

Оглавление

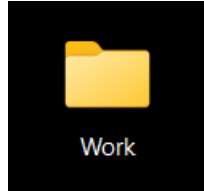
Оглавление	2
1 Создание проекта.....	4
1.1 Копирование папки Work на Рабочий стол.....	4
1.2 Создание репозитория	4
1.3 Установка имени и электронной почты	4
1.4 Добавление страницы в репозиторий	4
2 Проверка состояния.....	5
2.1 Проверка состояния репозитория	5
3 Внесение изменений	5
3.1 Изменение страницы.....	5
3.2 Проверка состояния	5
4 Индексация изменений	6
4.1 Добавление изменений	6
5 Индексация и коммит	6
6 Коммит изменений	6
6.1 Коммит ранее проиндексированного	6
6.2 Проверка состояния	7
7 Работа git с изменениями, а не файлами	7
7.1 Первое изменение: добавление стандартных тегов страницы	7
7.2 Добавление изменения.....	8
7.3 Второе изменение: добавление заголовков HTML	8
7.4 Проверка текущего статуса	9
7.5 Коммит.....	9
7.6 Добавление второго изменения	9
7.7 Коммит второго изменения	10
8 История.....	10
8.1 Однострочная история.....	11
8.3 Конечный формат лога.....	13
9 Алиасы.....	14
10 Получение старых версий.....	14
10.1 Получение хэша предыдущих версий	14
10.2 Возвращение к последней версии в ветке master.....	15
11 Создание тегов версий	16

11.1 Создание тега первой версии	16
11.2 Теги для предыдущих версий.....	16
11.3 Переключение по имени тега	17
11.4 Просмотр тегов с помощью команды tag.....	17
11.5 Просмотр Тегов в логах	18
12. Отмена локальных изменений (до индексации).....	18
12.1 Переключение на ветку Master	18
12.2 Изменение hello.html.....	18
12.3 Проверка состояния	19
12.4 Отмена изменений в рабочем каталоге	19
13. Отмена проиндексированных изменений (перед коммитом).....	20
13.1 Изменение файла и индексация изменения.....	20
13.2 Проверка состояния	21
13.3 Выполнение сброса буферной зоны.....	21
13.4 Переключение на версию коммита.....	22
14 Отмена коммитов.....	22
14.1 Отмена коммитов.....	22
14.2 Изменение файла и коммит	22
14.3 Коммит с новыми изменениями, отменяющими предыдущие.....	23
14.4 Проверка лога	23
15. Удаление коммитов из ветки.....	24
15.1 Команда reset.....	24
15.2. Проверка истории.....	25
15.3 Отметка ветки	25
15.4 Сброс коммитов к предшествующим коммиту Oops.....	25
15.5 Ничего никогда не теряется.....	26
16. Удаление тега badcom	26
16.1 Удаление тега badcom	26
17 Внесение изменений в коммиты	27
17.1 Изменение страницы с последующим коммитом	27
17.2 Обновление страницы hello.....	27
17.4 Просмотр истории.....	28

1 Создание проекта

1.1 Копирование папки Work на Рабочий стол

Создадим папку Work на Рабочем столе.



1.2 Создание репозитория

Создадим репозиторий.

```
MINGW64:/c/Users/33319/Desktop/Work

33319@Komp MINGW64 ~/Desktop/work (main)
$ git init
Initialized empty Git repository in C:/Users/33319/Desktop/work/.git/

33319@Komp MINGW64 ~/Desktop/work (master)
$ |
```

1.3 Установка имени и электронной почты

Установим имя и электронную почту.

```
MINGW64:/c/Users/33319/Desktop/Work

33319@Komp MINGW64 ~/Desktop/work (master)
$ git config --local user.name "Pavel5198"
git config --local user.email "333196@mail.ru"

33319@Komp MINGW64 ~/Desktop/work (master)
$ |
```

1.4 Добавление страницы в репозиторий

Добавим в репозиторий страницу «Hello, World».

```
MINGW64:/c/Users/33319/Desktop/Work
33319@Komp MINGW64 ~/Desktop/Work (master)
$ git add hello.html
git commit -m "First Commit"
[master (root-commit) fcfacf9] First Commit
1 file changed, 36 insertions(+)
create mode 100644 hello.html

33319@Komp MINGW64 ~/Desktop/Work (master)
$ |
```

2 Проверка состояния

2.1 Проверка состояния репозитория

Проверим текущее состояние репозитория.

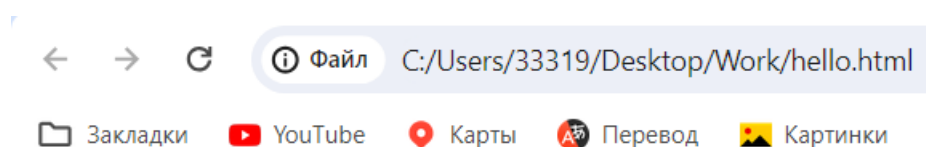
```
MINGW64:/c/Users/33319/Desktop/Work
33319@Komp MINGW64 ~/Desktop/Work (master)
$ git status
On branch master
nothing to commit, working tree clean

33319@Komp MINGW64 ~/Desktop/Work (master)
$ |
```

3 Внесение изменений

3.1 Изменение страницы

Изменим содержимое файла hello.html на `<h1>Hello, World!</h1>`.



Hello, World!

3.2 Проверка состояния

Проверим состояние рабочего каталога согласно рисунка 3.2.

```
MINGW64:/c/Users/33319/Desktop/Work
33319@Komp MINGW64 ~/Desktop/Work (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")
33319@Komp MINGW64 ~/Desktop/Work (master)
$
```

4 Индексация изменений

4.1 Добавление изменений

Дадим команду git проиндексировать изменения и проверим состояние.

```
MINGW64:/c/Users/33319/Desktop/Work
33319@Komp MINGW64 ~/Desktop/Work (master)
$ git add hello.html
git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   hello.html

33319@Komp MINGW64 ~/Desktop/Work (master)
$
```

5 Индексация и коммит

Отдельный шаг индексации в git позволяет продолжать вносить изменения в рабочий каталог, а затем, в момент, когда понадобится взаимодействовать с версионным контролем, git позволит записать изменения в малых коммитах, которые фиксируют то, что было сделано.

6 Коммит изменений

6.1 Коммит ранее проиндексированного

Сделаем коммит ранее проиндексированного в репозиторий. Выполним команду git commit для перехода в редактор.

```
MINGW64:/c/Users/33319/Desktop/Work

# Please enter the commit message for your changes. Lines starting
# with '#' will be ignored, and an empty message aborts the commit.
#
# On branch master
# Changes to be committed:
#   modified:   hello.html
#
~
~
.git/COMMIT_EDITMSG [unix] (08:58 06/02/2024) 1,0-1 All
```

В первой строке введем комментарий: «Added h1 tag», сохраним файл и выйдем из редактора.

```
MINGW64:/c/Users/33319/Desktop/Work

(pid 17400)
[master 2f517b2] Added h1 tag
1 file changed, 1 insertion(+), 1 deletion(-)
```

6.2 Проверка состояния

Проверим чистоту рабочего каталога, как показано на рисунке 6.3.

```
MINGW64:/c/Users/33319/Desktop/Work

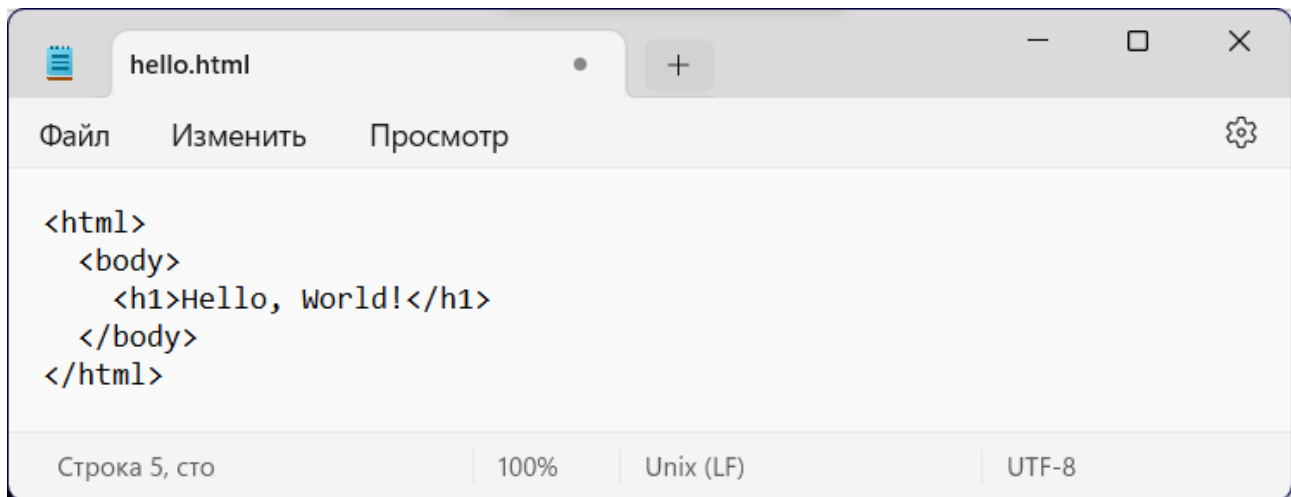
33319@Komp MINGW64 ~/Desktop/Work (master)
$ git commit
On branch master
nothing to commit, working tree clean

33319@Komp MINGW64 ~/Desktop/Work (master)
$
```

7 Работа git с изменениями, а не файлами

7.1 Первое изменение: добавление стандартных тегов страницы

Внесем изменения в файл hello.html.

