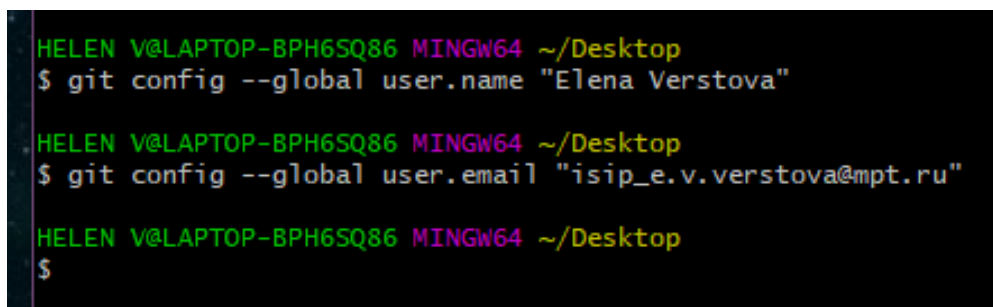


Работа с Git

Подготовка

Если git никогда ранее не использовался пользователем, то необходимо установить имя и электронную почту. Для этого требуется выполнить команды, показанные на рисунке 1.



```
HELEN V@LAPTOP-BPH6SQ86 MINGW64 ~/Desktop
$ git config --global user.name "Elena Verstova"

HELEN V@LAPTOP-BPH6SQ86 MINGW64 ~/Desktop
$ git config --global user.email "isip_e.v.verstova@yandex.ru"

HELEN V@LAPTOP-BPH6SQ86 MINGW64 ~/Desktop
$
```

Рисунок 1 - Установка имени и электронной почты

Затем нужно указать параметры установки окончаний строк (рисунок 2).



```
HELEN V@LAPTOP-BPH6SQ86 MINGW64 ~/Desktop
$ git config --global core.autocrlf true

HELEN V@LAPTOP-BPH6SQ86 MINGW64 ~/Desktop
$ git config --global core.safecrlf true

HELEN V@LAPTOP-BPH6SQ86 MINGW64 ~/Desktop
$
```

Рисунок 2 - Параметры окончаний строк

И последним пунктом идет установка отображения Unicode, показанная на рисунке 3.

```
HELEN V@LAPTOP-BPH6SQ86 MINGW64 ~/Desktop
$ git config --global core.quotepath off

HELEN V@LAPTOP-BPH6SQ86 MINGW64 ~/Desktop
$
```

Рисунок 3 - Установка отображения Unicode

Ответ -

```
kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop
$ git config --global user.name "Fedor Kozhurin"

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop
$ git config --global user.email "kozhurinf@inbox.ru"

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop
$ git config --global core.autocrlf true

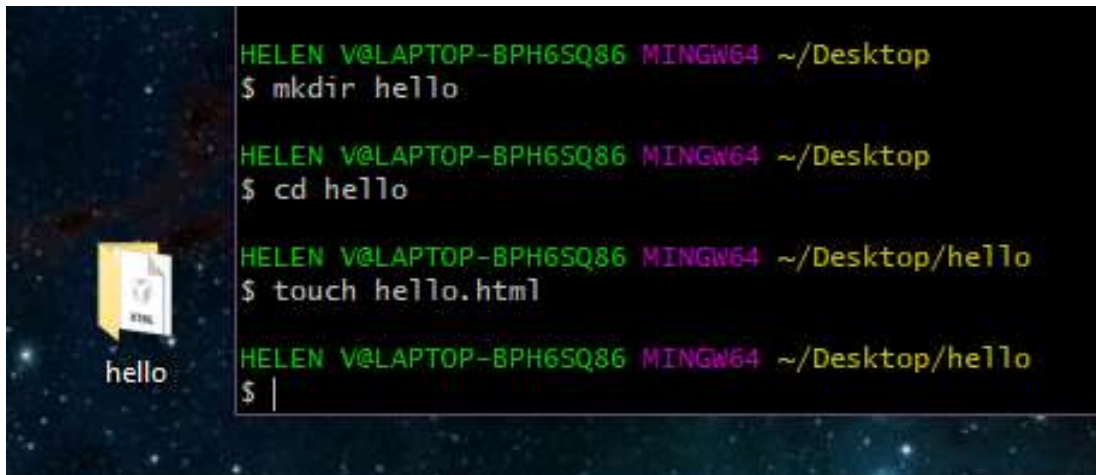
kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop
$ git config --global core.safecrlf true

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop
$ git config --global core.quotepath off

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop
$
```

Создание проекта

Сначала нужно создать пустой каталог и внутри него файл `hello.html` (рисунок 4).

A screenshot of a terminal window with a dark background and a space-themed wallpaper. On the left, a file explorer icon shows a folder named 'hello'. The terminal shows the following commands and output:

```
HELEN V@LAPTOP-BPH6SQ86 MINGW64 ~/Desktop
$ mkdir hello

HELEN V@LAPTOP-BPH6SQ86 MINGW64 ~/Desktop
$ cd hello

HELEN V@LAPTOP-BPH6SQ86 MINGW64 ~/Desktop/hello
$ touch hello.html

HELEN V@LAPTOP-BPH6SQ86 MINGW64 ~/Desktop/hello
$ |
```

Рисунок 4 - Создание каталога и файла

После этого в файл необходимо ввести данные, например, «Hello, world», как показано на рисунке 5.

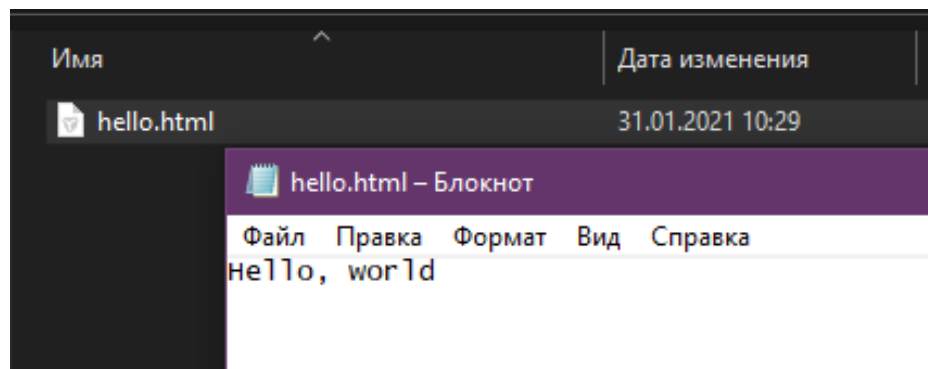


Рисунок 5 - Содержание файла

Для создания репозитория используется команда `git init` (рисунок 6).

A screenshot of a terminal window showing the execution of the `git init` command. The output indicates that an empty Git repository has been initialized in the specified directory.

```
HELEN V@LAPTOP-BPH6SQ86 MINGW64 ~/Desktop/hello
$ git init
Initialized empty Git repository in C:/Users/HELEN V/Desktop/hello/.git/

HELEN V@LAPTOP-BPH6SQ86 MINGW64 ~/Desktop/hello (master)
$
```

Рисунок 6 - Создание репозитория

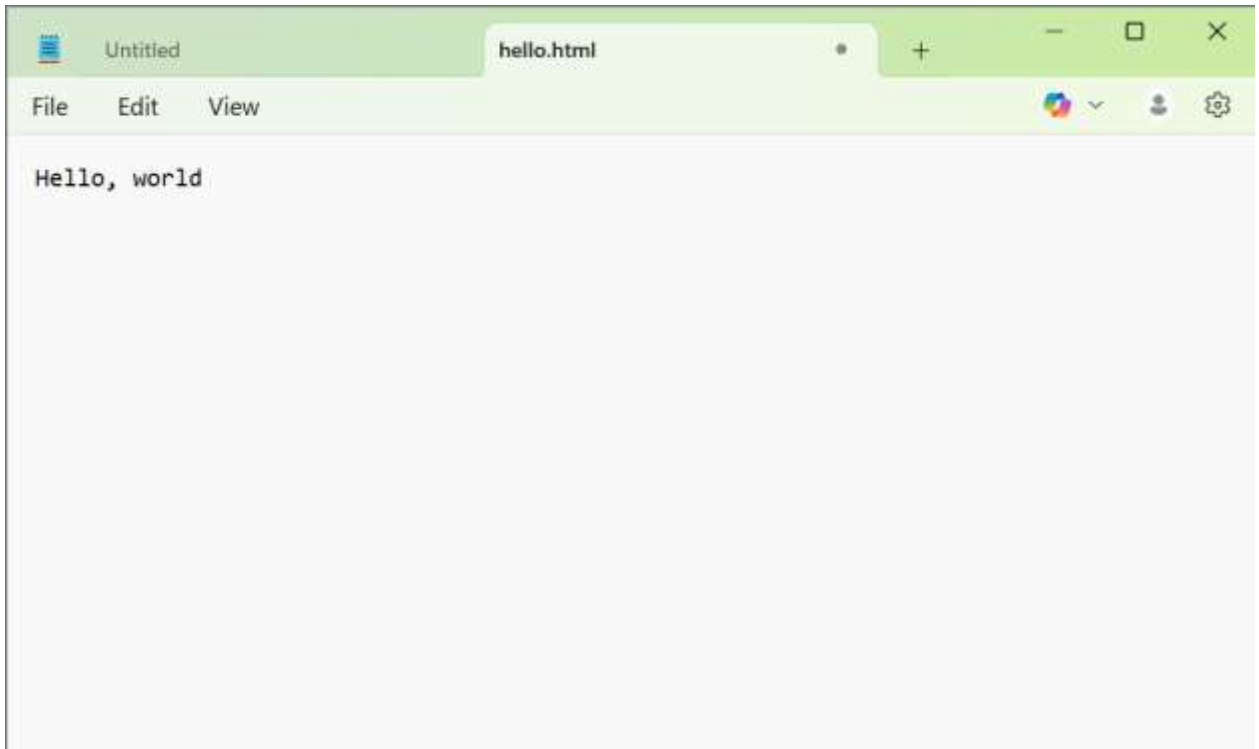
Для добавления страницы в репозиторий необходима команда `git add` (рисунок 7).

```
HELEN V@LAPTOP-BPH6SQ86 MINGW64 ~/Desktop/hello (master)  
$ git add hello.html
```

```
HELEN V@LAPTOP-BPH6SQ86 MINGW64 ~/Desktop/hello (master)  
$ git commit -m "First Commit"  
[master (root-commit) 9234a79] First Commit  
1 file changed, 1 insertion(+)  
create mode 100644 hello.html
```

```
HELEN V@LAPTOP-BPH6SQ86 MINGW64 ~/Desktop/hello (master)  
$ |
```

Рисунок 7 - Добавление в репозиторий



```
kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop
$ mkdir hello

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop
$ cd hello

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello
$ touch hello.html

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello
$ git init
Initialized empty Git repository in C:/Users/kozur/Desktop/hello/.git/

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ git add hello.html

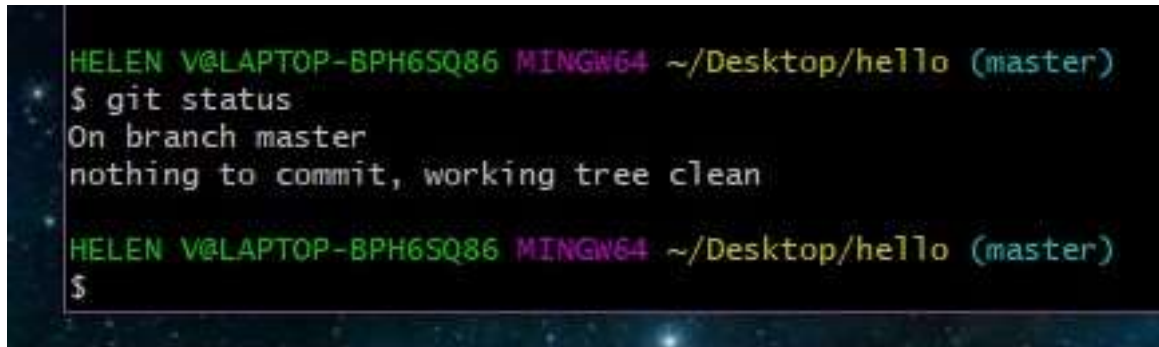
kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ git commit -m "First Commit"
[master (root-commit) 9d515cd] First Commit
1 file changed, 1 insertion(+)
create mode 100644 hello.html

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$
```

Проверка состояния

Проверка состояния репозитория осуществляется с помощью команды `git status`. Если в репозитории хранится текущее состояние рабочего каталога

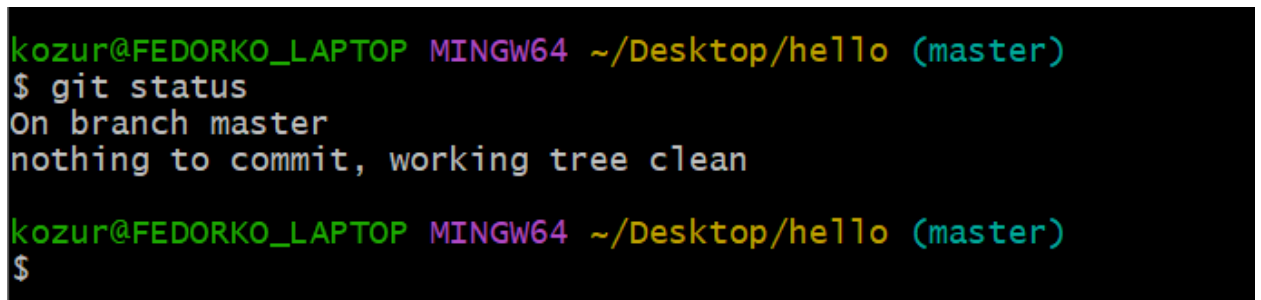
и нет изменений, ожидающих записи, будет показано сообщение, как на рисунке 8.



```
HELEN V@LAPTOP-BPH6SQ86 MINGW64 ~/Desktop/hello (master)
$ git status
On branch master
nothing to commit, working tree clean

HELEN V@LAPTOP-BPH6SQ86 MINGW64 ~/Desktop/hello (master)
$
```

Рисунок 8 - Проверка состояния репозитория



```
kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ git status
On branch master
nothing to commit, working tree clean

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$
```

Внесение изменений

Сначала необходимо внести изменения в файл (рисунок 9).

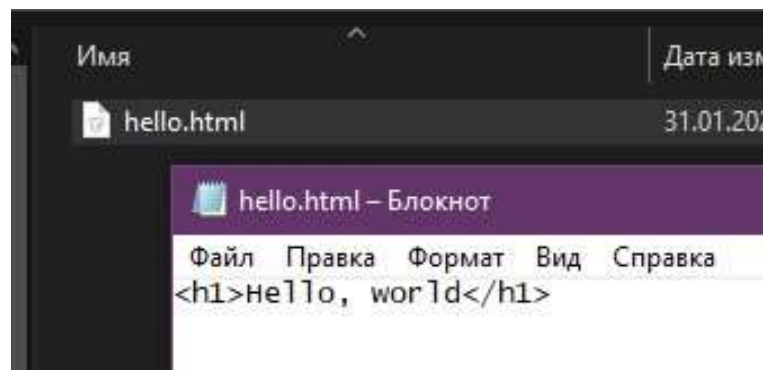


Рисунок 9 - Внесение изменений в файл

Если после предыдущего пункта осуществить проверку состояния репозитория, то будет показано данное сообщение (рисунок 10).

```
HELEN V@LAPTOP-BPH6SQ86 MINGW64 ~/Desktop/hello (master)
$ git status
On branch master
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   hello.html

no changes added to commit (use "git add" and/or "git commit -a")

HELEN V@LAPTOP-BPH6SQ86 MINGW64 ~/Desktop/hello (master)
$ |
```

Рисунок 10 - Сообщение о незафиксированных изменениях



```
kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ git status
On branch master
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   hello.html

no changes added to commit (use "git add" and/or "git commit -a")

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ |
```

Индексация изменений

Для того, чтобы проиндексировать изменения, нужно осуществить действия, показанные на рисунке 11.


```
HELEN V@LAPTOP-BPH6SQ86 MINGW64 ~/Desktop/hello (master)
$ git add hello.html

HELEN V@LAPTOP-BPH6SQ86 MINGW64 ~/Desktop/hello (master)
$ git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   hello.html

HELEN V@LAPTOP-BPH6SQ86 MINGW64 ~/Desktop/hello (master)
$ |
```

Рисунок 11 - Команды для индексации изменений

После этого изменения файлы были проиндексированы. Это значит, что пока изменения не записаны в репозиторий. Если изменения позже не нужно будет фиксировать, то индексацию можно снять командой `git reset`.

```
kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ git add hello.html

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   hello.html

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ |
```


Индексация и коммит

Можно зафиксировать изменения отдельными коммитами. Как это сделать, показано на рисунках 12-14.

Имя	Дата изменения	Тип	Размер
a.html	31.01.2021 10:48	Yandex Browser H...	1 КБ
b.html	31.01.2021 10:49	Yandex Browser H...	1 КБ
hello.html	31.01.2021 10:40	Yandex Browser H...	1 КБ
v.html	31.01.2021 10:49	Yandex Browser H...	1 КБ

Рисунок 12 - Создано 3 файла

```
HELEN V@LAPTOP-BPH65Q86 MINGW64 ~/Desktop/hello (master)
$ git add a.html

HELEN V@LAPTOP-BPH65Q86 MINGW64 ~/Desktop/hello (master)
$ git add b.html

HELEN V@LAPTOP-BPH65Q86 MINGW64 ~/Desktop/hello (master)
$ git commit -m "Changes for a and b"
[master 00106a5] Changes for a and b
3 files changed, 3 insertions(+), 1 deletion(-)
create mode 100644 a.html
create mode 100644 b.html

HELEN V@LAPTOP-BPH65Q86 MINGW64 ~/Desktop/hello (master)
$ :
```


Рисунок 13 - Индексация и коммит для 2 файлов

```
HELEN V@LAPTOP-BPH65Q86 MINGW64 ~/Desktop/hello (master)
$ git add v.html

HELEN V@LAPTOP-BPH65Q86 MINGW64 ~/Desktop/hello (master)
$ git commit -m "Unrelated change for v"
[master cf86455] Unrelated change for v
1 file changed, 1 insertion(+)
create mode 100644 v.html

HELEN V@LAPTOP-BPH65Q86 MINGW64 ~/Desktop/hello (master)
$ :
```

Рисунок 14 - Индексация и коммит для третьего файла

	a	6/6/2025 5:59 PM	Chrome HTML Docu...	0 KB
	b	6/7/2025 1:15 PM	Chrome HTML Docu...	0 KB
	hello	6/6/2025 5:52 PM	Chrome HTML Docu...	1 KB
	v	6/7/2025 1:16 PM	Chrome HTML Docu...	0 KB

```
kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ git add a.html

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ git add b.html

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ git commit -m "Changes for a and b"
[master 02333b6] Changes for a and b
3 files changed, 1 insertion(+), 1 deletion(-)
create mode 100644 a.html
create mode 100644 b.html

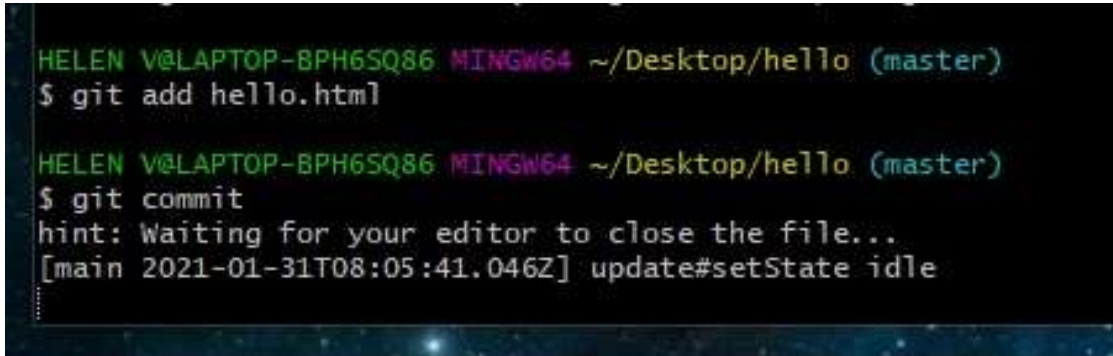
kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ git add v.html

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ git commit -m "Unrelated change for v"
[master ecf6696] Unrelated change for v
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 v.html

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$
```

Коммит изменений

Для того, чтобы редактировать комментарий коммита, нужно использовать команду `git commit` без метки `-m`.

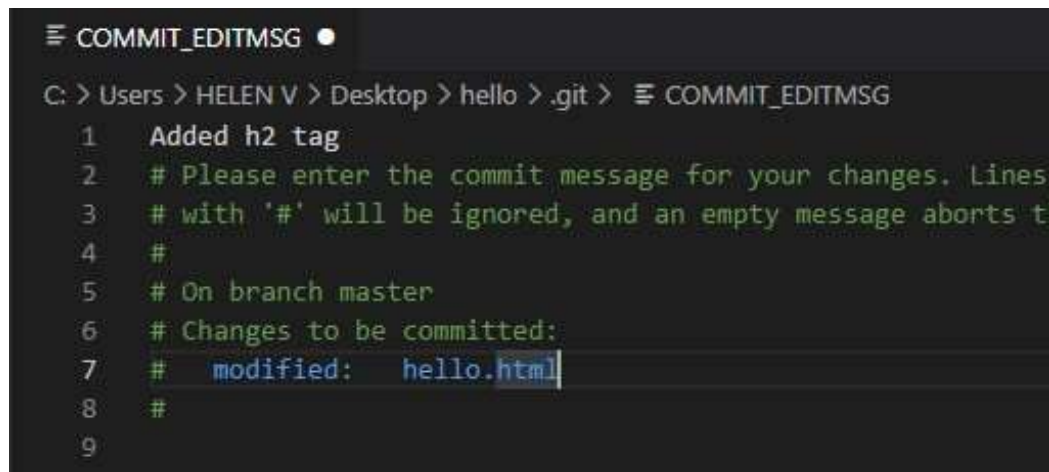


```
HELEN V@LAPTOP-BPH6SQ86 MINGW64 ~/Desktop/hello (master)
$ git add hello.html

HELEN V@LAPTOP-BPH6SQ86 MINGW64 ~/Desktop/hello (master)
$ git commit
hint: Waiting for your editor to close the file...
[main 2021-01-31T08:05:41.046Z] update#setState idle
```

Рисунок 15 - Коммит изменений

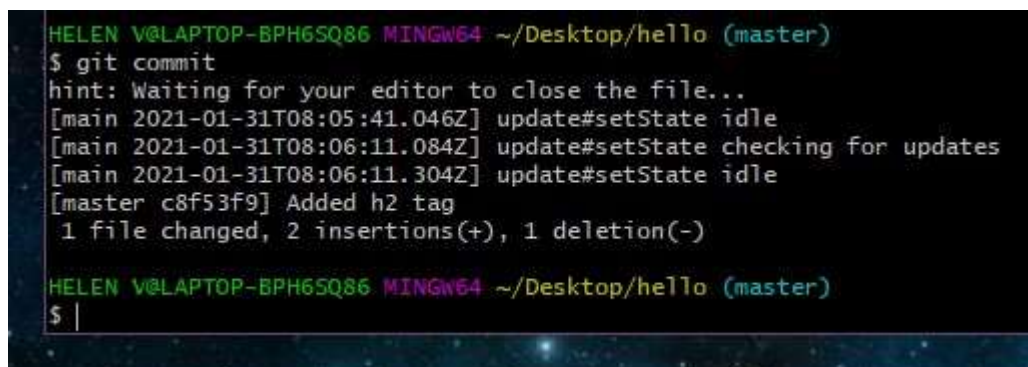
После этого будет открыт редактор, в котором необходимо на 1 строке записать комментарий коммита (рисунок 16).



```
COMMIT_EDITMSG
C: > Users > HELEN V > Desktop > hello > .git > COMMIT_EDITMSG
1 Added h2 tag
2 # Please enter the commit message for your changes. Lines
3 # with '#' will be ignored, and an empty message aborts t
4 #
5 # On branch master
6 # Changes to be committed:
7 #   modified:   hello.html
8 #
9
```

Рисунок 16 - Ввод комментария

После выхода из текстового редактора будет указано следующее сообщение (рисунок 17).

