

Лабораторная работа №1

Работа с git

Кудякова С. А.

19 февраля 2026

Российский университет дружбы народов, Москва, Россия

- Кудякова София Андреевна
- студентка
- группа НФИбд-01-23
- Российский университет дружбы народов
- 1132236033@rudn.ru
- <https://github.com/sakudyakova>

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

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.

Выполнение лабораторной работы

```
sakudyakova@dk3n55 ~/work/study/2025-2026/Имитационное моделирование/simmod $ cd  
sakudyakova@dk3n55 ~ $ git config --global core.autocrlf input  
sakudyakova@dk3n55 ~ $ git config --global core.safecrlf true  
sakudyakova@dk3n55 ~ $ git config --global core.quotepath off
```

Рис. 1: Настройка git

```
sakudyakova@dk3n55 ~ $ mkdir hello
sakudyakova@dk3n55 ~ $ cd hello/
sakudyakova@dk3n55 ~/hello $ touch hello.html
sakudyakova@dk3n55 ~/hello $ echo "Hello, World!" > hello.html
sakudyakova@dk3n55 ~/hello $ git init
Инициализирован пустой репозиторий Git в /afs/.dk.sci.pfu.edu.ru/home/s/a/sakudyakova/hello/.git/
```

Рис. 2: Создание репозитория


```
sakudyakova@dk3n55 ~/hello $ git add hello.html
sakudyakova@dk3n55 ~/hello $ git commit -m "Initial Commit"

error: gpg failed to sign the data:
[GNUPG:] KEY_CONSIDERED B009F83EAE23FCAB710CA4B99909797C7EE26FB7 2
[GNUPG:] BEGIN_SIGNING H8
[GNUPG:] PINENTRY_LAUNCHED 9266 gnome3 1.3.2-unknown - xterm-256color :1 - 526
gpg: подписать не удалось: Операция отменена
[GNUPG:] FAILURE sign 83886179
gpg: signing failed: Операция отменена

fatal: сбой записи объекта коммита
sakudyakova@dk3n55 ~/hello $ git commit -m "Initial Commit"
[master (корневой коммит) 5fe58b8] Initial Commit
 1 file changed, 1 insertion(+)
 create mode 100644 hello.html
sakudyakova@dk3n55 ~/hello $ git status
Текущая ветка: master
ничего коммитить, нет изменений в рабочем каталоге
```

Рис. 3: Добавление файла в репозиторий

Изменим содержимое файла hello.html на:

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

```
sakudyakova@dk3n55 ~/hello $ nano hello.html
sakudyakova@dk3n55 ~/hello $ git add hello.html
sakudyakova@dk3n55 ~/hello $ git status
Текущая ветка: master
Изменения, которые будут включены в коммит:
  (используйте «git restore --staged <файл>...», чтобы убрать из индекса)
        изменено:      hello.html

sakudyakova@dk3n55 ~/hello $ git commit
[master f250f17] <html>                <body>                <h1>Hello, World!</h1>  </body> </html>
 1 file changed, 1 insertion(+), 1 deletion(-)
sakudyakova@dk3n55 ~/hello $ git status
Текущая ветка: master
ничего коммитить, нет изменений в рабочем каталоге
```

Рис. 4: Внесение изменений в содержимое репозитория

Изменим страницу «Hello, World», чтобы она содержала стандартные теги

и

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

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



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

Рис. 5: 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 проиндексированно.

```
sakudyakova@dk3n55 ~/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

sakudyakova@dk3n55 ~/hello $
```

Рис. 6: Внесение нескольких изменений в содержимое репозитория

Произведем коммит проиндексированного изменения. Проиндексируем оставшееся изменение, посмотрим статус и прокоммитим его тоже

```
sakudyakova@dk3n55 ~/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

sakudyakova@dk3n55 ~/hello $
```

Рис. 7: Внесение нескольких изменений в содержимое репозитория

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

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

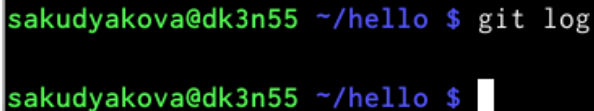
```
git log --pretty=oneline --since='5 minutes ago'
```

```
git log --pretty=oneline --until='5 minutes ago'
```

```
sakudyakova@dk3n55 ~/hello $ git commit -m"Added standard HTML page tags"
[master b7d2665] Added standard HTML page tags
 1 file changed, 5 insertions(+), 1 deletion(-)
sakudyakova@dk3n55 ~/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»)
sakudyakova@dk3n55 ~/hello $ git add .
sakudyakova@dk3n55 ~/hello $ git status
Текущая ветка: master
Изменения, которые будут включены в коммит:
  (используйте «git restore --staged <файл>...», чтобы убрать из индекса)
    изменено:      hello.html
```


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

A terminal window with a black background. The prompt is 'sakudyakova@dk3n55 ~/hello \$'. The command 'git log' has been entered. A second line shows the same prompt with a white cursor block at the end, indicating the command is being executed or about to be.

```
sakudyakova@dk3n55 ~/hello $ git log  
  
sakudyakova@dk3n55 ~/hello $
```

Рис. 9: Просмотр разных версий репозитория

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

```
sakudyakova@dk3n55 ~/hello $ git tag v1
sakudyakova@dk3n55 ~/hello $ git checkout v1^
git: «checkout» не является командой git. Смотрите «git --help».
```

Самые похожие команды:
checkout

```
sakudyakova@dk3n55 ~/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 `-c`. Например:

```
git switch -c <новая-ветка>
```

Или отмените эту операцию с помощью:

```
git switch -
```

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

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

A terminal window with a black background. The text 'v1' and 'v1-beta' is displayed on the first two lines. The third line shows 'lines 1-2/2 (END)' with a white cursor at the end of the text.