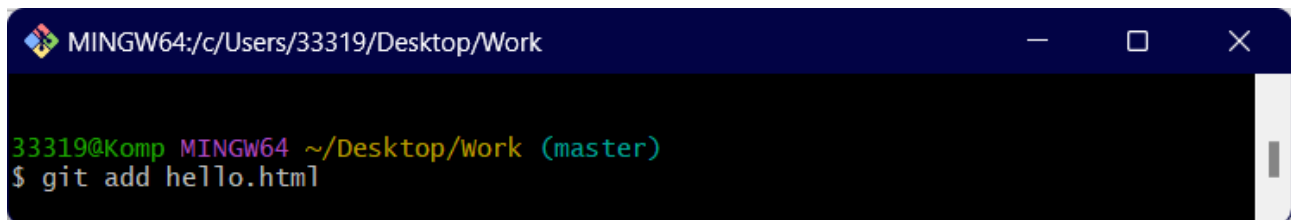


The screenshot shows a code editor window with a single tab titled 'hello.html'. The menu bar includes 'Файл', 'Изменить', 'Просмотр', and a settings icon. The code content is as follows:

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

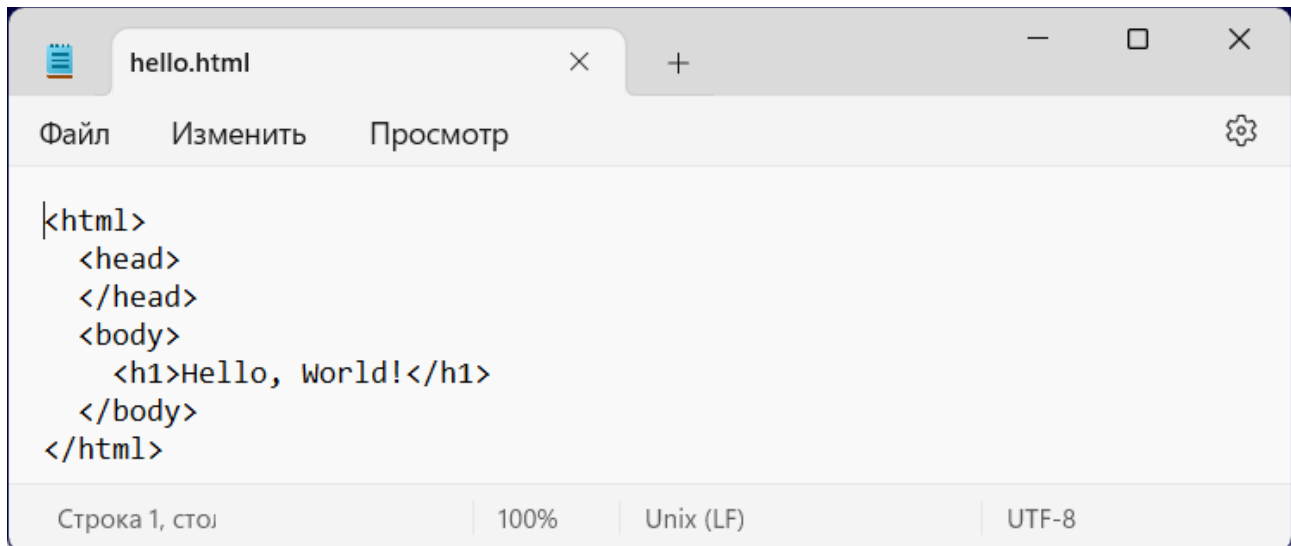
The status bar at the bottom indicates 'Строка 5, столбец 1', '100%', 'Unix (LF)', and 'UTF-8'.

7.2 Добавление изменения



The screenshot shows a terminal window with the title bar 'MINGW64:/c/Users/33319/Desktop/Work'. The prompt is '33319@Komp MINGW64 ~/Desktop/work (master)'. The command entered is '\$ git add hello.html'.

7.3 Второе изменение: добавление заголовков HTML

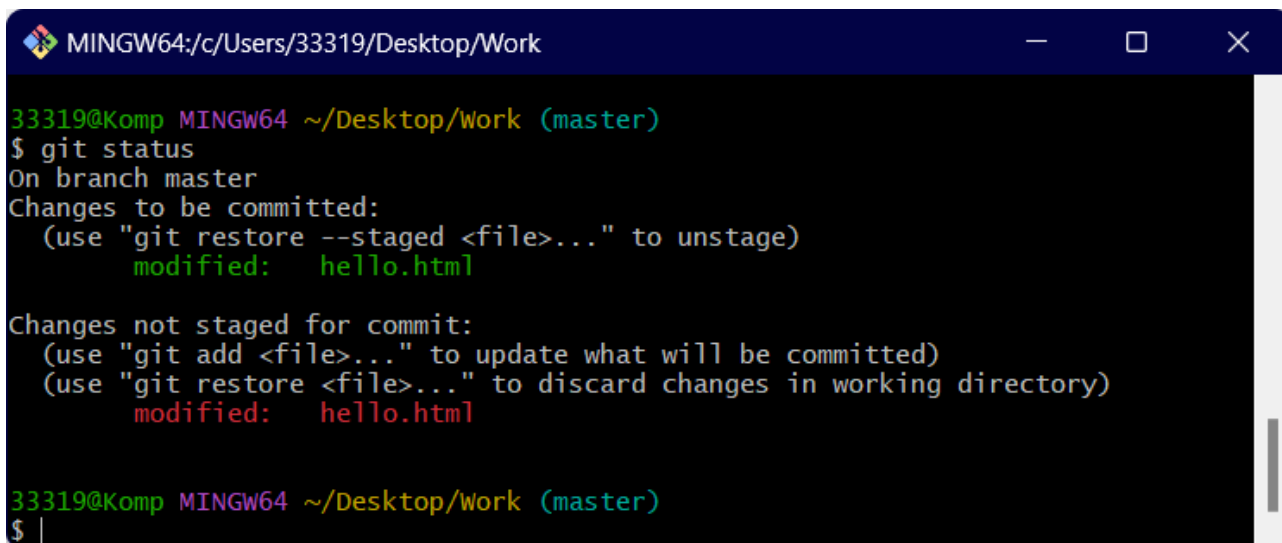


The screenshot shows the same code editor window with 'hello.html'. The code content is now updated to include a head section:

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

The status bar at the bottom indicates 'Строка 1, столбец 1', '100%', 'Unix (LF)', and 'UTF-8'.

7.4 Проверка текущего статуса

A terminal window titled 'MINGW64:/c/Users/33319/Desktop/Work' showing the output of the 'git status' command. The output indicates that the user is on the 'master' branch and that there are changes to be committed (hello.html modified) and changes not staged for commit (also hello.html modified).

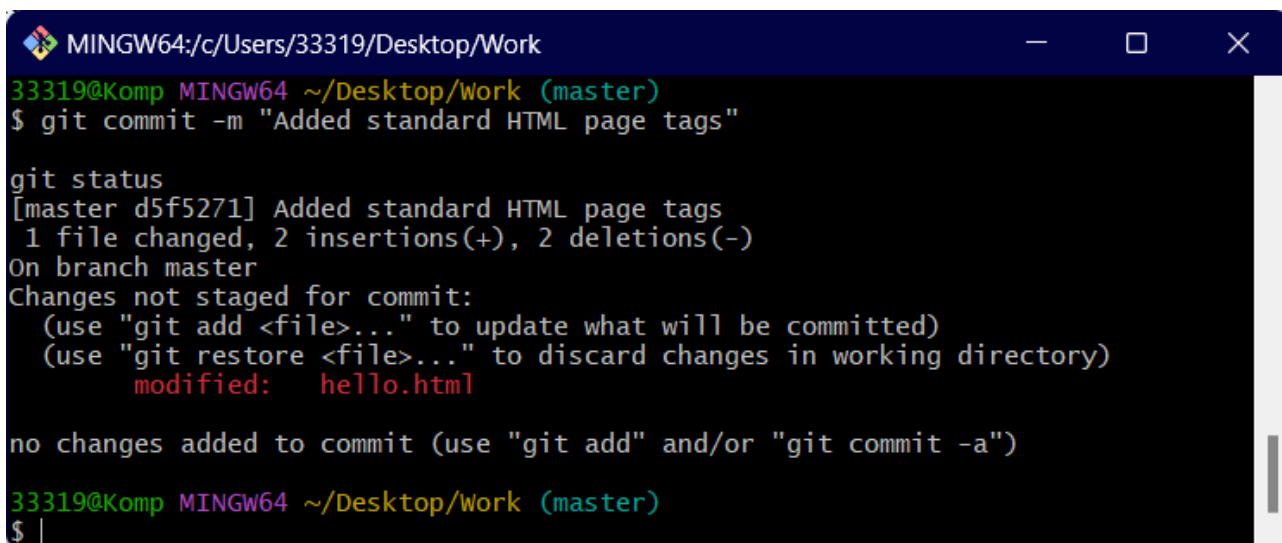
```
MINGW64:/c/Users/33319/Desktop/Work
33319@Komp MINGW64 ~/Desktop/Work (master)
$ git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   hello.html

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

33319@Komp MINGW64 ~/Desktop/Work (master)
$ |
```

7.5 Коммит

Произведем коммит проиндексированного изменения (значение по умолчанию), а затем еще раз проверим состояние.

A terminal window titled 'MINGW64:/c/Users/33319/Desktop/Work' showing the output of 'git commit -m "Added standard HTML page tags"' followed by 'git status'. The commit was successful, creating a new commit 'd5f5271'. The status shows that the changes are now committed and no changes are staged for commit.

```
MINGW64:/c/Users/33319/Desktop/Work
33319@Komp MINGW64 ~/Desktop/Work (master)
$ git commit -m "Added standard HTML page tags"

git status
[master d5f5271] Added standard HTML page tags
 1 file changed, 2 insertions(+), 2 deletions(-)
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")

33319@Komp MINGW64 ~/Desktop/Work (master)
$ |
```

Состояние команды говорит о том, что hello.html имеет незафиксированные изменения, но уже не в буферной зоне.

