

Вариативные задания 1

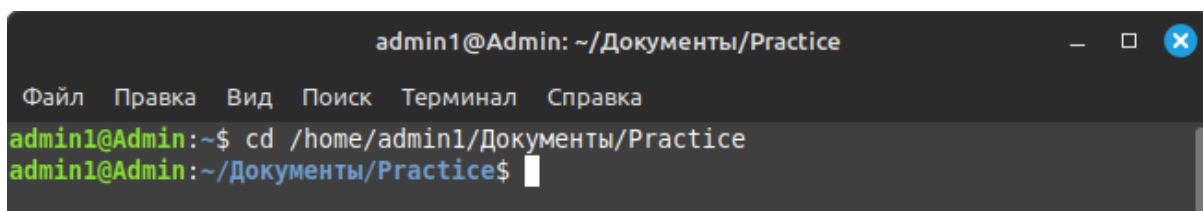
Задание 2.1

Задание выполняется после проведения вебинара «Использование системы контроля версий Git».

На примере основных действий, продемонстрированных на вебинаре по Git, освоить работу (и продемонстрировать выполнение показанных на вебинаре действий) с одним из визуальных клиентов для работы с Git на выбор:

Вариант 4. Терминал (Terminal) или Командная строка (или Command Shell Git).

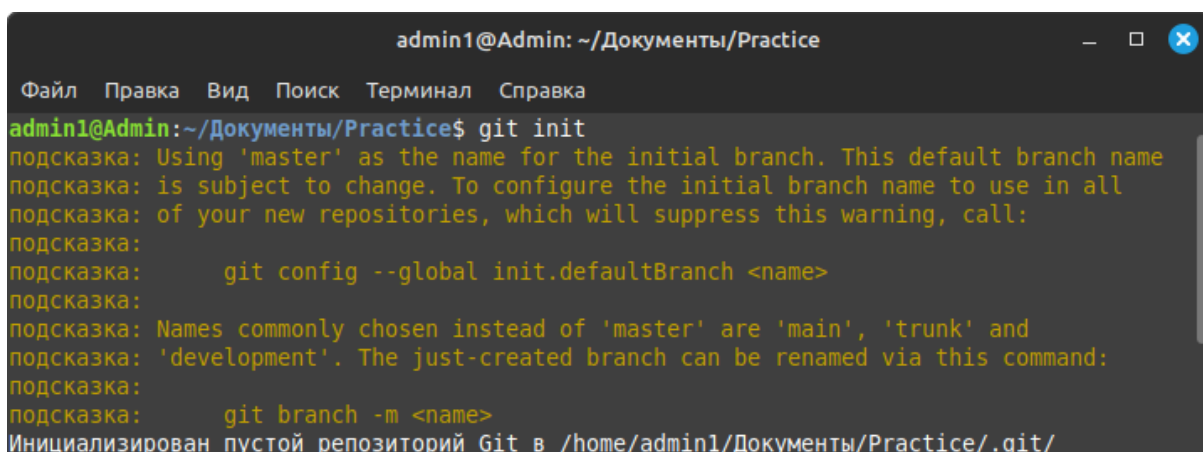
1. `cd` + путь, где будет находиться репозиторий - Выбор места, где будет находиться проект на диске.



```
admin1@Admin: ~/Документы/Practice
Файл  Правка  Вид  Поиск  Терминал  Справка
admin1@Admin:~$ cd /home/admin1/Документы/Practice
admin1@Admin:~/Документы/Practice$
```

Рисунок 1.

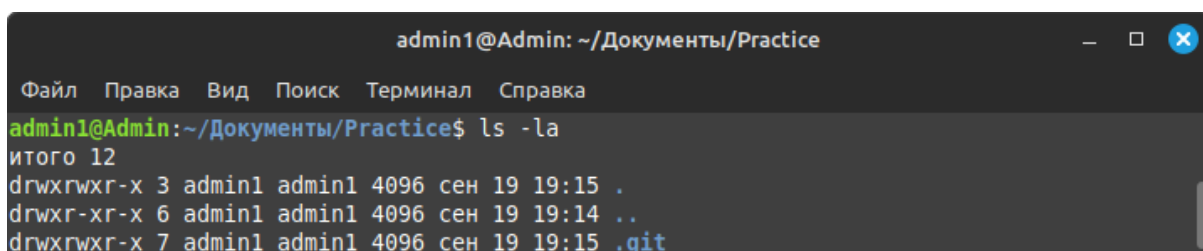
2. `git init` - Инициализирует пустой репозиторий.



```
admin1@Admin: ~/Документы/Practice
Файл  Правка  Вид  Поиск  Терминал  Справка
admin1@Admin:~/Документы/Practice$ git init
подсказка: Using 'master' as the name for the initial branch. This default branch name
подсказка: is subject to change. To configure the initial branch name to use in all
подсказка: of your new repositories, which will suppress this warning, call:
подсказка:
подсказка:     git config --global init.defaultBranch <name>
подсказка:
подсказка: Names commonly chosen instead of 'master' are 'main', 'trunk' and
подсказка: 'development'. The just-created branch can be renamed via this command:
подсказка:
подсказка:     git branch -m <name>
Инициализирован пустой репозиторий Git в /home/admin1/Документы/Practice/.git/
```

Рисунок 2.

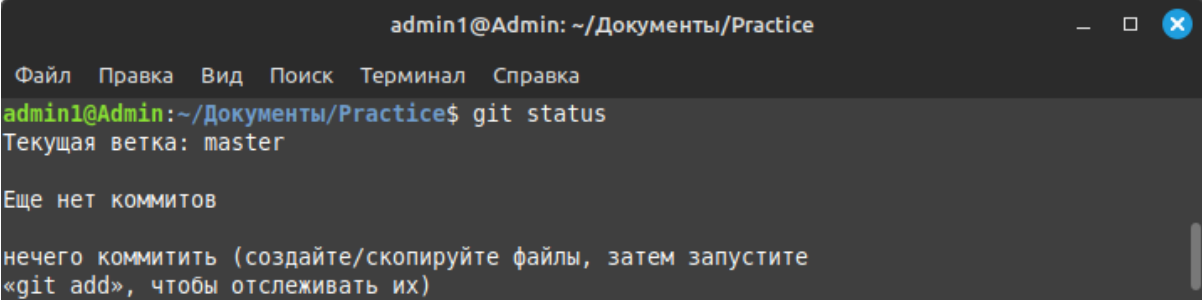
3. `ls -la` - Позволяет увидеть, что репозиторий создан.



```
admin1@Admin: ~/Документы/Practice
Файл  Правка  Вид  Поиск  Терминал  Справка
admin1@Admin:~/Документы/Practice$ ls -la
итого 12
drwxrwxr-x 3 admin1 admin1 4096 сен 19 19:15 .
drwxr-xr-x 6 admin1 admin1 4096 сен 19 19:14 ..
drwxrwxr-x 7 admin1 admin1 4096 сен 19 19:15 .git
```

Рисунок 3.

4. `git status` - Показывает статус репозитория.



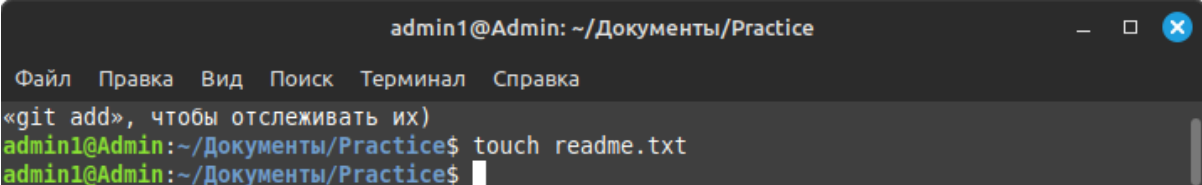
```
admin1@Admin: ~/Документы/Practice
Файл  Правка  Вид  Поиск  Терминал  Справка
admin1@Admin:~/Документы/Practice$ git status
Текущая ветка: master

Еще нет коммитов

нечего коммитить (создайте/скопируйте файлы, затем запустите
«git add», чтобы отслеживать их)
```

Рисунок 4.

5. `touch + название файла` - Создает файл в репозитории.



```
admin1@Admin: ~/Документы/Practice
Файл  Правка  Вид  Поиск  Терминал  Справка
«git add», чтобы отслеживать их)
admin1@Admin:~/Документы/Practice$ touch readme.txt
admin1@Admin:~/Документы/Practice$
```

Рисунок 5.

Файл создан.

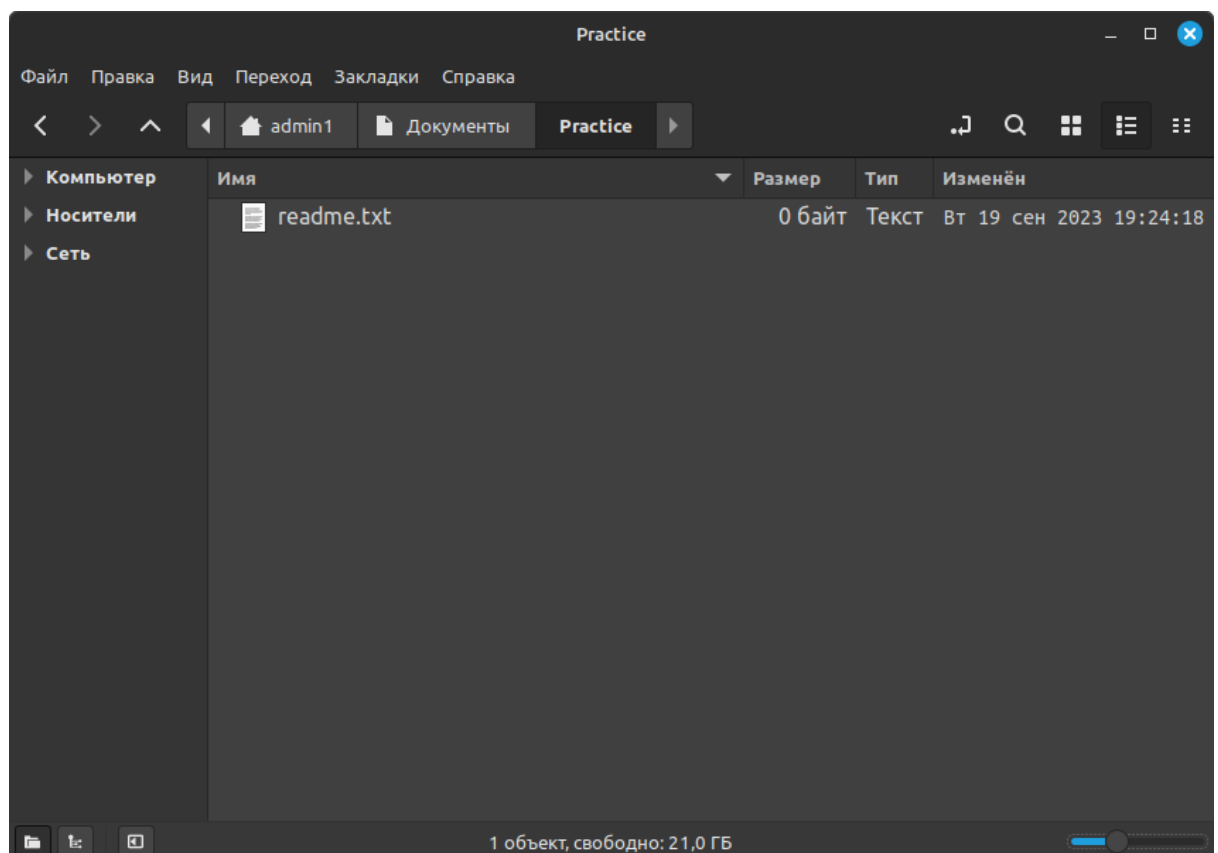


Рисунок 6.

6. nano readme.txt - Позволяет отредактировать файл readme.txt в gnu интерфейсе.

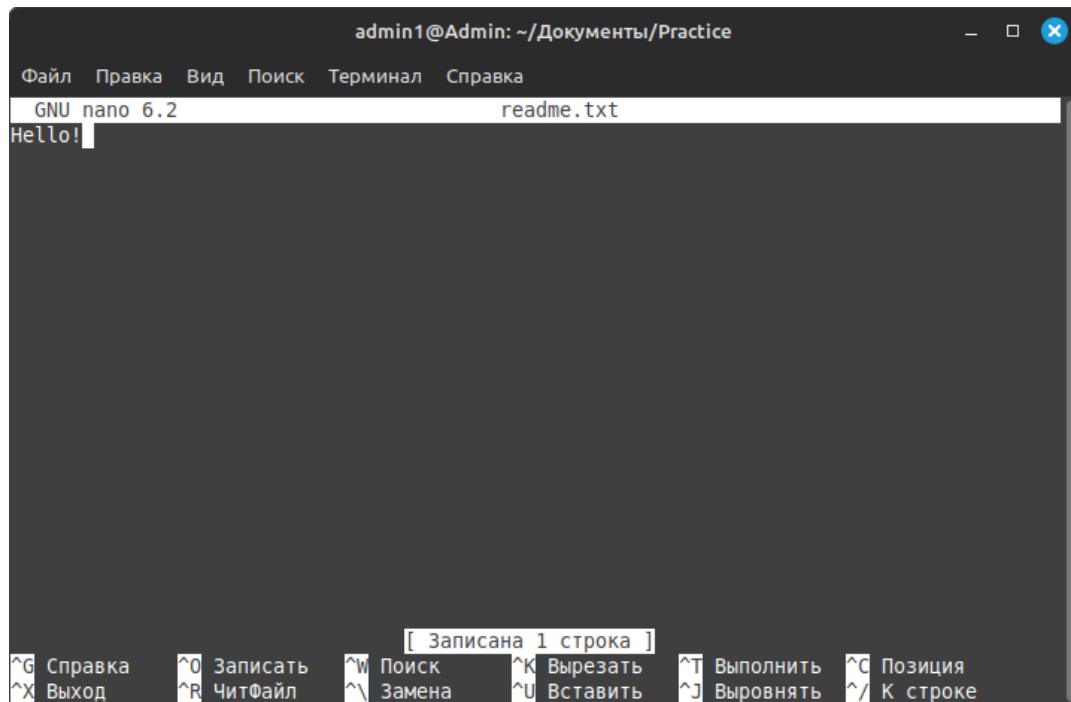


Рисунок 7.

Отредактированный файл в каталоге.

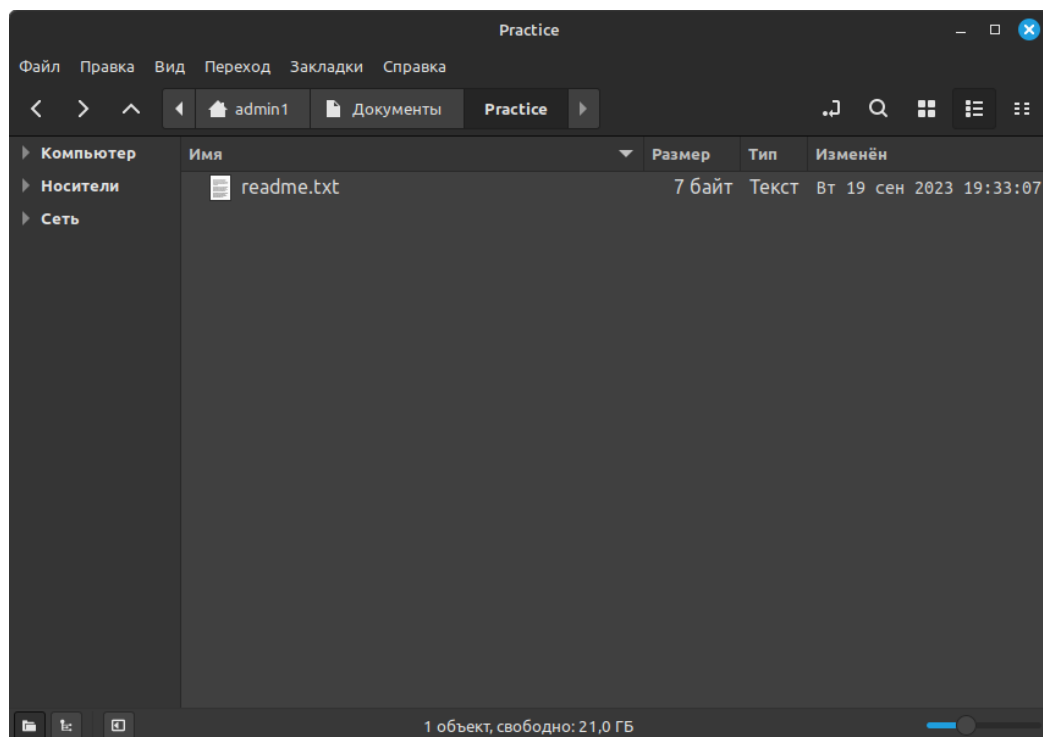
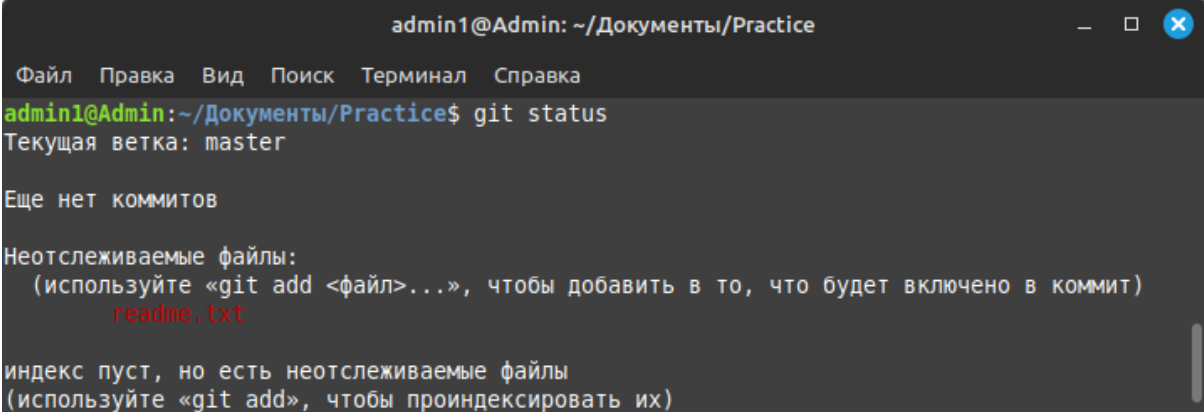


Рисунок 8.

