

Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего образования
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт цифрового развития
Кафедра инфокоммуникаций

**«Исследование возможностей Git для работы с локальными
репозиториями»**

**ОТЧЕТ
по лабораторной работе №2
дисциплины
«Основы программной инженерии»**

Выполнил:
Мизин Глеб Егорович
2 курс, группа ПИЖ-б-о-21-1,
09.03.04 «Программная инженерия»,
направленность (профиль) «Разработка
и сопровождение программного
обеспечения», очная форма обучения

(подпись)

Проверил:

(подпись)

Отчет защищен с оценкой _____ Дата защиты _____

Ставрополь, 2022 г.

1. Просмотр истории коммитов

В git можно просматривать историю коммитов используя команду git log

```
C:\Users\student-09-525\Lab-1>git log
commit 121d2b62b84403c1bc9d5d845c58ed504555f23a <HEAD -> master, origin/master,
origin/HEAD>
Author: Glek <glebmizin1@gmail.com>
Date: Sat Sep 10 02:24:23 2022 +0300

    Загрузил отчёт

commit 1b19aae025066d99a3bb82c88cc9acd3c6514c0c
Author: Glek <glebmizin1@gmail.com>
Date: Sat Sep 10 01:29:27 2022 +0300

    Добавил папку для отчёта

commit ab515eea66067bb03cbbcb1de0fcd606306dac07
Author: Glek <glebmizin1@gmail.com>
Date: Fri Sep 9 23:24:38 2022 +0300

    добавил лицензию

commit 58f9611f6e5543779c3d57c86356aefe30039736
Author: Glek <glebmizin1@gmail.com>
Date: Fri Sep 9 23:10:36 2022 +0300

    I changed it again :)
:---skipping---
commit 121d2b62b84403c1bc9d5d845c58ed504555f23a <HEAD -> master, origin/master,
origin/HEAD>
Author: Glek <glebmizin1@gmail.com>
Date: Sat Sep 10 02:24:23 2022 +0300

    Загрузил отчёт

commit 1b19aae025066d99a3bb82c88cc9acd3c6514c0c
Author: Glek <glebmizin1@gmail.com>
Date: Sat Sep 10 01:29:27 2022 +0300

    Добавил папку для отчёта
```

Рисунок 1 – Использование команды git log

Так же можно добавлять параметры к запросу, так на следующем рисунке демонстрируется параметр -patch который отображает патч для каждого коммита, и параметр -2 при помощи которого можно вывести определенное количество коммитов

```

C:\Users\student-09-525\Lab-1>git log -p -2
commit 87668dab8426a6807f2ab680195add841d4435ec <HEAD -> master, origin/master,
origin/HEAD>
Author: Glek <glebmizin1@gmail.com>
Date: Sat Sep 10 11:55:00 2022 +0300

    Изменил proj.py

diff --git a/proj.py b/proj.py
index 05c5e59..fdb392a 100644
--- a/proj.py
+++ b/proj.py
@@ -1,1 @@
-print(Hello!!!!)
\ No newline at end of file
+print("Hello!!!!")
\ No newline at end of file

commit 81495839b8b84bbff36ac47c3109e3a7a164b3fe
Author: GlebMizin <112752101+GlebMizin@users.noreply.github.com>
Date: Sat Sep 10 11:50:04 2022 +0300

    Update LICENSE

diff --git a/LICENSE b/LICENSE
index 6cd6b9e..eb896de 100644
--- a/LICENSE
+++ b/LICENSE
@@ -1,6 +1,6 @@
 MIT License

```

Рисунок 2 – Использование команды git log с параметрами -p и -2

Так же можно использовать параметр --stat, чтобы отобразить статистику измененных файлов для каждого коммита.

```

C:\Users\student-09-525\Lab-1>git log --stat
commit 87668dab8426a6807f2ab680195add841d4435ec <HEAD -> master, origin/master,
origin/HEAD>
Author: Glek <glebmizin1@gmail.com>
Date: Sat Sep 10 11:55:00 2022 +0300

    Изменил proj.py

proj.py | 2 +-
1 file changed, 1 insertion(+), 1 deletion(-)

commit 81495839b8b84bbff36ac47c3109e3a7a164b3fe
Author: GlebMizin <112752101+GlebMizin@users.noreply.github.com>
Date: Sat Sep 10 11:50:04 2022 +0300

    Update LICENSE

LICENSE | 2 +-
1 file changed, 1 insertion(+), 1 deletion(-)

commit 121d2b62b84403c1bc9d5d845c58ed504555f23a
Author: Glek <glebmizin1@gmail.com>
Date: Sat Sep 10 02:24:23 2022 +0300

    Загрузил отчёт

...20\265\320\275\320\265\321\200\320\270\320\270.pdf" : Bin 0 -> 352865 bytes
1 file changed, 0 insertions(+), 0 deletions(-)

commit 1b19aae025066d99a3bb82c88cc9acd3c6514c0c
Author: Glek <glebmizin1@gmail.com>
Date: Sat Sep 10 01:29:27 2022 +0300

    Добавил папку для отчёта