7.6 Добавление второго изменения

Добавим второе изменение в индекс, а затем проверим состояние с помощью команды git status.


```
MINGW64:/c/Users/33319/Desktop/Work
33319@Komp MINGW64 ~/Desktop/Work (master)
$ git add .
git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   hello.html

33319@Komp MINGW64 ~/Desktop/Work (master)
$ |
```

7.7 Коммит второго изменения

```
MINGW64:/c/Users/33319/Desktop/Work
33319@Komp MINGW64 ~/Desktop/Work (master)
$ git commit -m "Added HTML header"
[master 43912b9] Added HTML header
1 file changed, 2 insertions(+)

33319@Komp MINGW64 ~/Desktop/Work (master)
$ |
```

8 История

Получим список произведенных изменений используя функцию команды `git log`.

```
MINGW64:/c/Users/33319/Desktop/Work
commit 43912b94303a883ebc5f97854c9ea8c1093f4c2e (HEAD -> master)
Author: Pavel5198 <333196@mail.ru>
Date: Tue Feb 6 14:13:43 2024 +0300

    Added HTML header

commit d5f5271cf02c3f77d53b633db36d7f6656477bb5
Author: Pavel5198 <333196@mail.ru>
Date: Tue Feb 6 14:09:40 2024 +0300

    Added standard HTML page tags

commit 041c3ba7c0f1e062d0b37cb3aa30112e04191b64
Author: Pavel5198 <333196@mail.ru>
Date: Tue Feb 6 13:26:19 2024 +0300

    Added standard HTML page tags

commit 2f517b233acc80dcb1fc8c17e6bd775a1cf1baef
Author: Pavel5198 <333196@mail.ru>
Date: Tue Feb 6 08:58:11 2024 +0300

    Added h1 tag

commit 0c6a8f750d8678be13ce01ea59b05a1acdd3d0c1
Author: Pavel5198 <333196@mail.ru>
Date: Tue Feb 6 05:05:54 2024 +0300

    Changes for a and b

commit fcfacf99f2ee9f98473160c363b83fb015ceb248
Author: Pavel5198 <333196@mail.ru>
Date: Tue Feb 6 04:21:46 2024 +0300

    First Commit

(END)
```

8.1 Однострочная история

Используем однострочный формат.

```
MINGW64:/c/Users/33319/Desktop/Work
33319@Komp MINGW64 ~/Desktop/Work (master)
$ git log --pretty=oneline
43912b94303a883ebc5f97854c9ea8c1093f4c2e (HEAD -> master) Added HTML header
d5f5271cf02c3f77d53b633db36d7f6656477bb5 Added standard HTML page tags
041c3ba7c0f1e062d0b37cb3aa30112e04191b64 Added standard HTML page tags
2f517b233acc80dcb1fc8c17e6bd775a1cf1baef Added h1 tag
0c6a8f750d8678be13ce01ea59b05a1acdd3d0c1 Changes for a and b
fcfacf99f2ee9f98473160c363b83fb015ceb248 First Commit

33319@Komp MINGW64 ~/Desktop/Work (master)
$
```

```
MINGW64:/c/Users/33319/Desktop/Work

33319@Komp MINGW64 ~/Desktop/Work (master)
$ git log --pretty=oneline --max-count=2
43912b94303a883ebc5f97854c9ea8c1093f4c2e (HEAD -> master) Added HTML header
d5f5271cf02c3f77d53b633db36d7f6656477bb5 Added standard HTML page tags

33319@Komp MINGW64 ~/Desktop/Work (master)
$ |
```

```
MINGW64:/c/Users/33319/Desktop/Work

33319@Komp MINGW64 ~/Desktop/Work (master)
$ git log --pretty=oneline --since='5 minutes ago'

33319@Komp MINGW64 ~/Desktop/Work (master)
$
```

```
MINGW64:/c/Users/33319/Desktop/Work

33319@Komp MINGW64 ~/Desktop/Work (master)
$ git log --pretty=oneline --until='5 minutes ago'
43912b94303a883ebc5f97854c9ea8c1093f4c2e (HEAD -> master) Added HTML header
d5f5271cf02c3f77d53b633db36d7f6656477bb5 Added standard HTML page tags
041c3ba7c0f1e062d0b37cb3aa30112e04191b64 Added standard HTML page tags
2f517b233acc80dcb1fc8c17e6bd775a1cf1baef Added h1 tag
0c6a8f750d8678be13ce01ea59b05a1acdd3d0c1 Changes for a and b
fcfacf99f2ee9f98473160c363b83fb015ceb248 First Commit

33319@Komp MINGW64 ~/Desktop/Work (master)
$ |
```

```
MINGW64:/c/Users/33319/Desktop/Work

33319@Komp MINGW64 ~/Desktop/Work (master)
$ git log --pretty=oneline --author=<your name>
bash: syntax error near unexpected token `newline'

33319@Komp MINGW64 ~/Desktop/Work (master)
$ |
```

```
MINGW64:/c/Users/33319/Desktop/Work
bash: syntax error near unexpected token `newline'

33319@Komp MINGW64 ~/Desktop/Work (master)
$ git log --pretty=oneline --all
43912b94303a883ebc5f97854c9ea8c1093f4c2e (HEAD -> master) Added HTML header
d5f5271cf02c3f77d53b633db36d7f6656477bb5 Added standard HTML page tags
041c3ba7c0f1e062d0b37cb3aa30112e04191b64 Added standard HTML page tags
2f517b233acc80dcb1fc8c17e6bd775a1cf1baef Added h1 tag
0c6a8f750d8678be13ce01ea59b05a1acdd3d0c1 Changes for a and b
:...skipping...
43912b94303a883ebc5f97854c9ea8c1093f4c2e (HEAD -> master) Added HTML header
d5f5271cf02c3f77d53b633db36d7f6656477bb5 Added standard HTML page tags
041c3ba7c0f1e062d0b37cb3aa30112e04191b64 Added standard HTML page tags
2f517b233acc80dcb1fc8c17e6bd775a1cf1baef Added h1 tag
0c6a8f750d8678be13ce01ea59b05a1acdd3d0c1 Changes for a and b
fcfacf99f2ee9f98473160c363b83fb015ceb248 First Commit
```

Для просмотра изменений, сделанных за последнюю неделю можно использовать следующую конструкцию: добавим имя автора --author=ivashkovskayatk, если хотим увидеть только изменения, которые сделаны именно им.

```
MINGW64:/c/Users/33319/Desktop/Work

33319@Komp MINGW64 ~/Desktop/Work (master)
$ git log --all --pretty=format:"%h %cd %s (%an)" --since='7 days ago'
43912b9 Tue Feb 6 14:13:43 2024 +0300 Added HTML header (Pavel5198)
d5f5271 Tue Feb 6 14:09:40 2024 +0300 Added standard HTML page tags (Pavel5198)
041c3ba Tue Feb 6 13:26:19 2024 +0300 Added standard HTML page tags (Pavel5198)
2f517b2 Tue Feb 6 08:58:11 2024 +0300 Added h1 tag (Pavel5198)
0c6a8f7 Tue Feb 6 05:05:54 2024 +0300 Changes for a and b (Pavel5198)
fcfacf9 Tue Feb 6 04:21:46 2024 +0300 First Commit (Pavel5198)

33319@Komp MINGW64 ~/Desktop/Work (master)
$
```

8.3 Конечный формат лога

```
MINGW64:/c/Users/33319/Desktop/Work

33319@Komp MINGW64 ~/Desktop/Work (master)
$ git log --pretty=format:"%h %ad | %s [%an]" --graph --date=short
* 43912b9 2024-02-06 | Added HTML header (HEAD -> master) [Pavel5198]
* d5f5271 2024-02-06 | Added standard HTML page tags [Pavel5198]
* 041c3ba 2024-02-06 | Added standard HTML page tags [Pavel5198]
* 2f517b2 2024-02-06 | Added h1 tag [Pavel5198]
* 0c6a8f7 2024-02-06 | Changes for a and b [Pavel5198]
* fcfacf9 2024-02-06 | First Commit [Pavel5198]

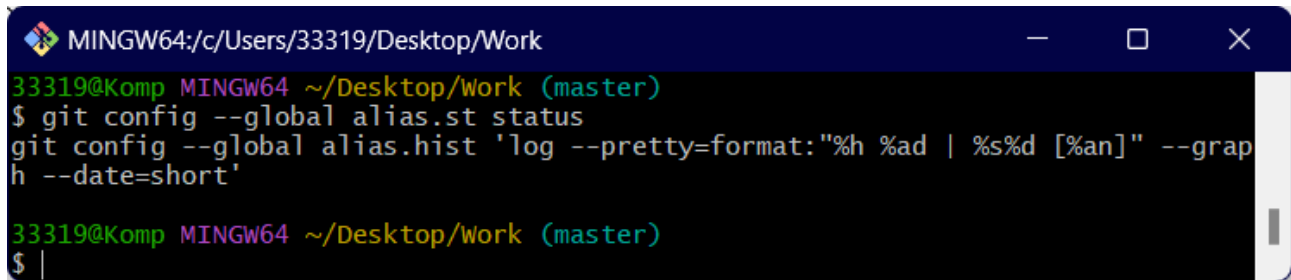
33319@Komp MINGW64 ~/Desktop/Work (master)
$
```

Рассмотрим его в деталях:

- --pretty="..." — определяет формат вывода.
- %h — укороченный хэш коммита

- %d — дополнения коммита («головы» веток или теги)
- %ad — дата коммита
- %s — комментарий
- %an — имя автора
- --graph — отображает дерево коммитов в виде ASCII-графика
- --date=short — сохраняет формат даты коротким и симпатичным

9 Алиасы



```

MINGW64:/c/Users/33319/Desktop/Work
33319@Komp MINGW64 ~/Desktop/Work (master)
$ git config --global alias.st status
git config --global alias.hist 'log --pretty=format:"%h %ad | %s%d [%an]" --graph --date=short'
33319@Komp MINGW64 ~/Desktop/Work (master)
$

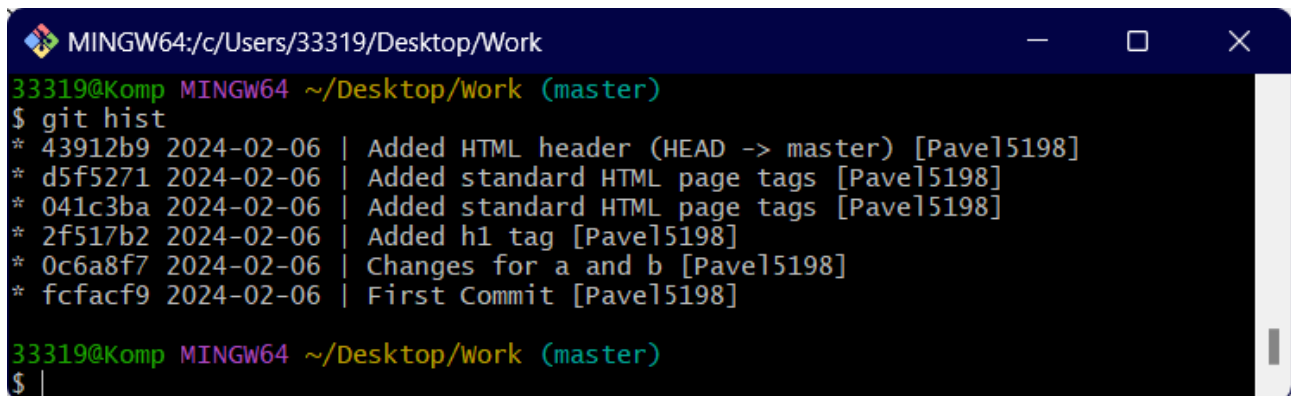
```

Команда `git hist` позволит избежать ввода очень длинной команды `log`.

10 Получение старых версий

Научимся возвращать рабочий каталог к любому предыдущему состоянию. Возвращаться назад в историю очень просто. Команда `checkout` скопирует любой снимок из репозитория в рабочий каталог.

10.1 Получение хэша предыдущих версий



```

MINGW64:/c/Users/33319/Desktop/Work
33319@Komp MINGW64 ~/Desktop/Work (master)
$ git hist
* 43912b9 2024-02-06 | Added HTML header (HEAD -> master) [Pavel5198]
* d5f5271 2024-02-06 | Added standard HTML page tags [Pavel5198]
* 041c3ba 2024-02-06 | Added standard HTML page tags [Pavel5198]
* 2f517b2 2024-02-06 | Added h1 tag [Pavel5198]
* 0c6a8f7 2024-02-06 | Changes for a and b [Pavel5198]
* fcfacf9 2024-02-06 | First Commit [Pavel5198]
33319@Komp MINGW64 ~/Desktop/Work (master)
$

```

Проверим содержимое файла `hello.html`.

```
MINGW64:/c/Users/33319/Desktop/Work
33319@Komp MINGW64 ~/Desktop/Work (master)
$ git checkout <hash>
cat hello.html
bash: syntax error near unexpected token `newline'
<html>
  <head>
  </head>
  <body>
    <h1>Hello, World!</h1>
  </body>
</html>

33319@Komp MINGW64 ~/Desktop/Work (master)
$
```

Подставим необходимое значение хэш для нашего репозитория.

```
MINGW64:/c/Users/33319/Desktop/Work
33319@Komp MINGW64 ~/Desktop/Work (master)
$ git checkout fcfacf9
Note: switching to 'fcfacf9'.

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 fcfacf9 First Commit

33319@Komp MINGW64 ~/Desktop/Work ((fcfacf9...))
$ |
```

10.2 Возвращение к последней версии в ветке master

«master» — имя ветки по умолчанию. Переключая имена веток, мы попадаем на последнюю версию выбранной ветки.

```
MINGW64:/c/Users/33319/Desktop/Work
33319@Komp MINGW64 ~/Desktop/Work ((fcfacf9...))
$ git checkout master
cat hello.html
Previous HEAD position was fcfacf9 First Commit
Switched to branch 'master'
<html>
  <head>
  </head>
  <body>
    <h1>Hello, World!</h1>
  </body>
</html>
33319@Komp MINGW64 ~/Desktop/Work (master)
$
```

11 Создание тегов версий

Узнаем, как создавать теги для коммитов для использования в будущем. Назовем текущую версию страницы hello первой (v1).

11.1 Создание тега первой версии

```
MINGW64:/c/Users/33319/Desktop/Work
33319@Komp MINGW64 ~/Desktop/Work (master)
$ git tag v1
33319@Komp MINGW64 ~/Desktop/Work (master)
$
```

Теперь текущая версия страницы называется v1.

11.2 Теги для предыдущих версий

Создадим тег для версии, которая идет перед текущей версией и назовем его v1-beta. В первую очередь нам надо переключиться на предыдущую версию. Вместо поиска до хэш, мы будем использовать ^, обозначающее «родитель v1».


```
MINGW64:/c/Users/33319/Desktop/Work
33319@Komp MINGW64 ~/Desktop/Work (master)
$ git checkout v1^
cat hello.html
Note: switching to 'v1^'.

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 d5f5271 Added standard HTML page tags
<html>
  <body>
    <h1>Hello, World!</h1>
  </body>
</html>

33319@Komp MINGW64 ~/Desktop/Work ((d5f5271...))
$ |
```

11.3 Переключение по имени тега

Попробуем попереключаться между двумя отмеченными версиями.

```
MINGW64:/c/Users/33319/Desktop/Work
33319@Komp MINGW64 ~/Desktop/Work ((d5f5271...))
$ git checkout v1
git checkout v1-beta
Previous HEAD position was d5f5271 Added standard HTML page tags
HEAD is now at 43912b9 Added HTML header
error: pathspec 'v1-beta' did not match any file(s) known to git

33319@Komp MINGW64 ~/Desktop/Work ((v1))
$
```

11.4 Просмотр тегов с помощью команды tag

```
MINGW64:/c/Users/33319/Desktop/Work
33319@Komp MINGW64 ~/Desktop/Work ((v1))
$ git tag
v1

33319@Komp MINGW64 ~/Desktop/Work ((v1))
$
```


11.5 Просмотр Тегов в логах

```
MINGW64:/c/Users/33319/Desktop/Work

33319@Komp MINGW64 ~/Desktop/Work ((v1))
$ git hist master --all
* 43912b9 2024-02-06 | Added HTML header (HEAD, tag: v1, master) [Pavel5198]
* d5f5271 2024-02-06 | Added standard HTML page tags [Pavel5198]
* 041c3ba 2024-02-06 | Added standard HTML page tags [Pavel5198]
* 2f517b2 2024-02-06 | Added h1 tag [Pavel5198]
* 0c6a8f7 2024-02-06 | Changes for a and b [Pavel5198]
...skipping...
* 43912b9 2024-02-06 | Added HTML header (HEAD, tag: v1, master) [Pavel5198]
* d5f5271 2024-02-06 | Added standard HTML page tags [Pavel5198]
* 041c3ba 2024-02-06 | Added standard HTML page tags [Pavel5198]
* 2f517b2 2024-02-06 | Added h1 tag [Pavel5198]
* 0c6a8f7 2024-02-06 | Changes for a and b [Pavel5198]
* fcfacf9 2024-02-06 | First Commit [Pavel5198]
~
~
```

Мы можем видеть теги (v1 и v1-beta) в логе вместе с именем ветки (master). Кроме того, HEAD показывает коммит, на который вы переключились (на данный момент это v1-beta).

12. Отмена локальных изменений (до индексации)

Научиться отменять изменения в рабочем каталоге.

