита

4.13 Перемещение файлов

Переместим наш файл в каталог `lib`. Для этого создадим его и используем команду `git mv`, сделаем коммит этого перемещения (рис. 4.22).

```

Initial Commit
evdvorkina@evdvorkina:~/hello$ mkdir lib
evdvorkina@evdvorkina:~/hello$ git mv hello.html lib/
evdvorkina@evdvorkina:~/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

evdvorkina@evdvorkina:~/hello$ git commit -m "Moved hello.html to lib"
[master 16ac787] Moved hello.html to lib
1 file changed, 0 insertions(+), 0 deletions(-)
rename hello.html => lib/hello.html (100%)

```

Рис. 4.22: Создание структуры репозитория

4.14 Подробнее о структуре

Добавим файл `index.html` в наш репозиторий, заполним файл:

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

Проиндексируем файл и сделаем коммит (рис. 4.23).

```
evdvorkina@evdvorkina:~/hello$ touch index.html
evdvorkina@evdvorkina:~/hello$ cat index.html
<html>
  <body>
    <iframe src="lib/hello.html" width="200" height="200" />
  </body>
</html>
evdvorkina@evdvorkina:~/hello$ git add index.html
evdvorkina@evdvorkina:~/hello$ git commit -m "Added index.html"
[master 4b1b645] Added index.html
 1 file changed, 5 insertions(+)
 create mode 100644 index.html
evdvorkina@evdvorkina:~/hello$
```

Рис. 4.23: Создание, индексация, коммит нового файла

Теперь при открытии `index.html` увидим кусок страницы `hello` в маленьком окошке (рис. 4.24).



Рис. 4.24: Результат открытия `index.html`

4.15 Git внутри: Каталог .git

Посмотрим каталог, в котором хранится вся информация `git ls -C .git/`

Затем посмотрим набор каталогов, имена которых состоят из 2 символов `ls -C .git/objects`

Имена каталогов являются первыми двумя буквами хэша `sha1` объекта, хранящегося в `git`. Посмотрим в один из каталогов с именем из 2 букв. `ls -C .git/objects/ac` Увидим файлы с именами из 38 символов. Это файлы, содержащие объекты, хранящиеся в `git`.

Посмотрим файл конфигурации, создающийся для каждого конкретного проекта `cat .git/config`. Затем посмотрим подкаталоги `.git/refs/heads` и `.git/refs/tags`, а также содержимое файла `v1`, в нём хранится хэш коммита, привязанный к тегу. Также посмотрим содержимое файла `HEAD`, который содержит ссылку на текущую ветку, в данный момент это ветка `master` (рис. 4.25).

```
evdvorkina@evdvorkina:~/hello$ ls -C .git/
COMMIT_EDITMSG  config  description  HEAD  hooks  index  info  logs  objects  ORIG_HEAD  refs
evdvorkina@evdvorkina:~/hello$ ls -C .git/objects/
0d  16  2c  4b  5d  63  6d  8a  ac  b3  b8  cd  f7  info
14  22  48  52  62  6a  7c  ab  ae  b4  bd  e9  fb  pack
evdvorkina@evdvorkina:~/hello$ ls -C .git/objects/ac
efaf69070826755b1b795af2947e5bc290428e
evdvorkina@evdvorkina:~/hello$ ls -C .git/objects/bd
cc2b691e4d9ca643fad3d512350c3d08bad4d5
evdvorkina@evdvorkina:~/hello$ cat .git/config
[core]
  repositoryformatversion = 0
  filemode = true
  bare = false
  logallrefupdates = true
evdvorkina@evdvorkina:~/hello$ ls .git/refs/
heads  tags
evdvorkina@evdvorkina:~/hello$ ls .git/refs/heads/
master
evdvorkina@evdvorkina:~/hello$ ls .git/refs/tags/
v1  v1-beta
evdvorkina@evdvorkina:~/hello$ cat .git/refs/tags/v1
7c6fbe6e06364b2a0fd1ef94bc90f38433f48096
evdvorkina@evdvorkina:~/hello$ cat .git/HEAD
ref: refs/heads/master
evdvorkina@evdvorkina:~/hello$
```

Рис. 4.25: Каталог .git

4.16 Работа непосредственно с объектами git

Найдем последний коммит через `log` и выведем его, используя `SHA1` хэш. Затем посмотрим дерево каталогов, ссылка на который идёт в последнем коммите,

выведем каталог lib и файл hello.html (рис. 4.26).

```
evdvorkina@evdvorkina:~/hello$ git log --max-count=1
commit 4b1b645228ac1e72cdbaa34005486061e6672017 (HEAD -> master)
Author: evdvorkina <1132226447@pfur.ru>
Date:   Fri Feb 21 23:22:55 2025 +0300

    Added index.html
evdvorkina@evdvorkina:~/hello$ git cat-file -t 4b1b645228ac1e72cdbaa34005486061e6672017
commit
evdvorkina@evdvorkina:~/hello$ git cat-file -p 4b1b645228ac1e72cdbaa34005486061e6672017
tree b85b3ac2ed8a623a49a8e23386245874cecc5f15
parent 16ac7871f9c6dc7bbb4daba298542920b02b2852
author evdvorkina <1132226447@pfur.ru> 1740169375 +0300
committer evdvorkina <1132226447@pfur.ru> 1740169375 +0300

Added index.html
evdvorkina@evdvorkina:~/hello$ git cat-file -p b85b3ac2ed8a623a49a8e23386245874cecc5f15
100644 blob aeeb538cd0610f43516914c29faae04e80a870d    index.html
040000 tree fb1d127ad9ca678db77f8e19ec462d30c51bba28    lib
evdvorkina@evdvorkina:~/hello$ git cat-file -p fb1d127ad9ca678db77f8e19ec462d30c51bba28
100644 blob 6a75e2eaf296b620149458cff0c01bbc43c81016    hello.html
evdvorkina@evdvorkina:~/hello$ git cat-file -p ^C
evdvorkina@evdvorkina:~/hello$ git cat-file -p 6a75e2eaf296b620149458cff0c01bbc43c81016
<!-- Author: Eva V. Dvorkina (1132226447@pfur.ru) -->
<html>
<head>
</head>
<body>
<h1>Hello, World!</h1>
</body>
</html>
evdvorkina@evdvorkina:~/hello$
```