```
v1  
v1-beta  
lines 1-2/2 (END)
```

Рис. 11: Переключение по имени тега и просмотр доступных тегов

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



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

Рис. 12: Внесение нежелательного комментария в hello.html (до индексации)

Отмена локальных изменений (до индексации)

```
sakudyakova@dk3n55 ~/hello $ git checkout master
error: Ваши локальные изменения в указанных файлах будут перезаписаны при переключении на состояние:
      hello.html
Сделайте коммит или спрячьте ваши изменения перед переключением веток.
Прерываю
sakudyakova@dk3n55 ~/hello $ git status
Отсоединённый указатель HEAD указывает на v1-beta
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»)
sakudyakova@dk3n55 ~/hello $ git checkout hello.html
Updated 1 path from the index
sakudyakova@dk3n55 ~/hello $ git status
Отсоединённый указатель HEAD указывает на v1-beta
нечего коммитить, нет изменений в рабочем каталоге
sakudyakova@dk3n55 ~/hello $ cat hello.html
<html>
  <body>
    <h1>Hello, World!</h1>
  </body>
</html>
sakudyakova@dk3n55 ~/hello $
```

Рис. 13: Отмена локальных изменений (до индексации)

Отмена проиндексированных изменений (перед коммитом)

```
sakudyakova@dk3n55 ~/hello $ cat hello.html
<html>
  <head>
    <!-- This is a bad comment. We want to revert it. -->
  </head>
  <body>
    <h1>Hello, World!</h1>
  </body>
</html>
sakudyakova@dk3n55 ~/hello $ git add hello.html
sakudyakova@dk3n55 ~/hello $ git status
bash: git: команда не найдена
sakudyakova@dk3n55 ~/hello $ git status
Отсоединённый указатель HEAD указывает на v1-beta
Изменения, которые будут включены в коммит:
  изменено:   hello.html

sakudyakova@dk3n55 ~/hello $ git reset HEAD hello.html
Непроиндексированные изменения после сброса:
M       hello.html
sakudyakova@dk3n55 ~/hello $ git checkout hello.html
Updated 1 path from the index
sakudyakova@dk3n55 ~/hello $ git status
Отсоединённый указатель HEAD указывает на v1-beta
ничего коммитить, нет изменений в рабочем каталоге
sakudyakova@dk3n55 ~/hello $
```

Рис. 14: Отмена проиндексированных изменений (перед коммитом)

Изменим файл hello.html на следующий.

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

```
commit 50b500a227a3d03cc199b1f70df86edff7fdd00d (HEAD)
Author: "Sofia <"1132236033@pfur.ru">
Date:   Wed Feb 18 15:24:48 2026 +0300

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

    This reverts commit e85cadec4edf7a645d1c37350c9543503c6df743.
```

```
commit e85cadec4edf7a645d1c37350c9543503c6df743
Author: "Sofia <"1132236033@pfur.ru">
Date:   Wed Feb 18 15:24:33 2026 +0300
```

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

```
commit b7d26657abeccf381ea0f7b0e91fd9470e0453b6 (tag: v1-beta)
Author: "Sofia <"1132236033@pfur.ru">
Date:   Thu Feb 12 18:20:36 2026 +0300
```

```
    Added standard HTML page tags
```

```
commit f250f177df85ae90ce99985e60c0b206c7de29e3
Author: "Sofia <"1132236033@pfur.ru">
Date:   Thu Feb 12 17:51:45 2026 +0300
```

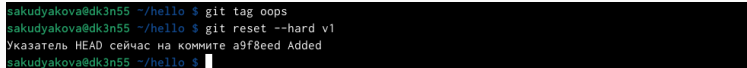
```
    <html>
      <body>
        <h1>Hello, World!</h1>
      </body>
    </html>
```

```
commit 5fe58b87d62fe19235d8ebd767bdbbfff327311a0
Author: "Sofia <"1132236033@pfur.ru">
Date:   Thu Feb 12 17:37:54 2026 +0300
```

```
    Initial Commit
```

```
lines 1-35/35 (END)
```


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

A terminal window with a black background and green text. It shows a sequence of three commands and their output. The first command is 'git tag oops', the second is 'git reset --hard v1', and the third is 'git status'. The output of the second command indicates that HEAD is now at commit a9f8eed and that two commits have been added.

```
sakudyakova@dk3n55 ~/hello $ git tag oops
sakudyakova@dk3n55 ~/hello $ git reset --hard v1
Указатель HEAD сейчас на коммите a9f8eed Added
sakudyakova@dk3n55 ~/hello $
```

Рис. 16: Удаление коммитов из ветки

Удаление коммиттов из ветки

Теперь в логе их нет, но если посмотреть логи с опцией `--all` можно их увидеть.

```

commit 50b500a227a3d03cc199b1f70df86edff7fdd00d (tag: oops)
Author: "Sofia <1132236033@pfur.ru>"
Date:   Wed Feb 18 15:24:48 2026 +0300

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

    This reverts commit e85cadec4edf7a645d1c37350c9543503c6df743.

commit e85cadec4edf7a645d1c37350c9543503c6df743
Author: "Sofia <1132236033@pfur.ru>"
Date:   Wed Feb 18 15:24:33 2026 +0300

    Oops, we didn't want this commit

commit a9f8eedbbe9be94a6eedd0fc4a0a6100e6f0fcbf (HEAD, tag: v1, master)
Author: "Sofia <1132236033@pfur.ru>"
Date:   Thu Feb 12 18:21:30 2026 +0300

    Added

commit b7d26657abeccf381ea0f7b0e91fd9470e0453b6 (tag: v1-beta)
Author: "Sofia <1132236033@pfur.ru>"
Date:   Thu Feb 12 18:20:36 2026 +0300

    Added standard HTML page tags

commit f250f177df85ae90ce99985e60c0b206c7de29e3
Author: "Sofia <1132236033@pfur.ru>"
Date:   Thu Feb 12 17:51:45 2026 +0300

    <html>

    <body>

    <h1>Hello World!</h1>

```

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

```
sakudyakova@dk3n55 ~/hello $ git log --all
sakudyakova@dk3n55 ~/hello $ git tag -d oops
Метка «oops» удалена (была 50b500a)
sakudyakova@dk3n55 ~/hello $ git log --all
```

Рис. 18: Удаление тега oops

Изменение предыдущего коммита

Добавим в страницу комментарий автора