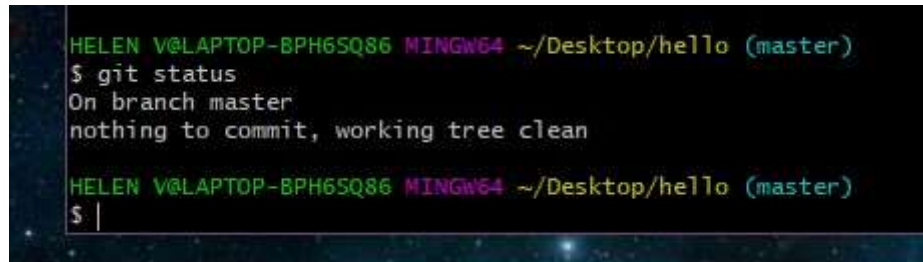


```
HELEN V@LAPTOP-BPH6SQ86 MINGW64 ~/Desktop/hello (master)
$ git commit
hint: Waiting for your editor to close the file...
[main 2021-01-31T08:05:41.046Z] update#setState idle
[main 2021-01-31T08:06:11.084Z] update#setState checking for updates
[main 2021-01-31T08:06:11.304Z] update#setState idle
[master c8f53f9] Added h2 tag
1 file changed, 2 insertions(+), 1 deletion(-)

HELEN V@LAPTOP-BPH6SQ86 MINGW64 ~/Desktop/hello (master)
$ |
```

Рисунок 17 - Коммит-сообщения

После этого еще раз нужно проверить состояние репозитория (рисунок 18).



```
HELEN V@LAPTOP-BPH6SQ86 MINGW64 ~/Desktop/hello (master)
$ git status
On branch master
nothing to commit, working tree clean


HELEN V@LAPTOP-BPH6SQ86 MINGW64 ~/Desktop/hello (master)
$ |
```

Рисунок 18 - Проверка состояния

Изменения, а не файлы

Для того, чтобы понять, что git фокусируется на изменениях в файле, а не на самом файле, можно проделать следующие действия.

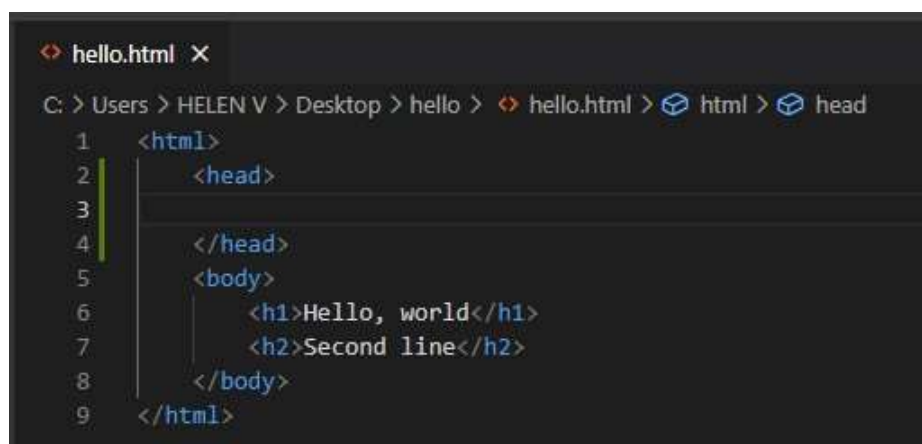
Сначала нужно в файл hello.html добавить теги html и body (рисунок 19), а затем проиндексировать изменения.



```
hello.html X
C: > Users > HELEN V > Desktop > hello > hello.html > html > body > h2
1  <html>
2    <body>
3      <h1>Hello, world</h1>
4      <h2>Second line</h2>
5    </body>
6  </html>
```

Рисунок 19 - Добавление тегов html и body

Затем еще раз нужно добавить изменения в файл (добавить тег head), но изменения не индексировать (рисунок 20).



```
hello.html X
C: > Users > HELEN V > Desktop > hello > hello.html > html > head
1  <html>
2    <head>
3
4  </head>
5    <body>
6      <h1>Hello, world</h1>
7      <h2>Second line</h2>
8    </body>
9  </html>
```

Рисунок 20 - Добавление тега head

Далее нужно проверить статус. На рисунке 21 видно, что файл hello.html указан дважды: первое изменение проиндексировано и готово к коммиту, а второе – нет.

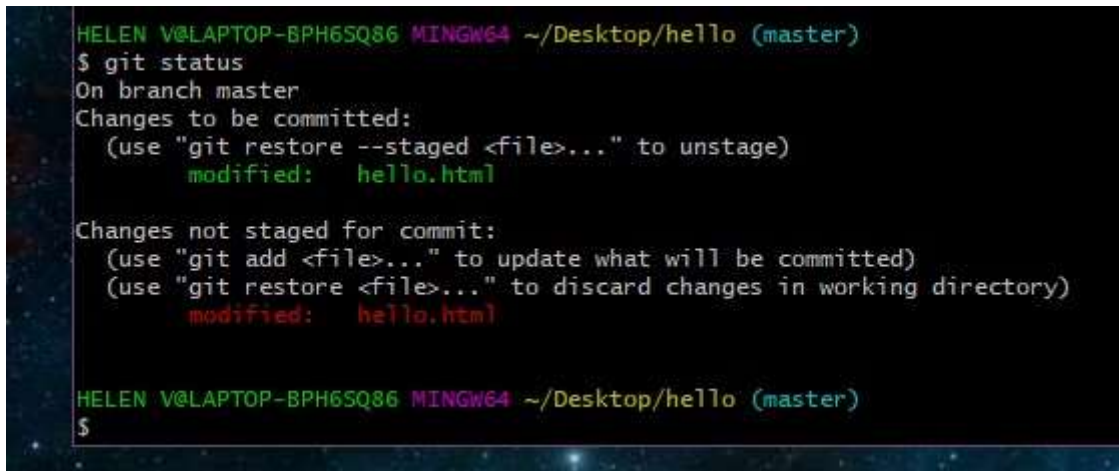
A screenshot of a terminal window with a dark background and a starry space theme. The prompt is 'HELEN V@LAPTOP-BPH65Q86 MINGW64 ~/Desktop/hello (master)'. The user enters '\$ git status'. The output shows 'On branch master' and 'Changes to be committed:' with 'modified: hello.html' in green. It also shows 'Changes not staged for commit:' with 'modified: hello.html' in red. The prompt returns to '\$'.

Рисунок 21 - Проверка состояния

Далее надо произвести коммит проиндексированного изменения и затем еще раз проверить состояние (рисунок 22).

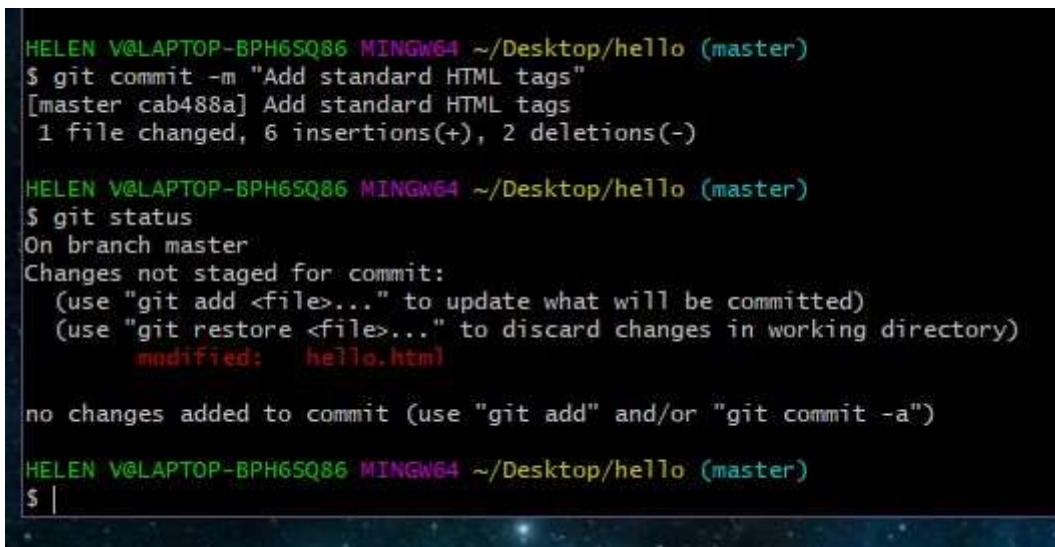
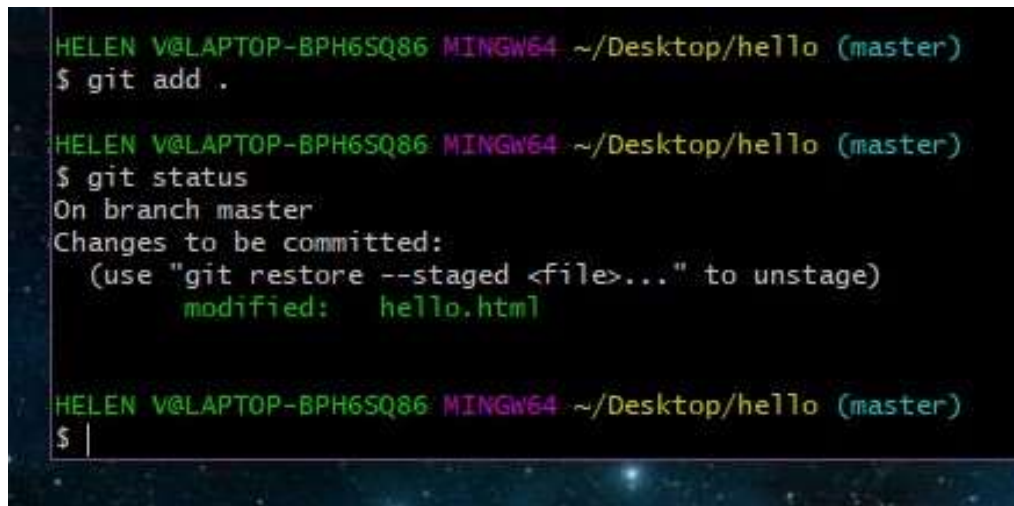
A screenshot of a terminal window with the same dark background and starry space theme. The prompt is 'HELEN V@LAPTOP-BPH65Q86 MINGW64 ~/Desktop/hello (master)'. The user enters '\$ git commit -m "Add standard HTML tags"'. The output shows the commit message and '1 file changed, 6 insertions(+), 2 deletions(-)'. The prompt returns to '\$'. The user then enters '\$ git status'. The output shows 'On branch master' and 'Changes not staged for commit:' with 'modified: hello.html' in red. It also says 'no changes added to commit'. The prompt returns to '\$'.

Рисунок 22 - Коммит 1 изменения и проверка состояния

Нужно добавить второе изменение в индекс и затем проверить состояние (рисунок 23).

A terminal window with a dark background and a starry space pattern at the bottom. The prompt is 'HELEN V@LAPTOP-BPH6SQ86 MINGW64 ~/Desktop/hello (master)'. The user enters '\$ git add .' and then '\$ git status'. The output shows 'On branch master' and 'Changes to be committed: (use "git restore --staged <file>..." to unstage) modified: hello.html'.

```
HELEN V@LAPTOP-BPH6SQ86 MINGW64 ~/Desktop/hello (master)
$ git add .

HELEN V@LAPTOP-BPH6SQ86 MINGW64 ~/Desktop/hello (master)
$ git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   hello.html

HELEN V@LAPTOP-BPH6SQ86 MINGW64 ~/Desktop/hello (master)
$ |
```

Рисунок 23 - Индексация 2 изменения и проверка состояния

После этого нужно сделать коммит второго изменения (рисунок 24).

A terminal window with a dark background and a starry space pattern at the bottom. The prompt is 'HELEN V@LAPTOP-BPH6SQ86 MINGW64 ~/Desktop/hello (master)'. The user enters '\$ git commit -m "Add HTML header"'. The output shows '[master d653372] Add HTML header' and '1 file changed, 3 insertions(+)'.

```
HELEN V@LAPTOP-BPH6SQ86 MINGW64 ~/Desktop/hello (master)
$ git commit -m "Add HTML header"
[master d653372] Add HTML header
1 file changed, 3 insertions(+)

HELEN V@LAPTOP-BPH6SQ86 MINGW64 ~/Desktop/hello (master)
$
```

Рисунок 24 - Коммит 2 изменения

```

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ git status
On branch master
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   hello.html

no changes added to commit (use "git add" and/or "git commit -a")

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ git commit -m "Add standart HTML tags"
On branch master
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   hello.html

no changes added to commit (use "git add" and/or "git commit -a")

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ git status
On branch master
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   hello.html

no changes added to commit (use "git add" and/or "git commit -a")

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ git add .

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   hello.html

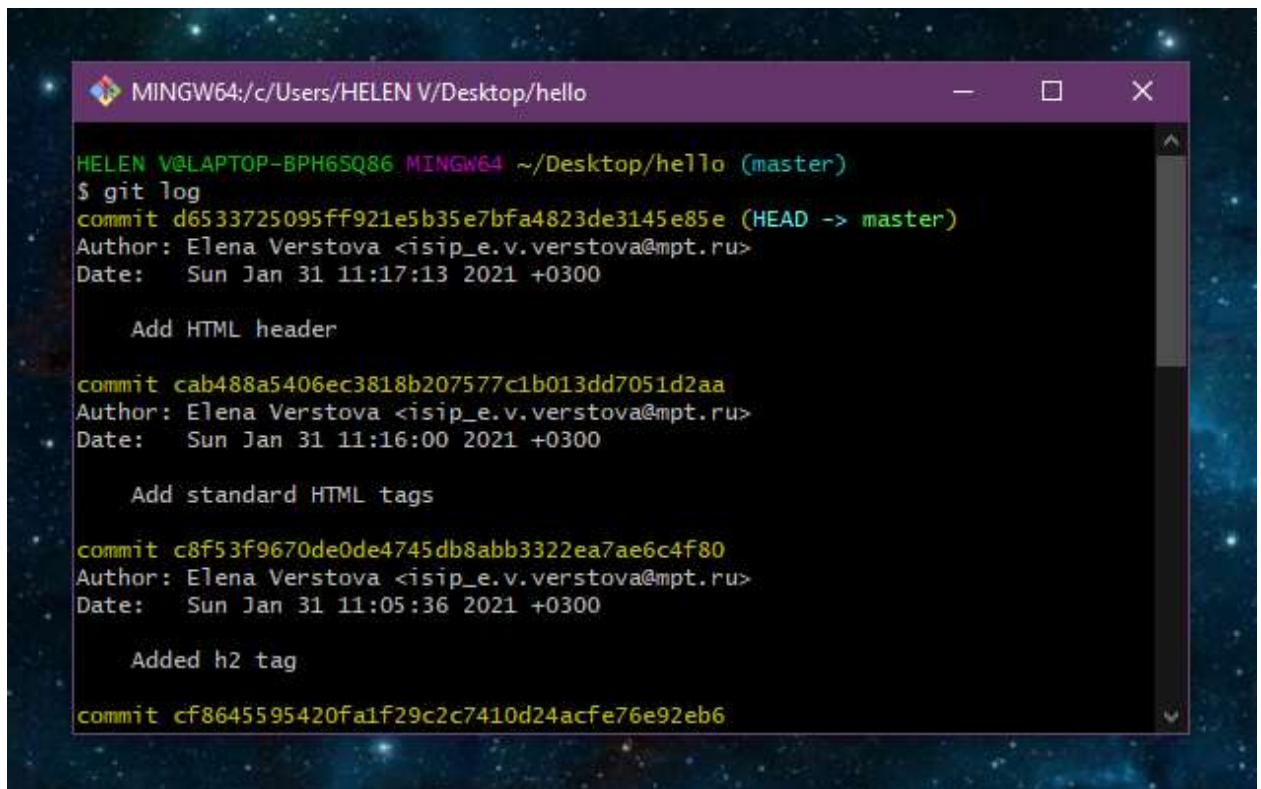
kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ git commit -m "Add HTML header"
> git commit -m "Add HTML header"
> svfknjbvibhiobvf
>

```

Работа с Git

История

Для того, чтобы просмотреть список произведенных изменений в проекте, используется команда `git log` (рисунок 1).



```
MINGW64:/c/Users/HELEN V/Desktop/hello

HELEN V@LAPTOP-BPH65Q86 MINGW64 ~/Desktop/hello (master)
$ git log
commit d6533725095ff921e5b35e7bfa4823de3145e85e (HEAD -> master)
Author: Elena Verstova <isip_e.v.verstova@mpt.ru>
Date: Sun Jan 31 11:17:13 2021 +0300

    Add HTML header

commit cab488a5406ec3818b207577c1b013dd7051d2aa
Author: Elena Verstova <isip_e.v.verstova@mpt.ru>
Date: Sun Jan 31 11:16:00 2021 +0300

    Add standard HTML tags

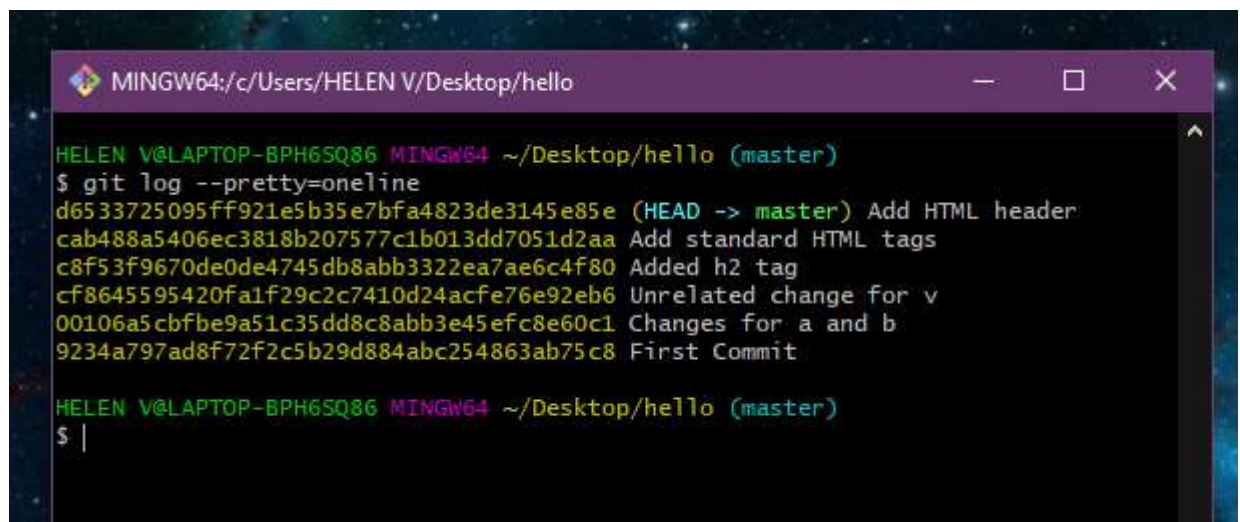
commit c8f53f9670de0de4745db8abb3322ea7ae6c4f80
Author: Elena Verstova <isip_e.v.verstova@mpt.ru>
Date: Sun Jan 31 11:05:36 2021 +0300

    Added h2 tag

commit cf8645595420fa1f29c2c7410d24acfe76e92eb6
```

Рисунок 25 - Просмотр истории изменений

На рисунке 1 была выведена полная история. Для того, чтобы увидеть однострочный формат используется команда `git log --pretty=oneline` (рисунок 2).



```
MINGW64:/c/Users/HELEN V/Desktop/hello

HELEN V@LAPTOP-BPH65Q86 MINGW64 ~/Desktop/hello (master)
$ git log --pretty=oneline
d6533725095ff921e5b35e7bfa4823de3145e85e (HEAD -> master) Add HTML header
cab488a5406ec3818b207577c1b013dd7051d2aa Add standard HTML tags
c8f53f9670de0de4745db8abb3322ea7ae6c4f80 Added h2 tag
cf8645595420fa1f29c2c7410d24acfe76e92eb6 Unrelated change for v
00106a5cbf9e9a51c35dd8c8abb3e45efc8e60c1 Changes for a and b
9234a797ad8f72f2c5b29d884abc254863ab75c8 First Commit

HELEN V@LAPTOP-BPH65Q86 MINGW64 ~/Desktop/hello (master)
$ |
```

Рисунок 26 - Однострочный формат вывода

Далее на рисунках 3-8 показано несколько вариантов вывода истории изменений.

```
HELEN V@LAPTOP-BPH65Q86 MINGW64 ~/Desktop/hello (master)
$ git log --pretty=oneline --max-count=2
d6533725095ff921e5b35e7bfa4823de3145e85e (HEAD -> master) Add HTML header
cab488a5406ec3818b207577c1b013dd7051d2aa Add standard HTML tags
```