```

Рисунок 3 – Использование команды git log с параметром --stat

Стоит отметить что в git есть возможность менять формат вывода, с использованием команды `--pretty`, которая имеет несколько опций `oneline`, `short`, `full` и `fuller`. При этом можно использовать опцию `format` которая позволяет вывести конкретную информацию.

```
C:\Users\student-09-525\Lab-1>git log --pretty=oneline
87668dab8426a6807f2ab680195add841d4435ec <HEAD -> master, origin/master, origin/HEAD> Изменил proj.py
81495839b8b84bbff36ac47c3109e3a7a164b3fe Update LICENSE
121d2b62b84403c1bc9d5d845c58ed504555f23a Загрузил отчёт
1b19aae025066d99a3bb82c88cc9acd3c6514c0c Добавил папку для отчёта
ab515eea66067bb03cbbcb1de0fcd606306dac07 добавил лицензию
58f9611f6e5543779c3d57c86356aefe30039736 I changed it again :)
a221a44724530d2968ff0cd45761b9da982c4580 Changed proj.py
261ae880eb6fe7a21e703f0de297d72aaa389f75 Delete 1 file
def5fccf2ec7d34471e7508c9f324e043abfd1a9 Add new files
170b840793e999dac48cf4981850ab409c4c9116 Update README.md
ee8406b6be5000891a102ec40f50fc1c5368202f Create .gitignore
9a205cd88b25750aa8d2fce44fd7c76153120f18 Add information about local repository in readme file
```

Рисунок 4 – Использование команды `git log` с параметром `--pretty` и опцией `online`

```
C:\Users\student-09-525\Lab-1>git log --pretty=format:"%h - %an, %ar : %s"
87668da - Glek, 9 minutes ago : Изменил proj.py
8149583 - GlebMizin, 14 minutes ago : Update LICENSE
121d2b6 - Glek, 10 hours ago : Загрузил отчёт
1b19aae - Glek, 11 hours ago : Добавил папку для отчёта
ab515ee - Glek, 13 hours ago : добавил лицензию
58f9611 - Glek, 13 hours ago : I changed it again :)
a221a44 - Glek, 13 hours ago : Changed proj.py
261ae88 - Glek, 13 hours ago : Delete 1 file
def5fcc - Glek, 13 hours ago : Add new files
170b840 - GlebMizin, 7 days ago : Update README.md
ee8406b - GlebMizin, 7 days ago : Create .gitignore
9a205cd - GlebMizin, 7 days ago : Add information about local repository in readme file
```

Рисунок 5 – Использование команды `git log` с параметром `--pretty` и опцией `format`