12.1 Переключение на ветку Master

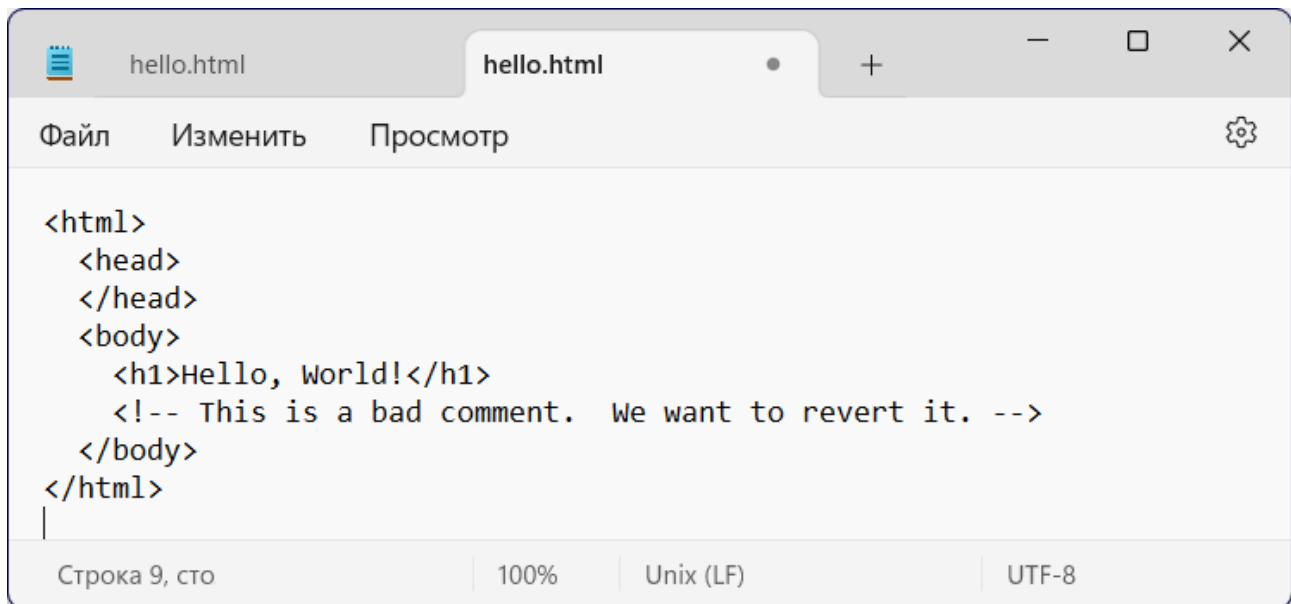
```
MINGW64:/c/Users/33319/Desktop/Work

33319@Komp MINGW64 ~/Desktop/Work ((v1))
$ git checkout master
Switched to branch 'master'

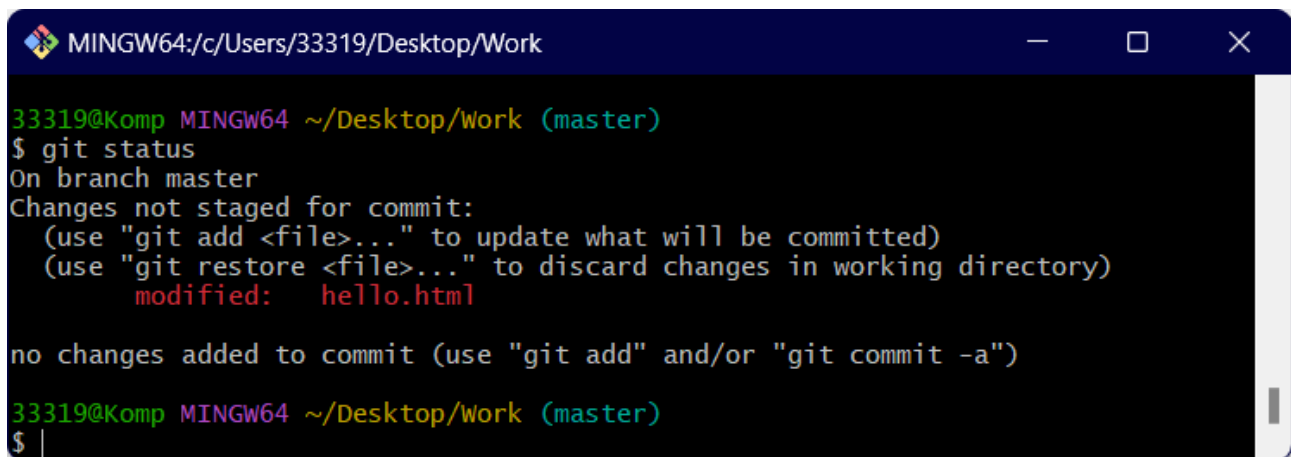
33319@Komp MINGW64 ~/Desktop/Work (master)
$
```

12.2 Изменение hello.html

Иногда случается, что мы изменили файл в рабочем каталоге, и хотим отменить последние коммиты. С этим справится команда checkout.



12.3 Проверка состояния



Мы видим, что файл hello.html был изменен, но еще не проиндексирован.

12.4 Отмена изменений в рабочем каталоге

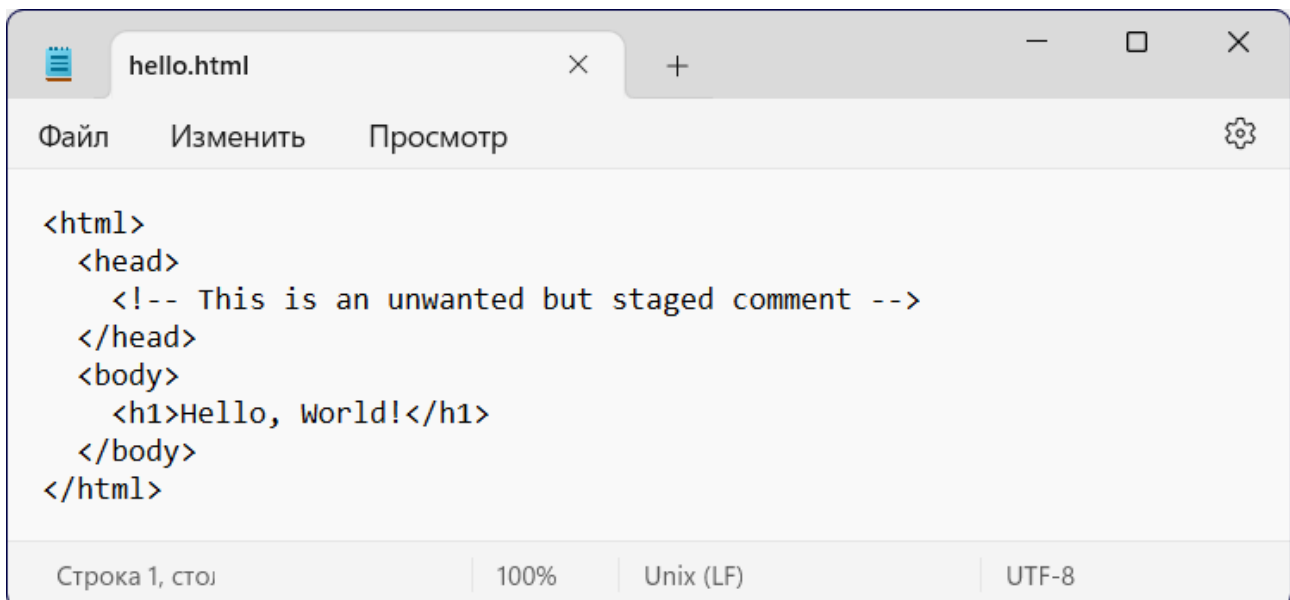
```
MINGW64:/c/Users/33319/Desktop/Work
33319@Komp MINGW64 ~/Desktop/Work (master)
$ git checkout hello.html
git status
cat hello.html
Updated 1 path from the index
On branch master
nothing to commit, working tree clean
<html>
  <head>
  </head>
  <body>
    <h1>Hello, world!</h1>
  </body>
</html>
33319@Komp MINGW64 ~/Desktop/Work (master)
$
```

Команда status показывает нам, что не было произведено никаких изменений, не зафиксированных в рабочем каталоге. И «нежелательный комментарий» больше не является частью содержимого файла.

13. Отмена проиндексированных изменений (перед коммитом)

Научимся отменять изменения, которые были проиндексированы

13.1 Изменение файла и индексация изменения



```
hello.html
Файл  Изменить  Просмотр
<html>
  <head>
    <!-- This is an unwanted but staged comment -->
  </head>
  <body>
    <h1>Hello, world!</h1>
  </body>
</html>
Строка 1, столбец 1  100%  Unix (LF)  UTF-8
```

Проиндексируем это изменение.

```
MINGW64:/c/Users/33319/Desktop/Work
33319@Komp MINGW64 ~/Desktop/Work (master)
$ git add hello.html

33319@Komp MINGW64 ~/Desktop/Work (master)
$
```

13.2 Проверка состояния

```
MINGW64:/c/Users/33319/Desktop/Work
33319@Komp MINGW64 ~/Desktop/Work (master)
$ git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   hello.html

33319@Komp MINGW64 ~/Desktop/Work (master)
$
```

Состояние показывает, что изменение было проиндексировано и готово к коммиту.

13.3 Выполнение сброса буферной зоны

К счастью, вывод состояния показывает нам именно то, что мы должны сделать для отмены индексации изменения.

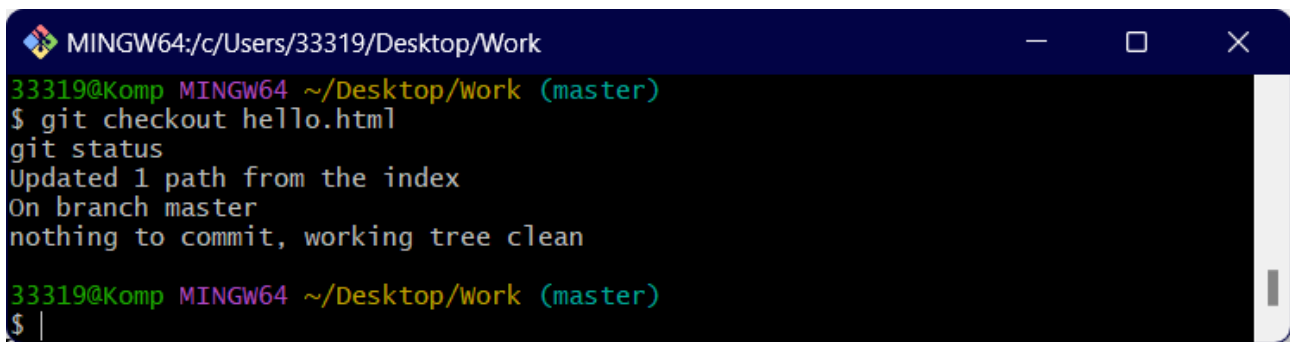
```
MINGW64:/c/Users/33319/Desktop/Work
33319@Komp MINGW64 ~/Desktop/Work (master)
$ git reset HEAD hello.html
Unstaged changes after reset:
M       hello.html

33319@Komp MINGW64 ~/Desktop/Work (master)
$
```

Команда `reset` сбрасывает буферную зону к HEAD. Это очищает буферную зону от изменений, которые мы только что проиндексировали.