Рис. 4.26: Работа непосредственно с объектами git

Исследуем git репозиторий вручную самостоятельно. Используя хэш родительского коммита последовательно дойдем до первой версии файла hello.html и посмотрим его(рис. 4.27).

```
commit 482de9fa98aff3d83f14194a97df3838a59e46ac
Author: evdvorkina <1132226447@pfur.ru>
Date:   Fri Feb 21 22:34:39 2025 +0300

    Initial Commit
evdvorkina@evdvorkina:~/hello$ git cat-file -p 482de9fa98aff3d83f14194a97df3838a59e46ac
tree 523cd318fc3b74ad3852cd34a74720ad6f7dd422
author evdvorkina <1132226447@pfur.ru> 1740166479 +0300
committer evdvorkina <1132226447@pfur.ru> 1740166479 +0300

Initial Commit
evdvorkina@evdvorkina:~/hello$ git cat-file -p 523cd318fc3b74ad3852cd34a74720ad6f7dd422
100644 blob 8ab686eafeb1f44702738c8b0f24f2567c36da6d    hello.html
evdvorkina@evdvorkina:~/hello$ git cat-file -p 8ab686eafeb1f44702738c8b0f24f2567c36da6d
Hello, World!
evdvorkina@evdvorkina:~/hello$
```

Рис. 4.27: Самостоятельная работа непосредственно с объектами git

4.17 Создание ветки

Создадим новую ветку «style» и перейдем в неё. Добавим туда файл стилей style.css и добавим его в репозиторий (рис. 4.28).

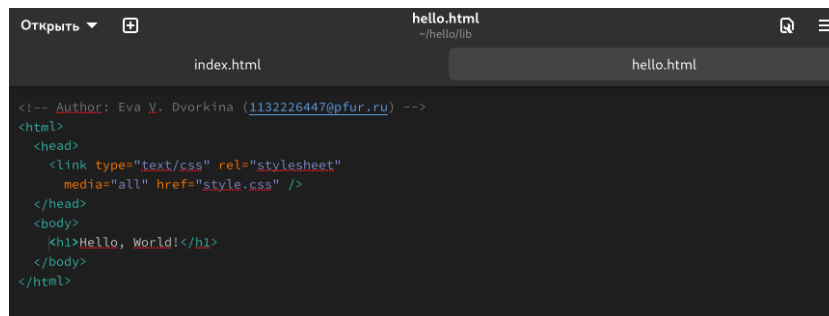
```

evdvorkina@evdvorkina:~/hello$ git checkout -b style
Переключились на новую ветку «style»
evdvorkina@evdvorkina:~/hello$ git status
Текущая ветка: style
нечего коммитить, нет изменений в рабочем каталоге
evdvorkina@evdvorkina:~/hello$ touch lib/style.css
evdvorkina@evdvorkina:~/hello$ cat lib/style.css
h1 {
  color: red;
}
evdvorkina@evdvorkina:~/hello$ git add lib/style.css
evdvorkina@evdvorkina:~/hello$ git commit -m "added css stylesheet"
[style 358c3b8] added css stylesheet
1 file changed, 3 insertions(+)
create mode 100644 lib/style.css
evdvorkina@evdvorkina:~/hello$

```

Рис. 4.28: Создание ветки

Обновим файл hello.html, чтобы использовать стили style.css и index.html (рис. 4.29).



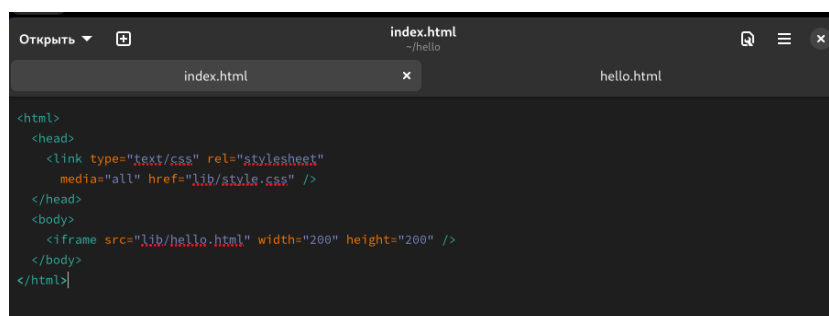
```

<!-- Author: Eva V. Dvorkina (113226447@pfur.ru) -->
<html>
<head>
  <link type="text/css" rel="stylesheet"
        media="all" href="style.css" />
</head>
<body>
  <h1>Hello, World!</h1>
</body>
</html>

```

Рис. 4.29: Редактирование файла hwllo.html

Обновив файл index.html (рис. 4.30).



```

<html>
<head>
  <link type="text/css" rel="stylesheet"
        media="all" href="lib/style.css" />
</head>
<body>
  <iframe src="lib/hello.html" width="200" height="200" />
</body>
</html>

```

Рис. 4.30: Редактирование файла index.html

Также обновим их в репозиторий (рис. 4.31).

```

evdvorkina@evdvorkina:~/hello$ git add lib/hello.html
evdvorkina@evdvorkina:~/hello$ git commit -m "Hello uses style.css"
[style d84fa62] Hello uses style.css
1 file changed, 2 insertions(+)
evdvorkina@evdvorkina:~/hello$ git add index.html
evdvorkina@evdvorkina:~/hello$ git commit -m "Updated Index.html"
[style 3cd9220] Updated Index.html
1 file changed, 4 insertions(+)
evdvorkina@evdvorkina:~/hello$