Рисунок 27 - Вывод последних 2 изменений

```
HELEN V@LAPTOP-BPH65Q86 MINGW64 ~/Desktop/hello (master)
$ git log --pretty=oneline --since='10 days ago'

HELEN V@LAPTOP-BPH65Q86 MINGW64 ~/Desktop/hello (master)
$ git log --pretty=oneline --since='12 days ago'
d6533725095ff921e5b35e7bfa4823de3145e85e (HEAD -> master) Add HTML header
cab488a5406ec3818b207577c1b013dd7051d2aa Add standard HTML tags
c8f53f9670de0de4745db8abb3322ea7ae6c4f80 Added h2 tag
cf8645595420fa1f29c2c7410d24acfe76e92eb6 Unrelated change for v
00106a5cbf9e9a51c35dd8c8abb3e45efc8e60c1 Changes for a and b
9234a797ad8f72f2c5b29d884abc254863ab75c8 First Commit
```

Рисунок 28 - Вывод изменений начиная с определенного времени

```
HELEN V@LAPTOP-BPH65Q86 MINGW64 ~/Desktop/hello (master)
$ git log --pretty=oneline --until='5 minutes ago'
d6533725095ff921e5b35e7bfa4823de3145e85e (HEAD -> master) Add HTML header
cab488a5406ec3818b207577c1b013dd7051d2aa Add standard HTML tags
c8f53f9670de0de4745db8abb3322ea7ae6c4f80 Added h2 tag
cf8645595420fa1f29c2c7410d24acfe76e92eb6 Unrelated change for v
00106a5cbf9e9a51c35dd8c8abb3e45efc8e60c1 Changes for a and b
9234a797ad8f72f2c5b29d884abc254863ab75c8 First Commit
```

Рисунок 29 - Вывод изменений до определенного времени

```
HELEN V@LAPTOP-BPH65Q86 MINGW64 ~/Desktop/hello (master)
$ git log --pretty=oneline --author='Elena Verstova'
d6533725095ff921e5b35e7bfa4823de3145e85e (HEAD -> master) Add HTML header
cab488a5406ec3818b207577c1b013dd7051d2aa Add standard HTML tags
c8f53f9670de0de4745db8abb3322ea7ae6c4f80 Added h2 tag
cf8645595420fa1f29c2c7410d24acfe76e92eb6 Unrelated change for v
00106a5cbf9e9a51c35dd8c8abb3e45efc8e60c1 Changes for a and b
9234a797ad8f72f2c5b29d884abc254863ab75c8 First Commit
```

Рисунок 30 - Вывод изменений, внесенных определенным автором

```
HELEN V@LAPTOP-BPH65Q86 MINGW64 ~/Desktop/hello (master)
$ git log --pretty=oneline --all
d6533725095ff921e5b35e7bfa4823de3145e85e (HEAD -> master) Add HTML header
cab488a5406ec3818b207577c1b013dd7051d2aa Add standard HTML tags
c8f53f9670de0de4745db8abb3322ea7ae6c4f80 Added h2 tag
cf8645595420fa1f29c2c7410d24acfe76e92eb6 Unrelated change for v
00106a5cbf9e9a51c35dd8c8abb3e45efc8e60c1 Changes for a and b
9234a797ad8f72f2c5b29d884abc254863ab75c8 First Commit
```

Рисунок 31 - Вывод всех изменений


```
HELEN V@LAPTOP-BPH65Q86 MINGW64 ~/Desktop/hello (master)
$ git log --all --pretty=format:"%h %cd %s (%an)" --since='12 days ago'
d653372 Sun Jan 31 11:17:13 2021 +0300 Add HTML header (Elena Verstova)
cab488a Sun Jan 31 11:16:00 2021 +0300 Add standard HTML tags (Elena Verstova)
c8f53f9 Sun Jan 31 11:05:36 2021 +0300 Added h2 tag (Elena Verstova)
cf86455 Sun Jan 31 10:50:51 2021 +0300 Unrelated change for v (Elena Verstova)
00106a5 Sun Jan 31 10:50:12 2021 +0300 Changes for a and b (Elena Verstova)
9234a79 Sun Jan 31 10:37:42 2021 +0300 First Commit (Elena Verstova)

HELEN V@LAPTOP-BPH65Q86 MINGW64 ~/Desktop/hello (master)
$
```

Рисунок 32 - Использование нескольких параметров

Алиасы

Для настройки алиасов используется команда, показанная на рисунке 9.

```
MINGW64/c/Users/HELEN V/Desktop/hello
HELEN V@LAPTOP-BPH65Q86 MINGW64 ~/Desktop/hello (master)
$ git config --global alias.co checkout

HELEN V@LAPTOP-BPH65Q86 MINGW64 ~/Desktop/hello (master)
$ git config --global alias.ci commit

HELEN V@LAPTOP-BPH65Q86 MINGW64 ~/Desktop/hello (master)
$ git config --global alias.st status

HELEN V@LAPTOP-BPH65Q86 MINGW64 ~/Desktop/hello (master)
$ git config --global alias.br branch

HELEN V@LAPTOP-BPH65Q86 MINGW64 ~/Desktop/hello (master)
$ git config --global alias.hist "log --pretty=format:'%h %ad | %s%d [%an]' --graph --date=short"

HELEN V@LAPTOP-BPH65Q86 MINGW64 ~/Desktop/hello (master)
$ git config --global alias.type 'cat-file -t'

HELEN V@LAPTOP-BPH65Q86 MINGW64 ~/Desktop/hello (master)
$ git config --global alias.dump 'cat-file -p'
```

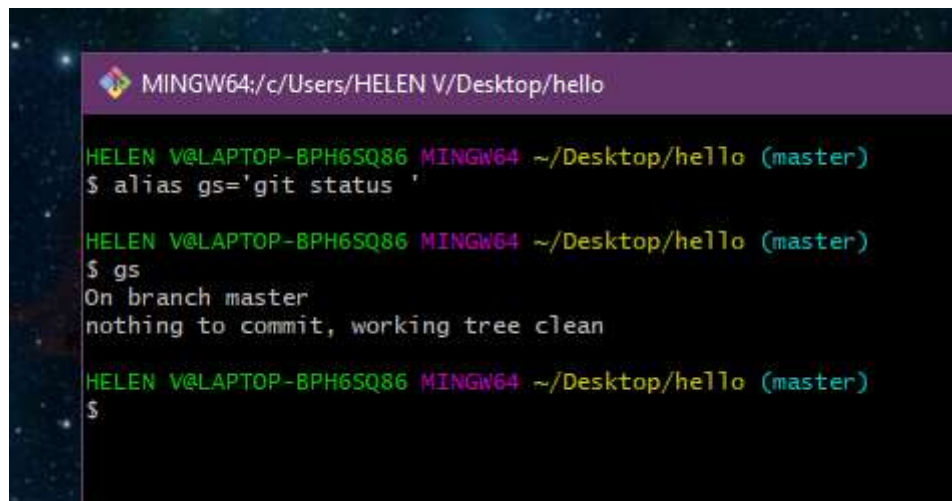
Рисунок 33 - Настройка алиасов для некоторых команд

При выполнении алиаса будет выполнена определенная команда и выведены нужные данные (рисунок 10).

```
HELEN V@LAPTOP-BPH65Q86 MINGW64 ~/Desktop/hello (master)
$ git config --global alias.hist "log --pretty=format:'%h %ad | %s%d [%an]' --graph --date=short"

HELEN V@LAPTOP-BPH65Q86 MINGW64 ~/Desktop/hello (master)
$ git hist
* d653372 2021-01-31 | Add HTML header (HEAD -> master) [Elena Verstova]
* cab488a 2021-01-31 | Add standard HTML tags [Elena Verstova]
* c8f53f9 2021-01-31 | Added h2 tag [Elena Verstova]
* cf86455 2021-01-31 | Unrelated change for v [Elena Verstova]
* 00106a5 2021-01-31 | Changes for a and b [Elena Verstova]
* 9234a79 2021-01-31 | First Commit [Elena Verstova]
```

Рисунок 34 - Выполнение алиаса hist



```
MINGW64:/c/Users/HELEN V/Desktop/hello

HELEN V@LAPTOP-BPH6SQ86 MINGW64 ~/Desktop/hello (master)
$ alias gs='git status '

HELEN V@LAPTOP-BPH6SQ86 MINGW64 ~/Desktop/hello (master)
$ gs
On branch master
nothing to commit, working tree clean

HELEN V@LAPTOP-BPH6SQ86 MINGW64 ~/Desktop/hello (master)
$
```

Рисунок 35 - Установка и выполнение алиаса gs

```

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop
$ cd hello

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ git hist
fatal: ambiguous argument 'graph': unknown revision or path not in the working tree.
Use '--' to separate paths from revisions, like this:
'git <command> [<revision>...] -- [<file>...]'

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ ^C

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ git config --global alias.hist "log -pretty=format:'%h %ad | %s%d [%an]' --graph --date=short"

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ git hist
fatal: unrecognized argument: -pretty=format:%h %ad | %s%d [%an]

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ git config --global alias.co checkout

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ git config --global alias.ci commit

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ git config --global alias.st status

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ git config --global alias.br branch

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ git config --global alias.hist "log -pretty=format:'%h %ad | %s%d [%an]' --graph --date=short"

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ git config --global alias.type 'cat-file -t'

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ git config --global alias.dump 'cat-file -p'

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ git config --global alias.hist "log --pretty=format:'%h %ad | %s%d [%an]' --graph --date=short"

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ git hist
* 895c6d9 2025-06-07 | Add HTML header (HEAD -> master) [Fedor Kozhurin]
* ecf6696 2025-06-07 | Unrelated change for v [Fedor Kozhurin]
* 02333b6 2025-06-07 | Changes for a and b [Fedor Kozhurin]
* 9d515cd 2025-06-06 | First Commit [Fedor Kozhurin]

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ alias gs='git status'

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ gs
On branch master
nothing to commit, working tree clean

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ |

```

Получение старых версий

Для того, чтобы вернуть рабочий каталог к предыдущему состоянию, можно использовать следующий способ: для начала нужно узнать хэши предыдущих версий, что можно сделать с помощью ранее заданного алиаса hist (рисунок 12).


```
MINGW64:/c/Users/HELEN V/Desktop/hello

HELEN V@LAPTOP-BPH6SQ86 MINGW64 ~/Desktop/hello (master)
$ git hist
* d653372 2021-01-31 | Add HTML header (HEAD -> master) [Elena Verstova]
* cab488a 2021-01-31 | Add standard HTML tags [Elena Verstova]
* c8f53f9 2021-01-31 | Added h2 tag [Elena Verstova]
* cf86455 2021-01-31 | Unrelated change for v [Elena Verstova]
* 00106a5 2021-01-31 | Changes for a and b [Elena Verstova]
* 9234a79 2021-01-31 | First Commit [Elena Verstova]
```

Рисунок 36 - Просмотр хэшей предыдущих версий

Далее нужно выполнить команду `git checkout` с номером нужного хэша (достаточно первых 7 знаков). После этого можно просмотреть содержимое файла с помощью команды `cat` (рисунок 13).

```
HELEN V@LAPTOP-BPH6SQ86 MINGW64 ~/Desktop/hello (master)
$ git checkout 9234a79
Note: switching to '9234a79'.

You are in 'detached HEAD' state. You can look around, make experimental
changes and commit them, and you can discard any commits you make in this
state without impacting any branches by switching back to a branch.

If you want to create a new branch to retain commits you create, you may
do so (now or later) by using -c with the switch command. Example:

    git switch -c <new-branch-name>

Or undo this operation with:

    git switch -

Turn off this advice by setting config variable advice.detachedHead to false

HEAD is now at 9234a79 First Commit

HELEN V@LAPTOP-BPH6SQ86 MINGW64 ~/Desktop/hello ((9234a79...))
$ cat hello.html
Hello, world
HELEN V@LAPTOP-BPH6SQ86 MINGW64 ~/Desktop/hello ((9234a79...))
$ |
```

Рисунок 37 - Возвращение к нужной версии и просмотр содержимого файла

Возвращение к последней версии в ветке master

Для возвращения к последней версии в ветке `master` (имя ветки по умолчанию) надо ввести команду `git checkout master`, что показано на рисунке 14.

```

HELEN V@LAPTOP-BPH6SQ86 MINGW64 ~/Desktop/hello ((9234a79...))
$ git checkout master
Previous HEAD position was 9234a79 First Commit
Switched to branch 'master'

HELEN V@LAPTOP-BPH6SQ86 MINGW64 ~/Desktop/hello (master)
$ cat hello.html
<html>
  <head>

  </head>
  <body>
    <h1>Hello, world</h1>
    <h2>Second line</h2>
  </body>
</html>
HELEN V@LAPTOP-BPH6SQ86 MINGW64 ~/Desktop/hello (master)
$

```

Рисунок 38 - Возвращение к последней версии в ветке master

```

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ git hist
* 895c6d9 2025-06-07 | Add HTML header (HEAD -> master) [Fedor Kozhurin]
* ecf6696 2025-06-07 | Unrelated change for v [Fedor Kozhurin]
* 02333b6 2025-06-07 | Changes for a and b [Fedor Kozhurin]
* 9d515cd 2025-06-06 | First Commit [Fedor Kozhurin]

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ git checkout 9d515cd
Note: switching to '9d515cd'.

You are in 'detached HEAD' state. You can look around, make experimental
changes and commit them, and you can discard any commits you make in this
state without impacting any branches by switching back to a branch.

If you want to create a new branch to retain commits you create, you may
do so (now or later) by using -c with the switch command. Example:

    git switch -c <new-branch-name>

Or undo this operation with:

    git switch -

Turn off this advice by setting config variable advice.detachedHead to false

HEAD is now at 9d515cd First Commit

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello ((9d515cd...))
$ cat hello.html
Hello, world
kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello ((9d515cd...))
$ git checkout master
Previous HEAD position was 9d515cd First Commit
Switched to branch 'master'

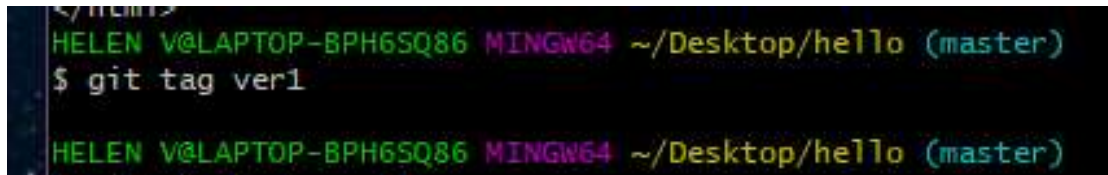
kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ cat hello.html
<html>
  <head>

  </head>
  <body>
    <h1>Hello, world</h1>
    <h2>Second line</h2>
  </body>
</html>

```


Создание тегов версий

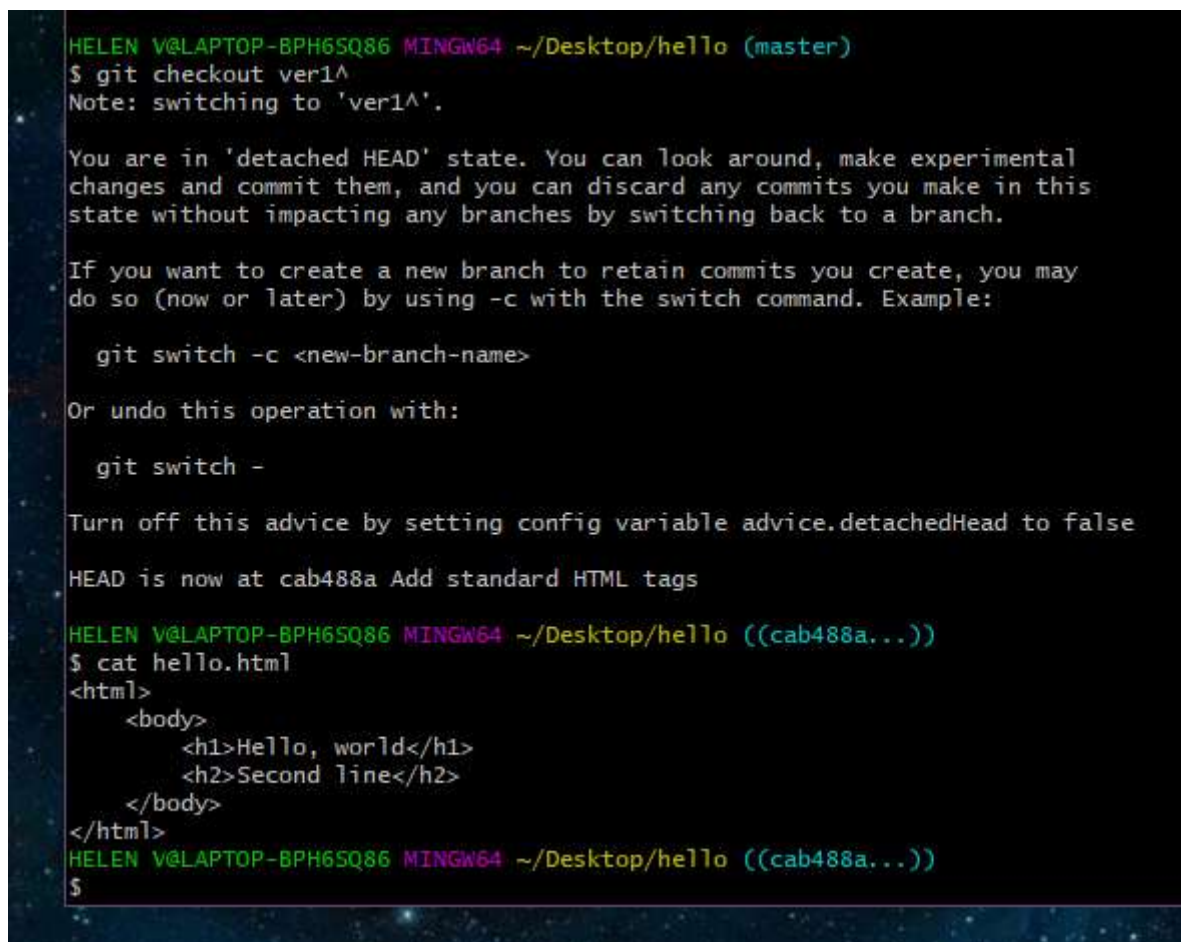
Для создания тега используется команда `git tag`. На рисунке 15 показано, тегом `ver1` была названа текущая версия страницы.



```
HELEN V@LAPTOP-BPH65Q86 MINGW64 ~/Desktop/hello (master)
$ git tag ver1
HELEN V@LAPTOP-BPH65Q86 MINGW64 ~/Desktop/hello (master)
```

Рисунок 39 - Задание тега

Чтобы перейти к предыдущей версии, можно использовать символ «`^`», который означает «родитель».



```
HELEN V@LAPTOP-BPH65Q86 MINGW64 ~/Desktop/hello (master)
$ git checkout ver1^
Note: switching to 'ver1^'.

You are in 'detached HEAD' state. You can look around, make experimental
changes and commit them, and you can discard any commits you make in this
state without impacting any branches by switching back to a branch.

If you want to create a new branch to retain commits you create, you may
do so (now or later) by using -c with the switch command. Example:

    git switch -c <new-branch-name>

Or undo this operation with:

    git switch -

Turn off this advice by setting config variable advice.detachedHead to false

HEAD is now at cab488a Add standard HTML tags
HELEN V@LAPTOP-BPH65Q86 MINGW64 ~/Desktop/hello ((cab488a...))
$ cat hello.html
<html>
  <body>
    <h1>Hello, world</h1>
    <h2>Second line</h2>
  </body>
</html>
HELEN V@LAPTOP-BPH65Q86 MINGW64 ~/Desktop/hello ((cab488a...))
$
```

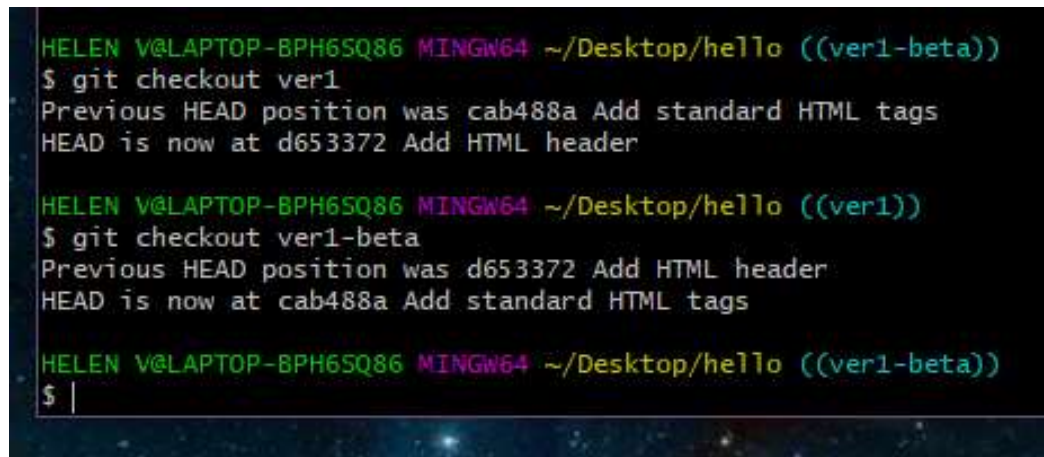
Рисунок 40 – Переход к предыдущей версии с помощью тега



```
HELEN V@LAPTOP-BPH65Q86 MINGW64 ~/Desktop/hello ((cab488a...))
$ git tag ver1-beta
HELEN V@LAPTOP-BPH65Q86 MINGW64 ~/Desktop/hello ((ver1-beta))
$
```

Рисунок 41 - Задание тега предыдущей версии

Теперь с помощью тегов можно переключаться между версиями (рисунок 18).



```
HELEN V@LAPTOP-BPH65Q86 MINGW64 ~/Desktop/hello ((ver1-beta))
$ git checkout ver1
Previous HEAD position was cab488a Add standard HTML tags
HEAD is now at d653372 Add HTML header

HELEN V@LAPTOP-BPH65Q86 MINGW64 ~/Desktop/hello ((ver1))
$ git checkout ver1-beta
Previous HEAD position was d653372 Add HTML header
HEAD is now at cab488a Add standard HTML tags

HELEN V@LAPTOP-BPH65Q86 MINGW64 ~/Desktop/hello ((ver1-beta))
$ |
```

Рисунок 42 - Переключение между версиями с помощью тегов

Для просмотра всех тегов используется команда `git tag` (рисунок 19).



```
HELEN V@LAPTOP-BPH65Q86 MINGW64 ~/Desktop/hello ((ver1-beta))
$ git tag
ver1
ver1-beta

HELEN V@LAPTOP-BPH65Q86 MINGW64 ~/Desktop/hello ((ver1-beta))
$ |
```