```
1 1 -- Author Sofia A. Kudyakova --
2 <html>
3   <head>
4   </head>
5   <body>
6     <h1>Hello, World!</h1>
7   </body>
8 </html>
```

Рис. 19: Изменение файла hello.html

Изменение предыдущего коммита

```
sakudyakova@dk3n55 ~/hello $ git add hello.html
sakudyakova@dk3n55 ~/hello $ git commit -m "Add an author comment"
[отделённый HEAD 31261b2] Add an author comment
1 file changed, 1 insertion(+)
sakudyakova@dk3n55 ~/hello $ cat hello.html
<!-- Author Sofia A. Kudyakova -->
<html>
  <head>
  </head>
  <body>
    <h1>Hello, World!</h1>
  </body>
</html>
sakudyakova@dk3n55 ~/hello $ cat hello.html
<!-- Author Sofia A. Kudyakova -->
<html>
  <head>
  </head>
  <body>
    <h1>Hello, World!</h1>
  </body>
</html>
sakudyakova@dk3n55 ~/hello $ git add hello.html
sakudyakova@dk3n55 ~/hello $ git commit --amend -m "Add an author/email comment"
[отделённый HEAD 4461cc0] Add an author/email comment
Date: Wed Feb 18 15:31:01 2026 +0300
1 file changed, 1 insertion(+)
sakudyakova@dk3n55 ~/hello $ git log
sakudyakova@dk3n55 ~/hello $
```

Рис. 20: Изменение предыдущего коммита

Перемещение файлов

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

```
sakudyakova@dk3n55 ~/hello $ mkdir lib
sakudyakova@dk3n55 ~/hello $ git mv hello.html lib/
sakudyakova@dk3n55 ~/hello $ git status
Отсоединённый указатель HEAD на цепочке коммитов начиная с v1-beta
Изменения, которые будут включены в коммит:
  (используйте «git restore --staged <файл>...», чтобы убрать из индекса)
    переименовано: hello.html -> lib/hello.html

sakudyakova@dk3n55 ~/hello $ git switch master
error: Ваши локальные изменения в указанных файлах будут перезаписаны при переключении на состояние:
      hello.html
Сделайте коммит или спрячьте ваши изменения перед переключением веток.
Прерываю
sakudyakova@dk3n55 ~/hello $ git status
Отсоединённый указатель HEAD на цепочке коммитов начиная с v1-beta
Изменения, которые будут включены в коммит:
  (используйте «git restore --staged <файл>...», чтобы убрать из индекса)
    переименовано: hello.html -> lib/hello.html

sakudyakova@dk3n55 ~/hello $ git checkout master
error: Ваши локальные изменения в указанных файлах будут перезаписаны при переключении на состояние:
      hello.html
Сделайте коммит или спрячьте ваши изменения перед переключением веток.
Прерываю
sakudyakova@dk3n55 ~/hello $ git status
Отсоединённый указатель HEAD на цепочке коммитов начиная с v1-beta
Изменения, которые будут включены в коммит:
  (используйте «git restore --staged <файл>...», чтобы убрать из индекса)
    переименовано: hello.html -> lib/hello.html
```

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

```
sakudyakova@dk3n55 ~/hello $ touch index.html
sakudyakova@dk3n55 ~/hello $ cat index.html
cat: index: Нет такого файла или каталога
cat: html: Нет такого файла или каталога
sakudyakova@dk3n55 ~/hello $ cat index.html
sakudyakova@dk3n55 ~/hello $ cat index.html
<html>
  <body>
    <iframe src="lib/hello.html" width=200" height="200" />
  </body>
</html>
sakudyakova@dk3n55 ~/hello $ git add index.html
sakudyakova@dk3n55 ~/hello $ git commit -m "Added index.html"
[отделённый HEAD 8ac67e4] Added index.html
 1 file changed, 5 insertions(+)
 create mode 100644 index.html
sakudyakova@dk3n55 ~/hello $
```

Рис. 22: Создание, индексация, коммит нового файла

Теперь при открытии index.html увидим кусок страницы hello в маленьком окошке

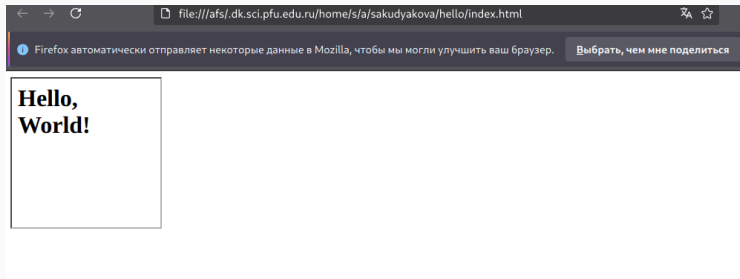


Рис. 23: Результат открытия index.html

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`

```
sakudyakova@dk3n55 ~/hello $ ls -C .git/
COMMIT_EDITMSG  config  description  HEAD  hooks  index  info  logs  objects  ORIG_HEAD  refs
sakudyakova@dk3n55 ~/hello $ ls -C .git/objects/
00 08 0d 15 28 31 3d 44 50 52 53 5f 62 63 6d 7e 8a 9e a9 ad b7 c3 e8 ec f2 f7 info pack
sakudyakova@dk3n55 ~/hello $ ls -C .git/objects/ac
ls: невозможно получить доступ к '.git/objects/ac': Нет такого файла или каталога
sakudyakova@dk3n55 ~/hello $ ls -C .git/objects/ad
2e97d369d7475f0ca497cc82aab706d12507b6
sakudyakova@dk3n55 ~/hello $ ls -C .git/objects/ec
2bda0c449aa398508b548fe1ca8cf1ca10fffe
sakudyakova@dk3n55 ~/hello $ cat .git/config
[core]
    repositoryformatversion = 0
    filemode = true
    bare = false
    logallrefupdates = true
sakudyakova@dk3n55 ~/hello $ ls .git/refs/
heads  tags
sakudyakova@dk3n55 ~/hello $ ls .git/refs/heads/
master
sakudyakova@dk3n55 ~/hello $ ls .git/refs/tags/
v1  v1-beta
sakudyakova@dk3n55 ~/hello $ ls .git/refs/tags/v1
.git/refs/tags/v1
sakudyakova@dk3n55 ~/hello $ cat .git/refs/tags/v1
a9f8eedbbe9be94a6eedd0fc4a0a6100e6f0fcbf
sakudyakova@dk3n55 ~/hello $ cat .git/HEAD
8ac67e4948fbdeeb71c2042c180efe392143353
```

Работа непосредственно с объектами git

Найдем последний коммит через `log` и выведем его, используя SHA1 хэш.