```

Рис. 4.31: Создание ветки

4.18 Навигация по веткам

Посмотрим все логи (рис. 4.32).

```

evdvorkina@evdvorkina:~/hello$ git log --all
commit 3cd922066dee1cd9836f72e2fa202664b9029885 (HEAD -> style)
Author: evdvorkina <1132226447@pfur.ru>
Date:   Fri Feb 21 23:46:21 2025 +0300

    Updated Index.html

commit d84fa62685a87a12dfda3598ae331d1fc70c79ea
Author: evdvorkina <1132226447@pfur.ru>
Date:   Fri Feb 21 23:44:59 2025 +0300

    Hello uses style.css

commit 358c3b829c3bf5250b45bb94e15648700d09864a
Author: evdvorkina <1132226447@pfur.ru>
Date:   Fri Feb 21 23:42:23 2025 +0300

    added css stylesheet

commit 4b1b645228ac1e72cdbaa34005486061e6672017 (master)
Author: evdvorkina <1132226447@pfur.ru>
Date:   Fri Feb 21 23:22:55 2025 +0300

    Added index.html

```

Рис. 4.32: Просмотр логов новой ветки

Переключимся обратно на основную ветку и посмотрим содержимое файла `lib/hello.html`, заметим, что он не использует стили, не изменился, также посмотрим содержимое этого файла в новой ветке `style`, файл в ней уже использует стили(рис. 4.33).

```

evdvorkina@evdvorkina:~/hello$ git checkout master
Переключились на ветку «master»
evdvorkina@evdvorkina:~/hello$ cat lib/hello.html
<!-- Author: Eva V. Dvorkina (1132226447@pfur.ru) -->
<html>
  <head>
  </head>
  <body>
    <h1>Hello, World!</h1>
  </body>
</html>
evdvorkina@evdvorkina:~/hello$ git checkout style
Переключились на ветку «style»
evdvorkina@evdvorkina:~/hello$ cat lib/hello.html
<!-- Author: Eva V. Dvorkina (1132226447@pfur.ru) -->
<html>
  <head>
    <link type="text/css" rel="stylesheet"
      media="all" href="style.css" />
  </head>
  <body>
    <h1>Hello, World!</h1>
  </body>
</html>
evdvorkina@evdvorkina:~/hello$

```

Рис. 4.33: Сравнение изменений файла hello.html

4.19 Изменения в ветке master

Вернемся в основную ветку и добавим файл README.md (рис. 4.34).

```

evdvorkina@evdvorkina:~/hello$ git checkout master
Переключились на ветку «master»
evdvorkina@evdvorkina:~/hello$ echo "This is the Hello World example from the git tutorial." > README.md
evdvorkina@evdvorkina:~/hello$ git add README.md
evdvorkina@evdvorkina:~/hello$ git commit -m "Added README"
[master a75cdf9] Added README
1 file changed, 1 insertion(+)
create mode 100644 README.md
evdvorkina@evdvorkina:~/hello$

```

Рис. 4.34: Изменения в ветке master

Посмотрим ветки и их различия `git log --graph --all` (рис. 4.35).

```

evdvorkina@evdvorkina:~/hello$ git log --graph --all
* commit a75cdf983041da983fb579238f8552687f625566 (HEAD -> master)
| Author: evdvorkina <1132226447@pfur.ru>
| Date:   Fri Feb 21 23:51:49 2025 +0300
|
|     Added README
|
| * commit 3cd922066dee1cd9836f72e2fa202664b9029885 (style)
| | Author: evdvorkina <1132226447@pfur.ru>
| | Date:   Fri Feb 21 23:46:21 2025 +0300
| |
| |     Updated Index.html
| |
| | * commit d84fa62685a87a12dfda3598ae331d1fc70c79ea
| | | Author: evdvorkina <1132226447@pfur.ru>
| | | Date:   Fri Feb 21 23:44:59 2025 +0300
| | |
| | |     Hello uses style.css
| | |
| | * commit 358c3b829c3bf5250b45bb94e15648700d09864a
| | / Author: evdvorkina <1132226447@pfur.ru>
| |   Date:   Fri Feb 21 23:42:23 2025 +0300
| |
| |     added css stylesheet
| |
| | * commit 4b1b645228ac1e72cdbaa34005486061e6672017
| | | Author: evdvorkina <1132226447@pfur.ru>
| | | Date:   Fri Feb 21 23:22:55 2025 +0300
| | |
| | |     Added index.html
| | |
| | * commit 16ac7871f9c6dc7bbb4daba298542920b02b2852
| | | Author: evdvorkina <1132226447@pfur.ru>
| | | Date:   Fri Feb 21 23:21:15 2025 +0300
| | |
| | |     Moved hello.html to lib