Команда `reset` (по умолчанию) не изменяет рабочий каталог. Поэтому рабочий каталог все еще содержит нежелательный комментарий. Мы можем использовать команду `checkout` из предыдущего раздела, чтобы удалить нежелательные изменения в рабочем каталоге.

13.4 Переключение на версию коммита



```
MINGW64:/c/Users/33319/Desktop/Work
33319@Komp MINGW64 ~/Desktop/Work (master)
$ git checkout hello.html
git status
Updated 1 path from the index
On branch master
nothing to commit, working tree clean

33319@Komp MINGW64 ~/Desktop/Work (master)
$ |
```

Наш рабочий каталог опять чист.

14 Отмена коммитов

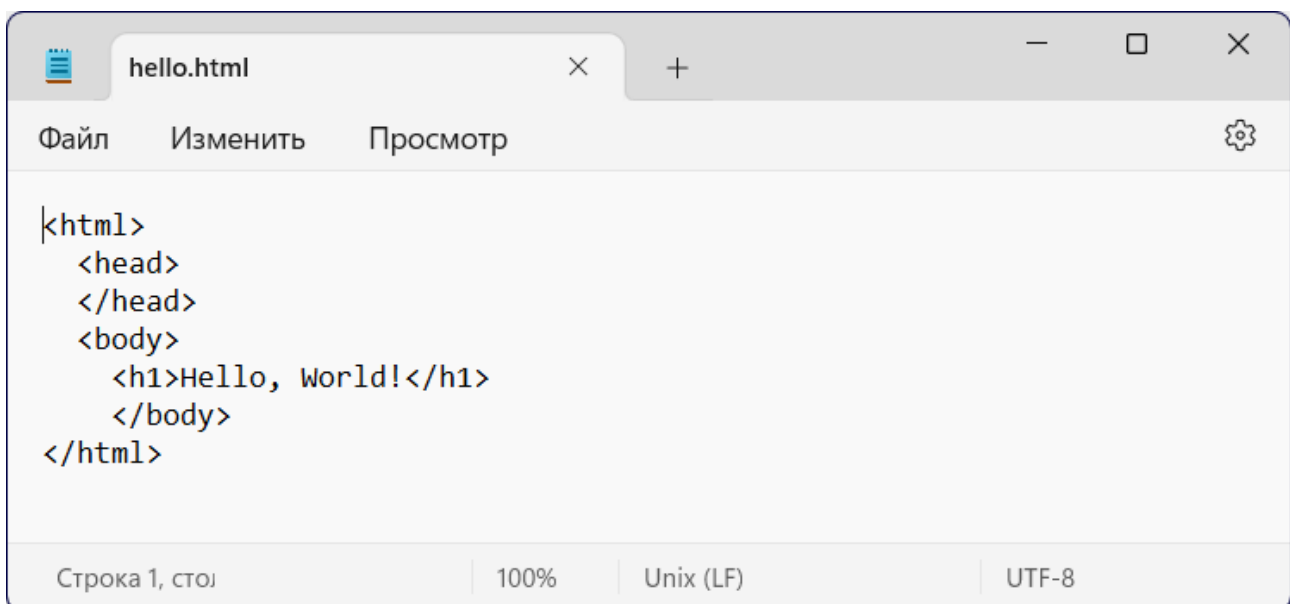
Научимся отменять коммиты в локальный репозиторий.

14.1 Отмена коммитов

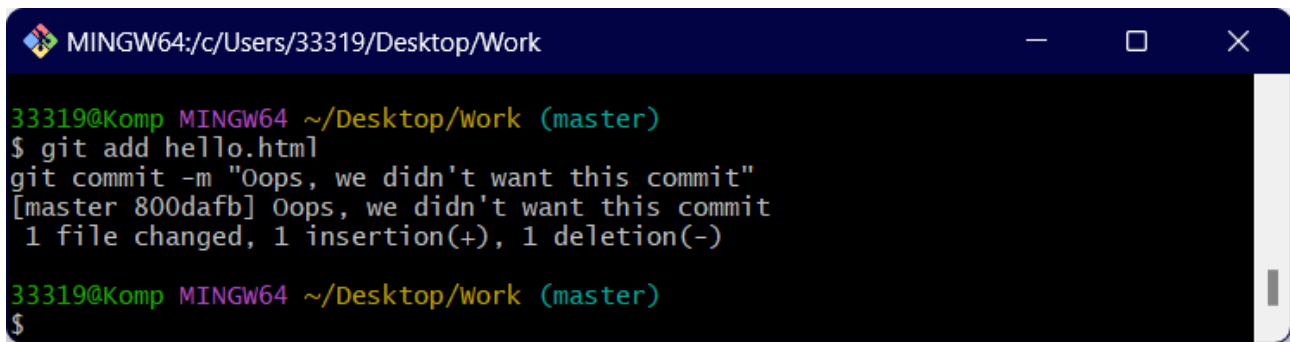
Иногда вы понимаете, что новые коммиты являются неверными, и хотите их отменить. Есть несколько способов решения этого вопроса, здесь мы будем использовать самый безопасный.

Отменим коммит путем создания нового коммита, отменяющего нежелательные изменения.

14.2 Изменение файла и коммит



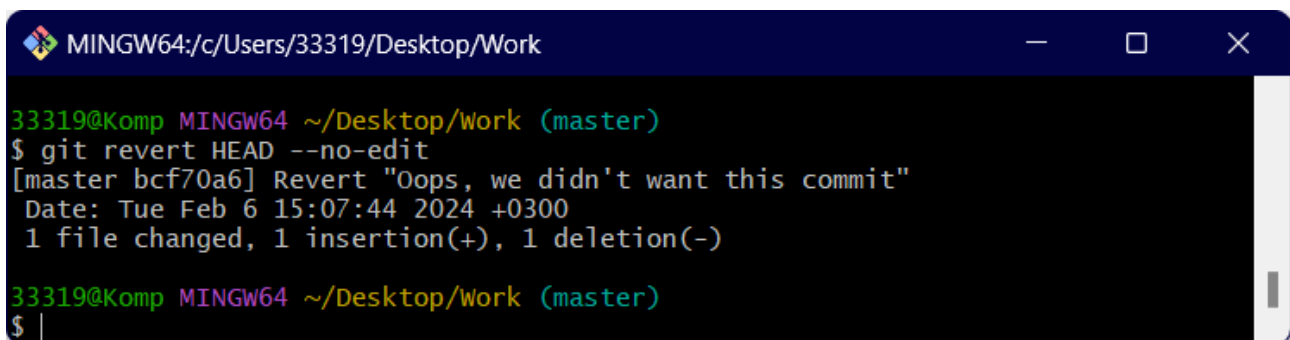
```
hello.html
Файл  Изменить  Просмотр
<html>
  <head>
  </head>
  <body>
    <h1>Hello, world!</h1>
  </body>
</html>
Строка 1, стол 1  100%  Unix (LF)  UTF-8
```

A terminal window titled 'MINGW64:/c/Users/33319/Desktop/Work' with standard window controls. The prompt is '33319@Komp MINGW64 ~/Desktop/Work (master)'. The user enters '\$ git add hello.html' followed by 'git commit -m "Oops, we didn't want this commit"'. The output shows '[master 800dafb] Oops, we didn't want this commit' and '1 file changed, 1 insertion(+), 1 deletion(-)'. The prompt returns to '\$'.

```
MINGW64:/c/Users/33319/Desktop/Work
33319@Komp MINGW64 ~/Desktop/Work (master)
$ git add hello.html
git commit -m "Oops, we didn't want this commit"
[master 800dafb] Oops, we didn't want this commit
1 file changed, 1 insertion(+), 1 deletion(-)
33319@Komp MINGW64 ~/Desktop/Work (master)
$
```

14.3 Коммит с новыми изменениями, отменяющими предыдущие

Чтобы отменить коммит, нам необходимо сделать коммит, который удаляет изменения, сохраненные нежелательным коммитом.

A terminal window titled 'MINGW64:/c/Users/33319/Desktop/Work' with standard window controls. The prompt is '33319@Komp MINGW64 ~/Desktop/Work (master)'. The user enters '\$ git revert HEAD --no-edit'. The output shows '[master bcf70a6] Revert "Oops, we didn't want this commit"', 'Date: Tue Feb 6 15:07:44 2024 +0300', and '1 file changed, 1 insertion(+), 1 deletion(-)'. The prompt returns to '\$'.

```
MINGW64:/c/Users/33319/Desktop/Work
33319@Komp MINGW64 ~/Desktop/Work (master)
$ git revert HEAD --no-edit
[master bcf70a6] Revert "Oops, we didn't want this commit"
Date: Tue Feb 6 15:07:44 2024 +0300
1 file changed, 1 insertion(+), 1 deletion(-)
33319@Komp MINGW64 ~/Desktop/Work (master)
$ |
```

Так как мы отменили самый последний произведенный коммит, мы смогли использовать **HEAD** в качестве аргумента для отмены. Мы можем отменить любой произвольной коммит в истории, указав его хэш-значение.