7. `git status` - Позволяет зафиксировать изменения. Файл создан, но не отслеживается системой.



```
admin1@Admin: ~/Документы/Practice
Файл  Правка  Вид  Поиск  Терминал  Справка
admin1@Admin:~/Документы/Practice$ git status
Текущая ветка: master

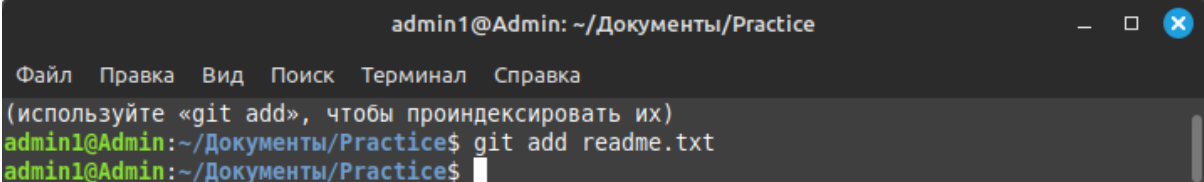
Еще нет коммитов

Неотслеживаемые файлы:
  (используйте «git add <файл>...», чтобы добавить в то, что будет включено в коммит)
  readme.txt

индекс пуст, но есть неотслеживаемые файлы
(используйте «git add», чтобы проиндексировать их)
```

Рисунок 9.

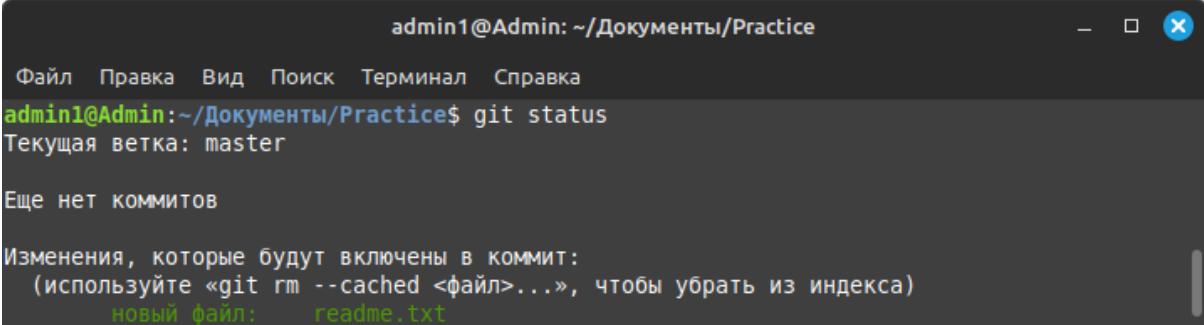
8. `git add` + имя файла - Делает файл отслеживаемым.



```
admin1@Admin: ~/Документы/Practice
Файл  Правка  Вид  Поиск  Терминал  Справка
(используйте «git add», чтобы проиндексировать их)
admin1@Admin:~/Документы/Practice$ git add readme.txt
admin1@Admin:~/Документы/Practice$
```

Рисунок 10.

9. `git status` - Проверяет новый статус файла. Файл становится отслеживаемым.



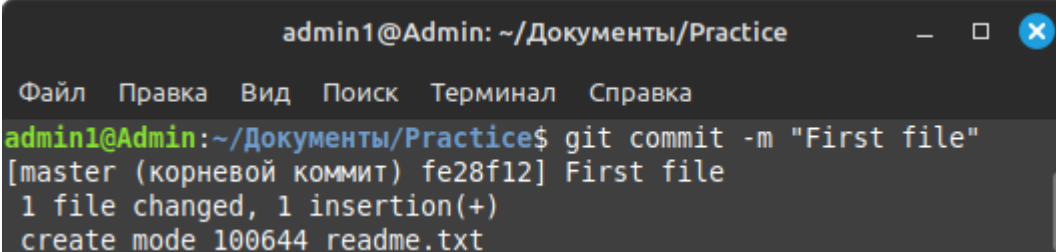
```
admin1@Admin: ~/Документы/Practice
Файл  Правка  Вид  Поиск  Терминал  Справка
admin1@Admin:~/Документы/Practice$ git status
Текущая ветка: master

Еще нет коммитов

Изменения, которые будут включены в коммит:
  (используйте «git rm --cached <файл>...», чтобы убрать из индекса)
  новый файл:   readme.txt
```

Рисунок 11.

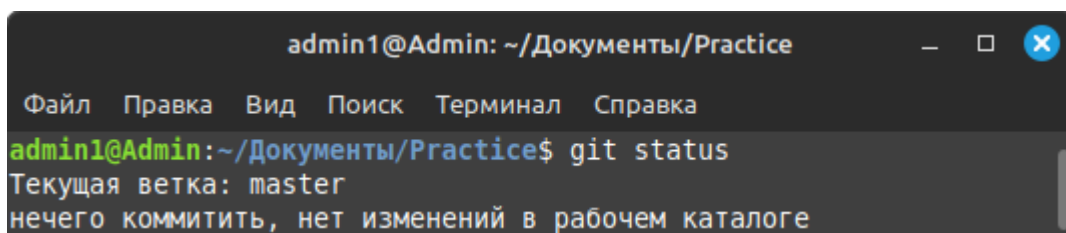
10. `git commit -m "First file"` - Фиксирует статус файла. Тем самым он хранится в репозитории.



```
admin1@Admin: ~/Документы/Practice
Файл  Правка  Вид  Поиск  Терминал  Справка
admin1@Admin:~/Документы/Practice$ git commit -m "First file"
[master (корневой коммит) fe28f12] First file
1 file changed, 1 insertion(+)
create mode 100644 readme.txt
```

Рисунок 12.

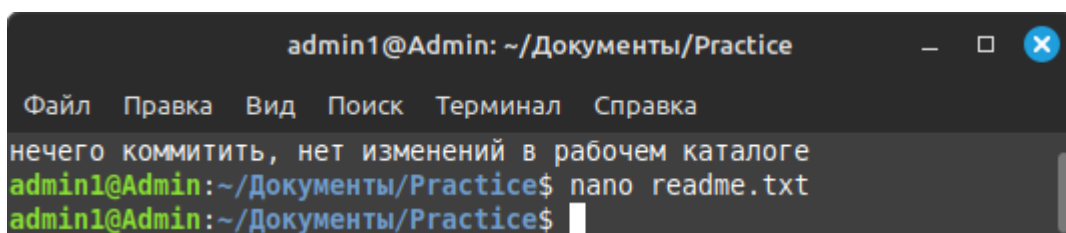
11. `git status` - Показывает статус файла. Т.к. файлов не было добавлено, нечего фиксировать.



```
admin1@Admin: ~/Документы/Practice
Файл  Правка  Вид  Поиск  Терминал  Справка
admin1@Admin:~/Документы/Practice$ git status
Текущая ветка: master
ничего коммитить, нет изменений в рабочем каталоге
```

Рисунок 13.

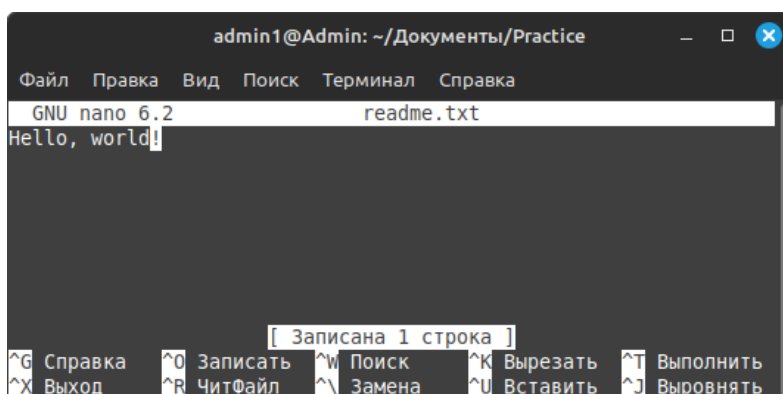
12. `nano readme.txt` - Ещё раз выполняем команду, чтобы изменить файл.



```
admin1@Admin: ~/Документы/Practice
Файл  Правка  Вид  Поиск  Терминал  Справка
ничего коммитить, нет изменений в рабочем каталоге
admin1@Admin:~/Документы/Practice$ nano readme.txt
admin1@Admin:~/Документы/Practice$
```

Рисунок 14.

Изменяем файл. Добавляем слово world.

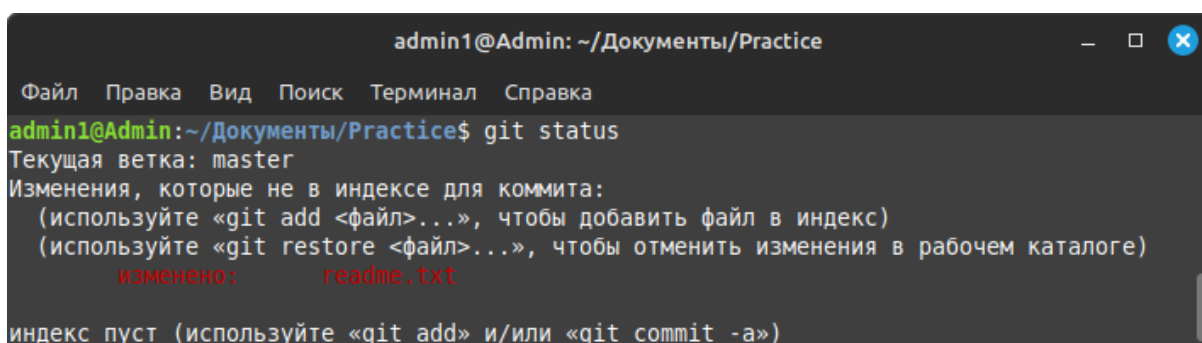


```
admin1@Admin: ~/Документы/Practice
Файл  Правка  Вид  Поиск  Терминал  Справка
GNU nano 6.2      readme.txt
Hello, world!

[ Записана 1 строка ]
^G Справка  ^O Записать  ^W Поиск    ^K Вырезать ^T Выполнить
^X Выход    ^R ЧитФайл  ^\ Замена   ^U Вставить ^J Выводить
```

Рисунок 15.

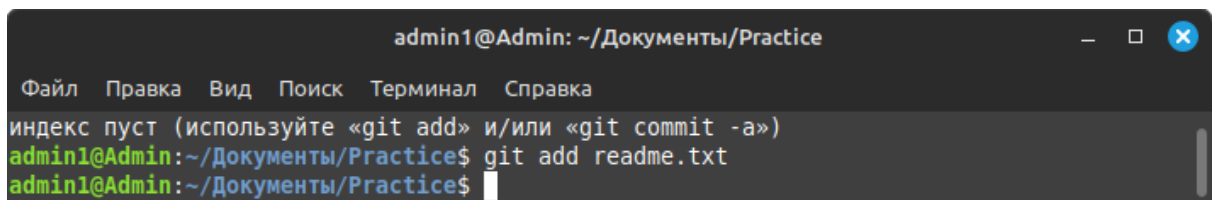
13. `git status` - Ещё раз проверяем статус файла. Файл отслеживается, но изменения не отмечены для добавления в commit.



```
admin1@Admin: ~/Документы/Practice
Файл  Правка  Вид  Поиск  Терминал  Справка
admin1@Admin:~/Документы/Practice$ git status
Текущая ветка: master
Изменения, которые не в индексе для коммита:
  (используйте «git add <файл>...», чтобы добавить файл в индекс)
  (используйте «git restore <файл>...», чтобы отменить изменения в рабочем каталоге)
        изменено:      readme.txt
индекс пуст (используйте «git add» и/или «git commit -a»)
```

Рисунок 16.

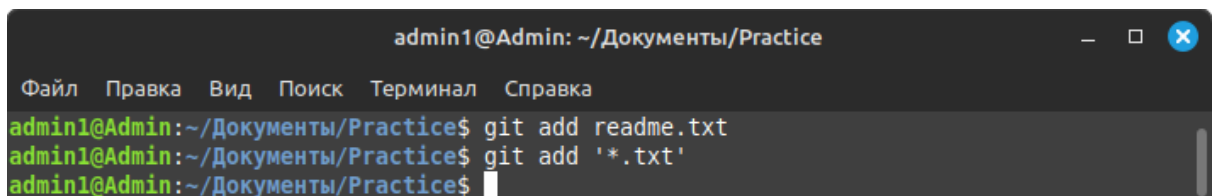
14. `git add` + имя файла - Добавляем файл в отслеживаемые staged area .



```
admin1@Admin: ~/Документы/Practice
Файл  Правка  Вид  Поиск  Терминал  Справка
индекс пуст (используйте «git add» и/или «git commit -a»)
admin1@Admin:~/Документы/Practice$ git add readme.txt
admin1@Admin:~/Документы/Practice$
```

Рисунок 17.

15. `git add '*.txt'` - добавляет только файлы текстового формата.



```
admin1@Admin: ~/Документы/Practice
Файл  Правка  Вид  Поиск  Терминал  Справка
admin1@Admin:~/Документы/Practice$ git add readme.txt
admin1@Admin:~/Документы/Practice$ git add '*.txt'
admin1@Admin:~/Документы/Practice$
```

Рисунок 18.

16. Продублируем файлы формата txt.