```

Рис. 4.35: Изменения в ветке master

4.20 Слияние

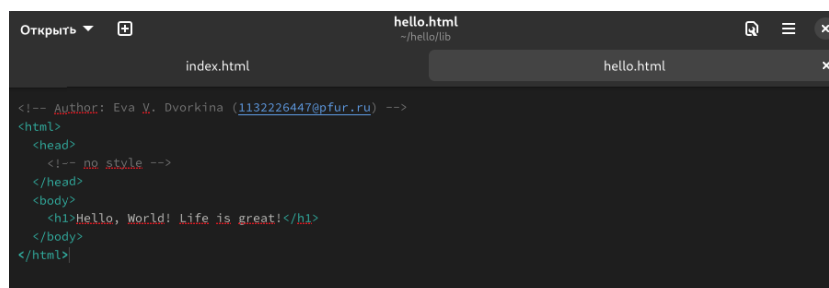
Слияние переносит изменения из двух веток в одну. Вернемся к ветке style и сольем master с style (рис. 4.36).

```
evdvorkina@evdvorkina:~/hello$ git checkout style
Переключились на ветку «style»
evdvorkina@evdvorkina:~/hello$ git merge master
Merge made by the 'ort' strategy.
 README.md | 1 +
 1 file changed, 1 insertion(+)
 create mode 100644 README.md
evdvorkina@evdvorkina:~/hello$ git log --graph --all
*   commit 54cd712e76d88684d410f28ff313c419f4630d56 (HEAD -> style)
| \ Merge: 3cd9220 a75cdf9
| | Author: evdvorkina <1132226447@pfur.ru>
| | Date:   Fri Feb 21 23:53:51 2025 +0300
| |
| | Merge branch 'master' into style
| |
| | * commit a75cdf983041da983fb579238f8552687f625566 (master)
| | | Author: evdvorkina <1132226447@pfur.ru>
| | | Date:   Fri Feb 21 23:51:49 2025 +0300
| | |
| | | Added README
| | |
| | * commit 3cd922066dee1cd9836f72e2fa202664b9029885
| | | Author: evdvorkina <1132226447@pfur.ru>
| | | Date:   Fri Feb 21 23:46:21 2025 +0300
| | |
| | | Updated Index.html
| | |
| | * commit d84fa62685a87a12dfda3598ae331d1fc70c79ea
| | | Author: evdvorkina <1132226447@pfur.ru>
| | | Date:   Fri Feb 21 23:44:59 2025 +0300
| | |
| | | Hello uses style.css
| | |
| | * commit 358c3b829c3bf5250b45bb94e15648700d09864a
| | | Author: evdvorkina <1132226447@pfur.ru>
| | | Date:   Fri Feb 21 23:42:23 2025 +0300
```

Рис. 4.36: Слияние веток

4.21 Создание конфликта

Вернемся в ветку master и создадим конфликт, внося изменения в файл hello.html (рис. 4.37).



```
Открыть ▾  hello.html
~/hello/lib

index.html  hello.html

<!-- Author: Eva Y. Dvorkina (1132226447@pfur.ru) -->
<html>
<head>
<!-- no style -->
</head>
<body>
<h1>Hello, World! Life is great!</h1>
</body>
</html>
```

Рис. 4.37: Создание конфликта

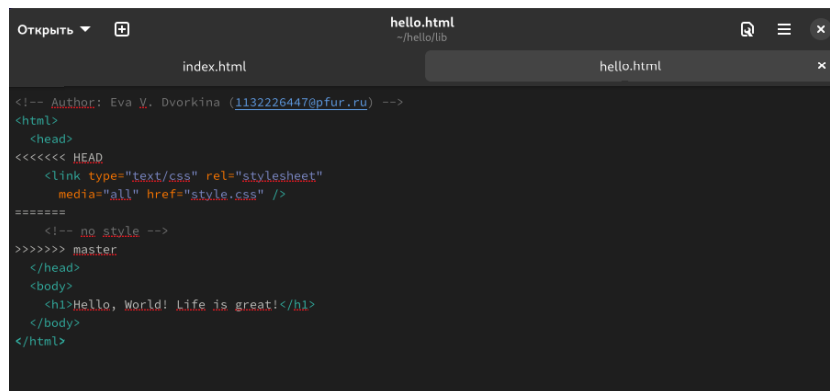
Просмотрим ветки. После коммита «Added README» ветка master была объединена с веткой style, но в настоящее время в master есть дополнительный коммит, который не был слит с style. Последнее изменение в master конфликтует с некоторыми изменениями в style (рис. 4.38).

```
evdvorkina@evdvorkina:~/hello$ git checkout master
Переключились на ветку «master»
evdvorkina@evdvorkina:~/hello$ git add lib/hello.html
evdvorkina@evdvorkina:~/hello$ git commit -m "Life is great"
[master d10534a] Life is great
1 file changed, 2 insertions(+), 1 deletion(-)
evdvorkina@evdvorkina:~/hello$ git log --graph --all
* commit d10534ad7571914b631573bd985ded8d9c10ef03 (HEAD -> master)
|
|   Life is great
|
| * commit 54cd712e76d88684d410f28ff313c419f4630d56 (style)
| |\ Merge: 3cd9220 a75cdf9
| |/ Author: evdvorkina <1132226447@pfur.ru>
|/  Date:   Fri Feb 21 23:53:51 2025 +0300
|
|   Merge branch 'master' into style
|
| * commit a75cdf983041da983fb579238f8552687f625566
| | Author: evdvorkina <1132226447@pfur.ru>
| | Date:   Fri Feb 21 23:51:49 2025 +0300
| |
| |   Added README
| |
| * commit 3cd922066deedcd9836f72e2fa202664b9029885
| | Author: evdvorkina <1132226447@pfur.ru>
| | Date:   Fri Feb 21 23:46:21 2025 +0300
| |
| |   Updated Index.html
| |
| * commit d84fa62685a87a12dfda3598ae331d1fc70c79ea
| | Author: evdvorkina <1132226447@pfur.ru>
| | Date:   Fri Feb 21 23:46:21 2025 +0300
| |
| |   Updated Index.html
```

Рис. 4.38: Создание конфликта

4.22 Разрешение конфликтов

Вернемся к ветке style и попытаемся объединить ее с новой веткой master. В файле lib/hello.html можно увидеть записи с обеих версий этого файла. Первый раздел — версия текущей ветки (style). Второй раздел — версия ветки master. Внесем изменения в lib/hello.html, оставив только необходимую нам запись и добавим этот файл в репозиторий, чтобы вручную разрешить конфликт (рис. 4.39, 4.40, 4.41).