14.4 Проверка лога

Проверка лога показывает нежелательные и отмененные коммиты в наш репозиторий.

```
MINGW64:/c/Users/33319/Desktop/Work
33319@Komp MINGW64 ~/Desktop/Work (master)
$ git hist
* bcf70a6 2024-02-06 | Revert "Oops, we didn't want this commit" (HEAD -> master) [Pavel5198]
* 800dafb 2024-02-06 | Oops, we didn't want this commit [Pavel5198]
* 43912b9 2024-02-06 | Added HTML header (tag: v1) [Pavel5198]
* d5f5271 2024-02-06 | Added standard HTML page tags [Pavel5198]
* 041c3ba 2024-02-06 | Added standard HTML page tags [Pavel5198]
* 2f517b2 2024-02-06 | Added h1 tag [Pavel5198]
* 0c6a8f7 2024-02-06 | Changes for a and b [Pavel5198]
...skipping...
* bcf70a6 2024-02-06 | Revert "Oops, we didn't want this commit" (HEAD -> master) [Pavel5198]
* 800dafb 2024-02-06 | Oops, we didn't want this commit [Pavel5198]
* 43912b9 2024-02-06 | Added HTML header (tag: v1) [Pavel5198]
* d5f5271 2024-02-06 | Added standard HTML page tags [Pavel5198]
* 041c3ba 2024-02-06 | Added standard HTML page tags [Pavel5198]
* 2f517b2 2024-02-06 | Added h1 tag [Pavel5198]
* 0c6a8f7 2024-02-06 | Changes for a and b [Pavel5198]
* fcfacf9 2024-02-06 | First Commit [Pavel5198]
```

15. Удаление коммитов из ветки

Научимся удалять самые последние коммиты из ветки

15.1 Команда reset

При получении ссылки на коммит (т.е. хэш, ветка или имя тега), команда reset...

1. Перепишет текущую ветку, чтобы она указывала на нужный коммит
2. Опционально сбросит буферную зону для соответствия с указанным коммитом
3. Опционально сбросит рабочий каталог для соответствия с указанным коммитом

15.2. Проверка истории

```
MINGW64:/c/Users/33319/Desktop/Work
33319@Komp MINGW64 ~/Desktop/Work (master)
$ git hist
* bcf70a6 2024-02-06 | Revert "Oops, we didn't want this commit" (HEAD -> master) [Pavel5198]
* 800dafb 2024-02-06 | Oops, we didn't want this commit [Pavel5198]
* 43912b9 2024-02-06 | Added HTML header (tag: v1) [Pavel5198]
* d5f5271 2024-02-06 | Added standard HTML page tags [Pavel5198]
* 041c3ba 2024-02-06 | Added standard HTML page tags [Pavel5198]
* 2f517b2 2024-02-06 | Added h1 tag [Pavel5198]
* 0c6a8f7 2024-02-06 | Changes for a and b [Pavel5198]
* fcfacf9 2024-02-06 | First Commit [Pavel5198]

33319@Komp MINGW64 ~/Desktop/Work (master)
$
```

Удалим два последних коммита в этой ветке с помощью сброса.

15.3 Отметка ветки

```
MINGW64:/c/Users/33319/Desktop/Work
33319@Komp MINGW64 ~/Desktop/Work (master)
$ git tag badcom

33319@Komp MINGW64 ~/Desktop/Work (master)
$
```

15.4 Сброс коммитов к предшествующим коммиту Oops

Глядя на историю лога (см. выше), мы видим, что коммит с тегом «v1» является коммитом, предшествующим ошибочному коммиту. Давайте сбросим ветку до этой точки. Поскольку ветка имеет тег, мы можем использовать имя тега в команде сброса (если она не имеет тега, мы можем использовать хэш-значение).

```
MINGW64:/c/Users/33319/Desktop/Work
33319@Komp MINGW64 ~/Desktop/Work (master)
$ git reset --hard v1
git hist
HEAD is now at 43912b9 Added HTML header
* 43912b9 2024-02-06 | Added HTML header (HEAD -> master, tag: v1) [Pavel5198]
* d5f5271 2024-02-06 | Added standard HTML page tags [Pavel5198]
* 041c3ba 2024-02-06 | Added standard HTML page tags [Pavel5198]
* 2f517b2 2024-02-06 | Added h1 tag [Pavel5198]
* 0c6a8f7 2024-02-06 | Changes for a and b [Pavel5198]
* fcfacf9 2024-02-06 | First Commit [Pavel5198]

33319@Komp MINGW64 ~/Desktop/Work (master)
$
```

Наша ветка master теперь указывает на коммит v1, а двух последних

коммитов в ветке уже нет. Параметр `--hard` указывает, что рабочий каталог должен быть обновлен в соответствии с новым `head` ветки

15.5 Ничего никогда не теряется

Что же случается с ошибочными коммитами? Оказывается, что коммиты все еще находятся в репозитории. На самом деле, мы все еще можем на них ссылаться. Помните, в начале этого раздела мы создали для отмененного коммита тег «`badcom`». Давайте посмотрим на все коммиты.

```
MINGW64:/c/Users/33319/Desktop/Work
33319@Komp MINGW64 ~/Desktop/Work (master)
$ git hist --all
* bcf70a6 2024-02-06 | Revert "Oops, we didn't want this commit" (tag: badcom) [Pavel5198]
* 800dafb 2024-02-06 | Oops, we didn't want this commit [Pavel5198]
* 43912b9 2024-02-06 | Added HTML header (HEAD -> master, tag: v1) [Pavel5198]
* d5f5271 2024-02-06 | Added standard HTML page tags [Pavel5198]
* 041c3ba 2024-02-06 | Added standard HTML page tags [Pavel5198]
* 2f517b2 2024-02-06 | Added h1 tag [Pavel5198]
* 0c6a8f7 2024-02-06 | Changes for a and b [Pavel5198]
* fcfacf9 2024-02-06 | First Commit [Pavel5198]

33319@Komp MINGW64 ~/Desktop/Work (master)
$
```

16. Удаление тега `badcom`

16.1 Удаление тега `badcom`

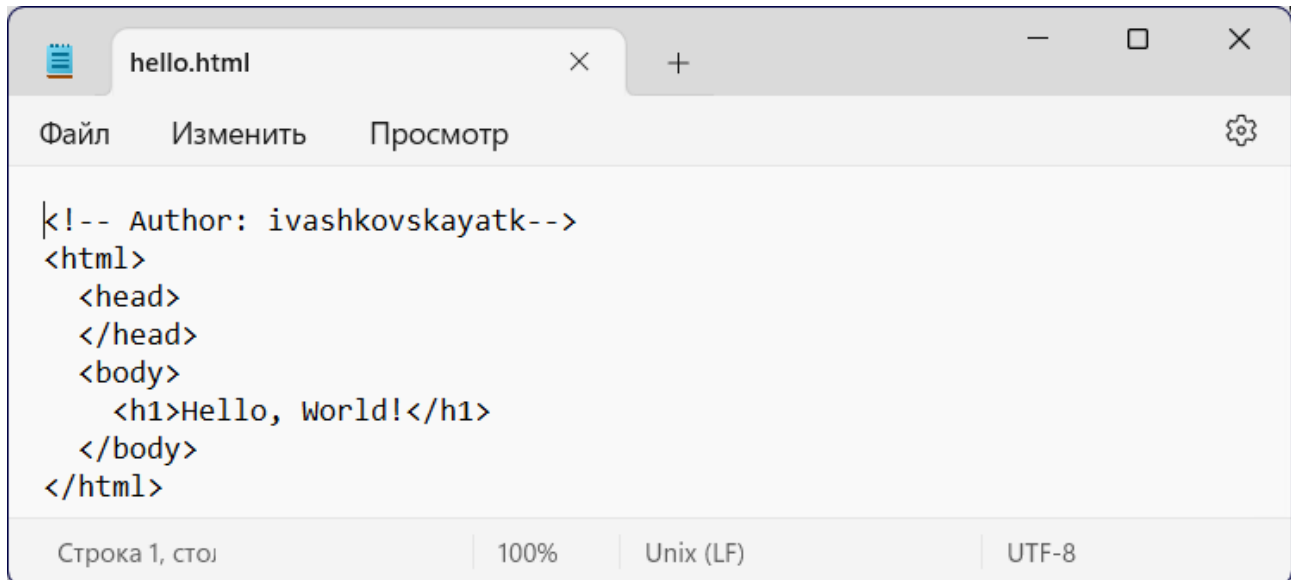
```
MINGW64:/c/Users/33319/Desktop/Work
33319@Komp MINGW64 ~/Desktop/Work (master)
$ git tag -d oops
git hist --all
error: tag 'oops' not found.
* bcf70a6 2024-02-06 | Revert "Oops, we didn't want this commit" (tag: badcom) [Pavel5198]
* 800dafb 2024-02-06 | Oops, we didn't want this commit [Pavel5198]
* 43912b9 2024-02-06 | Added HTML header (HEAD -> master, tag: v1) [Pavel5198]
* d5f5271 2024-02-06 | Added standard HTML page tags [Pavel5198]
* 041c3ba 2024-02-06 | Added standard HTML page tags [Pavel5198]
* 2f517b2 2024-02-06 | Added h1 tag [Pavel5198]
* 0c6a8f7 2024-02-06 | Changes for a and b [Pavel5198]
* fcfacf9 2024-02-06 | First Commit [Pavel5198]

33319@Komp MINGW64 ~/Desktop/Work (master)
$
```

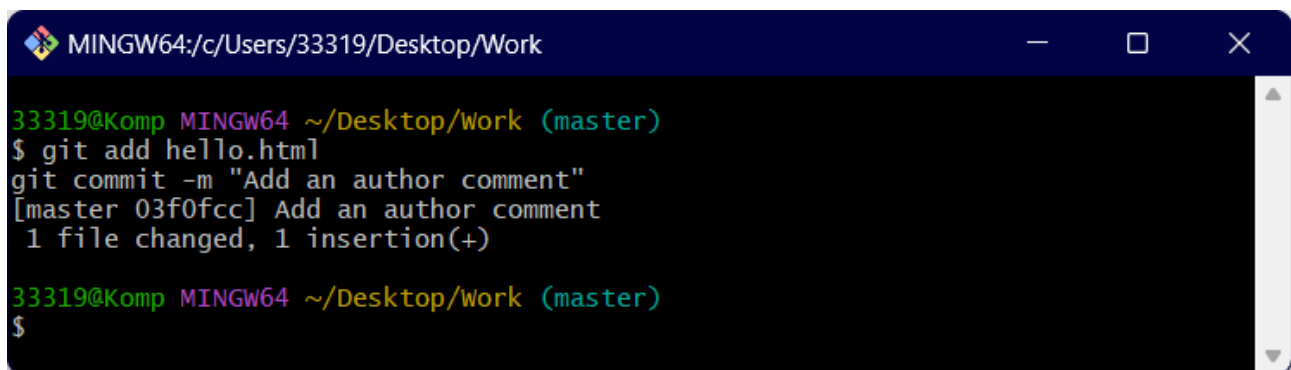
17 Внесение изменений в коммиты

17.1 Изменение страницы с последующим коммитом

Добавим в страницу комментарий автора.