Рисунок 43 - Просмотр тегов

Также можно просмотреть теги в логе, как показано на рисунке 20.

```
HELEN V@LAPTOP-BPH6SQ86 MINGW64 ~/Desktop/hello ((ver1-beta))
$ git hist master --all
* d653372 2021-01-31 | Add HTML header (tag: ver1, master) [Elena Verstova]
* cab488a 2021-01-31 | Add standard HTML tags (HEAD, tag: ver1-beta) [Elena Verstova]
* c8f53f9 2021-01-31 | Added h2 tag [Elena Verstova]
* cf86455 2021-01-31 | Unrelated change for v [Elena Verstova]
* 00106a5 2021-01-31 | Changes for a and b [Elena Verstova]
* 9234a79 2021-01-31 | First Commit [Elena Verstova]
HELEN V@LAPTOP-BPH6SQ86 MINGW64 ~/Desktop/hello ((ver1-beta))
```

Рисунок 44 - Просмотр тегов в логе

```
MINGW64:/c/Users/kozur/Desktop/hello
$ git tag ver1

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ git checkout ver1^
Note: switching to 'ver1^'.

You are in 'detached HEAD' state. You can look around, make experimental
changes and commit them, and you can discard any commits you make in this
state without impacting any branches by switching back to a branch.

If you want to create a new branch to retain commits you create, you may
do so (now or later) by using -c with the switch command. Example:

    git switch -c <new-branch-name>

Or undo this operation with:

    git switch -

Turn off this advice by setting config variable advice.detachedHead to false

HEAD is now at ecf6696 Unrelated change for v

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello ((ecf6696...))
$ cat hello.html
<h1>Hello, world</h1>
kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello ((ecf6696...))
$ ^C

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello ((ecf6696...))
$ cat hello.html
<html>
  <head>

  </head>
  <body>
    <h1>Hello, world</h1>
    <h2>Second line</h2>
  </body>
</html>

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello ((ecf6696...))
$ git tag ver1-beta

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello ((ver1-beta))
$ git checkout ver1
error: Your local changes to the following files would be overwritten by checkout:
    hello.html
Please commit your changes or stash them before you switch branches.
Aborting

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello ((ver1-beta))
$ git checkout ver1-beta
M    hello.html
HEAD is now at ecf6696 Unrelated change for v

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello ((ver1-beta))
$ git tag
ver1
ver1-beta

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello ((ver1-beta))
$ git hist master --all
* 895c6d9 2025-06-07 | Add HTML header (tag: ver1, master) [Fedor Kozhurin]
* ecf6696 2025-06-07 | Unrelated change for v (HEAD, tag: ver1-beta) [Fedor Kozhurin]
```

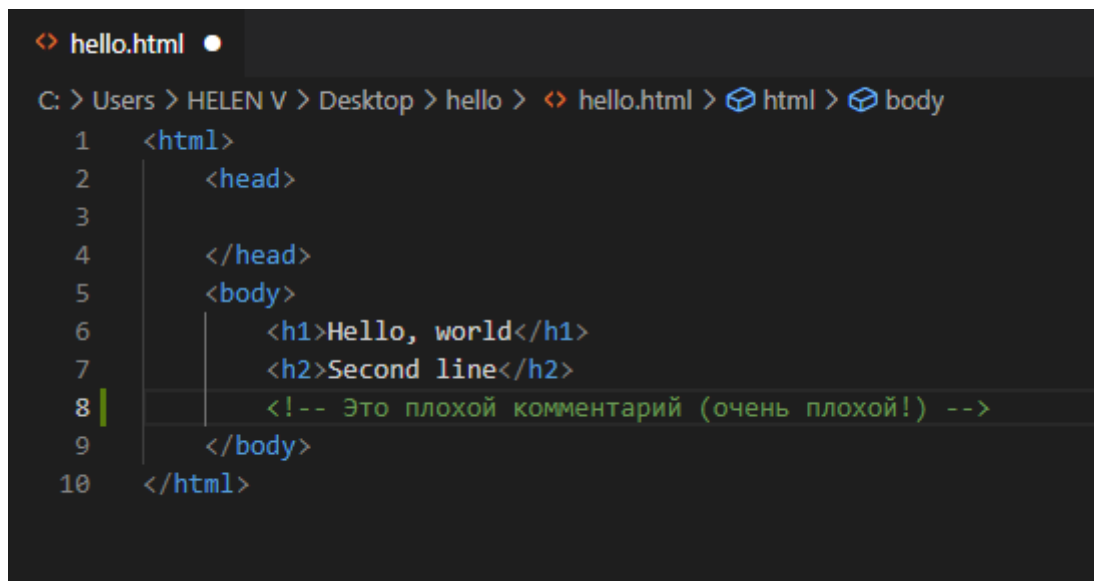
Отмена локальных изменений (до индексации)

Сначала нужно переключиться на последний коммит master (рисунок 21).

```
HELEN V@LAPTOP-BPH6SQ86 MINGW64 ~/Desktop/hello ((ver1-beta))
$ git checkout master
Previous HEAD position was cab488a Add standard HTML tags
Switched to branch 'master'
```

Рисунок 45 - Переключение на последний коммит

Далее для работы нужно внести изменение в файл (рисунок 22).



```
<> hello.html ●
C: > Users > HELEN V > Desktop > hello > <> hello.html > html > body
1  <html>
2      <head>
3
4      </head>
5      <body>
6          <h1>Hello, world</h1>
7          <h2>Second line</h2>
8      <!-- Это плохой комментарий (очень плохой!) -->
9      </body>
10 </html>
```

Рисунок 46 - Внесение изменения в файл

После выполнения команды `git status` будет показано, что есть не проиндексированное изменение (рисунок 23).

```
HELEN V@LAPTOP-BPH6SQ86 MINGW64 ~/Desktop/hello (master)
$ git st
On branch master
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   hello.html

no changes added to commit (use "git add" and/or "git commit -a")
HELEN V@LAPTOP-BPH6SQ86 MINGW64 ~/Desktop/hello (master)
$
```

Рисунок 47 - Не проиндексированное изменение

Для переключения в версию файла без изменений используется команда `git checkout hello.html` (рисунок 24). Команда `git status` покажет, что не было произведено изменений, не зафиксированных в каталоге.


```

HELEN V@LAPTOP-BPH6SQ86 MINGW64 ~/Desktop/hello (master)
$ git checkout hello.html
Updated 1 path from the index

HELEN V@LAPTOP-BPH6SQ86 MINGW64 ~/Desktop/hello (master)
$ git st
On branch master
nothing to commit, working tree clean

HELEN V@LAPTOP-BPH6SQ86 MINGW64 ~/Desktop/hello (master)
$ cat hello.html
<html>
  <head>

  </head>
  <body>
    <h1>Hello, world</h1>
    <h2>Second line</h2>
  </body>
</html>
HELEN V@LAPTOP-BPH6SQ86 MINGW64 ~/Desktop/hello (master)

```

Рисунок 48 - Возвращение к версии

```

Switched to branch 'master'
kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ git st
On branch master
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   hello.html

no changes added to commit (use "git add" and/or "git commit -a")

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ git checkout hello.html
Updated 1 path from the index

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ git st
On branch master
nothing to commit, working tree clean

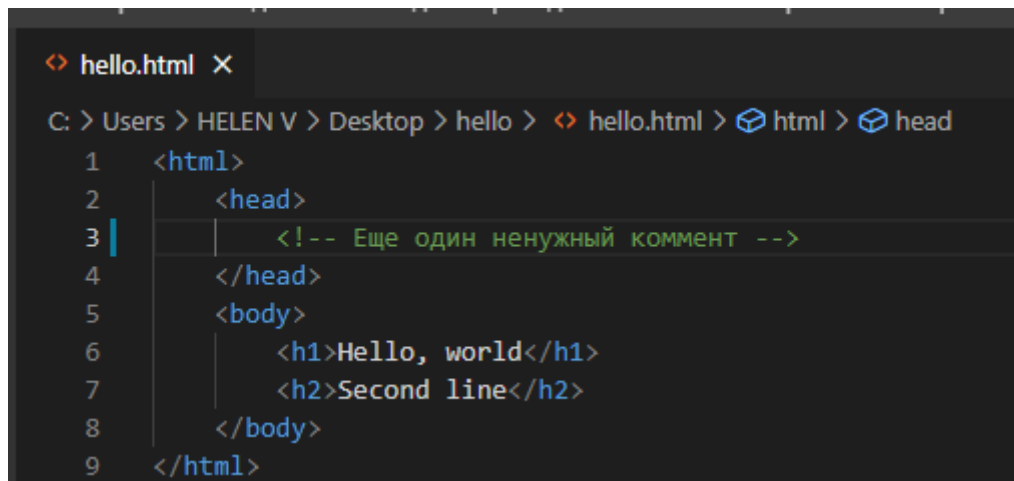
kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ cat hello.html
<html>
  <head>

  </head>
  <body>
    <h1>Hello, world</h1>
    <h2>Second line</h2>
  </body>
</html>
kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$

```

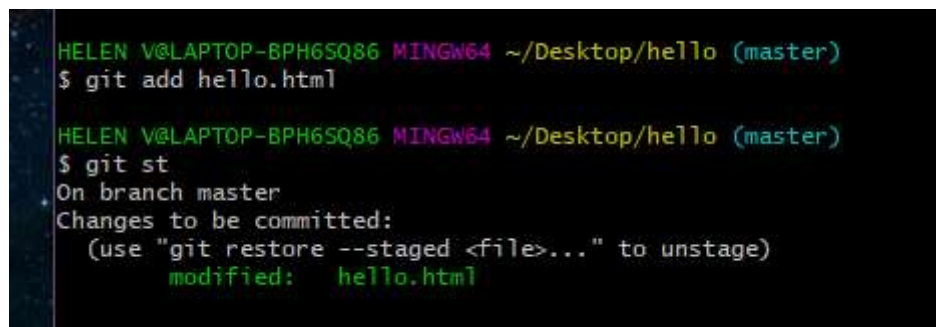
Отмена проиндексированных изменений (перед коммитом)

Для того, чтобы научиться отменять проиндексированные изменения, сначала нужно внести ненужное изменение в файл (рисунок 25). После этого производится индексация (рисунок 26).



```
<> hello.html X
C: > Users > HELEN V > Desktop > hello > <> hello.html > html > head
1 <html>
2   <head>
3   <!-- Еще один ненужный коммент -->
4   </head>
5   <body>
6     <h1>Hello, world</h1>
7     <h2>Second line</h2>
8   </body>
9 </html>
```

Рисунок 49 - Внесение ненужного изменения

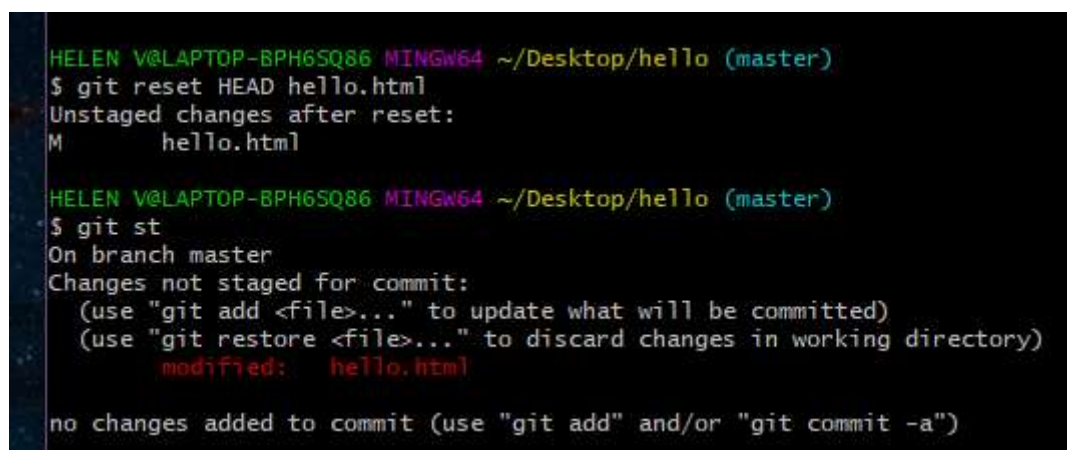


```
HELEN V@LAPTOP-BPH6SQ86 MINGW64 ~/Desktop/hello (master)
$ git add hello.html

HELEN V@LAPTOP-BPH6SQ86 MINGW64 ~/Desktop/hello (master)
$ git st
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    modified:   hello.html
```

Рисунок 50 - Индексация изменения

Для отмены индексация изменения используется команда `git reset HEAD hello.html` (рисунок 27). Команда `reset` сбрасывает буферную зону к HEAD и очищает ее от проиндексированных изменений. Но для удаления ненужного по-прежнему используется команда `git checkout` (рисунок 28).



```
HELEN V@LAPTOP-BPH6SQ86 MINGW64 ~/Desktop/hello (master)
$ git reset HEAD hello.html
Unstaged changes after reset:
M    hello.html

HELEN V@LAPTOP-BPH6SQ86 MINGW64 ~/Desktop/hello (master)
$ git st
On branch master
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
    modified:   hello.html

no changes added to commit (use "git add" and/or "git commit -a")
```

Рисунок 51 - Очистка буферной зоны


```

HELEN V@LAPTOP-BPH6SQ86 MINGW64 ~/Desktop/hello (master)
$ git checkout hello.html
Updated 1 path from the index

HELEN V@LAPTOP-BPH6SQ86 MINGW64 ~/Desktop/hello (master)
$ git st
On branch master
nothing to commit, working tree clean

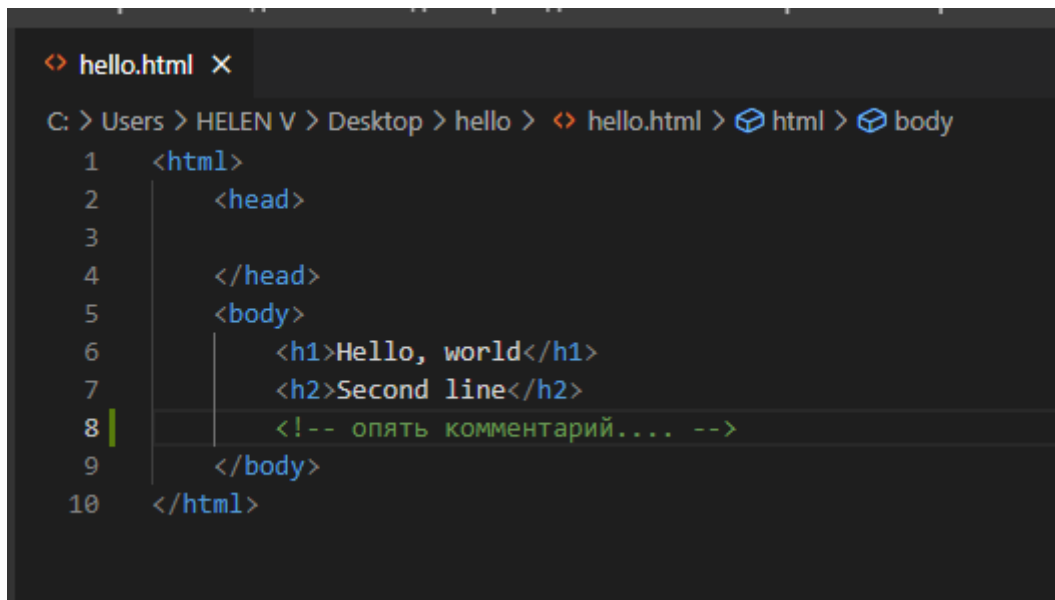
```

Рисунок 52 - Удаление ненужных изменений

Отмена коммитов

Для отмены коммита можно использовать способ создания нового коммита, отменяющего изменения.

Для начала надо внести изменение, проиндексировать его и записать коммит (рисунки 29-30).



```

<> hello.html X
C: > Users > HELEN V > Desktop > hello > <> hello.html > html > body
1  <html>
2      <head>
3
4      </head>
5      <body>
6          <h1>Hello, world</h1>
7          <h2>Second line</h2>
8      <!-- опять комментарий.... -->
9      </body>
10 </html>

```

Рисунок 53 - Внесение изменения в файл

```

HELEN V@LAPTOP-BPH6SQ86 MINGW64 ~/Desktop/hello (master)
$ git add hello.html

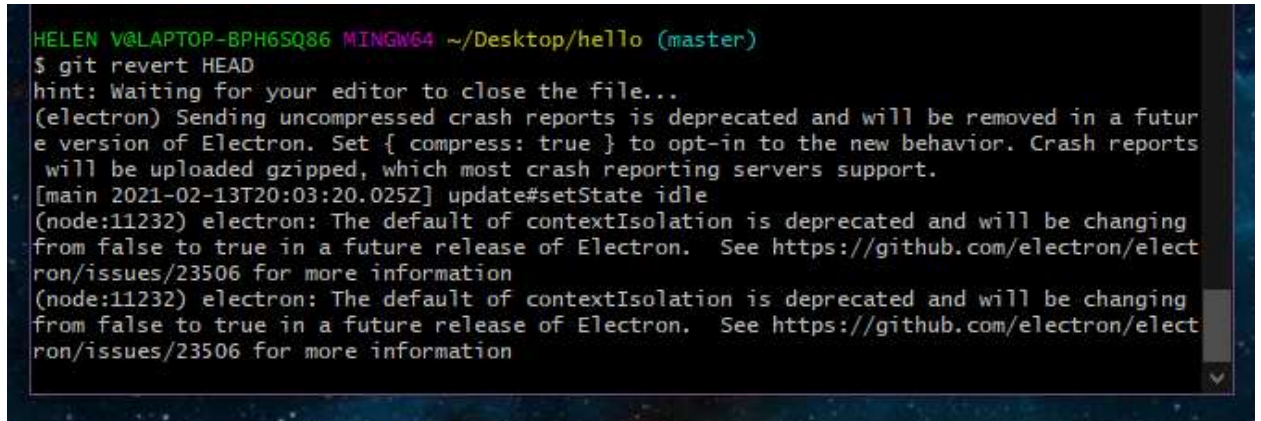
HELEN V@LAPTOP-BPH6SQ86 MINGW64 ~/Desktop/hello (master)
$ git commit -m "Oops, this is bad commit"
[master ce12d52] Oops, this is bad commit
1 file changed, 1 insertion(+)

HELEN V@LAPTOP-BPH6SQ86 MINGW64 ~/Desktop/hello (master)
$

```

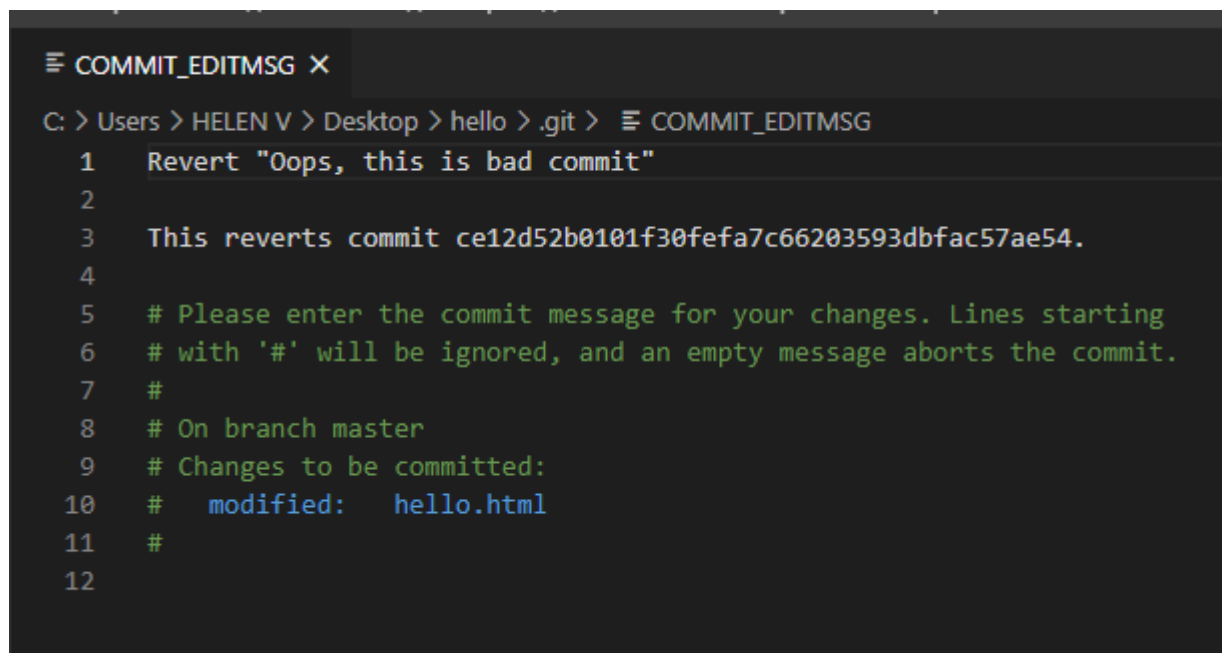
Рисунок 54 - Индексация и коммит

Для создания коммита, который удалит ненужные изменения, используется команда `git revert HEAD` (рисунок 31). После этого будет открыт редактор, в котором можно отредактировать коммит сообщение (рисунок 32), затем надо сохранить файл и закрыть редактор (рисунок 33).