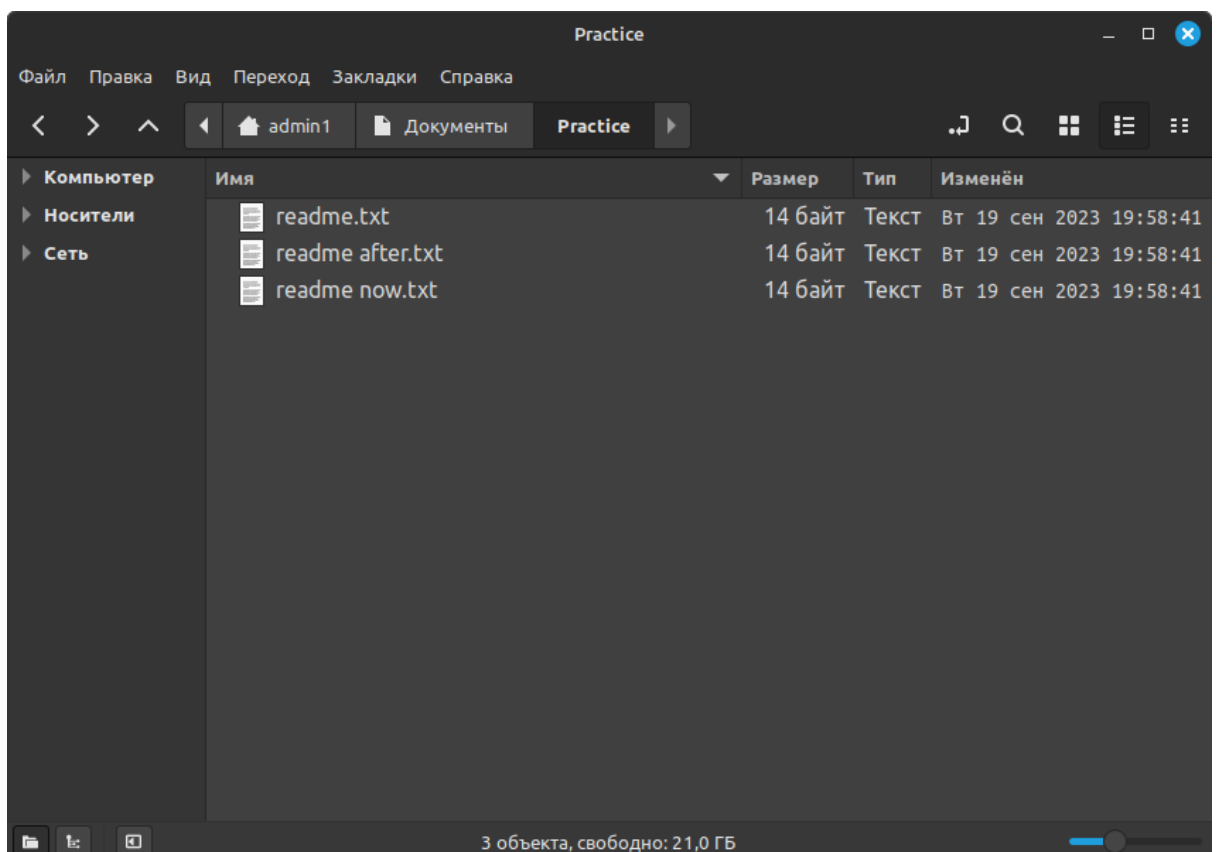
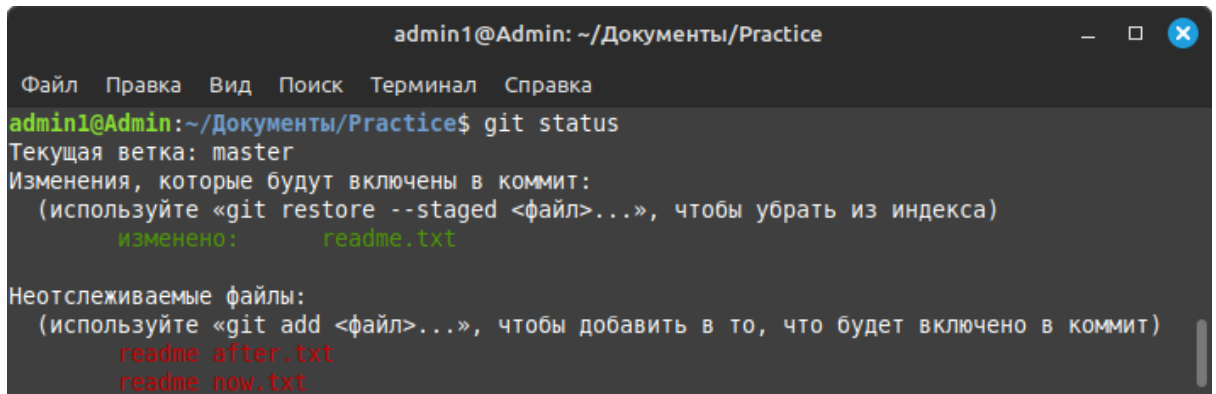


Рисунок 19.

17. `git status` - Смотрим статус добавленных файлов.

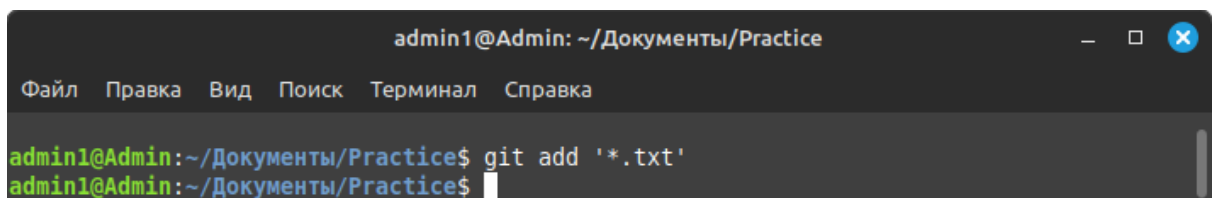


```
admin1@Admin: ~/Документы/Practice
Файл  Правка  Вид  Поиск  Терминал  Справка
admin1@Admin:~/Документы/Practice$ git status
Текущая ветка: master
Изменения, которые будут включены в коммит:
  (используйте «git restore --staged <файл>...», чтобы убрать из индекса)
        изменено:      readme.txt

Неотслеживаемые файлы:
  (используйте «git add <файл>...», чтобы добавить в то, что будет включено в коммит)
        readme after.txt
        readme now.txt
```

Рисунок 20.

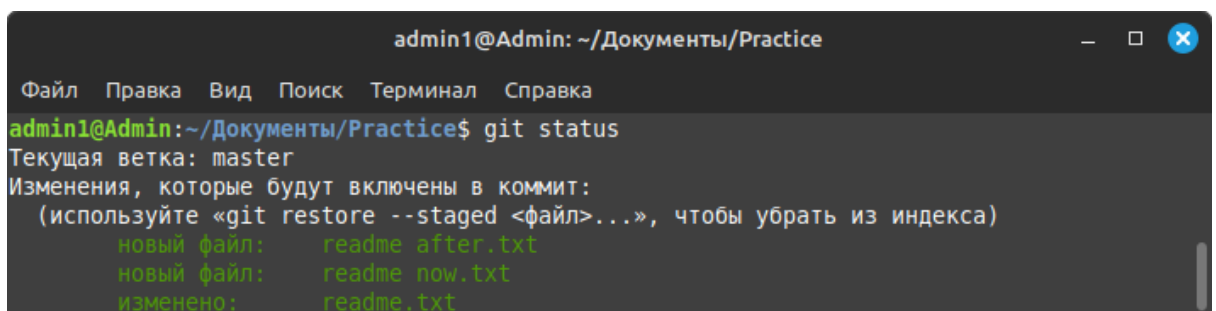
18. `git add '*.txt'` - Добавляем файлы текстового формата в staged area.



```
admin1@Admin: ~/Документы/Practice
Файл  Правка  Вид  Поиск  Терминал  Справка
admin1@Admin:~/Документы/Practice$ git add '*.txt'
admin1@Admin:~/Документы/Practice$
```

Рисунок 21.

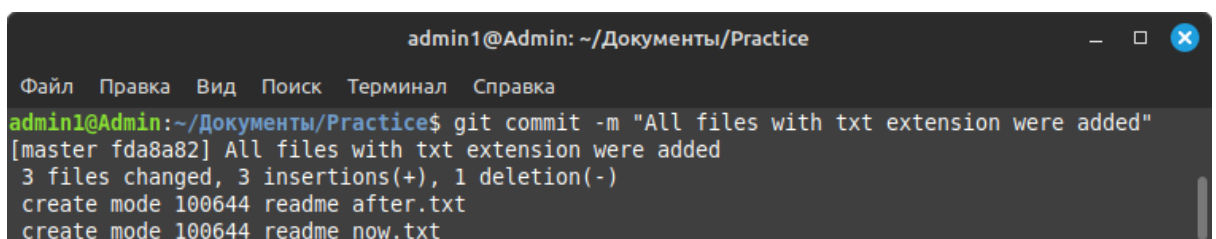
19. `git status` - Проверяем статус новых файлов.



```
admin1@Admin: ~/Документы/Practice
Файл  Правка  Вид  Поиск  Терминал  Справка
admin1@Admin:~/Документы/Practice$ git status
Текущая ветка: master
Изменения, которые будут включены в коммит:
  (используйте «git restore --staged <файл>...», чтобы убрать из индекса)
        новый файл:      readme after.txt
        новый файл:      readme now.txt
        изменено:      readme.txt
```

Рисунок 22.

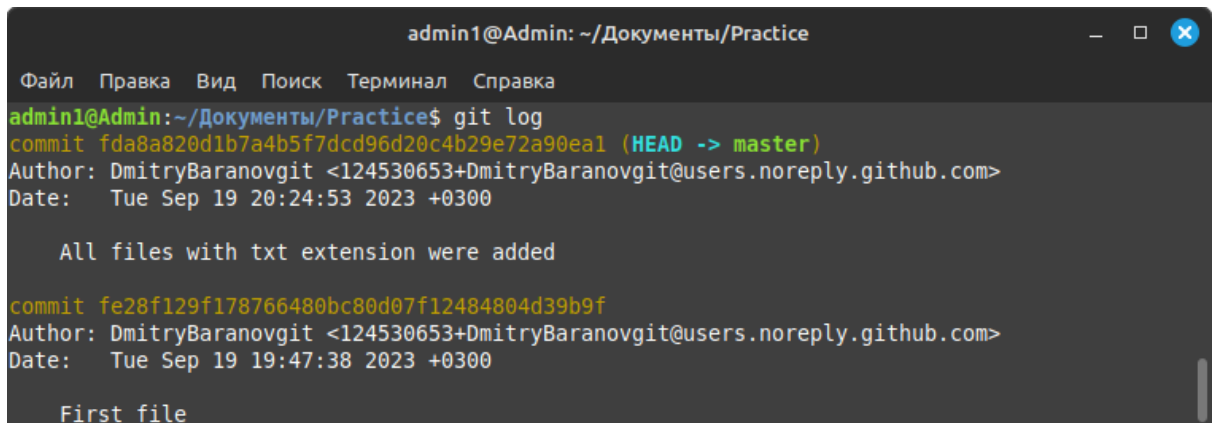
20. `git commit -m "All files with txt extension were added"` - Фиксируем статус файлов.



```
admin1@Admin: ~/Документы/Practice
Файл  Правка  Вид  Поиск  Терминал  Справка
admin1@Admin:~/Документы/Practice$ git commit -m "All files with txt extension were added"
[master fda8a82] All files with txt extension were added
3 files changed, 3 insertions(+), 1 deletion(-)
create mode 100644 readme after.txt
create mode 100644 readme now.txt
```

Рисунок 23.

21. `git log` - Показывает историю всех commit в обратном хронологическом порядке.



```
admin1@Admin: ~/Документы/Practice
Файл  Правка  Вид  Поиск  Терминал  Справка
admin1@Admin:~/Документы/Practice$ git log
commit fda8a820d1b7a4b5f7dcd96d20c4b29e72a90ea1 (HEAD -> master)
Author: DmitryBaranovgit <124530653+DmitryBaranovgit@users.noreply.github.com>
Date:   Tue Sep 19 20:24:53 2023 +0300

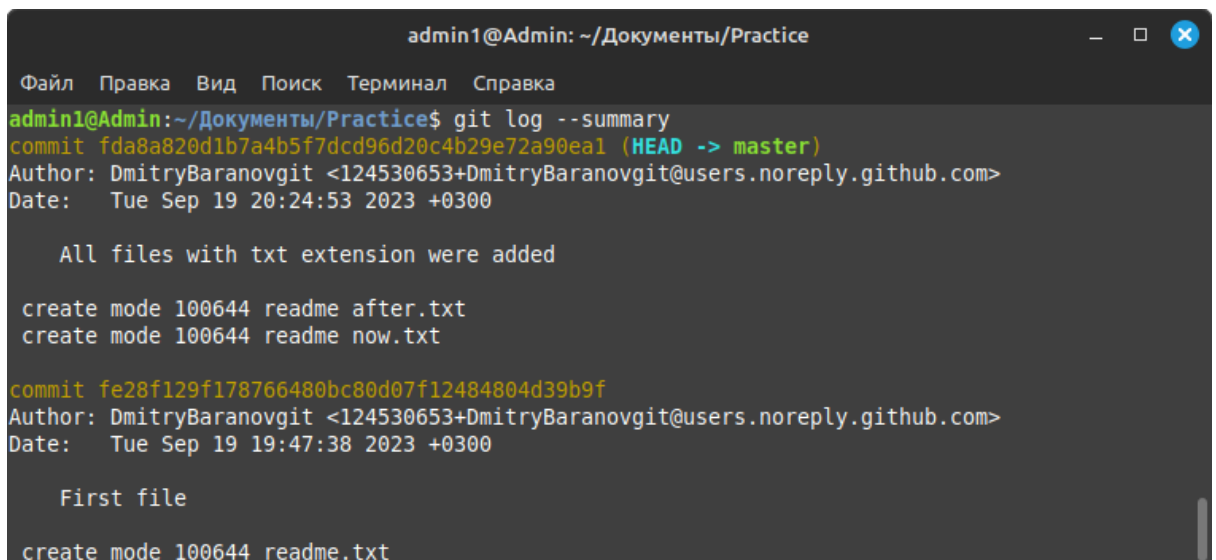
    All files with txt extension were added

commit fe28f129f178766480bc80d07f12484804d39b9f
Author: DmitryBaranovgit <124530653+DmitryBaranovgit@users.noreply.github.com>
Date:   Tue Sep 19 19:47:38 2023 +0300

    First file
```

Рисунок 24.

22. `git log --summary` - Показывает подробную историю всех изменений.



```
admin1@Admin: ~/Документы/Practice
Файл  Правка  Вид  Поиск  Терминал  Справка
admin1@Admin:~/Документы/Practice$ git log --summary
commit fda8a820d1b7a4b5f7dcd96d20c4b29e72a90ea1 (HEAD -> master)
Author: DmitryBaranovgit <124530653+DmitryBaranovgit@users.noreply.github.com>
Date:   Tue Sep 19 20:24:53 2023 +0300

    All files with txt extension were added

    create mode 100644 readme after.txt
    create mode 100644 readme now.txt

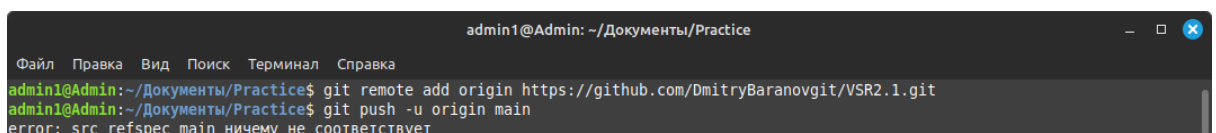
commit fe28f129f178766480bc80d07f12484804d39b9f
Author: DmitryBaranovgit <124530653+DmitryBaranovgit@users.noreply.github.com>
Date:   Tue Sep 19 19:47:38 2023 +0300

    First file

    create mode 100644 readme.txt
```

Рисунок 25.

23. `git remote add origin https://github.com/DmitryBaranovgit/VSR2.1.git` - Добавляет удалённый репозиторий.



```
admin1@Admin: ~/Документы/Practice
Файл  Правка  Вид  Поиск  Терминал  Справка
admin1@Admin:~/Документы/Practice$ git remote add origin https://github.com/DmitryBaranovgit/VSR2.1.git
admin1@Admin:~/Документы/Practice$ git push -u origin main
error: src refspec main ничему не соответствует
```

Рисунок 26.

24. Кликнув по аватарке в правом верхнем углу зайти в настройки GitHub.

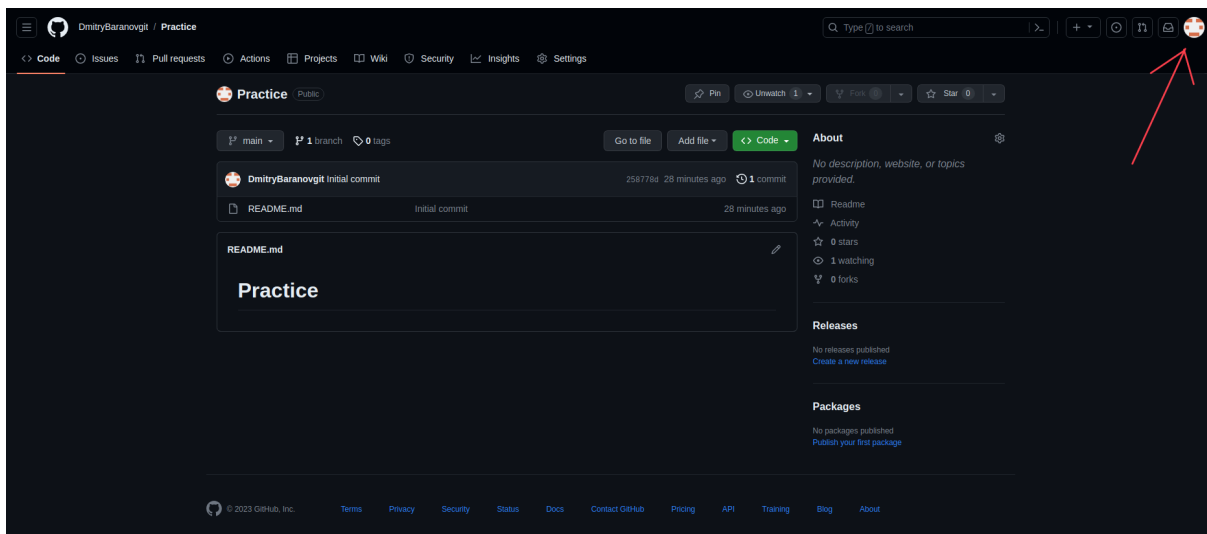


Рисунок 27.