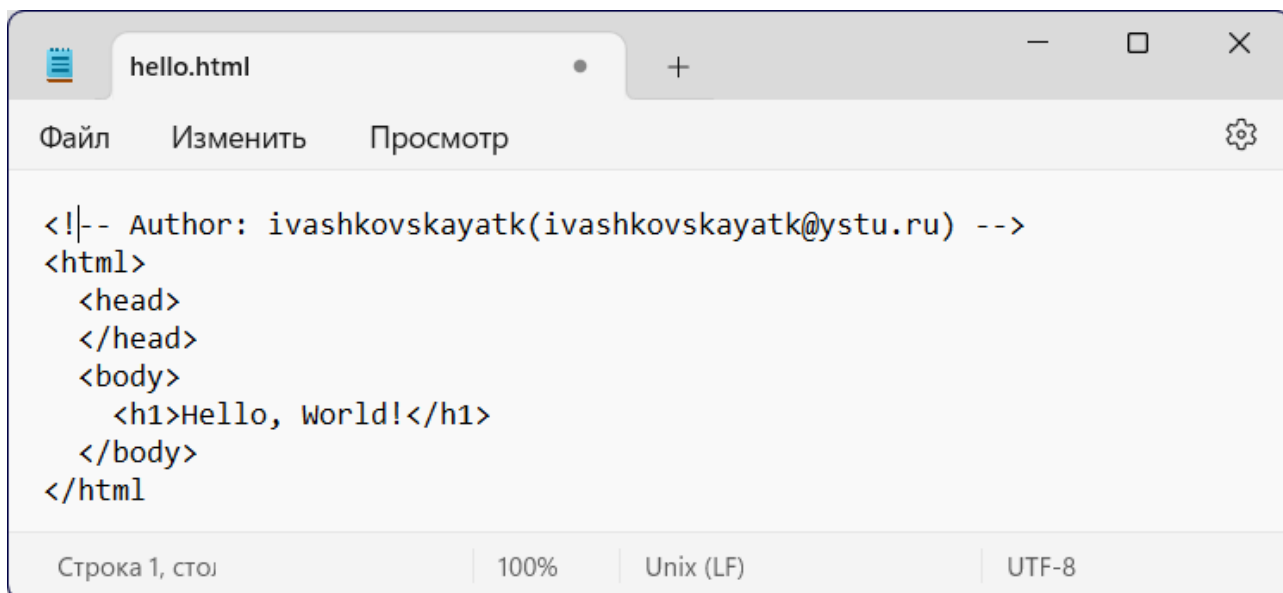


```
<!-- Author: ivashkovskayatk-->
<html>
  <head>
  </head>
  <body>
    <h1>Hello, world!</h1>
  </body>
</html>
```



```
MINGW64:/c/Users/33319/Desktop/Work
33319@Komp MINGW64 ~/Desktop/Work (master)
$ git add hello.html
git commit -m "Add an author comment"
[master 03f0fcc] Add an author comment
1 file changed, 1 insertion(+)
33319@Komp MINGW64 ~/Desktop/Work (master)
$
```

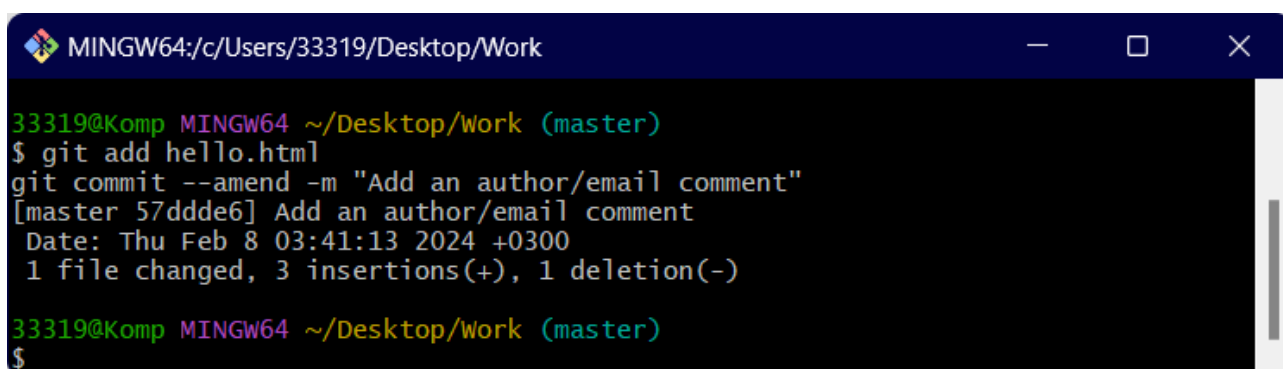
17.2 Обновление страницы hello



```
hello.html
Файл  Изменить  Просмотр
<!-- Author: ivashkovskayatk(ivashkovskayatk@ystu.ru) -->
<html>
  <head>
  </head>
  <body>
    <h1>Hello, World!</h1>
  </body>
</html>
Строка 1, столб 1  100%  Unix (LF)  UTF-8
```

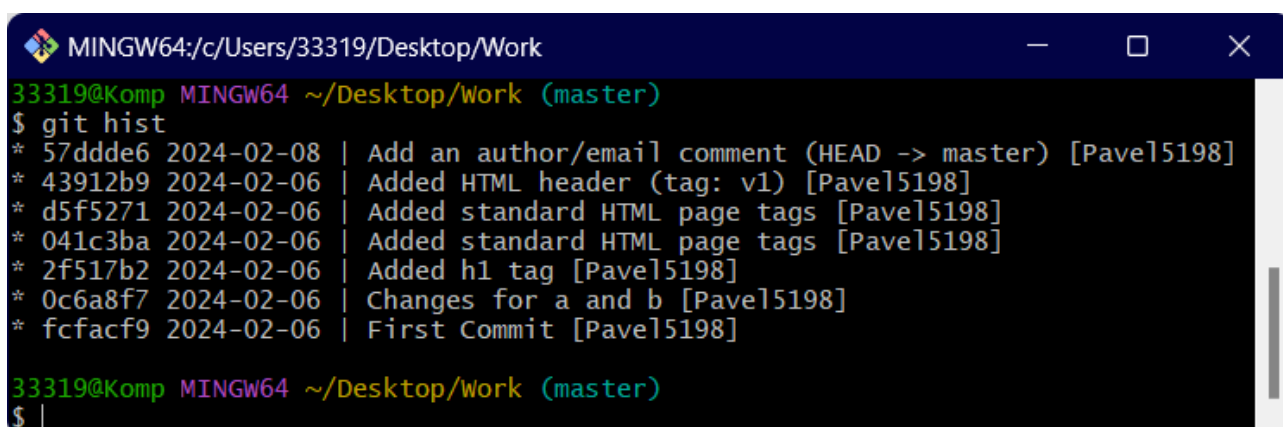
17.3 Изменение предыдущего коммита

Изменим предыдущий коммит, включив в него адрес электронной почты.



```
MINGW64:/c/Users/33319/Desktop/Work
33319@Komp MINGW64 ~/Desktop/Work (master)
$ git add hello.html
git commit --amend -m "Add an author/email comment"
[master 57ddde6] Add an author/email comment
Date: Thu Feb 8 03:41:13 2024 +0300
1 file changed, 3 insertions(+), 1 deletion(-)
33319@Komp MINGW64 ~/Desktop/Work (master)
$
```

17.4 Просмотр истории



```
MINGW64:/c/Users/33319/Desktop/Work
33319@Komp MINGW64 ~/Desktop/Work (master)
$ git hist
* 57ddde6 2024-02-08 | Add an author/email comment (HEAD -> master) [Pavel5198]
* 43912b9 2024-02-06 | Added HTML header (tag: v1) [Pavel5198]
* d5f5271 2024-02-06 | Added standard HTML page tags [Pavel5198]
* 041c3ba 2024-02-06 | Added standard HTML page tags [Pavel5198]
* 2f517b2 2024-02-06 | Added h1 tag [Pavel5198]
* 0c6a8f7 2024-02-06 | Changes for a and b [Pavel5198]
* fcfacf9 2024-02-06 | First Commit [Pavel5198]
33319@Komp MINGW64 ~/Desktop/Work (master)
$
```

Можем увидеть, что оригинальный коммит «автор» заменен коммитом «email». Этого же эффекта можно достичь путем сброса последнего коммита в ветке, и повторного коммита новых изменений.