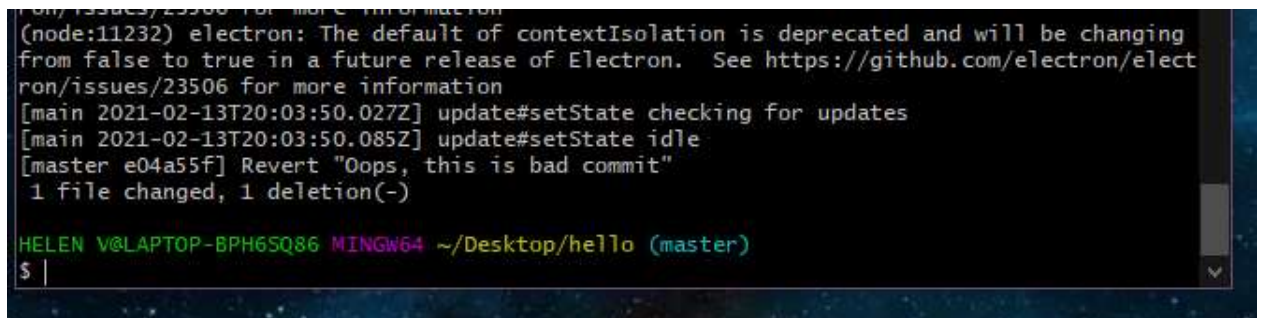
```
HELEN V@LAPTOP-BPH6SQ86 MINGW64 ~/Desktop/hello (master)
$ git revert HEAD
hint: Waiting for your editor to close the file...
(electron) Sending uncompressed crash reports is deprecated and will be removed in a future
version of Electron. Set { compress: true } to opt-in to the new behavior. Crash reports
will be uploaded gzipped, which most crash reporting servers support.
[main 2021-02-13T20:03:20.025Z] update#setState idle
(node:11232) electron: The default of contextIsolation is deprecated and will be changing
from false to true in a future release of Electron. See https://github.com/electron/elect
ron/issues/23506 for more information
(node:11232) electron: The default of contextIsolation is deprecated and will be changing
from false to true in a future release of Electron. See https://github.com/electron/elect
ron/issues/23506 for more information
```

Рисунок 55 - Выполнение команды `git revert`



```
COMMIT_EDITMSG X
C: > Users > HELEN V > Desktop > hello > .git > COMMIT_EDITMSG
1  Revert "Oops, this is bad commit"
2
3  This reverts commit ce12d52b0101f30fefa7c66203593dbfac57ae54.
4
5  # Please enter the commit message for your changes. Lines starting
6  # with '#' will be ignored, and an empty message aborts the commit.
7  #
8  # On branch master
9  # Changes to be committed:
10 #   modified:   hello.html
11 #
12
```

Рисунок 56 - Коммит сообщение в редакторе



```
(node:11232) electron: The default of contextIsolation is deprecated and will be changing
from false to true in a future release of Electron. See https://github.com/electron/elect
ron/issues/23506 for more information
[main 2021-02-13T20:03:50.027Z] update#setState checking for updates
[main 2021-02-13T20:03:50.085Z] update#setState idle
[master e04a55f] Revert "Oops, this is bad commit"
1 file changed, 1 deletion(-)

HELEN V@LAPTOP-BPH6SQ86 MINGW64 ~/Desktop/hello (master)
$ |
```

Рисунок 57 - Редактор закрыт

При проверке лога будут показаны все коммиты, в том числе и отмененные (рисунок 34).

```
HELEN V@LAPTOP-BPH6SQ86 MINGW64 ~/Desktop/hello (master)
$ git hist
* e04a55f 2021-02-13 | Revert "Oops, this is bad commit" (HEAD -> master) [Elena Verstova]
* ce12d52 2021-02-13 | Oops, this is bad commit [Elena Verstova]
* d653372 2021-01-31 | Add HTML header (tag: ver1) [Elena Verstova]
* cab488a 2021-01-31 | Add standard HTML tags (tag: ver1-beta) [Elena Verstova]
* c8f53f9 2021-01-31 | Added h2 tag [Elena Verstova]
* cf86455 2021-01-31 | Unrelated change for v [Elena Verstova]
* 00106a5 2021-01-31 | Changes for a and b [Elena Verstova]
* 9234a79 2021-01-31 | First Commit [Elena Verstova]

HELEN V@LAPTOP-BPH6SQ86 MINGW64 ~/Desktop/hello (master)
$
```

Рисунок 58 - Все коммиты при просмотре лога

Перед удалением коммита последний из них нужно отметить тегом, чтобы не потерять его (рисунок 35).

```
HELEN V@LAPTOP-BPH6SQ86 MINGW64 ~/Desktop/hello (master)
$ git tag oooops

HELEN V@LAPTOP-BPH6SQ86 MINGW64 ~/Desktop/hello (master)
$ git hist
* e04a55f 2021-02-13 | Revert "Oops, this is bad commit" (HEAD -> master, tag: oooops) [Elena Verstova]
* ce12d52 2021-02-13 | Oops, this is bad commit [Elena Verstova]
* d653372 2021-01-31 | Add HTML header (tag: ver1) [Elena Verstova]
* cab488a 2021-01-31 | Add standard HTML tags (tag: ver1-beta) [Elena Verstova]
* c8f53f9 2021-01-31 | Added h2 tag [Elena Verstova]
* cf86455 2021-01-31 | Unrelated change for v [Elena Verstova]
* 00106a5 2021-01-31 | Changes for a and b [Elena Verstova]
* 9234a79 2021-01-31 | First Commit [Elena Verstova]

HELEN V@LAPTOP-BPH6SQ86 MINGW64 ~/Desktop/hello (master)
$
```

Рисунок 59 - Отметка тегом коммита

Для сброса коммитов используется команда `git reset --hard ver1` (рисунок 36). Она сбрасывает ветку до версии с тегом `ver1`.


```
HELEN V@LAPTOP-BPH6SQ86 MINGW64 ~/Desktop/hello (master)
$ git reset --hard ver1
HEAD is now at d653372 Add HTML header

HELEN V@LAPTOP-BPH6SQ86 MINGW64 ~/Desktop/hello (master)
$ git hist
* d653372 2021-01-31 | Add HTML header (HEAD -> master, tag: ver1) [Elena Verstova]
* cab488a 2021-01-31 | Add standard HTML tags (tag: ver1-beta) [Elena Verstova]
* c8f53f9 2021-01-31 | Added h2 tag [Elena Verstova]
* cf86455 2021-01-31 | Unrelated change for v [Elena Verstova]
* 00106a5 2021-01-31 | Changes for a and b [Elena Verstova]
* 9234a79 2021-01-31 | First Commit [Elena Verstova]

HELEN V@LAPTOP-BPH6SQ86 MINGW64 ~/Desktop/hello (master)
$
```

Рисунок 60 - Сброс коммита

Но при просмотре лога с помощью команды `git hist --all` отмененные коммиты по-прежнему будут показываться, так как они всё еще находятся в репозитории (рисунок 37).

```
HELEN V@LAPTOP-BPH6SQ86 MINGW64 ~/Desktop/hello (master)
$ git hist --all
* e04a55f 2021-02-13 | Revert "Oops, this is bad commit" (tag: oooops) [Elena Verstova]
* ce12d52 2021-02-13 | Oops, this is bad commit [Elena Verstova]
* d653372 2021-01-31 | Add HTML header (HEAD -> master, tag: ver1) [Elena Verstova]
* cab488a 2021-01-31 | Add standard HTML tags (tag: ver1-beta) [Elena Verstova]
* c8f53f9 2021-01-31 | Added h2 tag [Elena Verstova]
* cf86455 2021-01-31 | Unrelated change for v [Elena Verstova]
* 00106a5 2021-01-31 | Changes for a and b [Elena Verstova]
* 9234a79 2021-01-31 | First Commit [Elena Verstova]

HELEN V@LAPTOP-BPH6SQ86 MINGW64 ~/Desktop/hello (master)
$
```

Рисунок 61 - Сброшенные коммиты находятся по-прежнему в репозитории

Удаление тега

Так как тег «oooops» больше не нужен, его и коммиты, на которые он указывает, можно удалить с помощью команды `git tag -d` (рисунок 38).

```
HELEN V@LAPTOP-BPH6SQ86 MINGW64 ~/Desktop/hello (master)
$ git tag -d oooops
Deleted tag 'oooops' (was e04a55f)

HELEN V@LAPTOP-BPH6SQ86 MINGW64 ~/Desktop/hello (master)
$ git hist --all
* d653372 2021-01-31 | Add HTML header (HEAD -> master, tag: ver1) [Elena Verstova]
* cab488a 2021-01-31 | Add standard HTML tags (tag: ver1-beta) [Elena Verstova]
* c8f53f9 2021-01-31 | Added h2 tag [Elena Verstova]
* cf86455 2021-01-31 | Unrelated change for v [Elena Verstova]
* 00106a5 2021-01-31 | Changes for a and b [Elena Verstova]
* 9234a79 2021-01-31 | First Commit [Elena Verstova]

HELEN V@LAPTOP-BPH6SQ86 MINGW64 ~/Desktop/hello (master)
$
```

Рисунок 62 - Удаление тега

```

[master f5b4187] Revert "Oops, this is bad commit"
1 file changed, 1 deletion(-)

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ git hist
* f5b4187 2025-06-07 | Revert "Oops, this is bad commit" (HEAD -> master) [Fedor Kozhurin]
* 24a8d27 2025-06-07 | Oops, this is bad commit [Fedor Kozhurin]
* 895c6d9 2025-06-07 | Add HTML header (tag: ver1) [Fedor Kozhurin]
* ecf6696 2025-06-07 | Unrelated change for v (tag: ver1-beta) [Fedor Kozhurin]
* 02333b6 2025-06-07 | Changes for a and b [Fedor Kozhurin]
* 9d515cd 2025-06-06 | First Commit [Fedor Kozhurin]

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ git tag oooops

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ git hist
* f5b4187 2025-06-07 | Revert "Oops, this is bad commit" (HEAD -> master, tag: oooops) [Fedor Kozhurin]
* 24a8d27 2025-06-07 | Oops, this is bad commit [Fedor Kozhurin]
* 895c6d9 2025-06-07 | Add HTML header (tag: ver1) [Fedor Kozhurin]
* ecf6696 2025-06-07 | Unrelated change for v (tag: ver1-beta) [Fedor Kozhurin]
* 02333b6 2025-06-07 | Changes for a and b [Fedor Kozhurin]
* 9d515cd 2025-06-06 | First Commit [Fedor Kozhurin]

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ git reset --hard ver1
HEAD is now at 895c6d9 Add HTML header

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ git hist
* 895c6d9 2025-06-07 | Add HTML header (HEAD -> master, tag: ver1) [Fedor Kozhurin]
* ecf6696 2025-06-07 | Unrelated change for v (tag: ver1-beta) [Fedor Kozhurin]
* 02333b6 2025-06-07 | Changes for a and b [Fedor Kozhurin]
* 9d515cd 2025-06-06 | First Commit [Fedor Kozhurin]

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ git hist --all
* f5b4187 2025-06-07 | Revert "Oops, this is bad commit" (tag: oooops) [Fedor Kozhurin]
* 24a8d27 2025-06-07 | Oops, this is bad commit [Fedor Kozhurin]
* 895c6d9 2025-06-07 | Add HTML header (HEAD -> master, tag: ver1) [Fedor Kozhurin]
* ecf6696 2025-06-07 | Unrelated change for v (tag: ver1-beta) [Fedor Kozhurin]
* 02333b6 2025-06-07 | Changes for a and b [Fedor Kozhurin]
* 9d515cd 2025-06-06 | First Commit [Fedor Kozhurin]

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ git tag -f oooops
Updated tag 'oooops' (was f5b4187)

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ git hist --all
* 895c6d9 2025-06-07 | Add HTML header (HEAD -> master, tag: ver1, tag: oooops) [Fedor Kozhurin]
* ecf6696 2025-06-07 | Unrelated change for v (tag: ver1-beta) [Fedor Kozhurin]
* 02333b6 2025-06-07 | Changes for a and b [Fedor Kozhurin]
* 9d515cd 2025-06-06 | First Commit [Fedor Kozhurin]

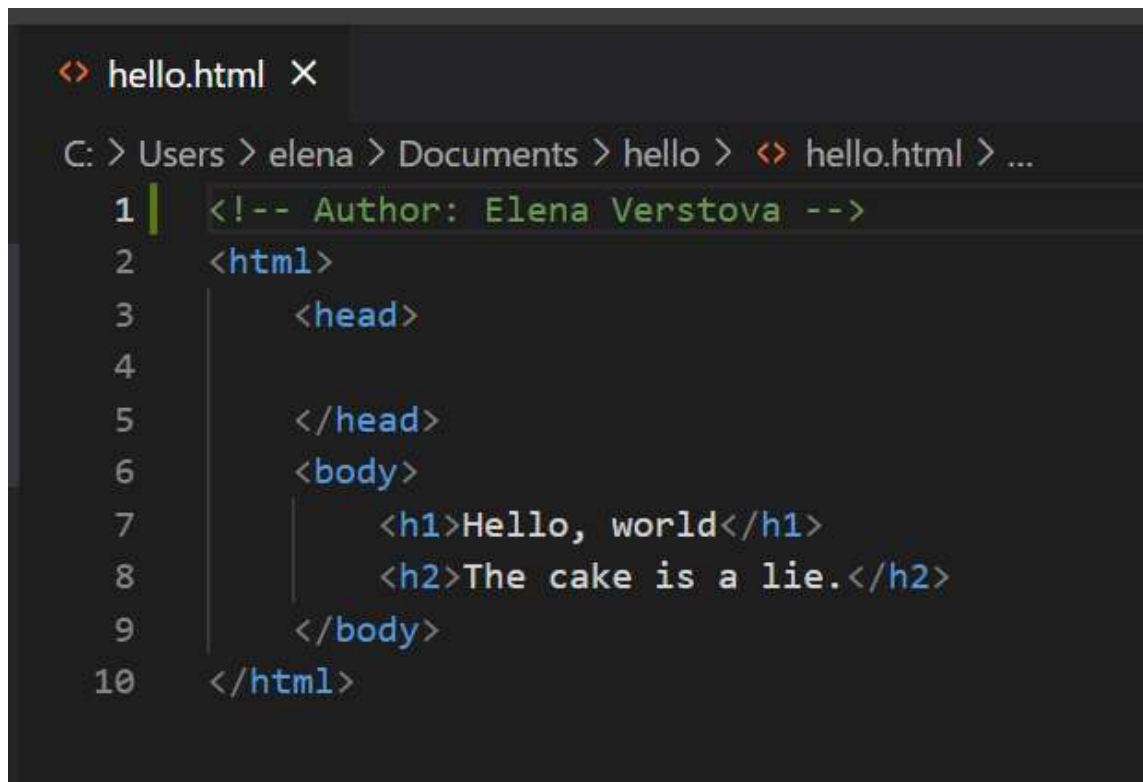
kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ |

```

Работа с Git

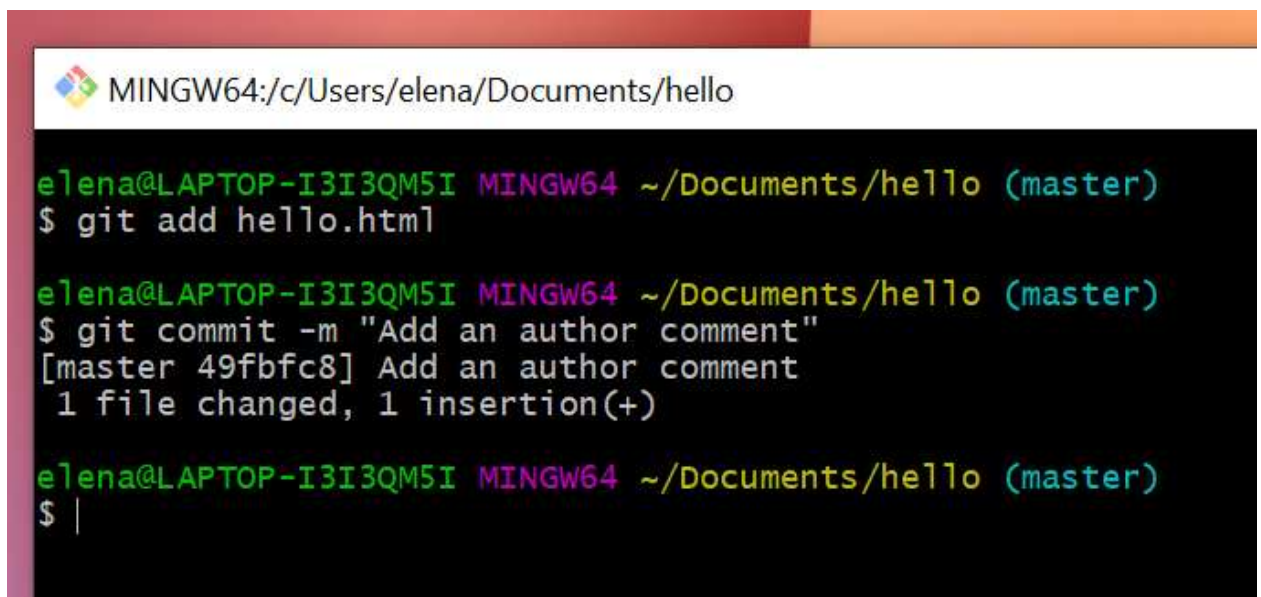
Внесение изменений в коммиты

Для начала будет создан коммит, в который позже будут внесены изменения. На рисунках 1 и 2 происходит добавление комментария в файл hello.html и его индексация и коммит.



```
<> hello.html X
C: > Users > elena > Documents > hello > <> hello.html > ...
1 | <!-- Author: Elena Verstova -->
2 | <html>
3 |   <head>
4 |
5 |   </head>
6 |   <body>
7 |     <h1>Hello, world</h1>
8 |     <h2>The cake is a lie.</h2>
9 |   </body>
10| </html>
```

Рисунок 63 - Добавление комментария в файл



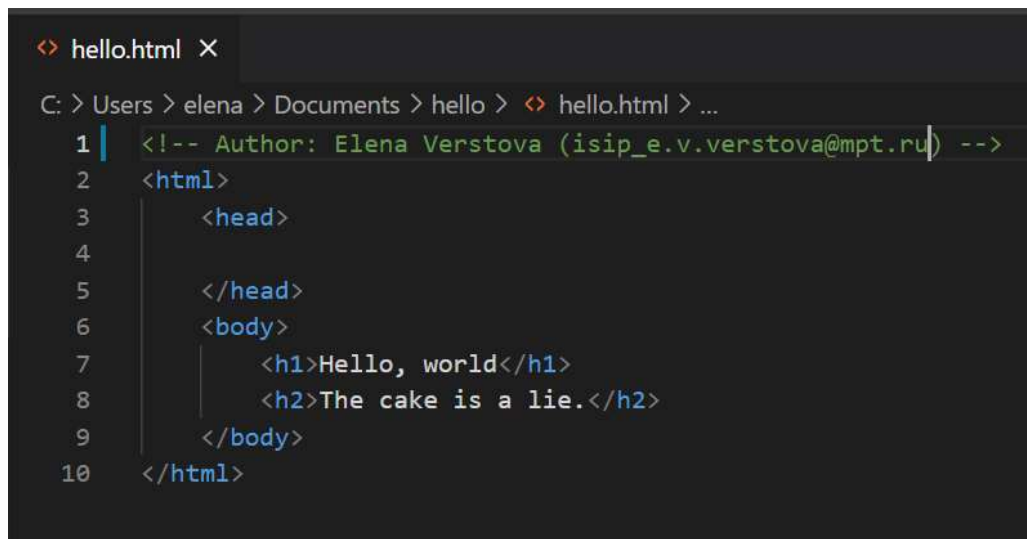
```
MINGW64:/c/Users/elena/Documents/hello
elena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (master)
$ git add hello.html

elena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (master)
$ git commit -m "Add an author comment"
[master 49fbfc8] Add an author comment
1 file changed, 1 insertion(+)

elena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (master)
$ |
```

Рисунок 64 - Индексация и коммит

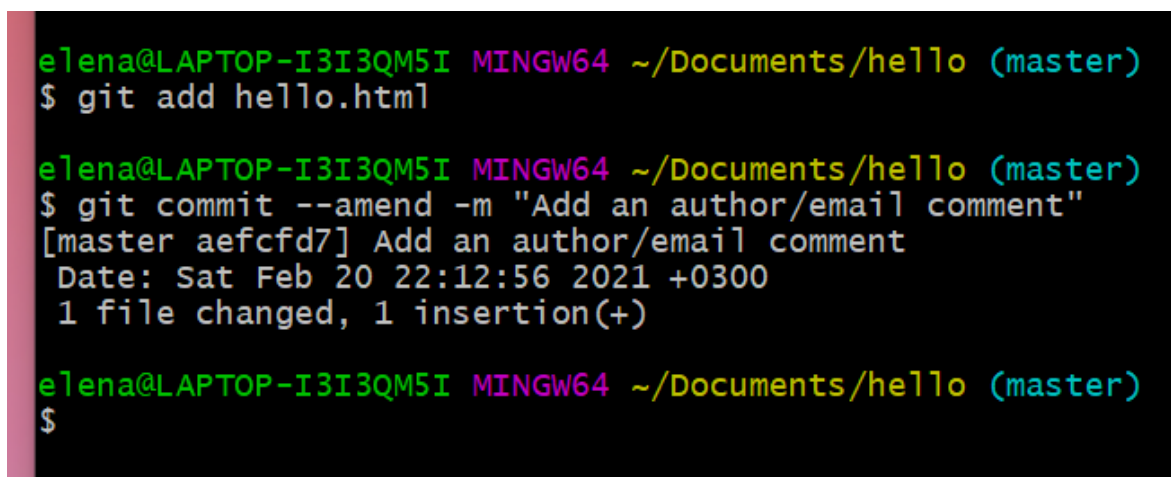
Далее необходимо добавить электронную почту в комментарий (рисунок 3).



```
<> hello.html X
C: > Users > elena > Documents > hello > <> hello.html > ...
1 | <!-- Author: Elena Verstova (isip_e.v.verstova@mpt.ru) -->
2 | <html>
3 |   <head>
4 |
5 |   </head>
6 |   <body>
7 |     <h1>Hello, world</h1>
8 |     <h2>The cake is a lie.</h2>
9 |   </body>
10| </html>
```

Рисунок 65 - Добавление электронной почты

Но для того, чтобы не создавать отдельный коммит ради электронной почты, можно изменить предыдущий так, как показано на рисунке 4.



```
e1ena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (master)
$ git add hello.html

e1ena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (master)
$ git commit --amend -m "Add an author/email comment"
[master aefcfd7] Add an author/email comment
Date: Sat Feb 20 22:12:56 2021 +0300
1 file changed, 1 insertion(+)

e1ena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (master)
$
```

Рисунок 66 - Индексация и изменение коммита

При просмотре истории можно будет заметить, что последний коммит был изменен.


```
eIena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (master)
$ git hist
* aefcfd7 2021-02-20 | Add an author/email comment (HEAD -> master) [ElenaVerstova]
* d243b6b 2021-02-17 | Added standard HTML tags (tag: ver1) [ElenaVerstova]
* 8de3e80 2021-02-17 | Added tag h2 (tag: ver1-beta) [ElenaVerstova]
* ce52614 2021-02-17 | Unrelated change for v [ElenaVerstova]
* e1f2276 2021-02-17 | Changes for a and b [ElenaVerstova]
* f302a0c 2021-02-17 | First Commit [ElenaVerstova]
```

Рисунок 67 - Последний коммит изменен

```
kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ git add hello.html

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ git commit -m "Add an author comment"
[master c13653f] Add an author comment
1 file changed, 1 insertion(+)

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ git add hello.html

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ git commit --amend -m "Add an author/email comment"
[master 0cd7990] Add an author/email comment
Date: Sat Jun 7 15:37:44 2025 +0300
1 file changed, 1 insertion(+)

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ git hist
* 0cd7990 2025-06-07 | Add an author/email comment (HEAD -> master) [Fedor Kozhurin]
* 895c6d9 2025-06-07 | Add HTML header (tag: ver1, tag: oooops) [Fedor Kozhurin]
* ecf6696 2025-06-07 | Unrelated change for v (tag: ver1-beta) [Fedor Kozhurin]
* 02333b6 2025-06-07 | Changes for a and b [Fedor Kozhurin]
* 9d515cd 2025-06-06 | First Commit [Fedor Kozhurin]

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$
```

Перемещение файлов

Для перемещения файлов в пределах репозитория используются команды, показанные на рисунке 6. После выполнения данных команды git индексирует эти изменения (удаление файла hello.html и создание файла lib/hello.html).

```
eIena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (master)
$ mkdir lib

eIena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (master)
$ git mv hello.html lib

eIena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (master)
$ git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        renamed:    hello.html -> lib/hello.html
```

Рисунок 68 - Перемещение файла

Далее надо осуществить коммит данного перемещения.

```
e1ena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (master)
$ git commit -m "Moved hello.html to lib"
[master b5bd827] Moved hello.html to lib
1 file changed, 0 insertions(+), 0 deletions(-)
rename hello.html => lib/hello.html (100%)

e1ena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (master)
$ |
```

Рисунок 69 – Коммит перемещения

```
kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ mkdir lib

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ git mv hello.html lib

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        renamed:    hello.html -> lib/hello.html

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ git commit -m "Moved hello.html to lib"
[master de3a855] Moved hello.html to lib
1 file changed, 0 insertions(+), 0 deletions(-)
rename hello.html => lib/hello.html (100%)

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ |
```

Подробнее о структуре

Необходимо добавить еще один файл в репозиторий. Это будет файл index.html с кодом, показанным на рисунке 8.

```
<> index.html ●
C: > Users > elena > Documents > hello > <> index.html > html
1  <html>
2    <body>
3      <iframe src="lib/hello.html" width="200" height="200"></iframe>
4    </body>
5  </html>
```

Рисунок 70 - Содержимое файла index.html

Далее нужно проиндексировать и закоммитить файл (рисунок 9).

```
e1ena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (master)
$ git add index.html

e1ena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (master)
$ git commit -m "Added index.html"
[master c2bcb76] Added index.html
1 file changed, 5 insertions(+)
create mode 100644 index.html

e1ena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (master)
$ |
```

Рисунок 71 - Индексация и коммит

При открытии файла index.html будет виден кусок страницы hello.html (рисунок 10).