Настройки.

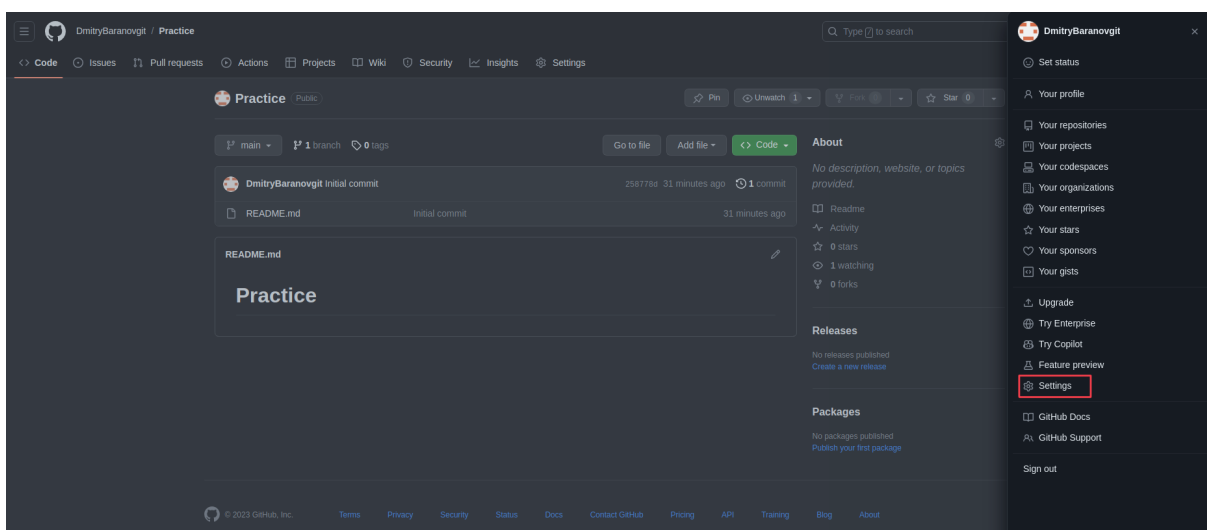


Рисунок 28.

25. Затем открыть настройки для разработчиков.

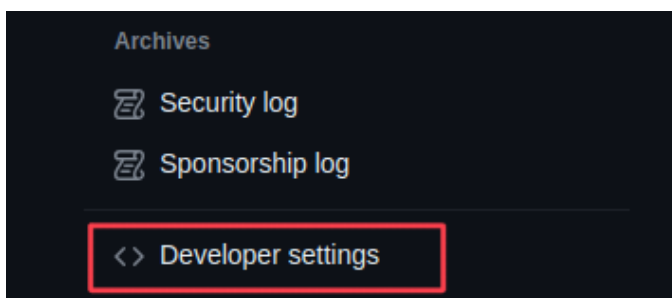


Рисунок 29.

Сгенерировать ключ.

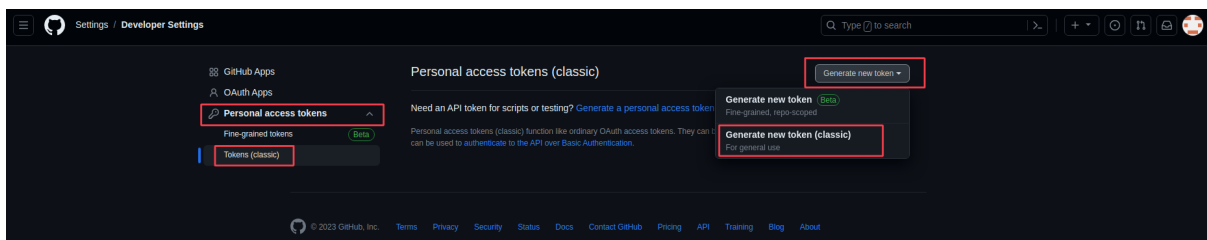


Рисунок 30.

Задать параметры.

Отметить операции с репозиторием.

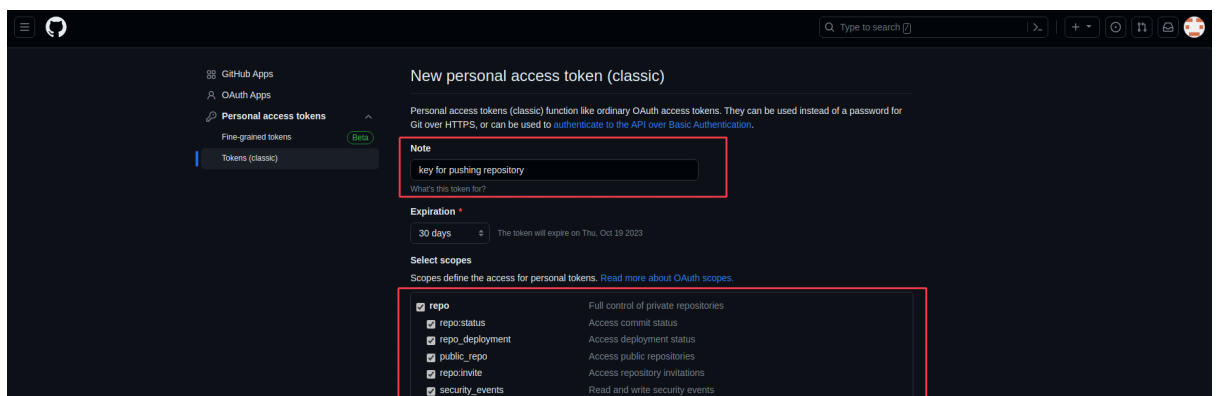


Рисунок 31.

Отметить операции администратора.

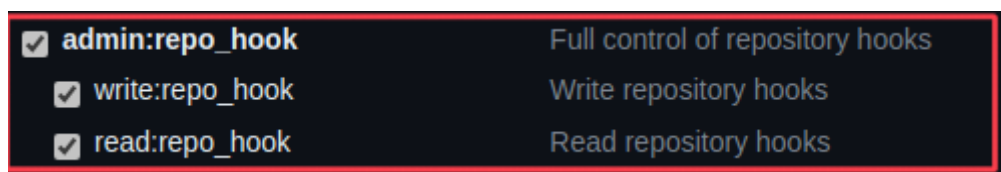


Рисунок 32.

Отметить удаление репозитория.



Рисунок 33.

Нажать создать ключ и скопировать сам ключ.

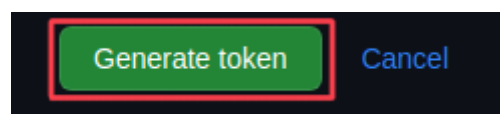
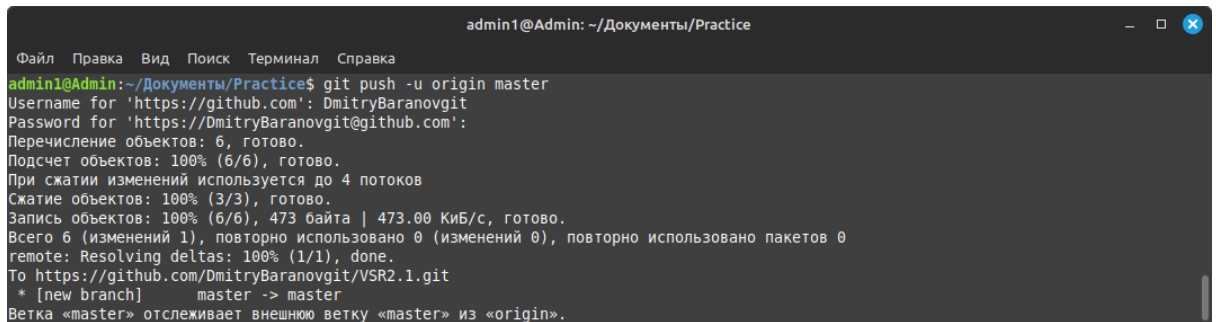


Рисунок 34.

26. `git push -u origin` + название ветки - Выгрузка в удалённый репозиторий.



```
admin1@Admin: ~/Документы/Practice
Файл  Правка  Вид  Поиск  Терминал  Справка
admin1@Admin:~/Документы/Practice$ git push -u origin master
Username for 'https://github.com': DmitryBaranovgit
Password for 'https://DmitryBaranovgit@github.com':
Перечисление объектов: 6, готово.
Подсчет объектов: 100% (6/6), готово.
При сжатии изменений используется до 4 потоков
Сжатие объектов: 100% (3/3), готово.
Запись объектов: 100% (6/6), 473 байта | 473.00 КиБ/с, готово.
Всего 6 (изменений 1), повторно использовано 0 (изменений 0), повторно использовано пакетов 0
remote: Resolving deltas: 100% (1/1), done.
To https://github.com/DmitryBaranovgit/VSR2.1.git
 * [new branch]      master -> master
Ветка «master» отслеживает внешнюю ветку «master» из «origin».
```

Рисунок 35.

27. Смотрим выгруженные файлы в GitHub.

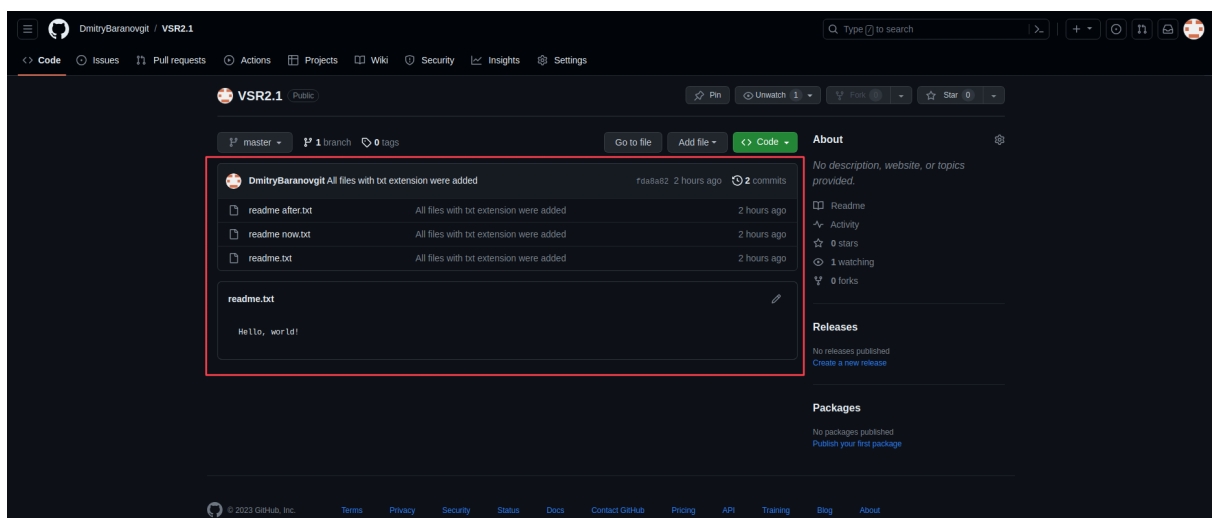


Рисунок 36.

28. Изменим в браузере readme.txt, нажав на специальный значок.

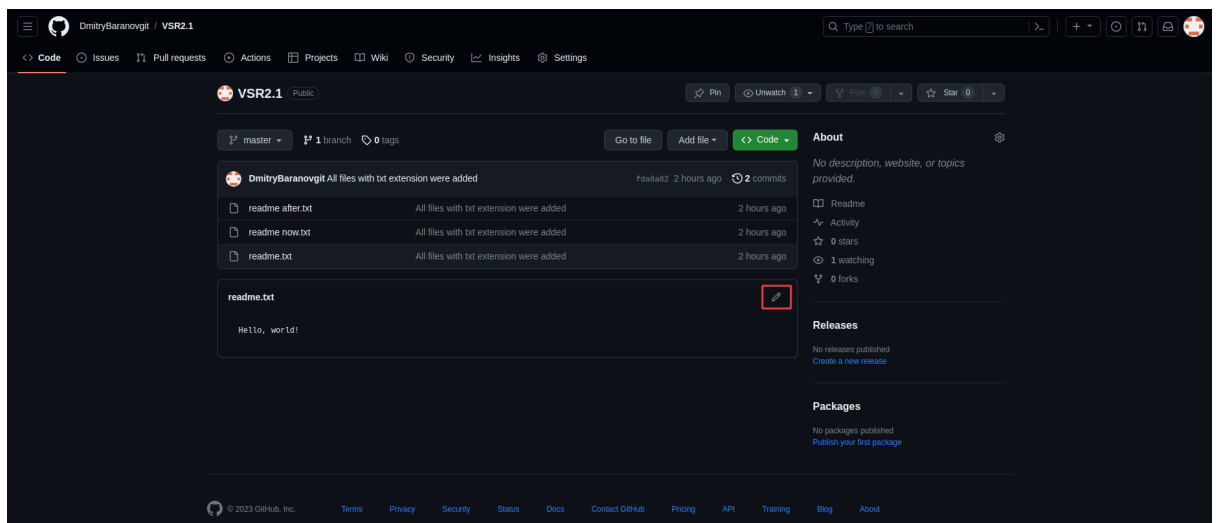


Рисунок 37.

Напишем ещё 2 раза Hello, world! и сохраним изменения.

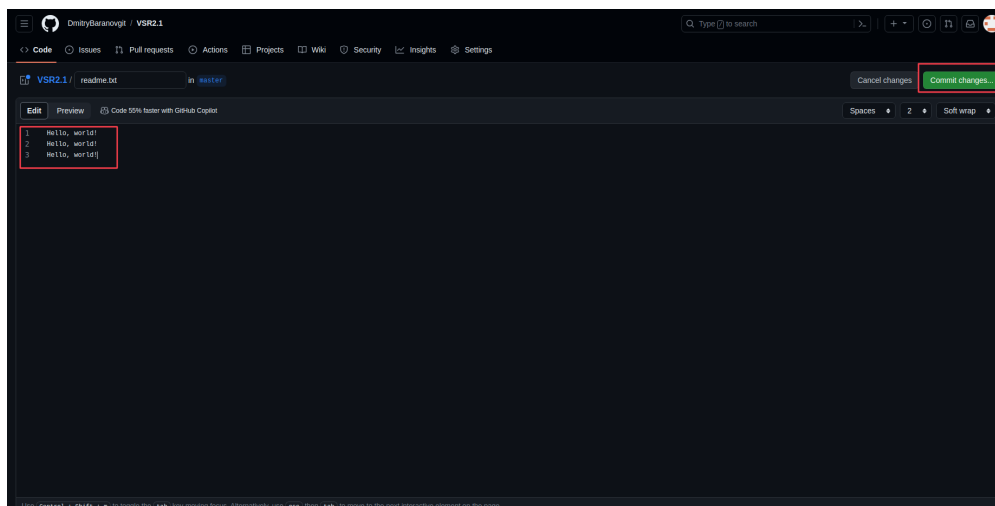


Рисунок 38.

29. git pull origin master - Позволяет посмотреть сделанные изменения в удаленном репозитории. Добавлено две новые строки.