```
sakudyakova@dk3n55 ~/hello $ git log --max-count=1
sakudyakova@dk3n55 ~/hello $ git cat-file -t 8ac67e4948fbdeeb71c2042c180efe392143353
commit
sakudyakova@dk3n55 ~/hello $ git cat-file -p 8ac67e4948fbdeeb71c2042c180efe392143353
tree 63bf71fb29dd6aa793da641114c4bed139646afe
parent 9ee4673e5a5d6cfdebd2ca4dbec533cd0ca582e3
author "Sofia <"1132236033@pfur.ru"> 1771418570 +0300
committer "Sofia <"1132236033@pfur.ru"> 1771418570 +0300
gpgsig -----BEGIN PGP SIGNATURE-----

iQIzBAABCAAdFiEEsAn4Pq4j/KtxDKS5mQl5fH7ib7cFAMmVs8oACGkQmQl5fH7i
b7c2zxAAEFigJpsb/OetXODzn/bzaYnzPZwFJ/PKosfbC6ryeqmJkLD9Yr3KTS9
rCPo7xkZ9yxJL2/uCWdKLyaPo+jXwETrqM8lvfj386XMrow/LwiE79RCKGBNUcp3
5Ub30junr+apOEI7dZmq+KNpY+TJReq20zZaSn2qsfNE9kvGQjFcF/RlafsJxG72
eZP+swPhsK9VlqiVGNmo5UfRH+ixr4Xdnu844pRGaRQagr37In+8/pHB5vjZE2Ih
PQOB5WuBENYftX5wreCdWy/gbkaTmkF1rbv1q6hKZjBa6wxAULmVROjNKq2IuUG
/caYC10s2LrL92hKeu1VEhq4RHoH2AUshdTjIK9jFNyysvpU6ppj6odmYE6UB0
Ql2NnmFVXl+TuUQBu3XPId/PN545r05SxeTdmF0a2bEAZ1A2j4XC0pmc4SiykRbj
ruMxpI56WEFVWX3qB4BjRQ5tqq/WfTE4bP8sbJ+GOH2mQ3xTbnEw2fx/H/ajQQQ1
r+C63mo9+1rapIthWZAc5iWubbvKpurKbCv88f8WdT/gnffCZYmzrvKEkTMvbsb3
k3qs8kpWCIzLrj0t35jPwTkOU2T90ze1XNjHppWYDqGL7CUf+R5KSxY0IYhMgS7u
IWTxtLQRagptV0h0isurBm/CClPpdlT8aGRlyPq8PWdtc0kq120=
=STGD
-----END PGP SIGNATURE-----
```

Added index.html

```
sakudyakova@dk3n55 ~/hello $ git cat-file -p 63bf71fb29dd6aa793da641114c4bed139646afe
100644 blob 3d78056d00d637b699e842b57d9d6ff611240e4b index.html
040000 tree ad2e97d369d7475f0ca497cc82aab706d12507b6 lib
sakudyakova@dk3n55 ~/hello $ git cat-file -p ad2e97d369d7475f0ca497cc82aab706d12507b6
100644 blob 28b42e2eb776169579ce01dfcc67fc4e5e3de715 hello.html
sakudyakova@dk3n55 ~/hello $ git cat-file -p 28b42e2eb776169579ce01dfcc67fc4e5e3de715
```

```
<!-- Author Sofia A. Kudyakova -->
```

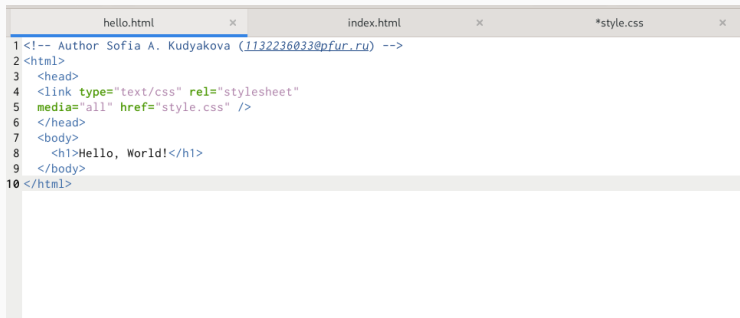
```
<html>
<head>
</head>
<body>
<h1>Hello, World!</h1>
```

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

```
sakudyakova@dk3n55 ~/hello $ fit checkout -b style
bash: fit: команда не найдена
sakudyakova@dk3n55 ~/hello $ git checkout -b style
ПереклЮчилиcь на новую ветку «style»
sakudyakova@dk3n55 ~/hello $ git status
Текущая ветка: style
нечего коммитить, нет изменений в рабочем каталоге
sakudyakova@dk3n55 ~/hello $ touch lib/style.css
sakudyakova@dk3n55 ~/hello $ cat lib/style.css
sakudyakova@dk3n55 ~/hello $ cat lib/style.css
h1 {
  color: red;
}
sakudyakova@dk3n55 ~/hello $ git add lib/style.css
sakudyakova@dk3n55 ~/hello $ git commit -m "Added css stylesheet"
[style e3c5414] Added css stylesheet
1 file changed, 3 insertions(+)
create mode 100644 lib/style.css
sakudyakova@dk3n55 ~/hello $
```

Рис. 27: Создание ветки

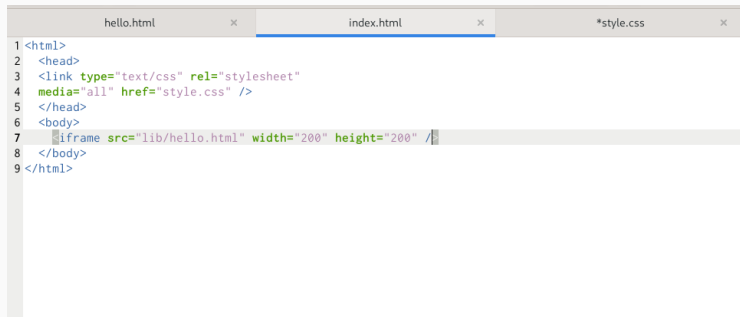
Обновим файл hello.html, чтобы использовать стили style.css и index.html