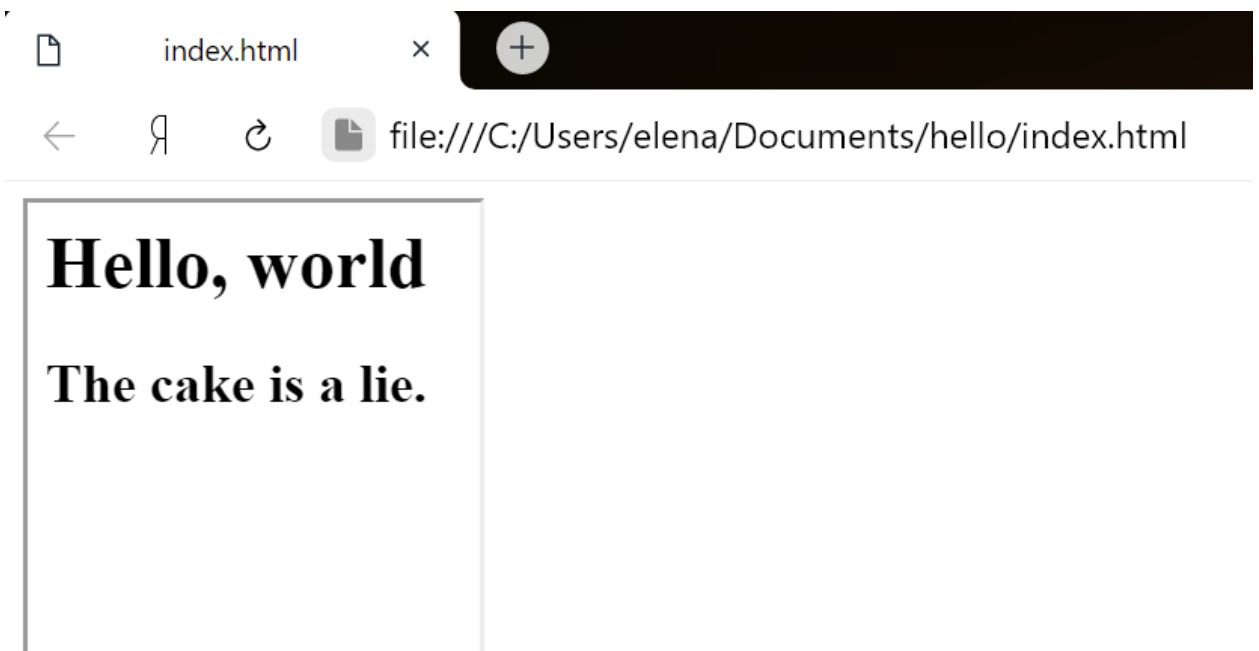
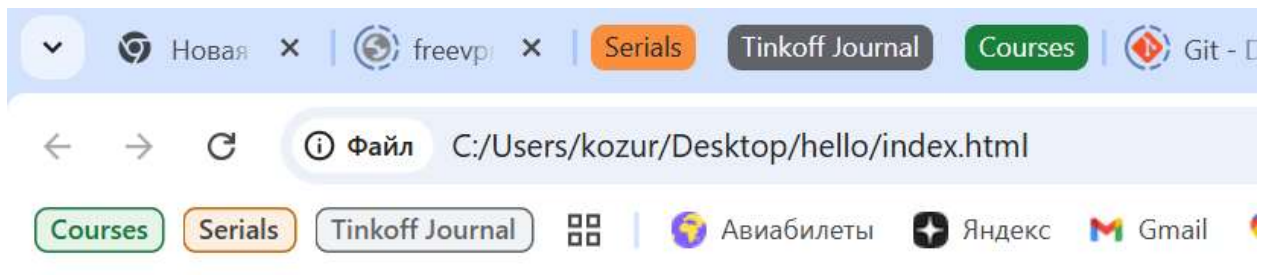


Рисунок 72 - Файл index.html, открытый в браузере



```
kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ git add index.html

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ git commit -m "Added index.html"
[master 741d84d] Added index.html
1 file changed, 5 insertions(+)
create mode 100644 index.html

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$
```

Каталог .git

Чтобы посмотреть структуру каталога .git необходимо выполнить команду, показанную на рисунке 11.

```
e1ena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (master)
$ ls -C .git
COMMIT_EDITMSG  ORIG_HEAD  description  index  logs/  refs/
HEAD           config     hooks/       info/  objects/

e1ena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (master)
$
```

Рисунок 73 - Содержание каталога .git

При аналогичном просмотре каталога objects можно будет увидеть множество каталогов с именами из 2 символов (рисунок 12). Имена каталогов являются первыми двумя буквами хэша.

```
e1ena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (master)
$ ls -C .git/objects
03/  16/  34/  49/  57/  6e/  88/  ae/  bc/  d2/  e1/  fd/
07/  1e/  37/  4e/  5c/  74/  8d/  af/  c2/  d8/  f3/  info/
12/  24/  44/  50/  67/  7f/  a0/  b5/  ce/  db/  f9/  pack/

e1ena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (master)
$
```

Рисунок 74 - Содержание каталога objects

При просмотре содержимого любого из каталогов будут показаны файлы, названия которых состоят из 38 символов (рисунок 13).

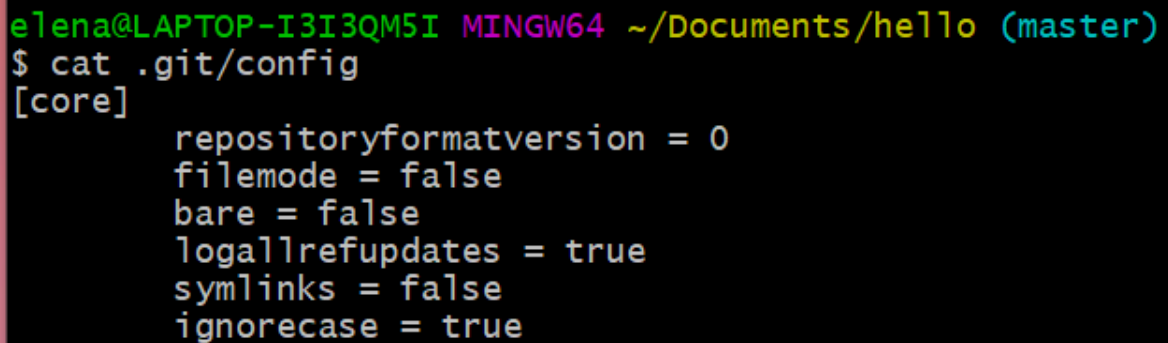
```
e1ena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (master)
$ ls -C .git/objects
03/  16/  34/  49/  57/  6e/  88/  ae/  bc/  d2/  e1/  fd/
07/  1e/  37/  4e/  5c/  74/  8d/  af/  c2/  d8/  f3/  info/
12/  24/  44/  50/  67/  7f/  a0/  b5/  ce/  db/  f9/  pack/

e1ena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (master)
$ ls -C .git/objects/8d
e3e8034c7ba4cc85a9ae26844a5c51be5f5e66

e1ena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (master)
$
```

Рисунок 75 - Просмотр каталога 8d

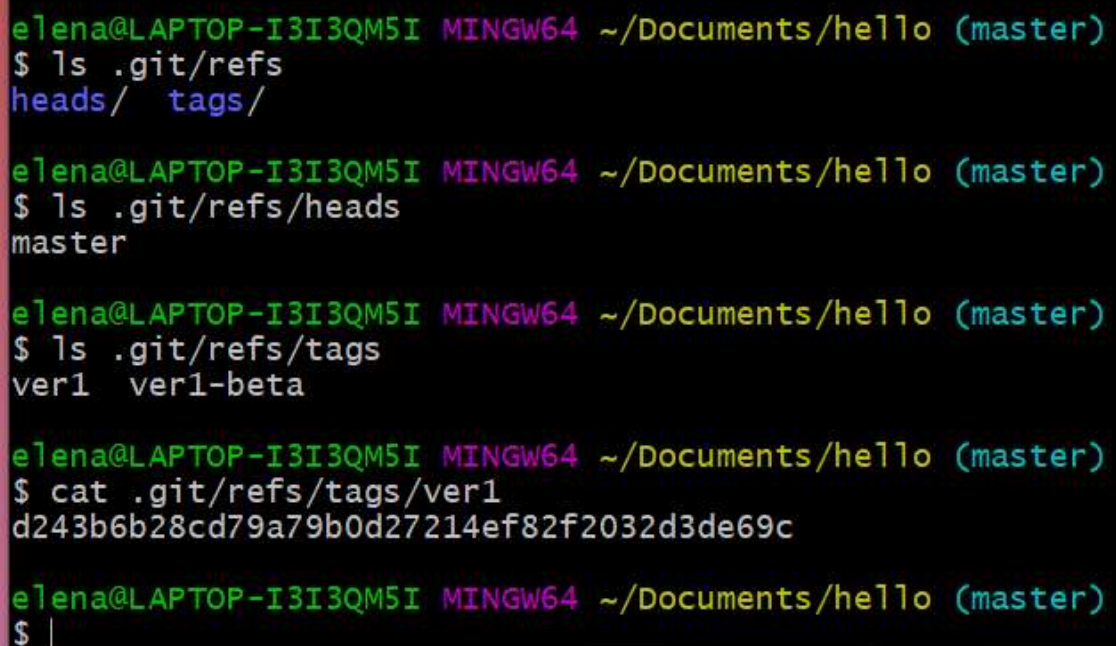
Далее требуется просмотреть файл конфигурации с помощью команды `cat` (рисунок 14).



```
e1ena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (master)
$ cat .git/config
[core]
    repositoryformatversion = 0
    filemode = false
    bare = false
    logallrefupdates = true
    symlinks = false
    ignorecase = true
```

Рисунок 76 - Просмотр файла конфигурации

На рисунке 14 показан просмотр файлов в подкаталоге `tags` и веток в каталоге `heads`.



```
e1ena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (master)
$ ls .git/refs
heads/  tags/

e1ena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (master)
$ ls .git/refs/heads
master

e1ena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (master)
$ ls .git/refs/tags
ver1  ver1-beta

e1ena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (master)
$ cat .git/refs/tags/ver1
d243b6b28cd79a79b0d27214ef82f2032d3de69c

e1ena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (master)
$ |
```

Рисунок 77 - Просмотр файлов и веток

Файл `HEAD` содержит ссылку на текущую ветку (рисунок 16).


```

elena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (master)
$ cat .git/HEAD
ref: refs/heads/master

elena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (master)
$ |

```

Рисунок 78 - Содержимое файла HEAD

```

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ ls -C .git
COMMIT_EDITMSG  ORIG_HEAD  description  index  logs/  refs/
HEAD           config     hooks/       info/  objects/

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ ls -C .git/objects
02/  05/  1e/  2d/  3e/  73/  7b/  89/  9d/  af/  c1/  d7/  db/  e6/  f3/  f7/  info/
03/  0c/  24/  39/  5c/  74/  87/  8f/  a3/  b6/  c8/  d8/  de/  ec/  f5/  fb/  pack/

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ ls -C .git/objects/8f
bb5aead8d8b0650186bf34555b5bec09dc6dc4

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ cat .git/config
[core]
    repositoryformatversion = 0
    filemode = false
    bare = false
    logallrefupdates = true
    symlinks = false
    ignorecase = true

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ ls .git/refs
heads/  tags/

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ ls .git/refs/heads
master

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ ls .git/refs/tags
oooops  ver1  ver1-beta

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ cat .git/refs/tags/ver1
895c6d957a5f17454336087ea5b6ad6080809ba1

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ cat .git/HEAD
ref: refs/heads/master

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$

```

Работа с объектами git

Для начала необходимо просмотреть последний коммит (рисунок 17).

```

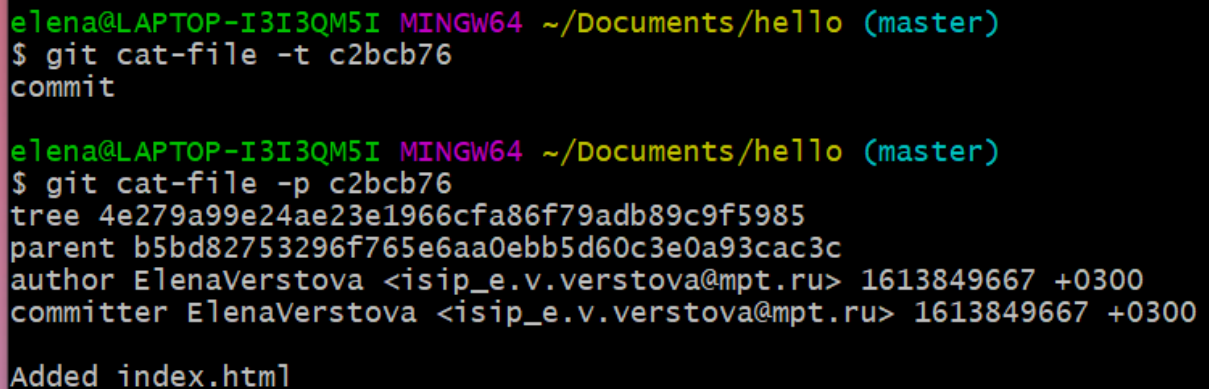
elena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (master)
$ git hist --max-count=1
* c2bcb76 2021-02-20 | Added index.html (HEAD -> master) [ElenaVerstova]

elena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (master)
$ |

```

Рисунок 79 - Последний коммит

Далее надо использовать хэш последнего коммита, используя команды `cat-file -p` и `cat-file -t` (рисунок 18) для просмотра объекта коммита. Также вместо длинных команд можно использовать сокращенные `type` и `dump`, если данные команды были заданы как алиасы.



```
e1ena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (master)
$ git cat-file -t c2bcb76
commit

e1ena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (master)
$ git cat-file -p c2bcb76
tree 4e279a99e24ae23e1966cfa86f79adb89c9f5985
parent b5bd82753296f765e6aa0ebb5d60c3e0a93cac3c
author ElenaVerstova <isip_e.v.verstova@mpt.ru> 1613849667 +0300
committer ElenaVerstova <isip_e.v.verstova@mpt.ru> 1613849667 +0300

Added index.html
```

Рисунок 80 - Просмотр объекта коммита



```
e1ena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (master)
$ git type c2bcb76
commit

e1ena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (master)
$ git dump c2bcb76
tree 4e279a99e24ae23e1966cfa86f79adb89c9f5985
parent b5bd82753296f765e6aa0ebb5d60c3e0a93cac3c
author ElenaVerstova <isip_e.v.verstova@mpt.ru> 1613849667 +0300
committer ElenaVerstova <isip_e.v.verstova@mpt.ru> 1613849667 +0300

Added index.html

e1ena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (master)
$ |
```

Рисунок 81 - Использование алиасов

Для просмотра дерева каталогов необходимо использовать его хэш (рисунок 20).

```
e1ena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (master)
$ git cat-file -p 4e279a9
100644 blob 1240583197c9a4507a2fb0d59eb1a82886844e57    a.html
100644 blob af60016690b87c1f1757db52ac1baf43ff3419a8    b.html
100644 blob 03ee26631ef36c173c9c1a9f7e51dd419f72482d    index.html
040000 tree 34c23e296c4e5b2207ddeb4a4aaf565abdf9e0d0    lib
100644 blob 1eded30425a18ead4e57682660b74c8645761b51    v.html

e1ena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (master)
$ |
```

Рисунок 82 - Просмотр дерева каталогов

Затем нужно просмотреть каталог lib (рисунок 21).

```
e1ena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (master)
$ git cat-file -p 34c23e296
100644 blob 07978d6ac75191ca3ce7aaca139e95e951fe230b    hello.html

e1ena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (master)
$
```

Рисунок 83 - Просмотр каталога lib

И затем требуется вывести содержимое файла hello.html (рисунок 22).

```
e1ena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (master)
$ git cat-file -p 07978d6a
<!-- Author: Elena Verstova (isip_e.v.verstova@mpt.ru) -->
<html>
  <head>

  </head>
  <body>
    <h1>Hello, world</h1>
    <h2>The cake is a lie.</h2>
  </body>
</html>
e1ena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (master)
$
```

Рисунок 84 - Вывод содержимого файла hello.html

Аналогичным образом можно просмотреть содержимое файла, каким оно было в самом первом коммите, как показано на рисунке 23. Для этого требуется использовать лишь нужные хэши.

```
elena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (master)
$ git hist
* c2bcb76 2021-02-20 | Added index.html (HEAD -> master, tag: ver2) [ElenaVerstova]
* b5bd827 2021-02-20 | Moved hello.html to lib [ElenaVerstova]
* aeefcd7 2021-02-20 | Add an author/email comment [ElenaVerstova]
* d243b6b 2021-02-17 | Added standard HTML tags (tag: ver1) [ElenaVerstova]
* 8de3e80 2021-02-17 | Added tag h2 (tag: ver1-beta) [ElenaVerstova]
* ce52614 2021-02-17 | Unrelated change for v [ElenaVerstova]
* e1f2276 2021-02-17 | Changes for a and b [ElenaVerstova]
* f302a0c 2021-02-17 | First Commit [ElenaVerstova]

elena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (master)
$ git cat-file -p f302a0c
tree d8a800db068656c3aafc962f7a438c987b49dd60
author ElenaVerstova <isip_e.v.verstova@mpt.ru> 1613562110 +0300
committer ElenaVerstova <isip_e.v.verstova@mpt.ru> 1613562110 +0300

First Commit

elena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (master)
$ git cat-file -p d8a800d
100644 blob dbe9dba55ea8fd4d5be3868b015e044be0848ec5    hello.html

elena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (master)
$ git cat-file -p dbe9dba
Hello, world
elena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (master)
$
```


Рисунок 85 - Просмотр содержимого файла при первом коммите

```
kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ git hist --max-count=1
* 741d84d 2025-06-07 | Added index.html (HEAD -> master) [Fedor Kozhurin]

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ git cat-file -t 741d84d
commit

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ git cat-file -p 741d84d
tree 8734f5df64b7977f65ab8e30f49d6858aefdbfd2
parent de3a855d29471dde54c1a12b9a729ed023f97e6d
author Fedor Kozhurin <kozhurinf@inbox.ru> 1749300305 +0300
committer Fedor Kozhurin <kozhurinf@inbox.ru> 1749300305 +0300

Added index.html

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ git type 741d84d
commit

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ git dump 741d84d
tree 8734f5df64b7977f65ab8e30f49d6858aefdbfd2
parent de3a855d29471dde54c1a12b9a729ed023f97e6d
author Fedor Kozhurin <kozhurinf@inbox.ru> 1749300305 +0300
committer Fedor Kozhurin <kozhurinf@inbox.ru> 1749300305 +0300

Added index.html

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ git cat-file -p 8734f5d
100644 blob e69de29bb2d1d6434b8b29ae775ad8c2e48c5391    a.html
100644 blob e69de29bb2d1d6434b8b29ae775ad8c2e48c5391    b.html
100644 blob 03ee26631ef36c173c9c1a9f7e51dd419f72482d    index.html
040000 tree fbb6df21aa704e87108e35533eccc00746c233d7    lib
100644 blob e69de29bb2d1d6434b8b29ae775ad8c2e48c5391    v.html

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ git cat-file -p fbb6df21a
100644 blob 7338f525763bb3b4857429c5c5bfb7babde415af    hello.html

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ git cat-file -p 7338f525
bash: git: command not found

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ git cat-file -p 7338f525
<!-- Author: Fedor Kozhurin (kozhurinf@inbox.ru)-->
<html>
  <head>

  </head>
  <body>
    <h1>Hello, world</h1>
    <h2>Second line</h2>
  </body>
</html>

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ git hist
* 741d84d 2025-06-07 | Added index.html (HEAD -> master) [Fedor Kozhurin]
* de3a855 2025-06-07 | Moved hello.html to lib [Fedor Kozhurin]
* 0cd7990 2025-06-07 | Add an author/email comment [Fedor Kozhurin]
* 895c6d9 2025-06-07 | Add HTML header (tag: ver1, tag: oooops) [Fedor Kozhurin]
```

```
kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ git cat-file -p 9d515cd
tree d8a800db068656c3aafc962f7a438c987b49dd60
author Fedor Kozhurin <kozhurinf@inbox.ru> 1749221469 +0300
committer Fedor Kozhurin <kozhurinf@inbox.ru> 1749221469 +0300

First Commit

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ git cat-file -p d8a800d
100644 blob dbe9dba55ea8fd4d5be3868b015e044be0848ec5    hello.html

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ git cat-file -p dbe9db
Hello, world
kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$
```

Работа с Git

Создание ветки

Для начала необходимо создать ветку style с помощью команды git checkout -b style (рисунок 1).

```
eIena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (master)
$ git checkout -b style
Switched to a new branch 'style'

eIena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (style)
$ git status
On branch style
nothing to commit, working tree clean

eIena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (style)
$
```

Рисунок 86 - Создание новой ветки style

Затем нужно создать файл стилей (рисунок 2) и внести в него код, показанный на рисунке 3.

```
eIena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (style)
$ touch lib/style.css

eIena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (style)
$
```

Рисунок 87 - Создание файла стилей

```
# style.css

C: > Users > elena > Documents > hello > lib > # style.css > h1
1  h1 {
2      color: red;
3  }
```

Рисунок 88 - Код style.css

После этого надо произвести индексацию и коммит (рисунок 4).

```
eIena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (style)
$ git add lib/style.css

eIena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (style)
$ git commit -m "Added css stylesheet"
[style ea17c64] Added css stylesheet
1 file changed, 3 insertions(+)
create mode 100644 lib/style.css

eIena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (style)
$ |
```

Рисунок 89 - Индексация и коммит нового файла

Далее требуется изменить основную страницу hello.html и закоммитить изменения (рисунки 5-6).

```
hello.html
C: > Users > elena > Documents > hello > lib > hello.html > html > head > link
1  <!-- Author: Elena Verstova (isip_e.v.verstova@mpt.ru) -->
2  <html>
3      <head>
4          <link rel="stylesheet" type="text/css" media="all" href="style.css">
5      </head>
6      <body>
7          <h1>Hello, world</h1>
8          <h2>The cake is a lie.</h2>
9      </body>
10 </html>
```

Рисунок 90 - Изменения в файле hello.html

```
eIena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (style)
$ git add lib/hello.html

eIena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (style)
$ git commit -m "Hello.html uses style.css"
[style 09ae13e] Hello.html uses style.css
1 file changed, 1 insertion(+), 1 deletion(-)

eIena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (style)
$
```

Рисунок 91 - Индексация и коммит

Далее аналогичные действия нужно осуществить с файлом index.html, как это показано на рисунках 7-8.