В git так же присутствует возможность фильтровать проверяемые коммиты по времени изменения, а также по расположению файлов.

```

C:\Users\student-09-525\Lab-1>git log --since=2.days
commit 87668dab8426a6807f2ab680195add841d4435ec <HEAD -> master, origin/master,
origin/HEAD>
Author: Glek <glebmizin1@gmail.com>
Date: Sat Sep 10 11:55:00 2022 +0300

    Изменил proj.py

commit 81495839b8b84bbff36ac47c3109e3a7a164b3fe
Author: GlebMizin <112752101+GlebMizin@users.noreply.github.com>
Date: Sat Sep 10 11:50:04 2022 +0300

    Update LICENSE

commit 121d2b62b84403c1bc9d5d845c58ed504555f23a
Author: Glek <glebmizin1@gmail.com>
Date: Sat Sep 10 02:24:23 2022 +0300

    Загрузил отчёт

commit 1b19aae025066d99a3bb82c88cc9acd3c6514c0c
Author: Glek <glebmizin1@gmail.com>
Date: Sat Sep 10 01:29:27 2022 +0300

    Добавил папку для отчёта

commit ab515eea66067bb03cbbbeb1de0fcd606306dac07
Author: Glek <glebmizin1@gmail.com>
Date: Fri Sep 9 23:24:38 2022 +0300

    добавил лицензию

commit 58f9611f6e5543779c3d57c86356aefe30039736
Author: Glek <glebmizin1@gmail.com>
Date: Fri Sep 9 23:10:36 2022 +0300

    I changed it again :)

commit a221a44724530d2968ff0cd45761b9da982c4580
Author: Glek <glebmizin1@gmail.com>
Date: Fri Sep 9 23:06:55 2022 +0300

    Changed proj.py
:....skipping...
commit 87668dab8426a6807f2ab680195add841d4435ec <HEAD -> master, origin/master,
origin/HEAD>
Author: Glek <glebmizin1@gmail.com>
Date: Sat Sep 10 11:55:00 2022 +0300

    Изменил proj.py

commit 81495839b8b84bbff36ac47c3109e3a7a164b3fe
Author: GlebMizin <112752101+GlebMizin@users.noreply.github.com>
Date: Sat Sep 10 11:50:04 2022 +0300

    Update LICENSE

commit 121d2b62b84403c1bc9d5d845c58ed504555f23a
Author: Glek <glebmizin1@gmail.com>
Date: Sat Sep 10 02:24:23 2022 +0300

    Загрузил отчёт

commit 1b19aae025066d99a3bb82c88cc9acd3c6514c0c
Author: Glek <glebmizin1@gmail.com>
Date: Sat Sep 10 01:29:27 2022 +0300

    Добавил папку для отчёта

commit ab515eea66067bb03cbbbeb1de0fcd606306dac07
Author: Glek <glebmizin1@gmail.com>
Date: Fri Sep 9 23:24:38 2022 +0300

    добавил лицензию

```

Рисунок 6 – Использование команды git log с фильтром since

```
C:\Users\student-09-525\Lab-1>git log -- C:\Users\student-09-525\Lab-1\doc
commit 121d2b62b84403c1bc9d5d845c58ed504555f23a
Author: Glek <glebmizin1@gmail.com>
Date: Sat Sep 10 02:24:23 2022 +0300

    Загрузил отчёт

commit 1b19aae025066d99a3bb82c88cc9acd3c6514c0c
Author: Glek <glebmizin1@gmail.com>
Date: Sat Sep 10 01:29:27 2022 +0300

    Добавил папку для отчёта
```

Рисунок 7 – Использование команды git log с фильтром «Path to file»

2. Просмотр содержимого коммитов

В git так же можно просматривать содержимое коммитов используя команду git show HEAD при этом ~1 выдаст предпоследний коммит, и так далее

```
C:\Users\УчебHa\Glek\Lab-1>git show HEAD
commit 3f85a7a2f1a93fd48bd3f5c13e57ef7e140298be (HEAD -> master, origin/master, origin/HEAD)
Author: Glek <glebmizin1@gmail.com>
Date: Mon Sep 26 13:19:16 2022 +0300

    + C#

diff --git "a/\320\236\321\202\321\207\321\221\321\202\321\213 \320\277\320\276 C#/\320\236\321\202\321\207\321\221\321\202 \342\204\2264.docx" "b/\320\236\321\202\321\207\321\221\321\202\321\213 \320\277\320\276 C#/\320\236\321\202\321\207\321\221\321\202 \342\204\2264.docx"
index 66430f8..ef8e7c6 100644
--- "a/\320\236\321\202\321\207\321\221\321\202\321\213 \320\277\320\276 C#/\320\236\321\202\321\207\321\221\321\202 \342\204\2264.docx"
+++ "b/\320\236\321\202\321\207\321\221\321\202\321\213 \320\277\320\276 C#/\320\236\321\202\321\207\321\221\321\202 \342\204\2264.docx"
@@ -181,15 +181,31 @@ namespace HelloWorld
    Рисунок 7 - Результат работы программы

3. Ответы на контрольные вопросы:
- 1.
- 2. Правильно объявленные массивы: int[] m; , System.String[] names = {"Piro","Nadi","Deny"}; , System.Int64[] big; , char[] c ={'z','y','x'}; , string [] str = new string [10];
- 3. В результате работы res будет равен 16
- 4. В результате работы программы в консоль будет выведено 1807074970
- 5. Количество элементов массива не может быть больше максимального индекса, минимальный индекс отвечает за первое значение массива
```