```
1 <!-- Author Sofia A. Kudyakova (1132236033@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. 28: Редактирование файла hwllo.html

Обновив файл index.html



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

Рис. 29: Редактирование файла index.html

Также обновим их в репозиторий

```
sakudyakova@dk3n55 ~/hello $ git add hello.html
sakudyakova@dk3n55 ~/hello $ git commit -m "Hello uses style.css"
[style 929c0ce] Hello uses style.css
1 file changed, 10 insertions(+)
create mode 100644 hello.html
sakudyakova@dk3n55 ~/hello $ git add index.html
sakudyakova@dk3n55 ~/hello $ git commit -m "Updated Index.html"
[style 29c9122] Updated Index.html
1 file changed, 5 insertions(+), 1 deletion(-)
sakudyakova@dk3n55 ~/hello $
```

Рис. 30: Создание ветки

Посмотрим все логи

```
commit 29c91225ad97cc12fd40deec6ac104fe903ba77a (HEAD -> style)
```

```
Author: "Sofia <1132236033@pfur.ru>"
```

```
Date:   Wed Feb 18 16:20:50 2026 +0300
```

```
    Updated Index.html
```

```
commit 929c0ce1776f86120b452cb91899fbc71a1c4a50
```

```
Author: "Sofia <1132236033@pfur.ru>"
```

```
Date:   Wed Feb 18 16:19:44 2026 +0300
```

```
    Hello uses style.css
```

```
commit e3c54146250db926bf4365ed81e5c52797a5065d
```

```
Author: "Sofia <1132236033@pfur.ru>"
```

```
Date:   Wed Feb 18 16:07:18 2026 +0300
```

```
    Added css stylesheet
```

```
commit 8ac67e4948fbedeeb71c2042c180efe392143353
```

```
Author: "Sofia <1132236033@pfur.ru>"
```

```
Date:   Wed Feb 18 15:42:50 2026 +0300
```

```
    Added index.html
```

```
commit 9ee4673e5a5d6cfdedb2ca4dbec533cd0ca582e3
```

```
Author: "Sofia <1132236033@pfur.ru>"
```

```
Date:   Wed Feb 18 15:38:24 2026 +0300
```

```
    Moved hello.html to lib
```

```
commit 4461cc0283957db84b67dfb1ff12eac2ba81ca77
```

```
Author: "Sofia <1132236033@pfur.ru>"
```

```
Date:   Wed Feb 18 15:31:01 2026 +0300
```


Вернемся в основную ветку и добавим файл README.md

```
sakudyakova@dk3n55 ~/hello $ git checkout master
Переключились на ветку «master»
sakudyakova@dk3n55 ~/hello $ echo "This is the Hello World example from the git tutorial" > README.md
sakudyakova@dk3n55 ~/hello $ git add README.md
sakudyakova@dk3n55 ~/hello $ git commit -m "Added README"
[master 5168be8] Added README
 1 file changed, 1 insertion(+)
 create mode 100644 README.md
```

Рис. 33: Изменения в ветке master

Изменения в ветке master

Посмотрим ветки и их различия `git log --graph --all`

```
* commit 5168be8b9e5e39d7c93bb184938cb458a90155fc (HEAD -> master)
| Author: "Sofia <1132236033@pfur.ru>"
| Date:   Wed Feb 18 16:37:20 2026 +0300
|
|     Added README
|
* commit 8b448f1f61bf514b6a21717649acf8d4ad58881f
| Author: "Sofia <1132236033@pfur.ru>"
| Date:   Wed Feb 18 16:32:25 2026 +0300
|
|     saving
|
* commit 29c91225ad97cc12fd40deec6ac104fe903ba77a (style)
| Author: "Sofia <1132236033@pfur.ru>"
| Date:   Wed Feb 18 16:20:50 2026 +0300
|
|     Updated Index.html
|
* commit 929c0ce1776f86120b452cb91899fbc71a1c4a50
| Author: "Sofia <1132236033@pfur.ru>"
| Date:   Wed Feb 18 16:19:44 2026 +0300
|
|     Hello uses style.css
|
* commit e3c54146250db926bf4365ed81e5c52797a5065d
| Author: "Sofia <1132236033@pfur.ru>"
| Date:   Wed Feb 18 16:07:18 2026 +0300
|
|     Added css stylesheet
|
* commit 8ac67e4948fbedeeb71c2042c180efe392143353
| Author: "Sofia <1132236033@pfur.ru>"
| Date:   Wed Feb 18 15:42:50 2026 +0300
```

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

```
sakudyakova@dk3n55 ~/hello $ git log --graph --all
sakudyakova@dk3n55 ~/hello $ git checkout style
Переключились на ветку «style»
sakudyakova@dk3n55 ~/hello $ git merge master
Автослияние lib/hello.html
КОНФЛИКТ (добавление/добавление): Конфликт слияния в lib/hello.html
Сбой автоматического слияния; исправьте конфликты, затем зафиксируйте результат.
sakudyakova@dk3n55 ~/hello $ git log --graph --all
```

Рис. 35: Слияние веток

Вернемся в ветку master и создадим конфликт, внося изменения в файл hello.html

```
1 <!-- Author Sofia A. Kudyakova (1132236033@pfur.ru) -->
2 <html>
3   <head>
4     <!-- no style -->
5 >>>>>> master
6   </head>
7   <body>
8     <h1>Hello, World! Life is great!</h1>
9   </body>
10 </html> |
```