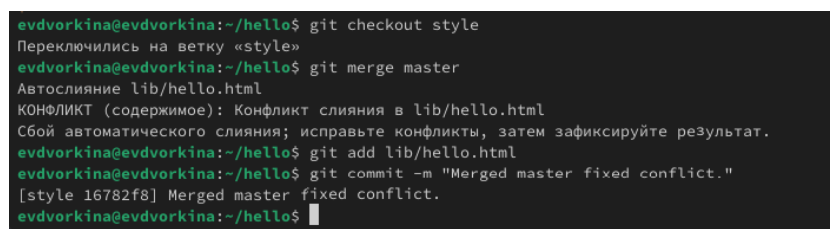
```
Открыть ▾  hello.html  
~/hello/lib  
index.html hello.html x  
  
<!-- Author: Eva V. Dvorkina (1132226447@pfur.ru) -->  
<html>  
  <head>  
    <link type="text/css" rel="stylesheet"  
      media="all" href="style.css" />  
    <!-- no style -->  
  </head>  
  <body>  
    <h1>Hello, World! Life is great!</h1>  
  </body>  
</html>
```

Рис. 4.39: Конфликт



```
Открыть ▾  hello.html  
~/hello/lib  
index.html hello.html x  
  
<!-- Author: Eva V. Dvorkina (1132226447@pfur.ru) -->  
<html>  
  <head>  
    <link type="text/css" rel="stylesheet"  
      media="all" href="style.css" />  
  </head>  
  <body>  
    <h1>Hello, World! Life is great!</h1>  
  </body>  
</html>
```

Рис. 4.40: Разрешение конфликта



```
evdvorkina@evdvorkina:~/hello$ git checkout style  
Переключились на ветку «style»  
evdvorkina@evdvorkina:~/hello$ git merge master  
Автослияние lib/hello.html  
КОНФЛИКТ (содержимое): Конфликт слияния в lib/hello.html  
Сбой автоматического слияния; исправьте конфликты, затем зафиксируйте результат.  
evdvorkina@evdvorkina:~/hello$ git add lib/hello.html  
evdvorkina@evdvorkina:~/hello$ git commit -m "Merged master fixed conflict."  
[style 16782f8] Merged master fixed conflict.  
evdvorkina@evdvorkina:~/hello$
```

Рис. 4.41: Коммит разрешения конфликта

4.23 Сброс ветки style

Вернемся на ветку style к точке перед тем, как мы слили ее с веткой master. Мы хотим вернуться в ветку style в точку перед слиянием с master. Нам необходимо найти последний коммит перед слиянием (рис. 4.42).

```

evdvorkina@evdvorkina:~/hello$ git checkout style
Уже на «style»
evdvorkina@evdvorkina:~/hello$ git log --graph
*   commit 16782f853317b035d322a5f154f23637996c83b0 (HEAD -> style)
| \
|  Merge: 54cd712 d10534a
|  Author: evdvorkina <1132226447@pfur.ru>
|  Date:   Sat Feb 22 00:00:00 2025 +0300
|
|      Merged master fixed conflict.
|
| *   commit d10534ad7571914b631573bd985ded8d9c10ef03 (master)
| |  Author: evdvorkina <1132226447@pfur.ru>
| |  Date:   Fri Feb 21 23:56:32 2025 +0300
| |
| |      Life is great
| |
| *   commit 54cd712e76d88684d410f28ff313c419f4630d56
| | \ Merge: 3cd9220 a75cdf9
| |  Author: evdvorkina <1132226447@pfur.ru>
| |  Date:   Fri Feb 21 23:53:51 2025 +0300
| |
| |      Merge branch 'master' into style
| |
| *   commit a75cdf983041da983fb579238f8552687f625566
| |  Author: evdvorkina <1132226447@pfur.ru>
| |  Date:   Fri Feb 21 23:51:49 2025 +0300
| |
| |      Added README
| |
| *   commit 3cd922066deed1cd9836f72e2fa202664b9029885
| |  Author: evdvorkina <1132226447@pfur.ru>
| |  Date:   Fri Feb 21 23:46:21 2025 +0300
| |
| |      Updated Index.html

```

Рис. 4.42: Поиск коммита перед слиянием

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

Поищем лог ветки style. Увидим, что у нас в истории больше нет коммитов слияний (рис. 4.43).

```
evdvorkina@evdvorkina:~/hello$ git reset --hard 3cd922066dee1cd9836f72e2fa202664b902988
Указатель HEAD сейчас на коммите 3cd9220 Updated Index.html
evdvorkina@evdvorkina:~/hello$ git log --graph --all
* commit d10534ad7571914b631573bd985ded8d9c10ef03 (master)
| Author: evdvorkina <1132226447@pfur.ru>
| Date: Fri Feb 21 23:56:32 2025 +0300
|
| Life is great
|
* commit a75cdf983041da983fb579238f8552687f625566
| Author: evdvorkina <1132226447@pfur.ru>
| Date: Fri Feb 21 23:51:49 2025 +0300
|
| Added README
|
* commit 3cd922066dee1cd9836f72e2fa202664b9029885 (HEAD -> style)
| Author: evdvorkina <1132226447@pfur.ru>
| Date: Fri Feb 21 23:46:21 2025 +0300
|
| Updated Index.html
|
* commit d84fa62685a87a12dfda3598ae331d1fc70c79ea
| Author: evdvorkina <1132226447@pfur.ru>
| Date: Fri Feb 21 23:44:59 2025 +0300
|
| Hello uses style.css
|
* commit 358c3b829c3bf5250b45bb94e15648700d09864a
| Author: evdvorkina <1132226447@pfur.ru>
| Date: Fri Feb 21 23:42:23 2025 +0300
```