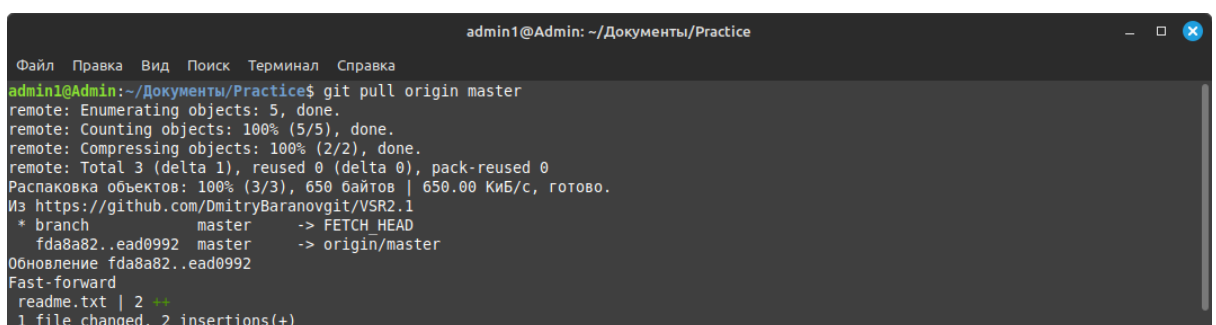
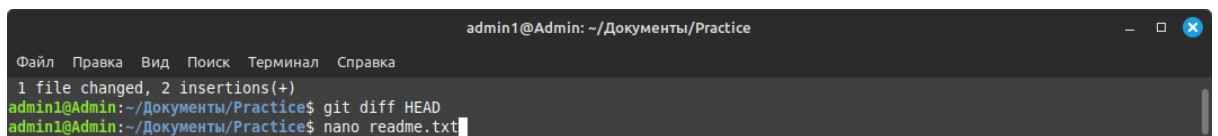


Рисунок 39.

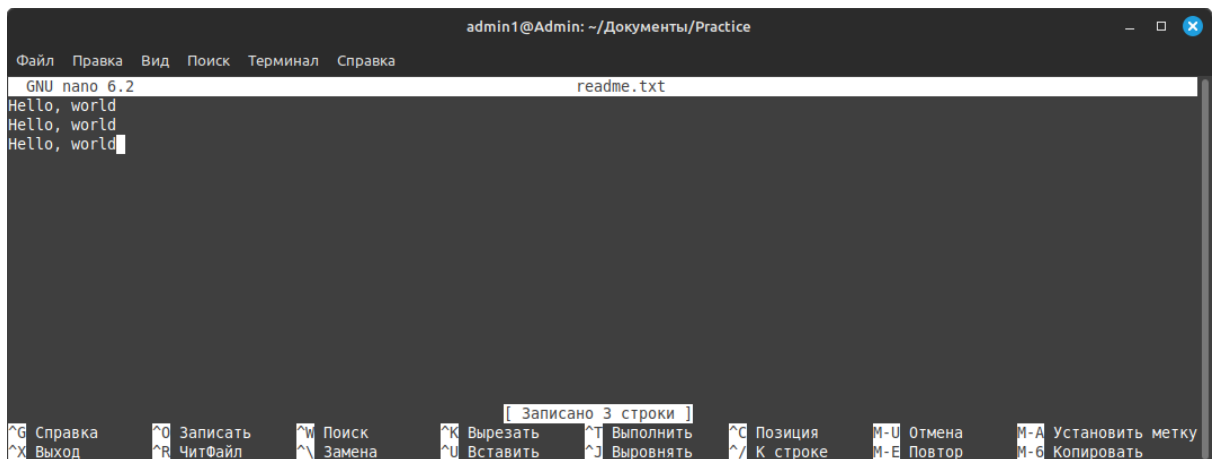
30. nano readme.txt - Добавим изменения в файл



```
admin1@Admin: ~/Документы/Practice
Файл  Правка  Вид  Поиск  Терминал  Справка
1 file changed, 2 insertions(+)
admin1@Admin:~/Документы/Practice$ git diff HEAD
admin1@Admin:~/Документы/Practice$ nano readme.txt
```

Рисунок 40.

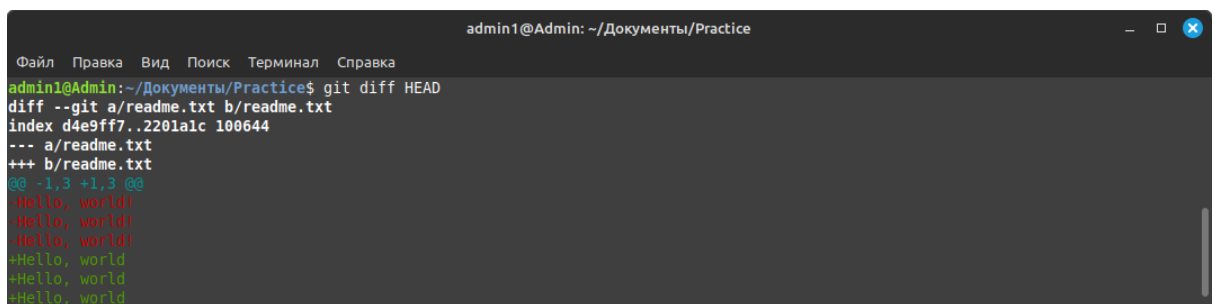
Изменяем файл. Удаляем ! знаки.



```
admin1@Admin: ~/Документы/Practice
Файл  Правка  Вид  Поиск  Терминал  Справка
GNU nano 6.2                                readme.txt
Hello, world
Hello, world
Hello, world
[ Записано 3 строки ]
^G Справка      ^O Записать     ^W Поиск       ^K Вырезать    ^T Выполнить   ^C Позиция     M-U Отмена     M-A Установить метку
^X Выход        ^R ЧитФайл     ^N Замена      ^U Вставить    ^J Выровнять   ^/_ К строке    M-E Повтор     M-G Копировать
```

Рисунок 41.

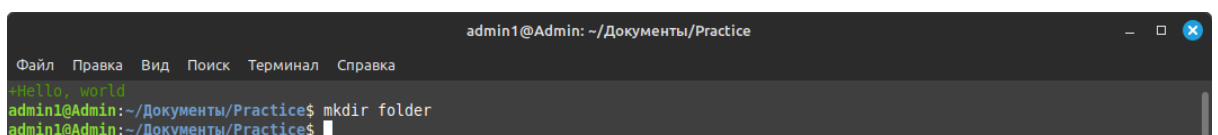
31. git diff HEAD - Позволяет увидеть изменения. HEAD - указатель на последнюю зафиксированную версию.



```
admin1@Admin: ~/Документы/Practice
Файл  Правка  Вид  Поиск  Терминал  Справка
admin1@Admin:~/Документы/Practice$ git diff HEAD
diff --git a/readme.txt b/readme.txt
index d4e9ff7..2201a1c 100644
--- a/readme.txt
+++ b/readme.txt
@@ -1,3 +1,3 @@
+Hello, world!
+Hello, world!
+Hello, world!
+Hello, world
+Hello, world
+Hello, world
```

Рисунок 42.

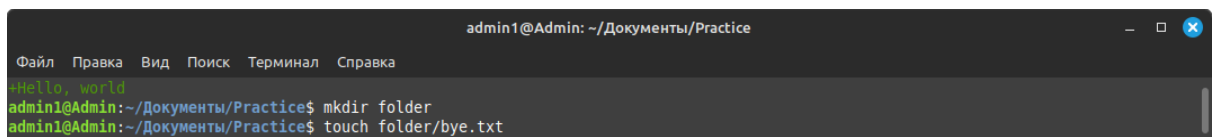
32. mkdir folder - Добавляет каталог folder.



```
admin1@Admin: ~/Документы/Practice
Файл  Правка  Вид  Поиск  Терминал  Справка
+Hello, world
admin1@Admin:~/Документы/Practice$ mkdir folder
admin1@Admin:~/Документы/Practice$
```

Рисунок 43.

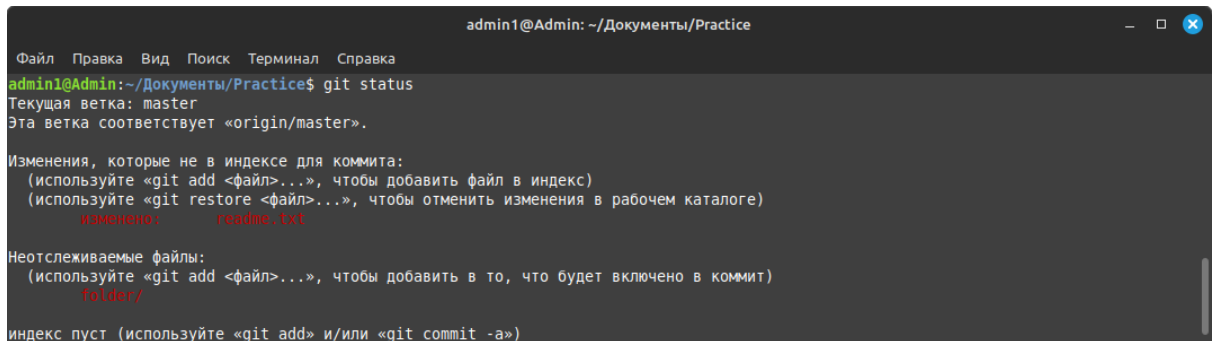
33. touch folder/bye.txt - Добавляет файл bye.txt в каталог folder.



```
admin1@Admin: ~/Документы/Practice
Файл  Правка  Вид  Поиск  Терминал  Справка
+Hello, world
admin1@Admin:~/Документы/Practice$ mkdir folder
admin1@Admin:~/Документы/Practice$ touch folder/bye.txt
```

Рисунок 44.

34. git status - Показывает статус. Каталог сделан, но не отслеживается в репозитории.



```
admin1@Admin: ~/Документы/Practice
Файл  Правка  Вид  Поиск  Терминал  Справка
admin1@Admin:~/Документы/Practice$ git status
Текущая ветка: master
Эта ветка соответствует «origin/master».

Изменения, которые не в индексе для коммита:
  (используйте «git add <файл>...», чтобы добавить файл в индекс)
  (используйте «git restore <файл>...», чтобы отменить изменения в рабочем каталоге)
      изменено:   readme.txt

Неотслеживаемые файлы:
  (используйте «git add <файл>...», чтобы добавить в то, что будет включено в коммит)
      folder/

индекс пуст (используйте «git add» и/или «git commit -a»)
```

Рисунок 45.

Посмотрим каталог.

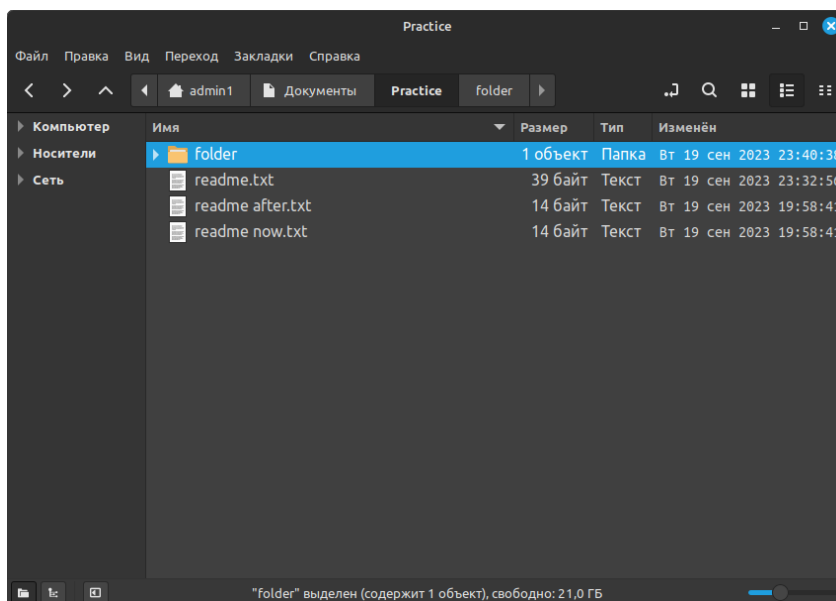


Рисунок 46.

И файл.

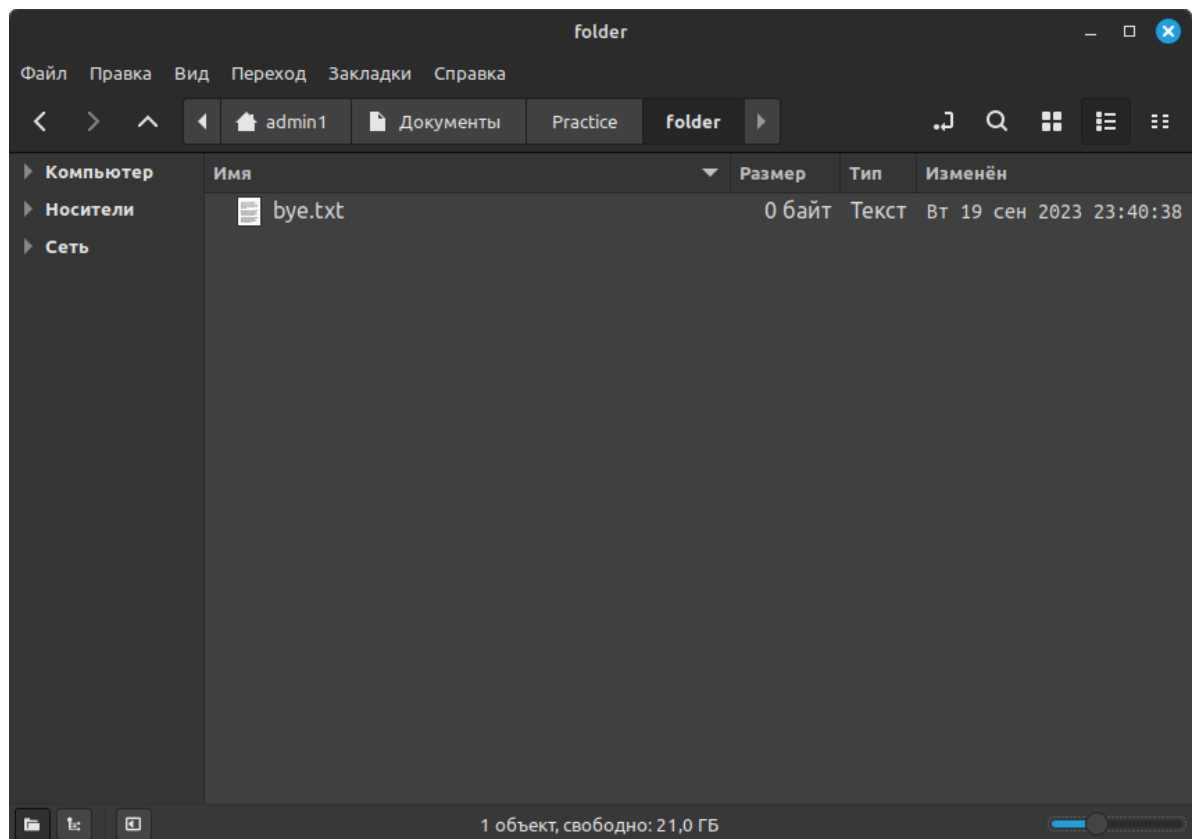


Рисунок 47.

35. `git add folder/bye.txt` - Добавляет каталог и файл в отслеживаемые.

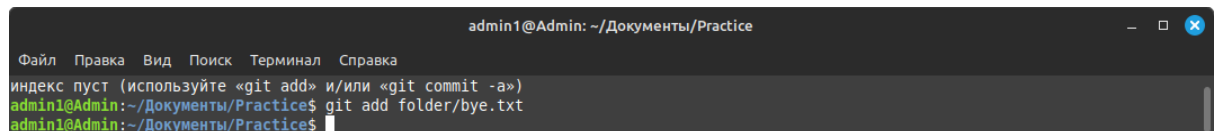


Рисунок 48.

36. `git status` - Статус файла.

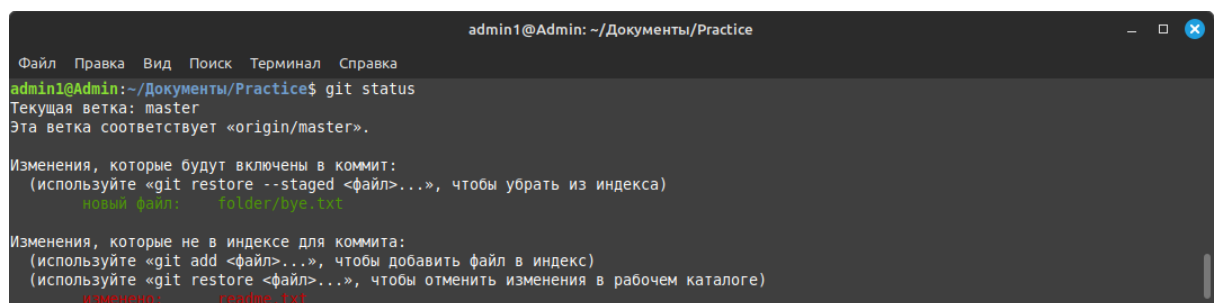
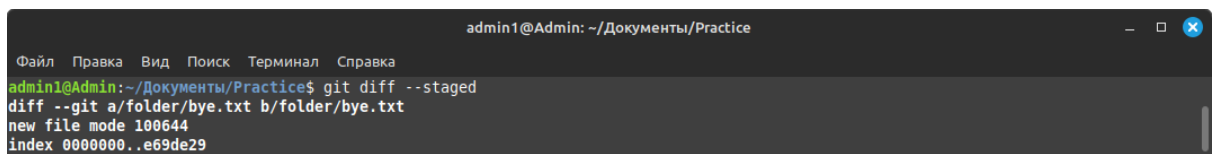


Рисунок 49.