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

Рисунок 92 - Изменения в файле index.html

```
e1ena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (style)
$ git add index.html

e1ena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (style)
$ git commit -m "Update index.html"
[style 2a8fdf7] Update index.html
1 file changed, 3 insertions(+)

e1ena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (style)
$ |
```

Рисунок 93 - Индексация и коммит

После выполнения предыдущих действий была создана новая ветка style

с

3

коммитами.


```

Hello, world
kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ git checkout -b style
Switched to a new branch 'style'

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (style)
$ git status
On branch style
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   lib/hello.html

no changes added to commit (use "git add" and/or "git commit -a")

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (style)
$ touch lib/style.css

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (style)
$ git add lib/style.css

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (style)
$ git commit -m "Added css stylesheet"
[style 700b85f] Added css stylesheet
1 file changed, 3 insertions(+)
create mode 100644 lib/style.css

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (style)
$ git add lib/hello.html

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (style)
$ git commit -m "Hello.html uses style.css"
[style a61025a] Hello.html uses style.css
1 file changed, 2 insertions(+), 2 deletions(-)

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (style)
$ git add index.html

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (style)
$ git commit -m "Update index.html"
> "
[style cd78f6f] Update index.html
1 file changed, 3 insertions(+)

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (style)
$ |

```

git

Навигация по веткам

При просмотре истории, как на рисунке 9, можно увидеть, что теперь в проекте 2 ветки.

```

elena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (style)
$ git hist --all
* 2a8fdf7 2021-02-24 | Update index.html (HEAD -> style) [ElenaVerstova]
* 09ae13e 2021-02-24 | Hello.html uses style.css [ElenaVerstova]
* ea17c64 2021-02-24 | Added css stylesheet [ElenaVerstova]
* c2bcb76 2021-02-20 | Added index.html (tag: ver2, master) [ElenaVerstova]
* b5bd827 2021-02-20 | Moved hello.html to lib [ElenaVerstova]
* aefcfd7 2021-02-20 | Add an author/email comment [ElenaVerstova]
* d243b6b 2021-02-17 | Added standard HTML tags (tag: ver1) [ElenaVerstova]
* 8de3e80 2021-02-17 | Added tag h2 (tag: ver1-beta) [ElenaVerstova]
* ce52614 2021-02-17 | Unrelated change for v [ElenaVerstova]
* e1f2276 2021-02-17 | Changes for a and b [ElenaVerstova]
* f302a0c 2021-02-17 | First Commit [ElenaVerstova]

elena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (style)
$

```

Рисунок 94 - Просмотр истории

Для переключения на ветку master используется команда `git checkout master` (рисунок 10). После переключения на нужную ветку при выводе файла `hello.html` можно увидеть, что изменения отсутствуют (по причине того, что они закоммичены в другой ветке).

```

elena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (style)
$ git checkout master
Switched to branch 'master'

elena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (master)
$ cat lib/hello.html
<!-- Author: Elena Verstova (isip_e.v.verstova@mpt.ru) -->
<html>
  <head>

  </head>
  <body>
    <h1>Hello, world</h1>
    <h2>The cake is a lie.</h2>
  </body>
</html>
elena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (master)
$

```

Рисунок 95 - Переключение на ветку master

При переключении на ветку `style` файл `hello.html` будет иметь другое содержание (рисунок 11).

```

elena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (master)
$ git checkout style
Switched to branch 'style'

elena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (style)
$ cat lib/hello.html
<!-- Author: Elena Verstova (isip_e.v.verstova@mpm.ru) -->
<html>
  <head>
    <link rel="stylesheet" type="text/css" media="all" href="style.css">
  </head>
  <body>
    <h1>Hello, world</h1>
    <h2>The cake is a lie.</h2>
  </body>
</html>
elena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (style)
$ |

```

Рисунок 96 - Переключение на ветку style

```

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (style)
$ git hist --all
* cd78f6f 2025-06-07 | Update index.html (HEAD -> style) [Fedor Kozhurin]
* a61025a 2025-06-07 | Hello.html uses style.css [Fedor Kozhurin]
* 700b85f 2025-06-07 | Added css stylesheet [Fedor Kozhurin]
* 741d84d 2025-06-07 | Added index.html (master) [Fedor Kozhurin]
* de3a855 2025-06-07 | Moved hello.html to lib [Fedor Kozhurin]
* 0cd7990 2025-06-07 | Add an author/email comment [Fedor Kozhurin]
* 895c6d9 2025-06-07 | Add HTML header (tag: ver1, tag: oooops) [Fedor Kozhurin]
* ecf6696 2025-06-07 | Unrelated change for v (tag: ver1-beta) [Fedor Kozhurin]
* 02333b6 2025-06-07 | Changes for a and b [Fedor Kozhurin]
* 9d515cd 2025-06-06 | First Commit [Fedor Kozhurin]

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (style)
$ git checkout master
Switched to branch 'master'

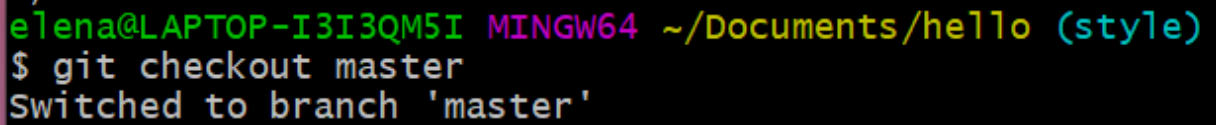
kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ cat lib/hello.html
<!-- Author: Fedor Kozhurin (kozhurinf@inbox.ru)-->
<html>
  <head>
  </head>
  <body>
    <h1>Hello, world</h1>
    <h2>Second line</h2>
  </body>
</html>
kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ git checkout style
Switched to branch 'style'

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (style)
$ cat lib/hello.html
<!-- Author: Fedor Kozhurin (kozhurinf@inbox.ru)-->
<html>
  <head>
    <link rel="stylesheet" type="text/css" media="all" href="style.css">
  </head>
  <body>
    <h1>Hello, world</h1>
    <h2>The cake is a lie.</h2>
  </body>
</html>

```

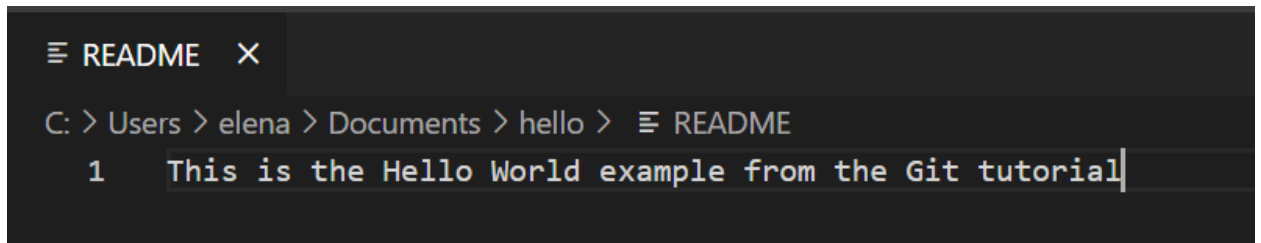
Изменения в ветке master

Необходимо переключиться на ветку master (рисунок 12) и добавить файл README (рисунки 13-14).



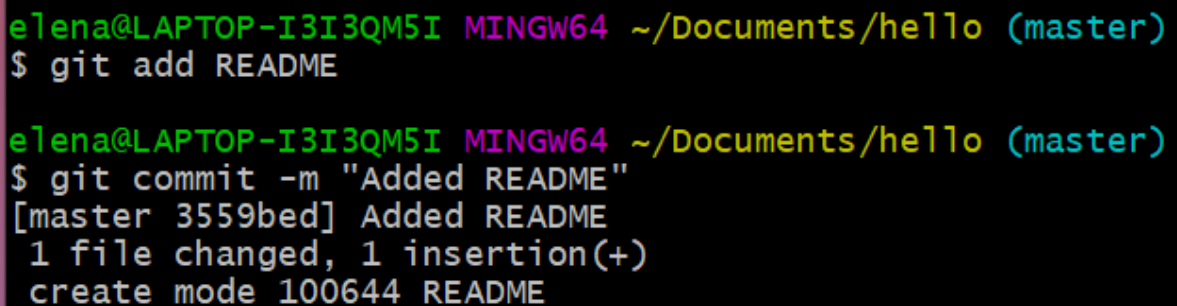
```
e1ena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (style)
$ git checkout master
Switched to branch 'master'
```

Рисунок 97 - Переключение на ветку master



```
≡ README ×
C: > Users > elena > Documents > hello > ≡ README
1 This is the Hello World example from the Git tutorial|
```

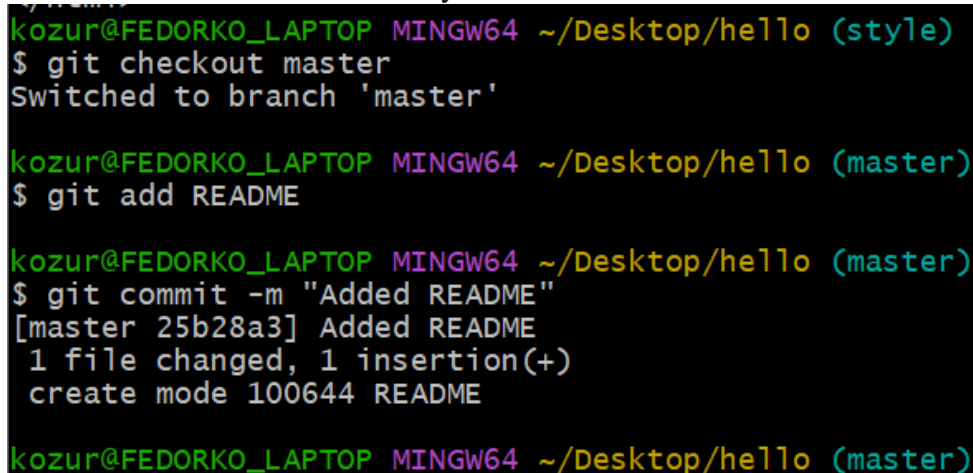
Рисунок 98 - Содержимое файла README



```
e1ena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (master)
$ git add README

e1ena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (master)
$ git commit -m "Added README"
[master 3559bed] Added README
1 file changed, 1 insertion(+)
create mode 100644 README
```

Рисунок 99 - Индексация и коммит



```
kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (style)
$ git checkout master
Switched to branch 'master'

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ git add README

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ git commit -m "Added README"
[master 25b28a3] Added README
1 file changed, 1 insertion(+)
create mode 100644 README

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
```

Просмотр отличающихся веток

На рисунке 15 можно увидеть дерево коммитов.

```
eIena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (master)
$ git hist --all
* 3559bed 2021-02-24 | Added README (HEAD -> master) [ElenaVerstova]
| * 2a8fdf7 2021-02-24 | Update index.html (style) [ElenaVerstova]
| * 09ae13e 2021-02-24 | Hello.html uses style.css [ElenaVerstova]
| * ea17c64 2021-02-24 | Added css stylesheet [ElenaVerstova]
|/
* c2bcb76 2021-02-20 | Added index.html (tag: ver2) [ElenaVerstova]
* b5bd827 2021-02-20 | Moved hello.html to lib [ElenaVerstova]
* aefcfd7 2021-02-20 | Add an author/email comment [ElenaVerstova]
* d243b6b 2021-02-17 | Added standard HTML tags (tag: ver1) [ElenaVerstova]
* 8de3e80 2021-02-17 | Added tag h2 (tag: ver1-beta) [ElenaVerstova]
* ce52614 2021-02-17 | Unrelated change for v [ElenaVerstova]
* e1f2276 2021-02-17 | Changes for a and b [ElenaVerstova]
* f302a0c 2021-02-17 | First Commit [ElenaVerstova]
```

Рисунок 100 - Дерево коммитов

Слияние

Слияние переносит изменения из двух веток в одну. Для слияния нужно перейти на ветку style и с помощью команды `git merge master` совместить ветки (рисунки 16-18).

```
eIena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (master)
$ git checkout style
Switched to branch 'style'

eIena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (style)
$ git merge master
```

Рисунок 101 - Переключение на ветку style

```
eIena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (style)
$ git merge master
hint: Waiting for your editor to close the file...
(electron) Sending uncompressed crash reports is deprecated and will be removed
in a future version of Electron. Set { compress: true } to opt-in to the new beh
avior. Crash reports will be uploaded gzipped, which most crash reporting server
s support.
[main 2021-02-24T11:53:30.275Z] update#setState idle
(node:45768) electron: The default of contextIsolation is deprecated and will be
changing from false to true in a future release of Electron. See https://githu
b.com/electron/electron/issues/23506 for more information
(node:45768) electron: The default of contextIsolation is deprecated and will be
changing from false to true in a future release of Electron. See https://githu
b.com/electron/electron/issues/23506 for more information
[main 2021-02-24T11:54:00.303Z] update#setState checking for updates
[main 2021-02-24T11:54:00.790Z] update#setState idle
Merge made by the 'recursive' strategy.
 README | 1 +
 1 file changed, 1 insertion(+)
 create mode 100644 README
```

Рисунок 102 - Слияние с веткой master


```
eIena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (style)
$ git hist --all
* 6efc8fb 2021-02-24 | Merge branch 'master' into style (HEAD -> style) [ElenaVerstova]
|
| * 3559bed 2021-02-24 | Added README (master) [ElenaVerstova]
| * 2a8fdf7 2021-02-24 | Update index.html [ElenaVerstova]
| * 09ae13e 2021-02-24 | Hello.html uses style.css [ElenaVerstova]
| * ea17c64 2021-02-24 | Added css stylesheet [ElenaVerstova]
|/
* c2bcb76 2021-02-20 | Added index.html (tag: ver2) [ElenaVerstova]
* b5bd827 2021-02-20 | Moved hello.html to lib [ElenaVerstova]
* aefcfd7 2021-02-20 | Add an author/email comment [ElenaVerstova]
* d243b6b 2021-02-17 | Added standard HTML tags (tag: ver1) [ElenaVerstova]
* 8de3e80 2021-02-17 | Added tag h2 (tag: ver1-beta) [ElenaVerstova]
* ce52614 2021-02-17 | Unrelated change for v [ElenaVerstova]
* e1f2276 2021-02-17 | Changes for a and b [ElenaVerstova]
* f302a0c 2021-02-17 | First Commit [ElenaVerstova]
```

Рисунок 103 - Просмотр истории

```
Merge made by the 'ort' strategy.
 README | 1 +
 1 file changed, 1 insertion(+)
 create mode 100644 README

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (style)
$ git hist --all
* e17527e 2025-06-07 | Merge branch 'master' into style (HEAD -> style) [Fedor Kozhurin]
|
| * 25b28a3 2025-06-07 | Added README (master) [Fedor Kozhurin]
| * cd78f6f 2025-06-07 | Update index.html [Fedor Kozhurin]
| * a61025a 2025-06-07 | Hello.html uses style.css [Fedor Kozhurin]
| * 700b85f 2025-06-07 | Added css stylesheet [Fedor Kozhurin]
|/
* 741d84d 2025-06-07 | Added index.html [Fedor Kozhurin]
* de3a855 2025-06-07 | Moved hello.html to lib [Fedor Kozhurin]
* 0cd7990 2025-06-07 | Add an author/email comment [Fedor Kozhurin]
* 895c6d9 2025-06-07 | Add HTML header (tag: ver1, tag: oooops) [Fedor Kozhurin]
* ecf6696 2025-06-07 | Unrelated change for v (tag: ver1-beta) [Fedor Kozhurin]
* 02333b6 2025-06-07 | Changes for a and b [Fedor Kozhurin]
* 9d515cd 2025-06-06 | First Commit [Fedor Kozhurin]
```

Создание конфликта

Для того, чтобы создать конфликт необходимо перейти в ветку master и внести изменения в файл hello.html (рисунки 19-21).

```
eIena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (style)
$ git checkout master
Switched to branch 'master'

eIena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (master)
$ |
```

Рисунок 104 - Переход в ветку master

```
<> hello.html ●
C: > Users > elena > Documents > hello > lib > <> hello.html > html > body > h1
1  <!-- Author: Elena Verstova (isip_e.v.verstova@mpt.ru) -->
2  <html>
3      <head>
4      |     <!-- no style -->
5      |
6      | </head>
7      | <body>
8      | |     <h1>Hello, world. Life is great</h1>
9      | |     <h2>The cake is a lie.</h2>
10     </body>
11 </html>
```

Рисунок 105 - Внесение изменений

```
elena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (master)
$ git add lib/hello.html

elena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (master)
$ git commit -m 'Life is great!'
[master 9dadf96] Life is great!
1 file changed, 2 insertions(+), 2 deletions(-)
```

Рисунок 106 - Индексация и коммит

После выполнения предыдущих действий при просмотре веток можно будет увидеть конфликт в виде, как на рисунке 22.

```

elena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (master)
$ git hist --all
* 9dadf96 2021-02-24 | Life is great! (HEAD -> master) [ElenaVerstova]
* 6efc8fb 2021-02-24 | Merge branch 'master' into style (style) [ElenaVerstova]
|
|/
* 3559bed 2021-02-24 | Added README [ElenaVerstova]
* 2a8fdf7 2021-02-24 | Update index.html [ElenaVerstova]
* 09ae13e 2021-02-24 | Hello.html uses style.css [ElenaVerstova]
* ea17c64 2021-02-24 | Added css stylesheet [ElenaVerstova]
|/
* c2bcb76 2021-02-20 | Added index.html (tag: ver2) [ElenaVerstova]
* b5bd827 2021-02-20 | Moved hello.html to lib [ElenaVerstova]
* aefcfd7 2021-02-20 | Add an author/email comment [ElenaVerstova]
* d243b6b 2021-02-17 | Added standard HTML tags (tag: ver1) [ElenaVerstova]
* 8de3e80 2021-02-17 | Added tag h2 (tag: ver1-beta) [ElenaVerstova]
* ce52614 2021-02-17 | Unrelated change for v [ElenaVerstova]
* e1f2276 2021-02-17 | Changes for a and b [ElenaVerstova]
* f302a0c 2021-02-17 | First Commit [ElenaVerstova]

```

Рисунок 107 - Конфликт изменений

```

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (style)
$ git checkout master
Switched to branch 'master'

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ git add lib/hello.html

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ git commit -m 'Life is great!'
[master 9c06b83] Life is great!
1 file changed, 3 insertions(+), 3 deletions(-)

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ git hist --all
* 9c06b83 2025-06-07 | Life is great! (HEAD -> master) [Fedor Kozhurin]
* e17527e 2025-06-07 | Merge branch 'master' into style (style) [Fedor Kozhurin]
|
|/
* 25b28a3 2025-06-07 | Added README [Fedor Kozhurin]
* cd78f6f 2025-06-07 | Update index.html [Fedor Kozhurin]
* a61025a 2025-06-07 | Hello.html uses style.css [Fedor Kozhurin]
* 700b85f 2025-06-07 | Added css stylesheet [Fedor Kozhurin]
|/
* 741d84d 2025-06-07 | Added index.html [Fedor Kozhurin]
* de3a855 2025-06-07 | Moved hello.html to lib [Fedor Kozhurin]
* 0cd7990 2025-06-07 | Add an author/email comment [Fedor Kozhurin]
* 895c6d9 2025-06-07 | Add HTML header (tag: ver1, tag: oooops) [Fedor Kozhurin]
* ecf6696 2025-06-07 | Unrelated change for v (tag: ver1-beta) [Fedor Kozhurin]
* 02333b6 2025-06-07 | Changes for a and b [Fedor Kozhurin]
* 9d515cd 2025-06-06 | First Commit [Fedor Kozhurin]

```

Разрешение конфликтов

При попытке объединить ветку style с master будет показана ошибка из-за конфликта, как показано на рисунке 23.

```

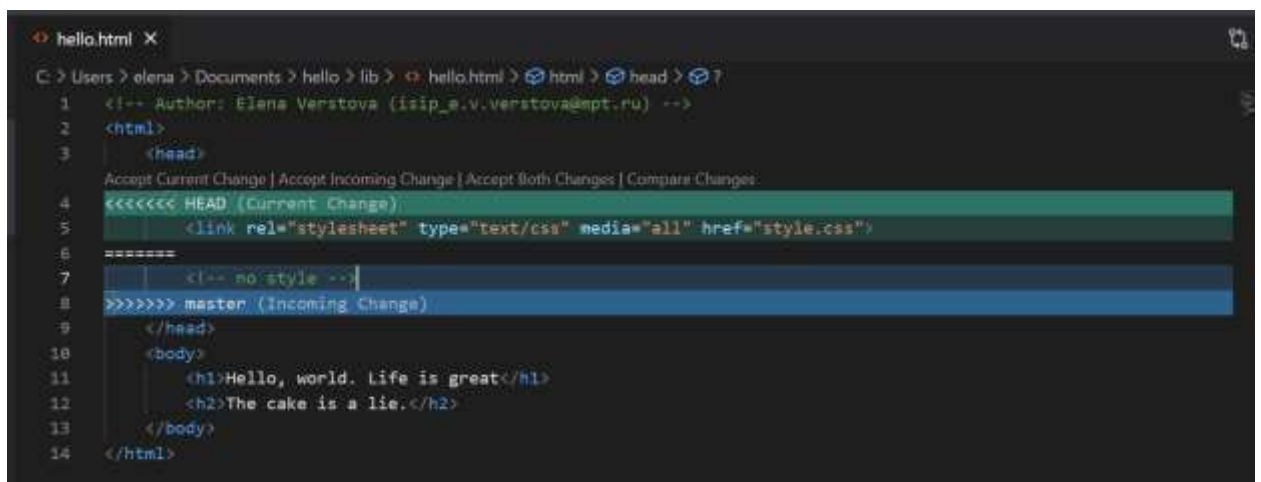
elena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (master)
$ git checkout style
Switched to branch 'style'

elena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (style)
$ git merge master
Auto-merging lib/hello.html
CONFLICT (content): Merge conflict in lib/hello.html
Automatic merge failed; fix conflicts and then commit the result.