Рис. 36: Создание конфликта

```
sakudyakova@dk3n55 ~/hello $ git checkout master
Переключились на ветку «master»
sakudyakova@dk3n55 ~/hello $ git add lib/hello.html
sakudyakova@dk3n55 ~/hello $ git commit -m ''
sakudyakova@dk3n55 ~/hello $ git commit -m 'Life is great'
Текущая ветка: master
нечего коммитить, нет изменений в рабочем каталоге
sakudyakova@dk3n55 ~/hello $ git log --graph -all
```

Рис. 37: Создание конфликта

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

Разрешение конфликтов

```
1 <!-- Author: Sofia A. Kudyakova (1132236033@pfur.ru) -->
2 <html>
3   <head>
4     <<<<<< HEAD
5       <link type="text/sss" 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>
```

Bracket match found on line: 14

HTML ▾

```
1 <!-- Author: Sofia A. Kudyakova (1132236033@pfur.ru) -->
2 <html>
3   <head>
4     <link type="text/sss" rel="stylesheet"
5       media="all" href="style-css" />
6   </head>
7   <body>
8     <h1>Hello, World! Life is great!</h1>
9   </body>
10 </html>
```



```
sakudyakova@dk3n55 ~/hello $ git checkout style
Переключились на ветку «style»
sakudyakova@dk3n55 ~/hello $ git merge master
Автослияние lib/hello.html
КОНФЛИКТ (содержимое): Конфликт слияния в lib/hello.html
Сбой автоматического слияния; исправьте конфликты, затем зафиксируйте результат.
sakudyakova@dk3n55 ~/hello $ git add lib/hello.html
sakudyakova@dk3n55 ~/hello $ git commit -m ""
Отмена коммита из-за пустого сообщения коммита.
sakudyakova@dk3n55 ~/hello $ git commit -m ""
Отмена коммита из-за пустого сообщения коммита.
sakudyakova@dk3n55 ~/hello $ git commit -m "Merged after fixed conflict"
[style 82d1dd3] Merged after fixed conflict
sakudyakova@dk3n55 ~/hello $
```

Рис. 40: Коммит разрешения конфликта

Сброс ветки style

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

```
* commit 82d1dd30578b51ddd57dae608de9c593c1bc9a3d (HEAD -> style)
| Merge: 1f24dc2 57ba7a0
| Author: "Sofia <"1132236033@pfur.ru">"
| Date:   Wed Feb 18 17:02:22 2026 +0300
|
|     Merged after fixed conflict
|
* commit 57ba7a079dbe6bfce9238a9b5aecb2fe1120c281 (master)
| Author: "Sofia <"1132236033@pfur.ru">"
| Date:   Wed Feb 18 17:01:13 2026 +0300
|
|     coo
|
* commit 1f24dc2a5d1dc5a6edc727003051933c60faa1f4
| Merge: 29c9122 5168be8
| Author: "Sofia <"1132236033@pfur.ru">"
| Date:   Wed Feb 18 16:45:14 2026 +0300
|
|     Merge branch 'master' into style
|
* commit 5168be8b9e5e39d7c93bb184938cb458a90155fc
| Author: "Sofia <"1132236033@pfur.ru">"
| Date:   Wed Feb 18 16:37:20 2026 +0300
|
|     Added README
|
* commit 8b448f1f61bf514b6a21717649acf8d4ad58881f
| Author: "Sofia <"1132236033@pfur.ru">"
```

Сброс ветки style

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

```
* commit 57ba7a079dbe6bfce9238a9b5aecb2fe1120c281 (master)
Author: "Sofia <1132236033@pfur.ru>"
Date:   Wed Feb 18 17:01:13 2026 +0300

    coo

* commit 5168be8b9e5e39d7c93bb184938cb458a90155fc
Author: "Sofia <1132236033@pfur.ru>"
Date:   Wed Feb 18 16:37:20 2026 +0300

    Added README

* commit 8b448f1f61bf514b6a21717649acf8d4ad58881f
Author: "Sofia <1132236033@pfur.ru>"
Date:   Wed Feb 18 16:32:25 2026 +0300

    saving

* commit 29c91225ad97cc12fd40deec6ac104fe903ba77a (HEAD -> style)
| Author: "Sofia <1132236033@pfur.ru>"
| Date:   Wed Feb 18 16:20:50 2026 +0300
|
|     Updated Index.html
|
* commit 929c0ce1776f86120b452cb91899fbc71a1c4a50
| Author: "Sofia <1132236033@pfur.ru>"
| Date:   Wed Feb 18 16:19:44 2026 +0300
|
|     Hello uses style.css
|
```

Сброс ветки master

Добавив интерактивный режим в ветку master, мы внесли изменения, конфликтующие с изменениями в ветке style.

```
* commit 57ba7a079dbe6bfce9238a9b5aecb2fe1120c281 (HEAD -> master)
Author: "Sofia <"1132236033@pfur.ru">"
Date:   Wed Feb 18 17:01:13 2026 +0300

    coo

* commit 5168be8b9e5e39d7c93bb184938cb458a90155fc
Author: "Sofia <"1132236033@pfur.ru">"
Date:   Wed Feb 18 16:37:20 2026 +0300

    Added README

* commit 8b448f1f61bf514b6a21717649acf8d4ad58881f
Author: "Sofia <"1132236033@pfur.ru">"
Date:   Wed Feb 18 16:32:25 2026 +0300

    saving

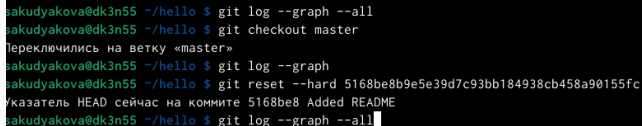
* commit a9f8eedbbe9be94a6eedd0fc4a0a6100e6f0fcfb (tag: v1)
Author: "Sofia <"1132236033@pfur.ru">"
Date:   Thu Feb 12 18:21:30 2026 +0300

    Added

* commit b7d26657abeccf381ea0f7b0e91fd9470e0453b6 (tag: v1-beta)
Author: "Sofia <"1132236033@pfur.ru">"
Date:   Thu Feb 12 18:20:36 2026 +0300

    Added standard HTML page tags
```