Рис. 4.43: Сброс ветки style

4.24 Сброс ветки master

Добавив интерактивный режим в ветку master, мы внесли изменения, конфликтующие с изменениями в ветке style. Давайте вернемся в ветку master в точку перед внесением конфликтующих изменений. Это позволяет нам продемонстрировать работу команды `git rebase`, не беспокоясь о конфликтах. Просмотрим коммиты ветки master (рис. 4.44).

```

evdvorkina@evdvorkina:~/hello$ git checkout master
Переключились на ветку «master»
evdvorkina@evdvorkina:~/hello$ git log --graph
* commit d10534ad7571914b631573bd985ded8d9c10ef03 (HEAD -> master)
| Author: evdvorkina <1132226447@pfur.ru>
| Date:   Fri Feb 21 23:56:32 2025 +0300
|
|     Life is great
|
* commit a75cdf983041da983fb579238f8552687f625566
| Author: evdvorkina <1132226447@pfur.ru>
| Date:   Fri Feb 21 23:51:49 2025 +0300
|
|     Added README
|
* commit 4b1b645228ac1e72cdbaa34005486061e6672017
| Author: evdvorkina <1132226447@pfur.ru>
| Date:   Fri Feb 21 23:22:55 2025 +0300
|
|     Added index.html
|
* commit 16ac7871f9c6dc7bbb4daba298542920b02b2852
| Author: evdvorkina <1132226447@pfur.ru>
| Date:   Fri Feb 21 23:21:15 2025 +0300
|
|     Moved hello.html to lib
|
* commit 5df24b3c99b26530caac26b692ecb838d4a20faf

```

Рис. 4.44: Поиск коммита перед конфликтом

Коммит «Added README» идет непосредственно перед коммитом конфликтующего интерактивного режима. Мы сбросим ветку master к коммиту «Added README» (рис. 4.45).

```

evdvorkina@evdvorkina:~/hello$ git reset --hard a75cdf983041da983fb579238f8552687f625566
Указатель HEAD сейчас на коммите a75cdf9 Added README
evdvorkina@evdvorkina:~/hello$ git log --graph --all
* commit a75cdf983041da983fb579238f8552687f625566 (HEAD -> master)
| Author: evdvorkina <1132226447@pfur.ru>
| Date:   Fri Feb 21 23:51:49 2025 +0300
|
|     Added README
|
* commit 3cd922066dee1cd9836f72e2fa202664b9029885 (style)
| Author: evdvorkina <1132226447@pfur.ru>
| Date:   Fri Feb 21 23:46:21 2025 +0300
|
|     Updated Index.html
|
* commit d84fa62685a87a12dfda3598ae331d1fc70c79ea
| Author: evdvorkina <1132226447@pfur.ru>
| Date:   Fri Feb 21 23:44:59 2025 +0300
|
|     Hello uses style.css
|
* commit 358c3b829c3bf5250b45bb94e15648700d09864a
| Author: evdvorkina <1132226447@pfur.ru>
| Date:   Fri Feb 21 23:42:23 2025 +0300
|
|     added css stylesheet

```

Рис. 4.45: Сброс ветки master

4.25 Перебазирование

Используем команду `rebase` вместо команды `merge`. Мы вернулись в точку до первого слияния и хотим перенести изменения из ветки `master` в нашу ветку `style`. На этот раз для переноса изменений из ветки `master` мы будем использовать команду `git rebase` вместо слияния (рис. 4.46).

```
evdvorkina@evdvorkina:~/hello$ git checkout style
Переключились на ветку «style»
evdvorkina@evdvorkina:~/hello$ git rebase master
Успешно перемещён и обновлён refs/heads/style.
evdvorkina@evdvorkina:~/hello$ git log --graph
* commit df5be5d433c96b5a2cfd3a9c1abd5451149b83b (HEAD -> style)
| Author: evdvorkina <1132226447@pfur.ru>
| Date: Fri Feb 21 23:46:21 2025 +0300
|
| Updated Index.html
|
* commit 2a093836880a847fa399fef7b169f09145e4ddd7
| Author: evdvorkina <1132226447@pfur.ru>
| Date: Fri Feb 21 23:44:59 2025 +0300
|
| Hello uses style.css
|
* commit 95f911dc372d6e2d9c6703e2022e0ae01bf2f12a
| Author: evdvorkina <1132226447@pfur.ru>
| Date: Fri Feb 21 23:42:23 2025 +0300
|
| added css stylesheet
|
* commit a75cdf983041da983fb579238f8552687f625566 (master)
| Author: evdvorkina <1132226447@pfur.ru>
| Date: Fri Feb 21 23:51:49 2025 +0300
|
| Added README
|
* commit 4b1b645228ac1e72cdbaa34005486061e6672017
| Author: evdvorkina <1132226447@pfur.ru>
```

Рис. 4.46: Перебазирование

4.26 Слияние в ветку master

Вернемся в ветку `master` и сольем ветку `style` в неё с помощью команды `git merge` (рис. 4.47).

```

evdvorkina@evdvorkina:~/hello$ git checkout master
Переключились на ветку «master»
evdvorkina@evdvorkina:~/hello$ git merge style
Обновление a75cdf9..df5be5d
Fast-forward
 index.html      | 4 ++++
 lib/hello.html  | 2 ++
 lib/style.css   | 3 +++
 3 files changed, 9 insertions(+)
 create mode 100644 lib/style.css
evdvorkina@evdvorkina:~/hello$ git log
commit df5be5d433c96b5a2cfd3a9c1abd5451149b83b (HEAD -> master, style)
Author: evdvorkina <1132226447@pfur.ru>
Date:   Fri Feb 21 23:46:21 2025 +0300

    Updated Index.html

commit 2a093836880a847fa399fef7b169f09145e4ddd7
Author: evdvorkina <1132226447@pfur.ru>
Date:   Fri Feb 21 23:44:59 2025 +0300

    Hello uses style.css

commit 95f011dc373d6a2d9c6763e7937e0ae01bf2f12a