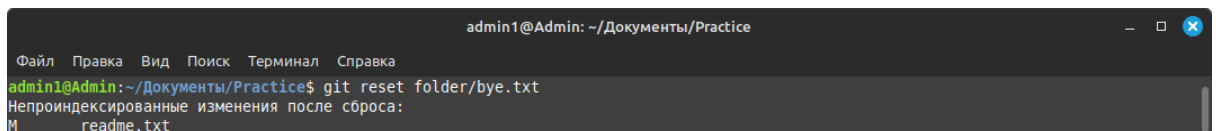
37. `git diff --staged` - Позволяет посмотреть изменения в стадии staged.



```
admin1@Admin: ~/Документы/Practice
Файл  Правка  Вид  Поиск  Терминал  Справка
admin1@Admin:~/Документы/Practice$ git diff --staged
diff --git a/folder/bye.txt b/folder/bye.txt
new file mode 100644
index 0000000..e69de29
```

Рисунок 50.

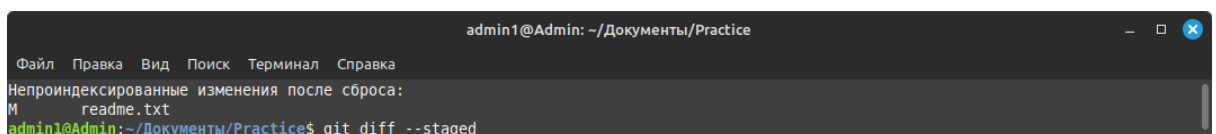
38. `git reset folder/bye.txt` - Отменяет действие.



```
admin1@Admin: ~/Документы/Practice
Файл  Правка  Вид  Поиск  Терминал  Справка
admin1@Admin:~/Документы/Practice$ git reset folder/bye.txt
Неприндексированные изменения после сброса:
M   readme.txt
```

Рисунок 51.

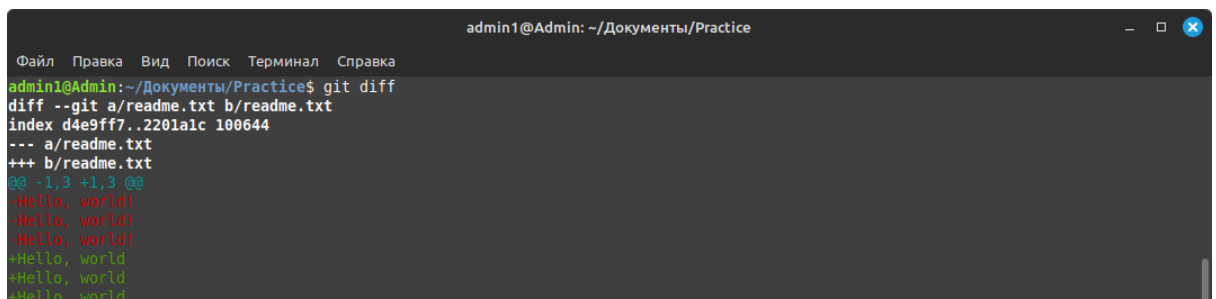
39. `git diff --staged` - Вернёт пустое значение.



```
admin1@Admin: ~/Документы/Practice
Файл  Правка  Вид  Поиск  Терминал  Справка
Неприндексированные изменения после сброса:
M   readme.txt
admin1@Admin:~/Документы/Practice$ git diff --staged
```

Рисунок 52.

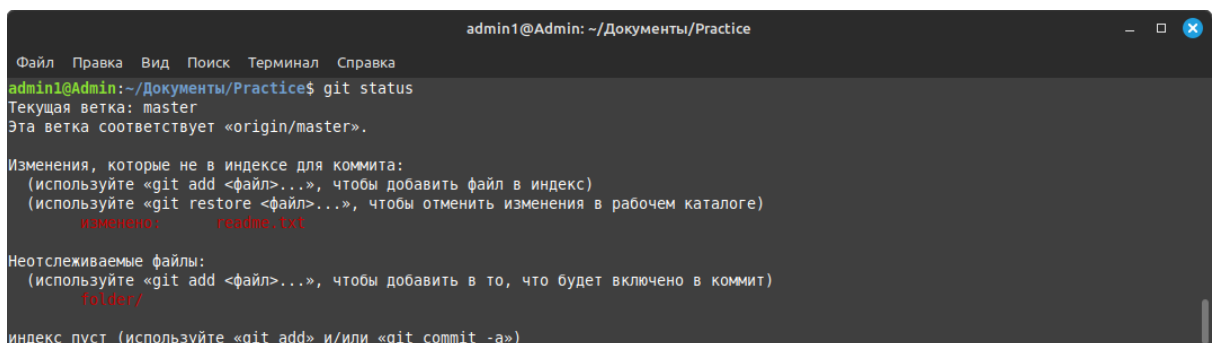
40. `git diff` - Показывает, что было без выполнения команды `git staged`.



```
admin1@Admin: ~/Документы/Practice
Файл  Правка  Вид  Поиск  Терминал  Справка
admin1@Admin:~/Документы/Practice$ git diff
diff --git a/readme.txt b/readme.txt
index d4e9ff7..2201alc 100644
--- a/readme.txt
+++ b/readme.txt
@@ -1,3 +1,3 @@
-Hello, world!
-Hello, world!
-Hello, world!
+Hello, world
+Hello, world
+Hello, world
```

Рисунок 53.

41. `git status` - Показывает, что снялось отслеживание с `readme.txt`.



```
admin1@Admin: ~/Документы/Practice
Файл  Правка  Вид  Поиск  Терминал  Справка
admin1@Admin:~/Документы/Practice$ git status
Текущая ветка: master
Эта ветка соответствует «origin/master».

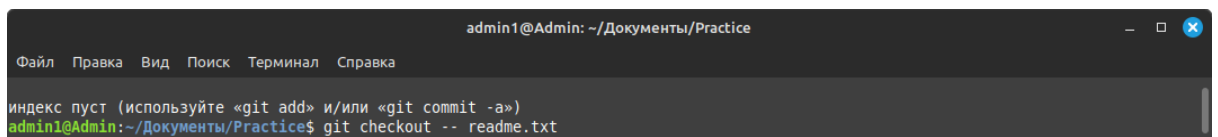
Изменения, которые не в индексе для коммита:
(используйте «git add <файл>...», чтобы добавить файл в индекс)
(используйте «git restore <файл>...», чтобы отменить изменения в рабочем каталоге)
изменено:   readme.txt

Неотслеживаемые файлы:
(используйте «git add <файл>...», чтобы добавить в то, что будет включено в коммит)
folder/

индекс пуст (используйте «git add» и/или «git commit -a»)
```

Рисунок 54.

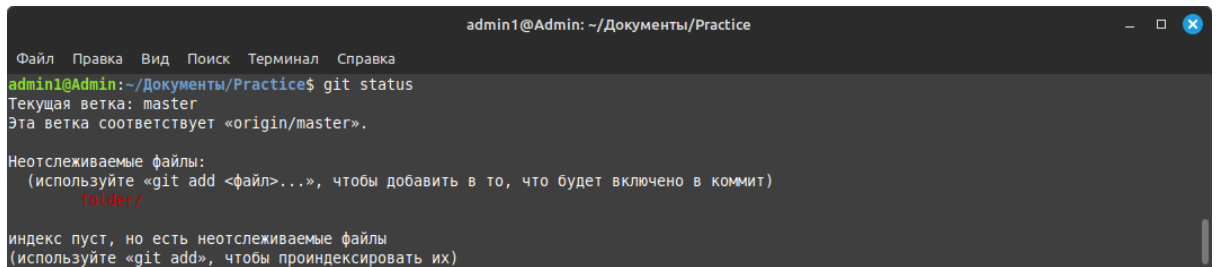
42. `git checkout -- readme.txt` - Возвращает изменения для файла `readme.txt`.



```
admin1@Admin: ~/Документы/Practice
Файл  Правка  Вид  Поиск  Терминал  Справка
индекс пуст (используйте «git add» и/или «git commit -a»)
admin1@Admin:~/Документы/Practice$ git checkout -- readme.txt
```

Рисунок 55.

43. `git status` - Показывает, что последние изменения файла убраны.



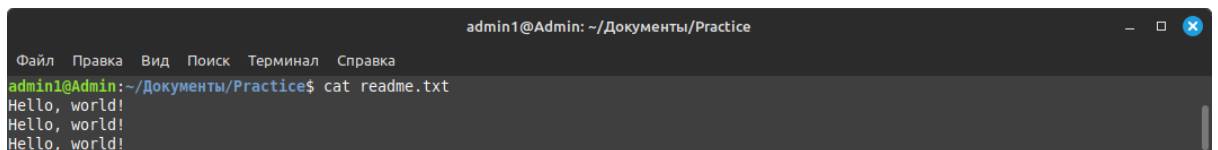
```
admin1@Admin: ~/Документы/Practice
Файл  Правка  Вид  Поиск  Терминал  Справка
admin1@Admin:~/Документы/Practice$ git status
Текущая ветка: master
Эта ветка соответствует «origin/master».

Неотслеживаемые файлы:
  (используйте «git add <файл>...», чтобы добавить в то, что будет включено в коммит)
  folder/

индекс пуст, но есть неотслеживаемые файлы
(используйте «git add», чтобы проиндексировать их)
```

Рисунок 56.

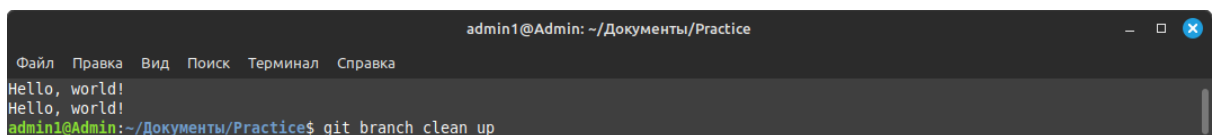
44. `cat readme.txt` - Показывает файл до изменений.



```
admin1@Admin: ~/Документы/Practice
Файл  Правка  Вид  Поиск  Терминал  Справка
admin1@Admin:~/Документы/Practice$ cat readme.txt
Hello, world!
Hello, world!
Hello, world!
```

Рисунок 57.

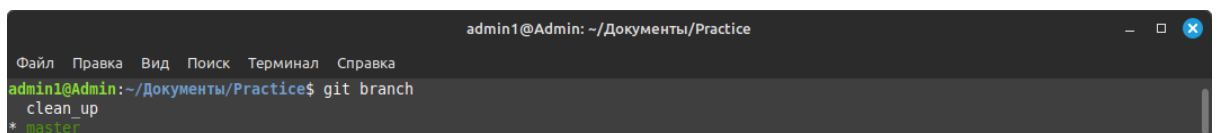
45. `git branch clean_up` - Создает ветку `clean_up`.



```
admin1@Admin: ~/Документы/Practice
Файл  Правка  Вид  Поиск  Терминал  Справка
Hello, world!
Hello, world!
admin1@Admin:~/Документы/Practice$ git branch clean_up
```

Рисунок 58.

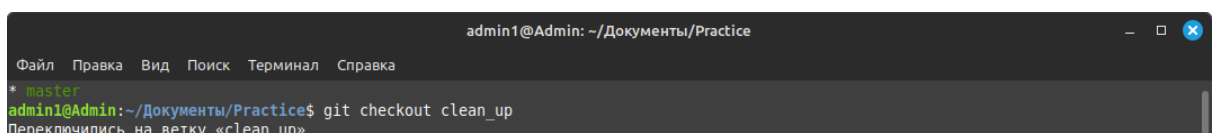
46. `git branch` - Показывает в какой ветке идет работа.



```
admin1@Admin: ~/Документы/Practice
Файл  Правка  Вид  Поиск  Терминал  Справка
admin1@Admin:~/Документы/Practice$ git branch
clean_up
* master
```

Рисунок 59.

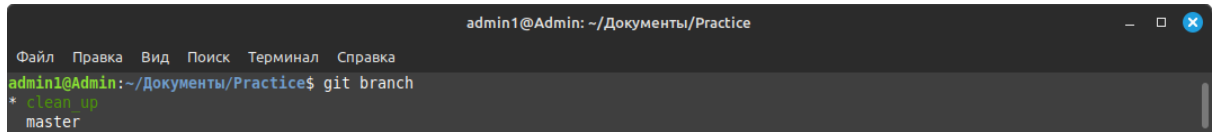
47. `git checkout clean_up` - Переключает на ветку `clean_up`.



```
admin1@Admin: ~/Документы/Practice
Файл  Правка  Вид  Поиск  Терминал  Справка
* master
admin1@Admin:~/Документы/Practice$ git checkout clean_up
Переключились на ветку «clean_up»
```

Рисунок 60.

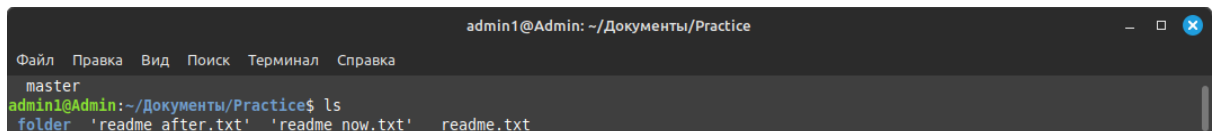
48. `git branch` - Показывает, что переключилось на ветку `clean_up`.



```
admin1@Admin: ~/Документы/Practice
Файл  Правка  Вид  Поиск  Терминал  Справка
admin1@Admin:~/Документы/Practice$ git branch
* clean_up
master
```

Рисунок 61.

49. `ls` - Показывает каталог.



```
admin1@Admin: ~/Документы/Practice
Файл  Правка  Вид  Поиск  Терминал  Справка
admin1@Admin:~/Документы/Practice$ ls
master
folder  'readme after.txt'  'readme now.txt'  readme.txt
```

Рисунок 62.

Каталог.

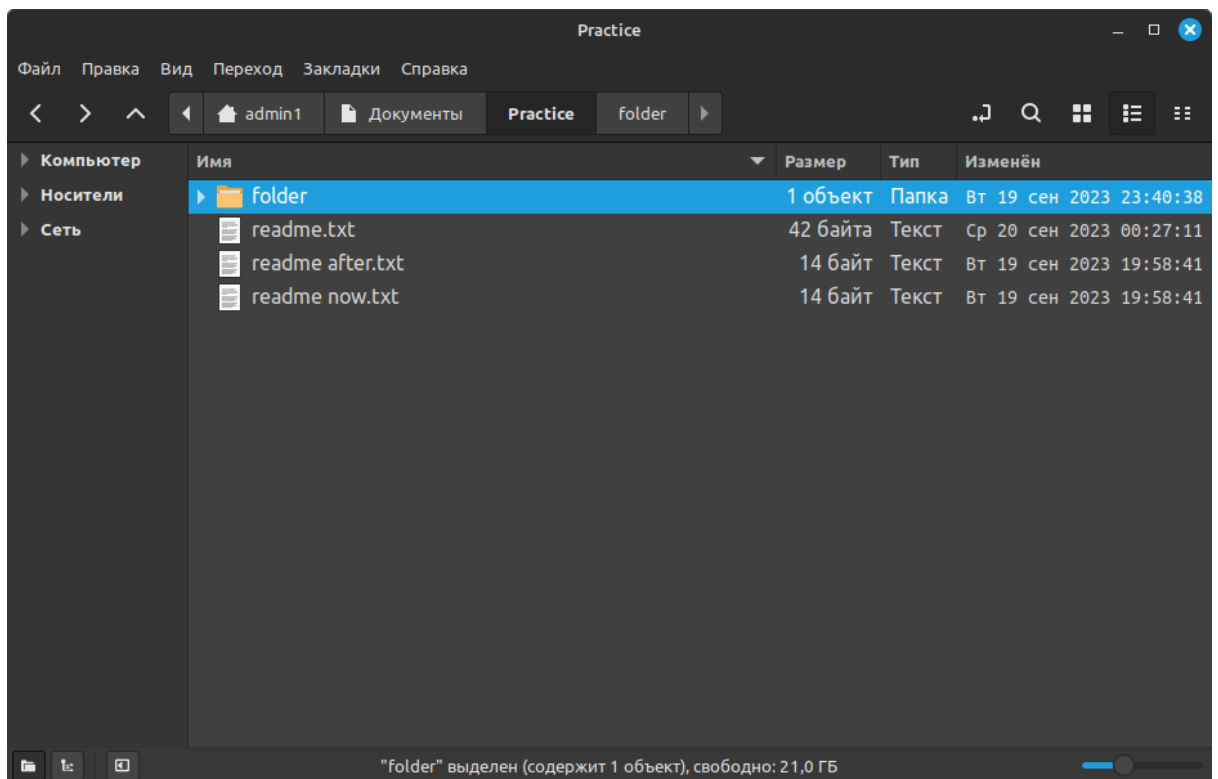


Рисунок 63.