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



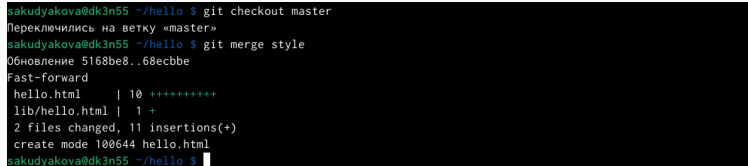
```
sakudyakova@dk3n55 ~/hello $ git log --graph --all
sakudyakova@dk3n55 ~/hello $ git checkout master
Переключились на ветку «master»
sakudyakova@dk3n55 ~/hello $ git log --graph
sakudyakova@dk3n55 ~/hello $ git reset --hard 5168be8b9e5e39d7c93bb184938cb458a90155fc
Указатель HEAD сейчас на коммите 5168be8 Added README
sakudyakova@dk3n55 ~/hello $ git log --graph --all
```

Рис. 44: Сброс ветки master

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

```
* commit 68ecbbe957c868927973706f58a53fc109e23eb7 (HEAD -> style)
| Author: "Sofia <"1132236033@pfur.ru">
| Date:   Wed Feb 18 16:19:44 2026 +0300
|
|     Hello uses style.css
|
* commit 3c24aa8c4381599456860619b679fc959752469a
| Author: "Sofia <"1132236033@pfur.ru">
| Date:   Wed Feb 18 15:31:01 2026 +0300
|
|     Add an author/email comment
|
* commit 5168be8b9e5e39d7c93bb184938cb458a90155fc (master)
| Author: "Sofia <"1132236033@pfur.ru">
| Date:   Wed Feb 18 16:37:20 2026 +0300
|
|     Added README
|
* commit 8b448f1f61bf514b6a21717649acf8d4ad58881f
| Author: "Sofia <"1132236033@pfur.ru">
| Date:   Wed Feb 18 16:32:25 2026 +0300
|
|     saving
|
* commit a9f8eedbbe9be94a6eedd0fc4a0a6100e6f0fcfb (tag: v1)
```

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

A terminal window with a black background and green text. The user is at a prompt 'sakudyakova@dk3n55 ~/hello \$'. They enter 'git checkout master', and the terminal shows 'Переключились на ветку «master»'. Then they enter 'git merge style', and the terminal shows 'Обновление 5168be8..68ecbbe' and 'Fast-forward'. A diff follows: 'hello.html | 10 ++++++++' and 'lib/hello.html | 1 +'. Below the diff, it says '2 files changed, 11 insertions(+)' and 'create mode 100644 hello.html'. The prompt returns to 'sakudyakova@dk3n55 ~/hello \$' with a cursor.

```
sakudyakova@dk3n55 ~/hello $ git checkout master
Переключились на ветку «master»
sakudyakova@dk3n55 ~/hello $ git merge style
Обновление 5168be8..68ecbbe
Fast-forward
 hello.html      | 10 ++++++++
 lib/hello.html  | 1 +
 2 files changed, 11 insertions(+)
 create mode 100644 hello.html
sakudyakova@dk3n55 ~/hello $
```

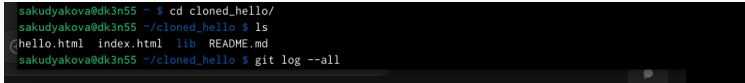
Рис. 46: Слияние style в master

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

```
sakudyakova@dk3n55 ~/hello $ cd ..
sakudyakova@dk3n55 ~ $ pwd
/afs/.dk.sci.pfu.edu.ru/home/s/a/sakudyakova
sakudyakova@dk3n55 ~ $ ls
121.sh  archive.tar  FILES.txt  input.txt  '-p'      tmp      Документы  Музыка      'Снимки :
122.sh  bin          GNUstep    lock.file  public    work     Загрузки   Общедоступные  Шаблоны
123.sh  blog         hello      output.txt public_html  Видео    Изображения 'Рабочий стол'
sakudyakova@dk3n55 ~ $ git clone hello cloned_hello
Клонирование в «cloned_hello»...
готово.
sakudyakova@dk3n55 ~ $ ls
121.sh  bin          GNUstep    output.txt  tmp      Загрузки   'Рабочий стол'
122.sh  blog         hello      '-p'        work     Изображения 'Снимки экрана'
123.sh  cloned_hello input.txt  public      Видео    Музыка      Шаблоны
archive.tar  FILES.txt  lock.file  public_html  Документы  Общедоступные
```

Рис. 47: Клонирование репозиториев

Затем посмотрим историю репозитория и увидим список всех коммитов в новый репозиторий, и он совпадает с историей коммитов в оригинальном репозитории.

A terminal window with a dark background and light green text. The prompt is 'sakudyakova@dk3n55 ~'. The user enters 'cd cloned_hello/' and the prompt changes to 'sakudyakova@dk3n55 ~/cloned_hello'. Then the user enters 'ls' and the output is 'hello.html index.html lib README.md'. Finally, the user enters 'git log --all'.

```
sakudyakova@dk3n55 ~ $ cd cloned_hello/  
sakudyakova@dk3n55 ~/cloned_hello $ ls  
hello.html index.html lib README.md  
sakudyakova@dk3n55 ~/cloned_hello $ git log --all
```

Рис. 48: Клонирование репозиторов

Что такое origin?