```

Рисунок 108 - Ошибка при слиянии

При открытии файла hello.html конфликт будет показан (рисунок 24).



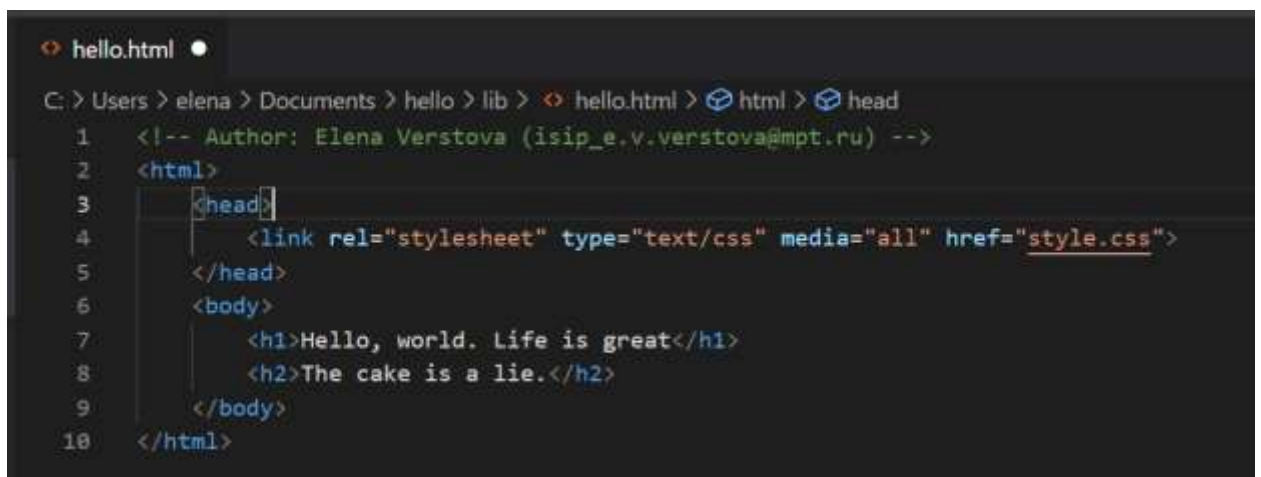
```

hello.html X
C: > Users > elena > Documents > hello > lib > hello.html > html > head > ?
1  <!-- Author: Elena Verstova (isip_e.v.verstova@mpt.ru) -->
2  <html>
3  <head>
4  <link rel="stylesheet" type="text/css" media="all" href="style.css">
5  </head>
6  <body>
7  <h1>Hello, world. Life is great</h1>
8  <h2>The cake is a lie.</h2>
9  </body>
10 </html>

```

Рисунок 109 - Просмотр файла hello.html при наличии конфликта

Чтобы решить конфликт, нужно внести изменения вручную (рисунок 25).



```

hello.html ●
C: > Users > elena > Documents > hello > lib > hello.html > html > head
1  <!-- Author: Elena Verstova (isip_e.v.verstova@mpt.ru) -->
2  <html>
3  <head>
4  <link rel="stylesheet" type="text/css" media="all" href="style.css">
5  </head>
6  <body>
7  <h1>Hello, world. Life is great</h1>
8  <h2>The cake is a lie.</h2>
9  </body>
10 </html>

```

Рисунок 110 - Решение конфликта вручную

Затем следует произвести индексацию и коммит (рисунок 26).

```
e1ena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (style|MERGING)
$ git add lib/hello.html

e1ena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (style|MERGING)
$ git commit -m "Merged master fixed conflict."
[style 945ce49] Merged master fixed conflict.
```

Рисунок 111 - Индексация и коммит

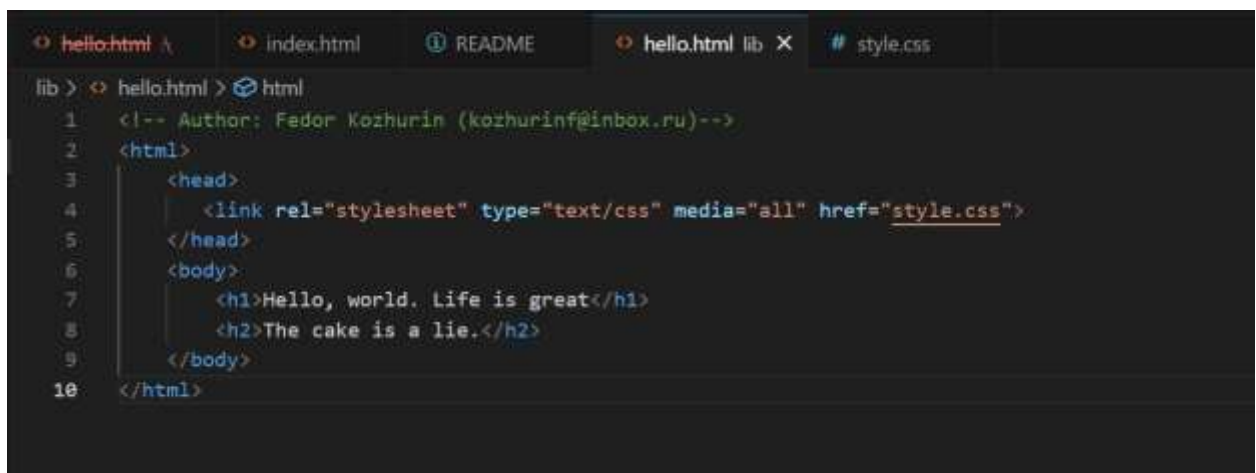
```
kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ git checkout style
Switched to branch 'style'

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (style)
$ git merge master
Auto-merging lib/hello.html
CONFLICT (content): Merge conflict in lib/hello.html
Automatic merge failed; fix conflicts and then commit the result.

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (style|MERGING)
$ git add lib/hello.html

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (style|MERGING)
$ git commit -m "Merged master fixed conflict."
[style 0d5677b] Merged master fixed conflict.

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (style)
$
```



```
lib > < hello.html > html
1  <!-- Author: Fedor Kozhurin (kozhurinf@inbox.ru)-->
2  <html>
3      <head>
4          <link rel="stylesheet" type="text/css" media="all" href="style.css">
5      </head>
6      <body>
7          <h1>Hello, world. Life is great</h1>
8          <h2>The cake is a lie.</h2>
9      </body>
10 </html>
```

Работа с Git

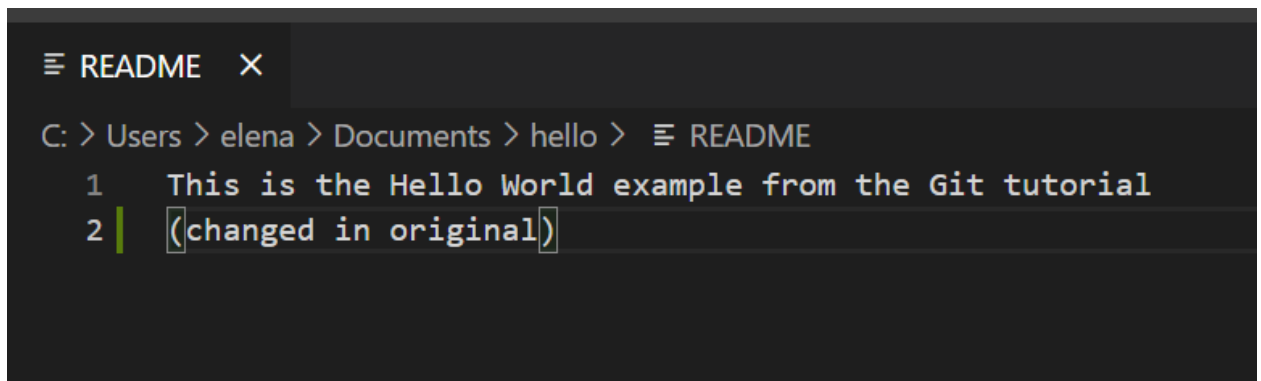
Сначала необходимо внести изменения в оригинальный репозиторий. Для этого нужно перейти в данный репозиторий (рисунок 1).


```
e1ena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/cloned_hello (master)
$ cd ../hello

e1ena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (master)
$
```

Рисунок 112 - Переход в оригинальный репозиторий

Далее надо внести изменения в файл README (рисунок 2) и затем произвести индексацию и коммит (рисунок 3).



The screenshot shows a text editor window with the title bar 'README' and a close button. The file path in the address bar is 'C: > Users > elena > Documents > hello > README'. The content of the file is as follows:

```
1 This is the Hello World example from the Git tutorial
2 | (changed in original)
```

Рисунок 113 - Изменения в файле README

```
e1ena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (master)
$ git add README

e1ena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (master)
$ git commit -m "Changed README in original repo"
[master 5906087] Changed README in original repo
1 file changed, 2 insertions(+), 1 deletion(-)

e1ena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (master)
$
```

Рисунок 114 - Индексация и коммит новых изменений

Далее требуется перейти в клонированный репозиторий и извлечь изменения с помощью команды `git fetch` (рисунок 4) и просмотреть историю (рисунок 5).

```
eIena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (master)
$ cd ../cloned_hello

eIena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/cloned_hello (master)
$ git fetch
remote: Enumerating objects: 5, done.
remote: Counting objects: 100% (5/5), done.
remote: Compressing objects: 100% (3/3), done.
remote: Total 3 (delta 1), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (3/3), 344 bytes | 11.00 KiB/s, done.
From C:/Users/elena/Documents/hello
    9dadf96..5906087  master    -> origin/master
```

Рисунок 115 - Извлечение изменений

```
eIena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/cloned_hello (master)
$ git hist --all
* 5906087 2021-03-10 | Changed README in original repo (origin/master, origin/HEAD) [ElenaVerstova]
* 9dadf96 2021-02-24 | Life is great! (HEAD -> master, origin/style) [ElenaVerstova]
* 3559bed 2021-02-24 | Added README [ElenaVerstova]
* c2bcb76 2021-02-20 | Added index.html (tag: ver2) [ElenaVerstova]
* b5bd827 2021-02-20 | Moved hello.html to lib [ElenaVerstova]
* aefcfd7 2021-02-20 | Add an author/email comment [ElenaVerstova]
* d243b6b 2021-02-17 | Added standard HTML tags (tag: ver1) [ElenaVerstova]
* 8de3e80 2021-02-17 | Added tag h2 (tag: ver1-beta) [ElenaVerstova]
* ce52614 2021-02-17 | Unrelated change for v [ElenaVerstova]
* e1f2276 2021-02-17 | Changes for a and b [ElenaVerstova]
* f302a0c 2021-02-17 | First Commit [ElenaVerstova]
```

Рисунок 116 - Просмотр истории

При попытке вывести содержимое файла README можно увидеть, что изменения не были внесены (рисунок 6).

```
eIena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/cloned_hello (master)
$ cat README
This is the Hello world example from the Git tutorial
eIena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/cloned_hello (master)
$
```

Рисунок 117 - Вывод содержимого файла README

Далее нужно слить извлеченные изменения в ветку master (рисунок 7).

```
eIena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/cloned_hello (master)
$ git merge origin/master
Updating 9dadf96..5906087
Fast-forward
 README | 3 ++-
 1 file changed, 2 insertions(+), 1 deletion(-)

eIena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/cloned_hello (master)
$ |
```

Рисунок 118 - Слияние изменений

И после выполнения предыдущего действия при выводе README можно будет увидеть последние изменения (рисунок 8).

```
e1ena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/cloned_hello (master)
$ cat README
This is the Hello world example from the Git tutorial
(changed in original)
e1ena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/cloned_hello (master)
$
```

Рисунок 119 - Вывод содержимого файла README

Также существует команда, объединяющая функции git fetch и git merge, которая показана на рисунке 9.

```
e1ena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/cloned_hello (master)
$ git pull
Already up to date.

e1ena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/cloned_hello (master)
$ |
```

Рисунок 120 - Команда git pull

Далее требуется добавить локальную ветку, которая будет отслеживать удаленную ветку (рисунок 10).

```
e1ena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/cloned_hello (master)
$ git branch --track style origin/style
Branch 'style' set up to track remote branch 'style' from 'origin'.

e1ena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/cloned_hello (master)
$ git branch -a
* master
  style
remotes/origin/HEAD -> origin/master
remotes/origin/master
remotes/origin/style

e1ena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/cloned_hello (master)
$ git hist --max-count=2
* 5906087 2021-03-10 | Changed README in original repo (HEAD -> master, origin/master, or
igin/HEAD) [ElenaVerstova]
* 9dadf96 2021-02-24 | Life is great! (origin/style, style) [ElenaVerstova]

e1ena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/cloned_hello (master)
$ |
```

Рисунок 121 - Добавление локальной ветки

Далее необходимо создать чистый репозиторий (рисунок 11).

```
e1ena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/cloned_hello (master)
$ cd ..

e1ena@LAPTOP-I3I3QM5I MINGW64 ~/Documents
$ git clone --bare hello hello.git
Cloning into bare repository 'hello.git'...
done.

e1ena@LAPTOP-I3I3QM5I MINGW64 ~/Documents
$ ls hello.git
HEAD config description hooks/ info/ objects/ packed-refs refs/

e1ena@LAPTOP-I3I3QM5I MINGW64 ~/Documents
$
```

Рисунок 122 - Создание чистого репозитория

Для добавления удаленного репозитория используется команда, показанная на рисунке 12.

```
e1ena@LAPTOP-I3I3QM5I MINGW64 ~/Documents
$ cd hello

e1ena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (master)
$ git remote add shared ../hello.git

e1ena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (master)
$ |
```

Рисунок 123 - Добавление удаленного репозитория

Затем требуется научиться отправлять изменения в удаленный репозиторий. Для этого сначала надо внести изменения, проиндексировать и произвести коммит (рисунок 13-14).

```
≡ README ×

C: > Users > elena > Documents > hello > ≡ README
1 This is the Hello World example from the Git tutorial
2 | (Changed in the original and pushed to shared)
```

Рисунок 124 - Внесение изменений в файл


```
eIena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (master)
$ git checkout master
Already on 'master'
M      README

eIena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (master)
$ git add README

eIena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (master)
$ git commit -m "Added shared comment to README"
[master c231127] Added shared comment to README
1 file changed, 1 insertion(+), 1 deletion(-)

eIena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (master)
$ |
```

Рисунок 125 - Индексация и коммит

Далее надо отправить изменения в общий репозиторий, используя команду `git push shared master` (рисунок 15).

```
eIena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (master)
$ git push shared master
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 4 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 376 bytes | 125.00 KiB/s, done.
Total 3 (delta 1), reused 0 (delta 0), pack-reused 0
To ../hello.git
    5906087..c231127  master -> master

eIena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (master)
$ |
```

Рисунок 126 - Отправка изменений в общий репозиторий

Для извлечения общих изменений нужно перейти в клонированный каталог и выполнить перечень команд (рисунок 16).

```
eIena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (master)
$ cd ../cloned_hello

eIena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/cloned_hello (master)
$
```

Рисунок 127 - Переход в клонированный репозиторий


```

elena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/cloned_hello (master)
$ git remote add shared ../hello.git

elena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/cloned_hello (master)
$ git branch --track shared master
Branch 'shared' set up to track local branch 'master'.

elena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/cloned_hello (master)
$ git pull shared master
remote: Enumerating objects: 5, done.
remote: Counting objects: 100% (5/5), done.
remote: Compressing objects: 100% (3/3), done.
remote: Total 3 (delta 1), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (3/3), 356 bytes | 11.00 KiB/s, done.
From ../hello
 * branch                master      -> FETCH_HEAD
 * [new branch]          master      -> shared/master
Updating 5906087..c231127
Fast-forward
 README | 2 +-
 1 file changed, 1 insertion(+), 1 deletion(-)

elena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/cloned_hello (master)
$ cat README
This is the Hello World example from the Git tutorial
(Changed in the original and pushed to shared)
elena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/cloned_hello (master)

```

Рисунок 128 - Команды, извлекающие общие изменения

Для настройки git сервера нужно выполнить команду, показанную на рисунке 18. Затем в другом окне можно проверить работу сервера, сделав копию проекта hello (рисунок 18).

```

elena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (master)
$ git daemon --verbose --export-all --base-path=.
[111720] Ready to rumble

```

Рисунок 129 - Настройка сервера

```
eIena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (master)
$ git clone ~/Documents/hello.git network_hello
Cloning into 'network_hello'...
done.

eIena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello (master)
$ cd network_hello

eIena@LAPTOP-I3I3QM5I MINGW64 ~/Documents/hello/network_hello (master)
$ ls
README  a.html  b.html  index.html  lib/  v.html
```

Рисунок 130 - Клонирование проекта

```

MINGW64:/c/Users/kozur/Desktop/cloned_hello

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/cloned_hello (master)
$ cd ../hello

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ git add README

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ git commit -m "Changed README in original repo"
[master 4c1f922] Changed README in original repo
1 file changed, 1 insertion(+)

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ cd ../cloned_hello

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/cloned_hello (master)
$ git fetch
remote: Enumerating objects: 5, done.
remote: Counting objects: 100% (5/5), done.
remote: Compressing objects: 100% (3/3), done.
remote: Total 3 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)
Unpacking objects: 100% (3/3), 347 bytes | 69.00 KiB/s, done.
From C:/Users/kozur/Desktop/hello
   9c06b83..4c1f922  master    -> origin/master

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/cloned_hello (master)
$ git hist --all
* 4c1f922 2025-06-07 | Changed README in original repo (origin/master, origin/HEAD) [Fedor Kozhurin]
* 9c06b83 2025-06-07 | Life is great! (HEAD -> master, origin/style) [Fedor Kozhurin]
* 25b28a3 2025-06-07 | Added README [Fedor Kozhurin]
* 741d84d 2025-06-07 | Added index.html [Fedor Kozhurin]
* de3a855 2025-06-07 | Moved hello.html to lib [Fedor Kozhurin]
* 0cd7990 2025-06-07 | Add an author/email comment [Fedor Kozhurin]
* 895c6d9 2025-06-07 | Add HTML header (tag: ver1, tag: oooops) [Fedor Kozhurin]
* ecf6696 2025-06-07 | Unrelated change for v (tag: ver1-beta) [Fedor Kozhurin]
* 02333b6 2025-06-07 | Changes for a and b [Fedor Kozhurin]
* 9d515cd 2025-06-06 | First Commit [Fedor Kozhurin]

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/cloned_hello (master)
$ cat README
This is the Hello World example from the Git tutorial

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/cloned_hello (master)
$ git merge origin/master
Updating 9c06b83..4c1f922
Fast-forward
 README | 1 +
1 file changed, 1 insertion(+)

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/cloned_hello (master)
$ cat README
This is the Hello World example from the Git tutorial
(changed in original)

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/cloned_hello (master)
$ git pull
Already up to date.

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/cloned_hello (master)
$ git branch --track style origin/style
branch 'style' set up to track 'origin/style'.

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/cloned_hello (master)
$ git branch -a
* master

```

```
MINGW64/c/Users/kozur/Desktop/cloned_hello
style
remotes/origin/HEAD -> origin/master
remotes/origin/master
remotes/origin/style

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/cloned_hello (master)
$ git hist --max-count=2
* 4c1f922 2025-06-07 | Changed README in original repo (HEAD -> master, origin/master, ori
EAD) [Fedor Kozhurin]
* 9c06b83 2025-06-07 | Life is great! (origin/style, style) [Fedor Kozhurin]

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/cloned_hello (master)
$ cd ..

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop
$ git clone --bare hello hello.git
Cloning into bare repository 'hello.git'...
done.

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop
$ ls hello.git
HEAD config description hooks/ info/ objects/ packed-refs refs/

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop
$ cd hello

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ git remote add shared ../hello.git

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ git checkout master
M README
Already on 'master'

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ git add README

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ git commit -m "Added shared comment to README"
[master 2313a0f] Added shared comment to README
1 file changed, 1 insertion(+), 1 deletion(-)

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ git push shared master
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 16 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 341 bytes | 341.00 KiB/s, done.
Total 3 (delta 2), reused 0 (delta 0), pack-reused 0 (from 0)
To ../hello.git
4c1f922..2313a0f master -> master

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ cd ../cloned_hello

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/cloned_hello (master)
$ git remote add shared ../hello.git

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/cloned_hello (master)
$ git branch --track shared master
branch 'shared' set up to track 'master'.

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/cloned_hello (master)
$ git pull shared master
```



```
kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/cloned_hello (master)
$ git pull shared master
remote: Enumerating objects: 5, done.
remote: Counting objects: 100% (5/5), done.
remote: Compressing objects: 100% (3/3), done.
remote: Total 3 (delta 2), reused 0 (delta 0), pack-reused 0 (from 0)
Unpacking objects: 100% (3/3), 321 bytes | 53.00 KiB/s, done.
From ../hello
 * branch                master      -> FETCH_HEAD
 * [new branch]          master      -> shared/master
Updating 4c1f922..2313a0f
Fast-forward
 README | 2 +-
 1 file changed, 1 insertion(+), 1 deletion(-)

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/cloned_hello (master)
$ car README
bash: car: command not found

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/cloned_hello (master)
$ cat README
This is the Hello World example from the Git tutorial
(changed in original and pushed to shared)
kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/cloned_hello (master)
$
```

```
kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ git clone ~/Desktop/hello.git network_hello
Cloning into 'network_hello'...
done.

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello (master)
$ cd network_hello

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello/network_hello (master)
$ ls
README  a.html  b.html  index.html  lib/  v.html

kozur@FEDORKO_LAPTOP MINGW64 ~/Desktop/hello/network_hello (master)
$
```