Файл каталога.

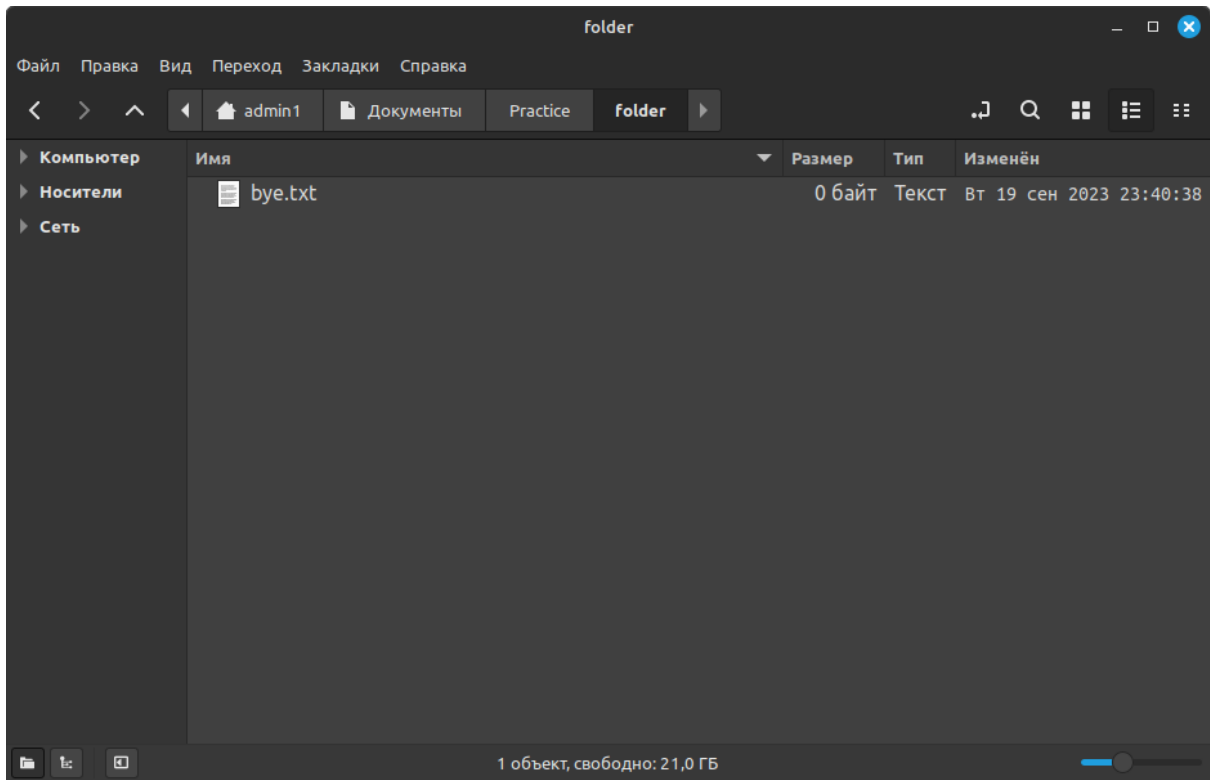


Рисунок 64.

50. `rm -r folder` - Удаляет каталог `folder` и файлы в нём.

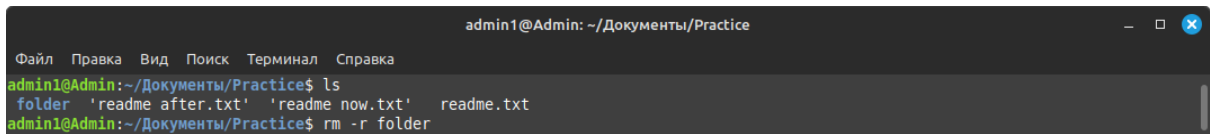


Рисунок 65.

Каталог.

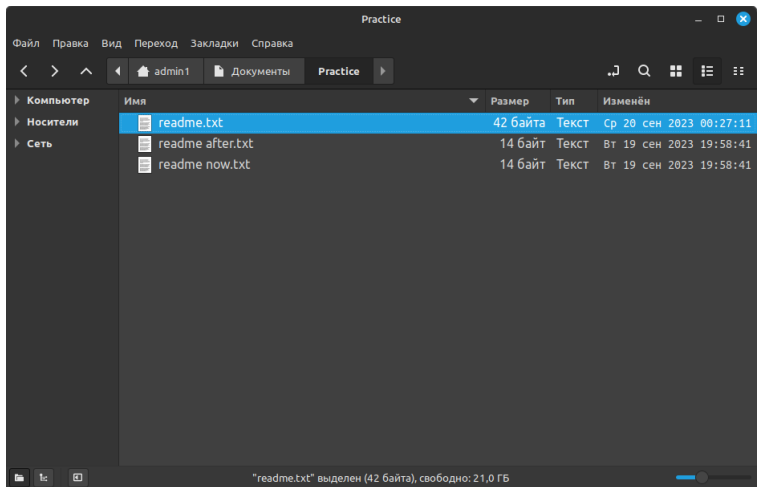
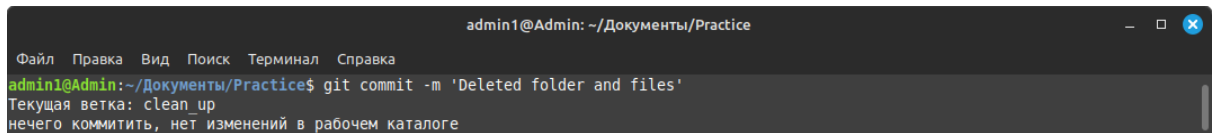


Рисунок 66.

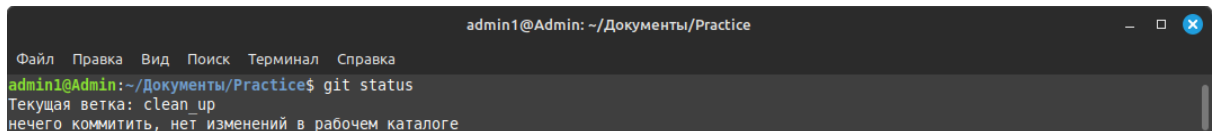
51. `git commit -m 'Deleted folder and files'` - Фиксирует изменения.



```
admin1@Admin: ~/Документы/Practice
Файл  Правка  Вид  Поиск  Терминал  Справка
admin1@Admin:~/Документы/Practice$ git commit -m 'Deleted folder and files'
Текущая ветка: clean_up
нечего коммитить, нет изменений в рабочем каталоге
```

Рисунок 67.

52. `git status` - Статус репозитория.



```
admin1@Admin: ~/Документы/Practice
Файл  Правка  Вид  Поиск  Терминал  Справка
admin1@Admin:~/Документы/Practice$ git status
Текущая ветка: clean_up
нечего коммитить, нет изменений в рабочем каталоге
```

Рисунок 68.

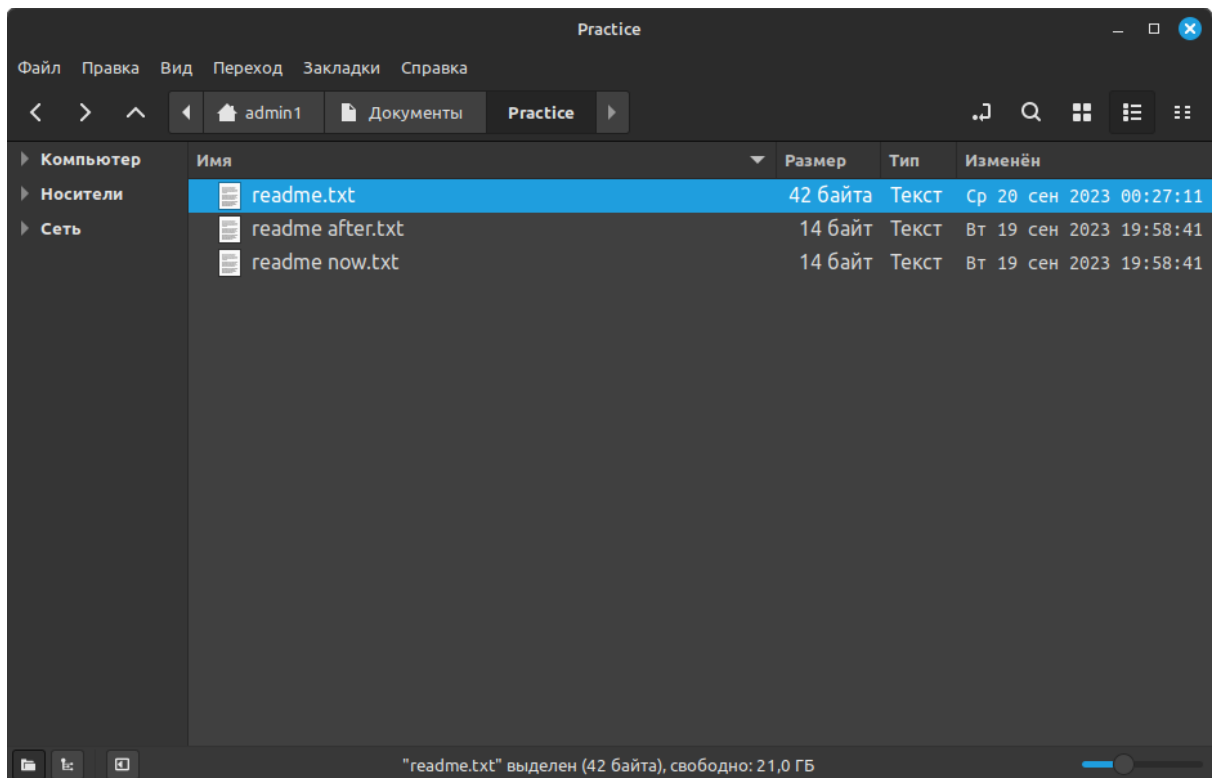
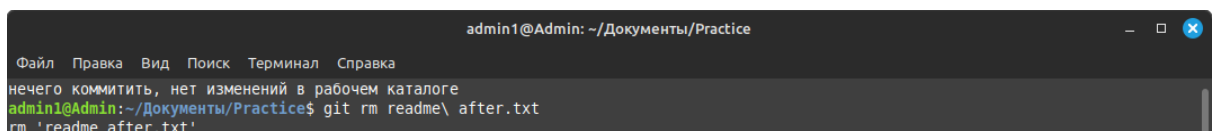


Рисунок 69.

53. `git rm readme\ after.txt` - Удаляет файл after.txt.



```
admin1@Admin: ~/Документы/Practice
Файл  Правка  Вид  Поиск  Терминал  Справка
нечего коммитить, нет изменений в рабочем каталоге
admin1@Admin:~/Документы/Practice$ git rm readme\ after.txt
rm 'readme after.txt'
```

Рисунок 70.

Каталог.

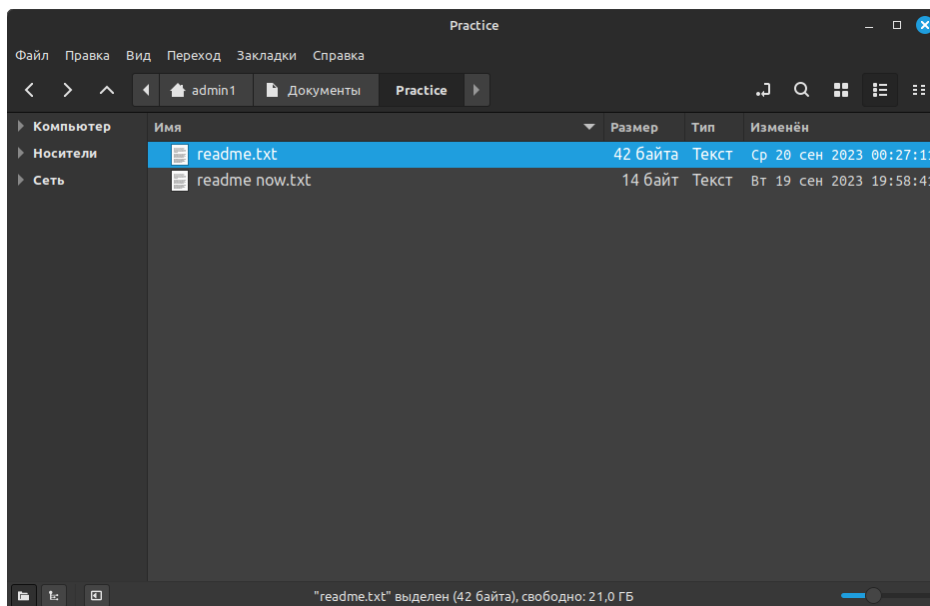


Рисунок 71.

54. `git rm readme\ now.txt` - Удаляет now.txt.

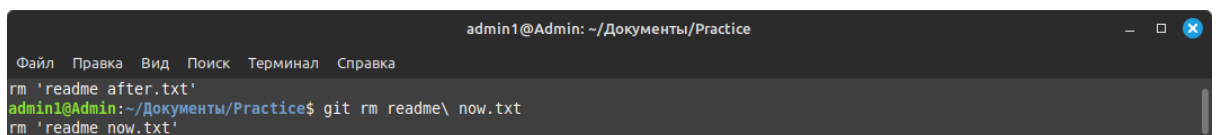


Рисунок 72.

Каталог.

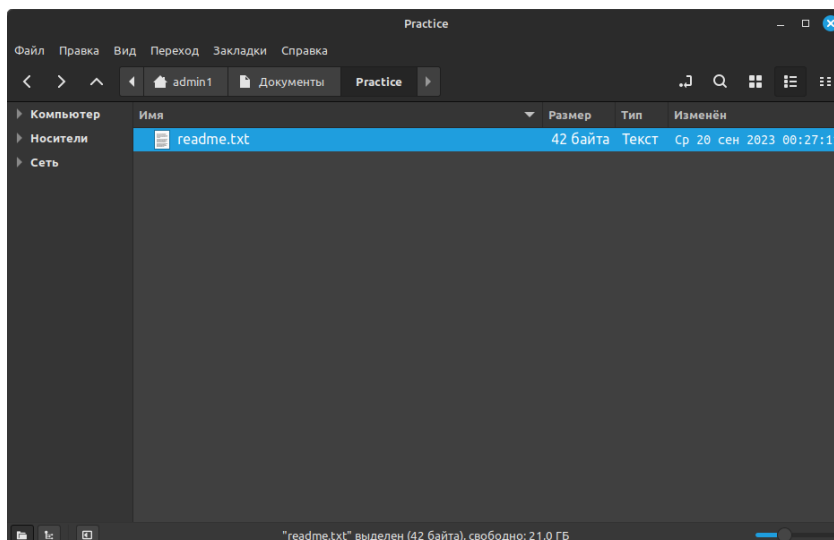
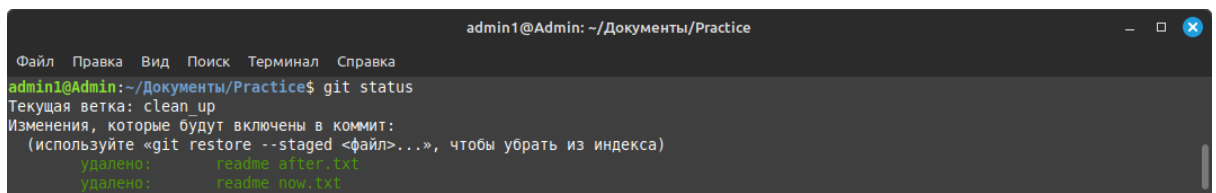


Рисунок 73.