Клонированный репозиторий знает об имени по умолчанию удаленного репозитория. Посмотрим, подробную информацию об имени по умолчанию. .

```
sakudyakova@dk3n55 ~/cloned_hello $ git remote
origin
sakudyakova@dk3n55 ~/cloned_hello $ git remote show origin
* внешний репозиторий origin
  URL для извлечения: /afs/.dk.sci.pfu.edu.ru/home/s/a/sakudyakova/hello
    URL для отправки: /afs/.dk.sci.pfu.edu.ru/home/s/a/sakudyakova/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)
```

Рис. 49: Просмотр имени по умолчанию удаленного репозитория

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

A terminal window with a dark background. The prompt is 'sakudyakova@dk3n55 ~/cloned_hello \$'. The first command 'git branch' is entered, followed by a second command 'git branch -a' on the next line.

```
sakudyakova@dk3n55 ~/cloned_hello $ git branch
sakudyakova@dk3n55 ~/cloned_hello $ git branch -a
```

Рис. 50: Просмотр веток

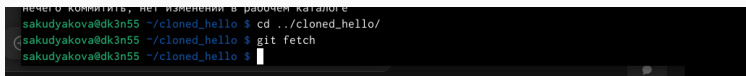
Изменение оригинального репозитория

Перейдем в репозиторий hello. Внесем изменения в файл README.md. Затем добавим их в репозиторий

```
sakudyakova@dk3n55 ~/cloned_hello $ cat README.md
This is the Hello World example from the git tutorial.
sakudyakova@dk3n55 ~/cloned_hello $ git add README.md
sakudyakova@dk3n55 ~/cloned_hello $ git commit -m "Add README file to the repository"
```

Рис. 51: 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

A terminal window with a dark background. The prompt is 'sakudyakova@dk3n55 ~/cloned_hello'. The first command is 'cd ../cloned_hello/' and the second is 'git fetch'. The output of 'git fetch' is partially visible at the top of the terminal: 'ничего коммитить, нет изменений в рабочем каталоге'.

```
sakudyakova@dk3n55 ~/cloned_hello $ cd ../cloned_hello/
sakudyakova@dk3n55 ~/cloned_hello $ git fetch
sakudyakova@dk3n55 ~/cloned_hello $
```

Рис. 52: Извлечение изменений

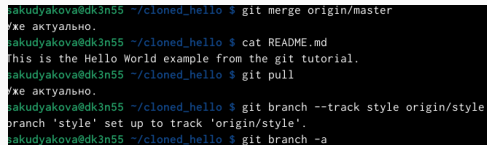
Сольем внесённые изменения в главную ветку с помощью `git pull`, которая является объединением `fetch` и `merge` в одну команду

A terminal window with a dark background and light green text. The prompt is 'sakudyakova@dk3n55 ~/cloned_hello \$'. The user enters 'cat README.md' and the output is 'This is the Hello World example from the git tutorial.'. Then the user enters 'git merge origin/master' and the output is 'уже актуально.'. Then the user enters 'cat README.md' and the output is 'This is the Hello World example from the git tutorial.'. Then the user enters 'git pull' and the output is 'уже актуально.'. Finally, the prompt 'sakudyakova@dk3n55 ~/cloned_hello \$' is shown with a cursor.

```
sakudyakova@dk3n55 ~/cloned_hello $ cat README.md
This is the Hello World example from the git tutorial.
sakudyakova@dk3n55 ~/cloned_hello $ git merge origin/master
уже актуально.
sakudyakova@dk3n55 ~/cloned_hello $ cat README.md
This is the Hello World example from the git tutorial.
sakudyakova@dk3n55 ~/cloned_hello $ git pull
уже актуально.
sakudyakova@dk3n55 ~/cloned_hello $
```

Рис. 53: Слияние извлеченных изменений

Добавление ветки наблюдения

A terminal window with a dark background and green text. The user is in a directory named ~/cloned_hello. The commands and their outputs are as follows:

```
sakudyakova@dk3n55 ~/cloned_hello $ git merge origin/master
Уже актуально.
sakudyakova@dk3n55 ~/cloned_hello $ cat README.md
This is the Hello World example from the git tutorial.
sakudyakova@dk3n55 ~/cloned_hello $ git pull
Уже актуально.
sakudyakova@dk3n55 ~/cloned_hello $ git branch --track style origin/style
branch 'style' set up to track 'origin/style'.
sakudyakova@dk3n55 ~/cloned_hello $ git branch -a
```

Рис. 54: Добавление ветки наблюдения

```
sakudyakova@dk3n55 ~/cloned_hello $ cd ..
sakudyakova@dk3n55 ~ $ git clone --bare hello hello.git
Клонирование в голый репозиторий «hello.git»...
готово.
sakudyakova@dk3n55 ~ $ ls hello.git
config  description  HEAD  hooks  info  objects  packed-refs  refs
sakudyakova@dk3n55 ~ $
```

Рис. 55: Создание чистого репозитория


```
sakudyakova@dk3n55 ~/hello $ cd ../cloned_hello
sakudyakova@dk3n55 ~/cloned_hello $ git remote add shared ../hello.git
sakudyakova@dk3n55 ~/cloned_hello $ git branch --track shared master
branch 'shared' set up to track 'master'.
sakudyakova@dk3n55 ~/cloned_hello $ git pull shared master
Иа ../hello
* branch          master      -> FETCH_HEAD
* [новая ветка]   master      -> shared/master
Уже актуально.
sakudyakova@dk3n55 ~/cloned_hello $ cat README.md
This is the Hello World example from the git tutorial.
sakudyakova@dk3n55 ~/cloned_hello $
```

Рис. 56: Отправка изменений

```
sakudyakova@dk3n55 ~/hello $ cd ../cloned_hello
sakudyakova@dk3n55 ~/cloned_hello $ git remote add shared ../hello.git
sakudyakova@dk3n55 ~/cloned_hello $ git branch --track shared master
branch 'shared' set up to track 'master'.
sakudyakova@dk3n55 ~/cloned_hello $ git pull shared master
Из ../hello
* branch          master      -> FETCH_HEAD
* [новая ветка]   master      -> shared/master
Уже актуально.
sakudyakova@dk3n55 ~/cloned_hello $ cat README.md
This is the Hello World example from the git tutorial.
sakudyakova@dk3n55 ~/cloned_hello $
```

Рис. 57: Извлечение изменений

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

Спасибо за внимание