```

Рис. 4.47: Слияние style в master

4.27 Клонирование репозитория

Перейдем в наш рабочий каталог и сделаем клон репозитория hello. Просмотрев его, увидим список всех файлов на верхнем уровне оригинального репозитория README.md, index.html и lib (рис. 4.48).

```

evdvorkina@evdvorkina:~/hello$ cd ..
evdvorkina@evdvorkina:~$ pwd
/home/evdvorkina
evdvorkina@evdvorkina:~$ ls
hello  Видео  Загрузки  Музыка  'Рабочий стол'
work  Документы  Изображения  Общедоступные  Шаблоны
evdvorkina@evdvorkina:~$ git clone hello cloned_hello
Клонирование в «cloned_hello»...
готово.
evdvorkina@evdvorkina:~$ ls
cloned_hello  work  Документы  Изображения  Общедоступные  Шаблоны
hello        Видео  Загрузки  Музыка  'Рабочий стол'
evdvorkina@evdvorkina:~$

```

Рис. 4.48: Клонирование репозитория

Затем посмотрим историю репозитория и увидим список всех коммитов в новый репозиторий, и он совпадает с историей коммитов в оригинальном репозитории. Единственное отличие в названиях веток (рис. 4.49).

```

evdvorkina@evdvorkina:~$ cd cloned_hello/
evdvorkina@evdvorkina:~/cloned_hello$ ls
index.html  lib  README.md
evdvorkina@evdvorkina:~/cloned_hello$ git log --all
commit df5be5d433c96b5a2cfd3a9c1abd5451149b83b (HEAD -> master, origin/style, origin/master, origin/HEAD)
Author: evdvorkina <1132226447@pfur.ru>
Date:   Fri Feb 21 23:46:21 2025 +0300

    Updated Index.html

commit 2a093836880a847fa399fef7b169f09145e4ddd7
Author: evdvorkina <1132226447@pfur.ru>
Date:   Fri Feb 21 23:44:59 2025 +0300

    Hello uses style.css

commit 95f911dc372d6e2d9c6703e2022e0ae01bf2f12a
Author: evdvorkina <1132226447@pfur.ru>
Date:   Fri Feb 21 23:42:23 2025 +0300

    added css stylesheet

commit a75cdf983041da983fb579238f8552687f625566
Author: evdvorkina <1132226447@pfur.ru>
Date:   Fri Feb 21 23:51:49 2025 +0300

```

Рис. 4.49: Клонирование репозитория

4.28 Что такое origin?

Клонированный репозиторий знает об имени по умолчанию удаленного репозитория. Посмотрим, подробную информацию об имени по умолчанию. Давайте посмотрим, можем ли мы получить более подробную информацию об имени по умолчанию (рис. 4.50).

```

evdvorkina@evdvorkina:~/cloned_hello$ git remote
origin
evdvorkina@evdvorkina:~/cloned_hello$ git remote show origin
* внешний репозиторий origin
  URL для извлечения: /home/evdvorkina/hello
  URL для отправки: /home/evdvorkina/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)
evdvorkina@evdvorkina:~/cloned_hello$

```

Рис. 4.50: Просмотр имени по умолчанию удаленного репозитория

4.29 Удаленные ветки

Посмотрим на ветки, доступные в нашем клонированном репозитории. Можно увидеть, что в списке только ветка master. Для того, чтобы увидеть все ветки используем опцию -a (рис. 4.51).

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

Рис. 4.51: Просмотр веток

4.30 Изменение оригинального репозитория

Перейдем в репозиторий hello. Внесем изменения в файл README.md. Затем добавим их в репозиторий (рис. 4.52).

```
evdvorkina@evdvorkina:~/hello$ cat README.md
This is the Hello World example from the git tutorial (changed).
evdvorkina@evdvorkina:~/hello$ git add README.md
evdvorkina@evdvorkina:~/hello$ git commit -m "Changed README in original repo"
[master d682093] Changed README in original repo
 1 file changed, 1 insertion(+), 1 deletion(-)
evdvorkina@evdvorkina:~/hello$
```

Рис. 4.52: Изменение оригинального репозитория

Перейдём в клон репозитория и используем команду `git fetch`, которая будет извлекать новые коммиты из удаленного репозитория, но не будет сливать их с наработками в локальных ветках (рис. 4.53).


```

evdvorkina@evdvorkina:~/hello$ cd ../cloned_hello/
evdvorkina@evdvorkina:~/cloned_hello$ git fetch
remote: Перечисление объектов: 5, готово.
remote: Подсчет объектов: 100% (5/5), готово.
remote: Сжатие объектов: 100% (3/3), готово.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
Распаковка объектов: 100% (3/3), 369 байтов | 73.00 КиБ/с, готово.
Из /home/evdvorkina/hello
   df5be5d..d682093  master    -> origin/master
evdvorkina@evdvorkina:~/cloned_hello$ git log --all
commit d6820935b64e50eb3938868f4ac2bc438a9b5927 (origin/master, origin/HEAD)
Author: evdvorkina <1132226447@pfur.ru>
Date:   Sat Feb 22 00:17:17 2025 +0300

    Changed README in original repo

commit df5be5d433c96b5a2cfd3a9c1abd5451149b83b (HEAD -> master, origin/style)
Author: evdvorkina <1132226447@pfur.ru>
Date:   Fri Feb 21 23:46:21 2025 +0300

    Updated Index.html

commit 2a093836880a847fa399fef7b169f09145e4ddd7
Author: evdvorkina <1132226447@pfur.ru>
Date:   Fri Feb 21 23:44:59 2025 +0300

    Hello uses style.css