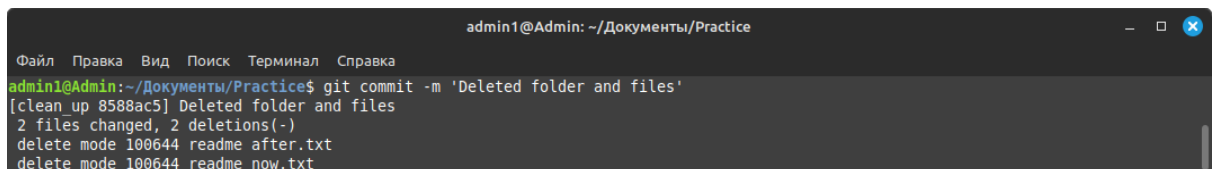
55. git status - Показывает, что файлы удалены.



```
admin1@Admin: ~/Документы/Practice
Файл  Правка  Вид  Поиск  Терминал  Справка
admin1@Admin:~/Документы/Practice$ git status
Текущая ветка: clean up
Изменения, которые будут включены в коммит:
(используйте «git restore --staged <файл>...», чтобы убрать из индекса)
удалено:      readme after.txt
удалено:      readme now.txt
```

Рисунок 74.

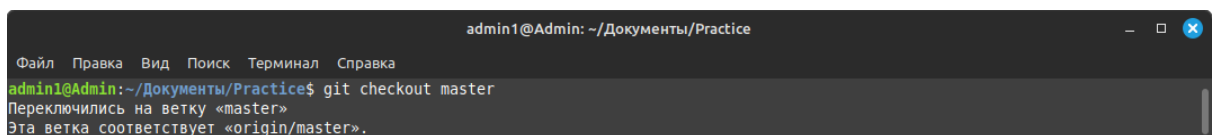
56. git commit -m 'Deleted folder and files' - Фиксируем, что файлы удалены из репозитория.



```
admin1@Admin: ~/Документы/Practice
Файл  Правка  Вид  Поиск  Терминал  Справка
admin1@Admin:~/Документы/Practice$ git commit -m 'Deleted folder and files'
[clean up 8588ac5] Deleted folder and files
2 files changed, 2 deletions(-)
delete mode 100644 readme after.txt
delete mode 100644 readme now.txt
```

Рисунок 75.

57. git checkout master - Переключение на ветку master.



```
admin1@Admin: ~/Документы/Practice
Файл  Правка  Вид  Поиск  Терминал  Справка
admin1@Admin:~/Документы/Practice$ git checkout master
Переключились на ветку «master»
Эта ветка соответствует «origin/master».
```

Рисунок 76.

Каталог.

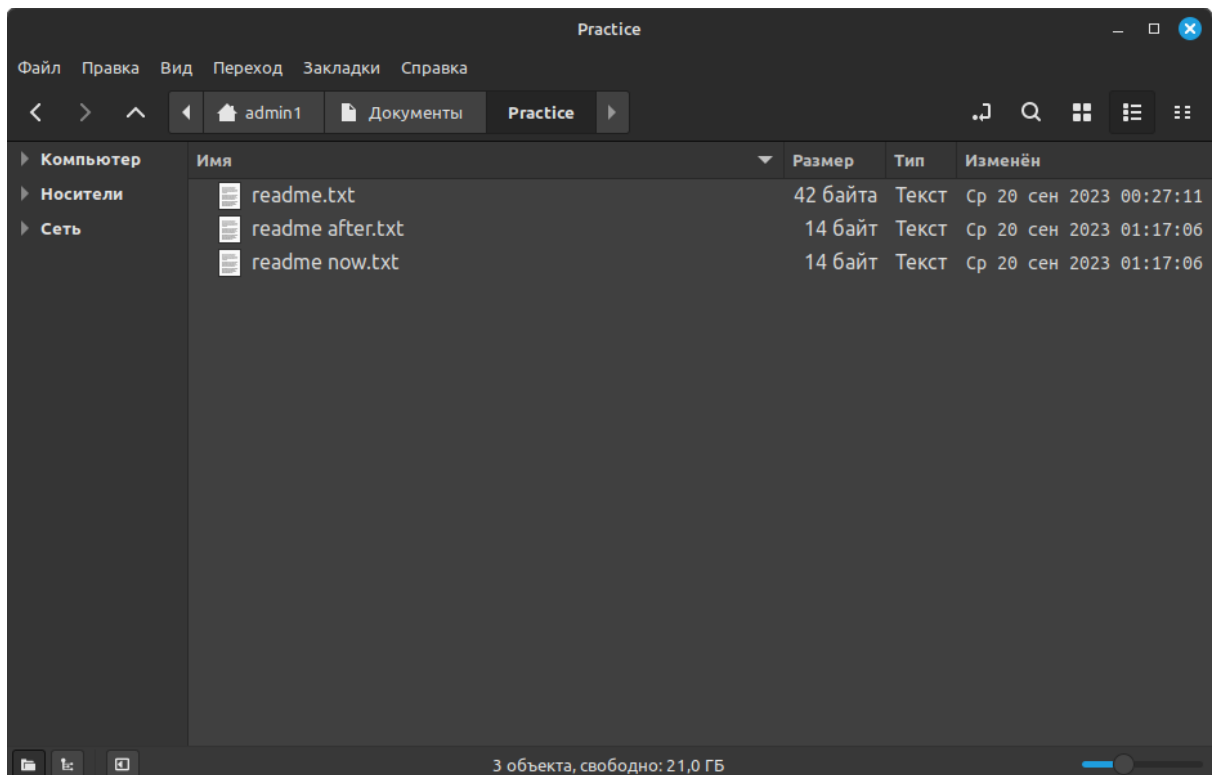
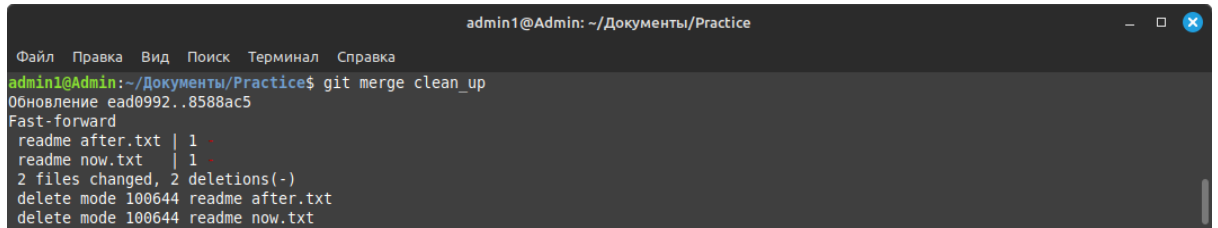


Рисунок 77.

58. `git merge clean_up` - Слияние с веткой `clean_up`.



```
admin1@Admin: ~/Документы/Practice
Файл  Правка  Вид  Поиск  Терминал  Справка
admin1@Admin:~/Документы/Practice$ git merge clean_up
Обновление ead0992..8588ac5
Fast-forward
 README after.txt | 1 +
 README now.txt   | 1 +
 2 files changed, 2 deletions(-)
 delete mode 100644 README after.txt
 delete mode 100644 README now.txt
```

Рисунок 78.

Каталог.

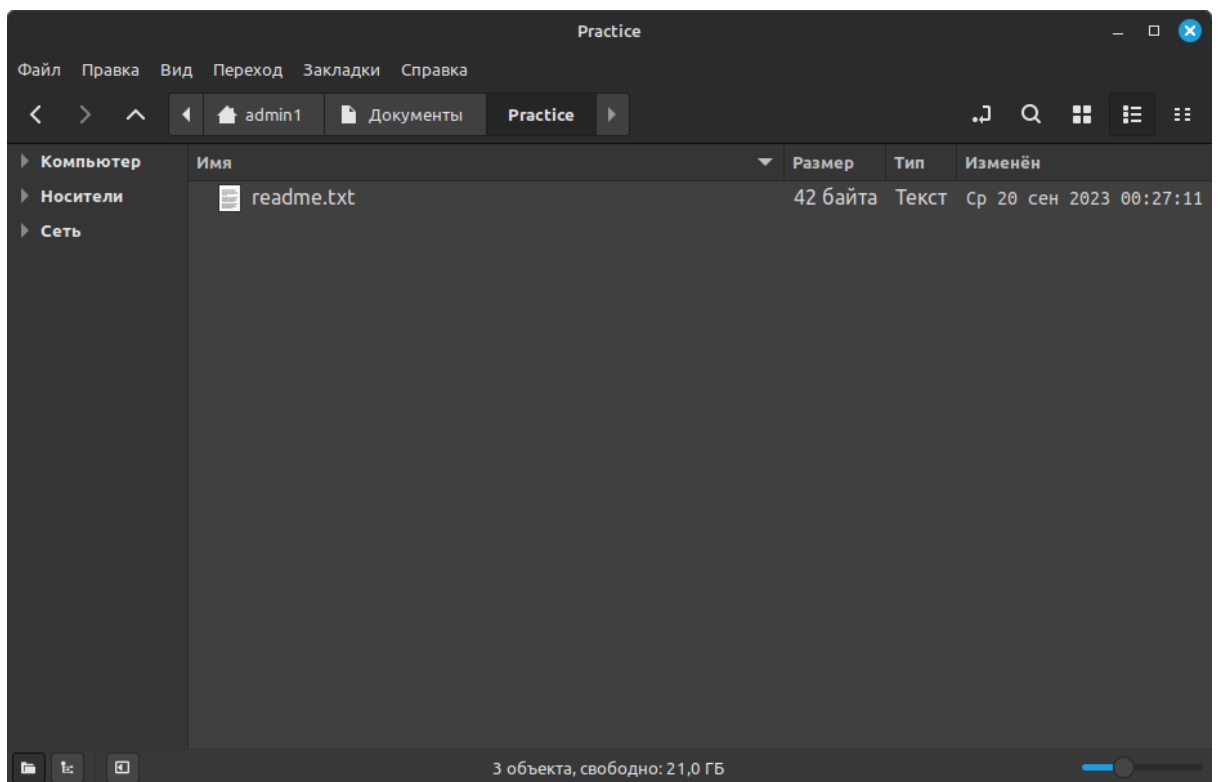
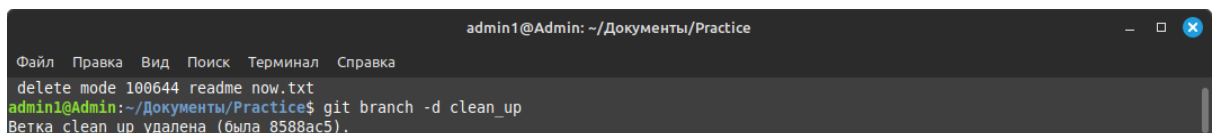


Рисунок 79.

59. `git branch -d clean_up` - Удаляем ветку `clean_up`.



```
admin1@Admin: ~/Документы/Practice
Файл  Правка  Вид  Поиск  Терминал  Справка
delete mode 100644 README now.txt
admin1@Admin:~/Документы/Practice$ git branch -d clean_up
Ветка clean up удалена (была 8588ac5).
```

Рисунок 80.

Каталог.

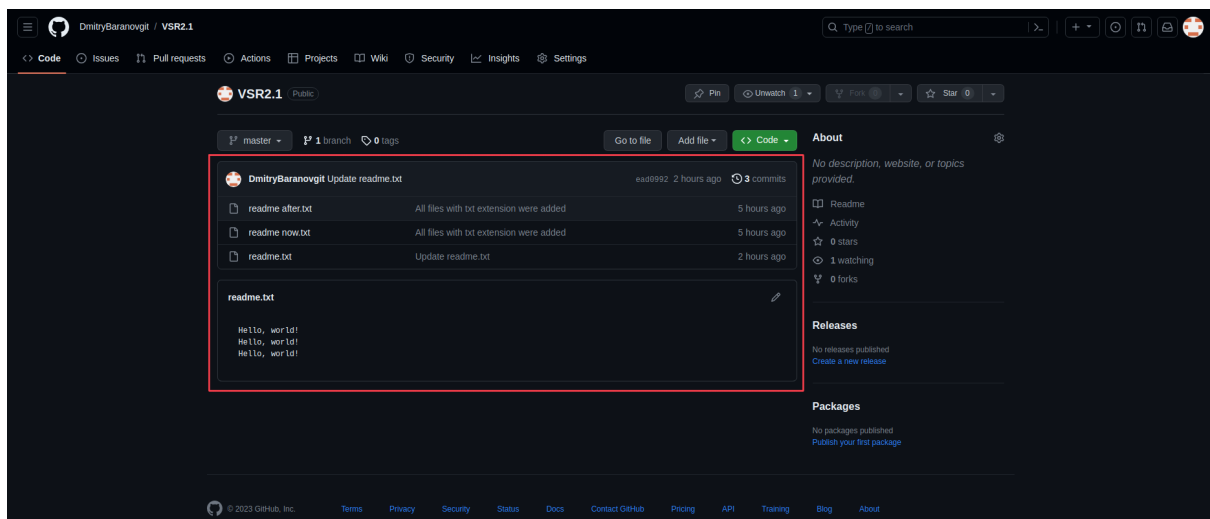


Рисунок 81.

60. git push - Выгружает последние изменения и показывает, что GitHub файлы были удалены.

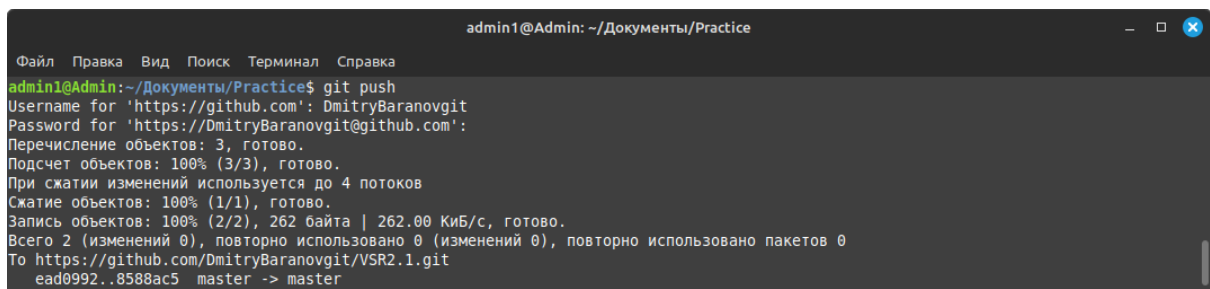


Рисунок 82.

GitHub. Последние изменения были добавлены.

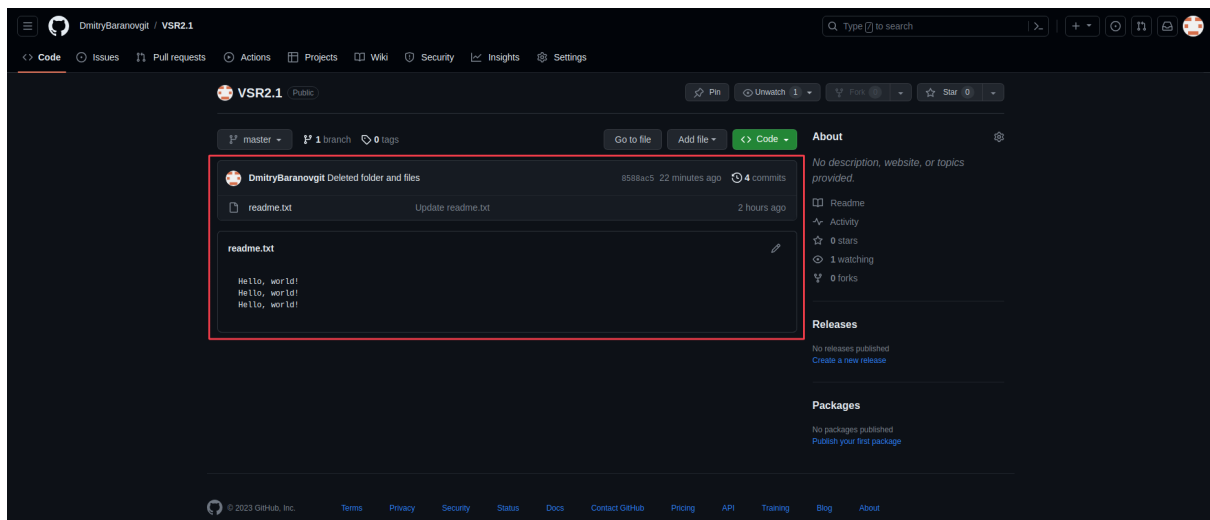


Рисунок 83.