```

Рис. 4.53: Извлечение изменений

4.31 Слияние извлеченных изменений

Сольем внесённые изменения в главную ветку с помощью `git pull`, которая является объединением `fetch` и `merge` в одну команду (рис. -fig. 4.54).

```

evdvorkina@evdvorkina:~/cloned_hello$ cat README.md
This is the Hello World example from the git tutorial.
evdvorkina@evdvorkina:~/cloned_hello$ git merge origin/master
Обновление df5be5d..d682093
Fast-forward
 README.md | 2 +-
 1 file changed, 1 insertion(+), 1 deletion(-)
evdvorkina@evdvorkina:~/cloned_hello$ cat README.md
This is the Hello World example from the git tutorial (changed).
evdvorkina@evdvorkina:~/cloned_hello$ git pull
Уже актуально.
evdvorkina@evdvorkina:~/cloned_hello$ _

```

Рис. 4.54: Слияние извлеченных изменений

4.32 Добавление ветки наблюдения

Добавим локальную ветку, которая отслеживает удаленную ветку, теперь мы можем видеть ветку `style` в списке веток и логе (рис. 4.55).

```

evdvorkina@evdvorkina:~/cloned_hello$ git branch --track style origin/style
branch 'style' set up to track 'origin/style'.
evdvorkina@evdvorkina:~/cloned_hello$ git branch -a
* master
  style
  remotes/origin/HEAD -> origin/master
  remotes/origin/master
  remotes/origin/style
evdvorkina@evdvorkina:~/cloned_hello$ git log --max-count=2
commit d6820935b64e50eb3938868f4ac2bc438a9b5927 (HEAD -> master, origin/master, origin/HEAD)
Author: evdvorkina <1132226447@pfur.ru>
Date: Sat Feb 22 00:17:17 2025 +0300

    Changed README in original repo

commit df5be5d433c96b5a2cfd3a9c1abd5451149b83b (origin/style, style)
Author: evdvorkina <1132226447@pfur.ru>
Date: Fri Feb 21 23:46:21 2025 +0300

    Updated Index.html

```

Рис. 4.55: Добавление ветки наблюдения

4.33 Создание чистого репозитория

Как правило, репозитории, оканчивающиеся на .git являются чистыми репозиториями. Создадим такой в рабочем каталоге. Затем добавим репозиторий hello.git к нашему оригинальному репозиторию (рис. 4.56).

```

evdvorkina@evdvorkina:~/cloned_hello$ cd ..
evdvorkina@evdvorkina:~$ git clone --bare hello hello.git
Клонирование в голый репозиторий «hello.git»...
готово.
evdvorkina@evdvorkina:~$ ls hello.git
config description HEAD hooks info objects packed-refs refs
evdvorkina@evdvorkina:~$

```

Рис. 4.56: Создание чистого репозитория

4.34 Отправка и извлечение изменений

Так как чистые репозитории, как правило, расшариваются на каком-нибудь сетевом сервере, нам необходимо отправить наши изменения в другие репозитории. Начнем с создания изменения для отправки. Отредактируем файл README.md и сделаем коммит, затем отправим изменения в общий репозиторий. Затем извлечем изменения из общего репозитория (рис. 4.57, 4.58).

```

evdvorkina@evdvorkina:~$ cd hello
evdvorkina@evdvorkina:~/hello$ git remote add shared ../hello.git
evdvorkina@evdvorkina:~/hello$ cat README.md
This is the Hello World example from the git tutorial (changed).
(Changed in the original and pushed to shared)
evdvorkina@evdvorkina:~/hello$ git checkout master
M       README.md
Уже на «master»
evdvorkina@evdvorkina:~/hello$ git add README.md
evdvorkina@evdvorkina:~/hello$ git commit -m "Added shared comment to readme"
[master 0e6108a] Added shared comment to readme
 1 file changed, 1 insertion(+)
evdvorkina@evdvorkina:~/hello$ git push shared master
Перечисление объектов: 5, готово.
Подсчет объектов: 100% (5/5), готово.
При сжатии изменений используется до 3 потоков
Сжатие объектов: 100% (3/3), готово.
Запись объектов: 100% (3/3), 409 байтов | 409.00 КиБ/с, готово.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To ../hello.git
   d682093..0e6108a  master -> master
evdvorkina@evdvorkina:~/hello$

```

Рис. 4.57: Отправка изменений

```

evdvorkina@evdvorkina:~/hello$ cd ../cloned_hello
evdvorkina@evdvorkina:~/cloned_hello$ git remote add shared ../hello.git
evdvorkina@evdvorkina:~/cloned_hello$ git branch --track shared master
branch 'shared' set up to track 'master'.
evdvorkina@evdvorkina:~/cloned_hello$ pull shared master
bash: pull: команда не найдена...
^[[A^C
evdvorkina@evdvorkina:~/cloned_hello$ git pull shared master
remote: Перечисление объектов: 5, готово.
remote: Подсчет объектов: 100% (5/5), готово.
remote: Сжатие объектов: 100% (3/3), готово.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
Распаковка объектов: 100% (3/3), 389 байтов | 35.00 КиБ/с, готово.
Из ../hello
 * branch            master      -> FETCH_HEAD
 * [новая ветка]     master      -> shared/master
Обновление d682093..0e6108a
Fast-forward
 README.md | 1 +
 1 file changed, 1 insertion(+)
evdvorkina@evdvorkina:~/cloned_hello$ cat README.md
This is the Hello World example from the git tutorial (changed).
(Changed in the original and pushed to shared)
evdvorkina@evdvorkina:~/cloned_hello$

```

Рис. 4.58: Извлечение изменений

5 Выводы

В процессе выполнения данной лабораторной работы я приобрела практические навыки работы с Git.

Список литературы

1. Git [Электронный ресурс]. URL: <https://ru.wikipedia.org/wiki/Git>.
2. Работа с git [Электронный ресурс]. URL: https://esystem.rudn.ru/pluginfile.php/2584275/mod_folder/content/0/git.pdf.