Рисунок 8 –Просмотр последнего коммита

```
C:\Users\УчебHa\Glek\Lab-1>git show HEAD~1
commit 3cd79ae7935249a0d48593a3a5e6e56d234351db
Author: Glek <glebmizin1@gmail.com>
Date: Mon Sep 26 12:49:04 2022 +0300

    +

diff --git "a/\320\236\321\202\321\207\321\221\321\202\321\213 \320\277\320\276 C#/\321\207\321\221\321\202 \342\204\2262.docx" "b/\320\236\321\202\321\207\321\221\321\202\321\213 \320\277\320\276 C#/\321\207\321\221\321\202 \342\204\2262.docx"
deleted file mode 100644
index 2bb7d2a..0000000
Failed to extract required information from <C:\Users\4B5F~1\AppData\Local\Temp\git-blob-a22160/~$чёр №2.docx>!
--- "a/\320\236\321\202\321\207\321\221\321\202\321\213 \320\277\320\276 C#/\321\207\321\221\321\202 \342\204\2262.docx"
+++ /dev/null
@@ -1 +0,0 @@
- "EAdmin"
n"
<FC>"M"t"
\ No newline at end of file
diff --git "a/\320\236\321\202\321\207\321\221\321\202\321\213 \320\277\320\276 C#/\321\207\321\221\321\202 \342\204\2263.docx" "b/\320\236\321\202\321\207\321\221\321\202\321\213 \320\277\320\276 C#/\321\207\321\221\321\202 \342\204\2263.docx"
deleted file mode 100644
index 2a62400..0000000
```

Рисунок 9 – Просмотр предпоследнего коммита

```
C:\Users\УчебНа\Glek\Lab-1>git show 1b19aae025066
commit 1b19aae025066d99a3bb82c88cc9acd3c6514c0c
Author: Glek <glebmizin1@gmail.com>
Date:   Sat Sep 10 01:29:27 2022 +0300

    Добавил папку для отчёта

diff --git a/doc/.gitkeep b/doc/.gitkeep
new file mode 100644
index 0000000..e69de29
```

Рисунок 10 – Просмотр коммита по хешу

3. Теги

В git есть возможность создавать теги, легковесные и аннотированные, отличие заключается в том, что первый вид, является просто ссылкой на коммит а второй хранится в базе git как полноценный объект.

```
C:\Users\student-09-525\Lab-1>git tag -a v1.1 -m "smth"
```

Рисунок 11 – Создание аннотированного тега

Теги так же можно просматривать при помощи команды git show

```
C:\Users\student-09-525\Lab-1>git show v1.1
tag v1.1
Tagger: Glek <glebmizin1@gmail.com>
Date:   Sat Sep 10 12:28:34 2022 +0300

smth

commit 87668dab8426a6807f2ab680195add841d4435ec <HEAD -> master, tag: v1.1, origin/master, origin/HEAD>
Author: Glek <glebmizin1@gmail.com>
Date:   Sat Sep 10 11:55:00 2022 +0300

    Изменил proj.py

diff --git a/proj.py b/proj.py
index 05c5e59..fdb392a 100644
--- a/proj.py
+++ b/proj.py
@@ -1,1 @@
-print<Hello!!!!!!>
\ No newline at end of file
+print<'Hello!!!!!!'>
\ No newline at end of file
```

Рисунок 12 – Просмотр тега

Так как теги не отправляются на удалённый сервер по умолчанию следует отправлять теги явно, с использованием команды `git push origins`

```
C:\Users\student-09-525\Lab-1>git push origin --tags
Enumerating objects: 8, done.
Counting objects: 100% (8/8), done.
Delta compression using up to 2 threads
Compressing objects: 100% (5/5), done.
Writing objects: 100% (6/6), 572 bytes | 286.00 KiB/s, done.
Total 6 (delta 2), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (2/2), completed with 1 local object.
To https://github.com/GlebMizin/Lab-1.git
 * [new tag]          v1.1 -> v1.1
 * [new tag]          v1.1.1 -> v1.1.1
 * [new tag]          v1.1.2 -> v1.1.2
```

Рисунок 13 – Отправка тега на удаленный сервер

Список тегов можно просмотреть при помощи команды `git tag` с опцией `-l`, так же можно фильтровать список выдаваемых тегов при помощи «*»

```
C:\Users\student-09-525\Lab-1>git tag -l
v1.1
v1.1.1
v1.1.2

C:\Users\student-09-525\Lab-1>git tag -l "v1.1.*"
v1.1.1
v1.1.2
```

Рисунок 14 – Вызов списка всех тегов и вызов отфильтрованного списка

Так же существует возможность удаления тегов командой `git tag -d` «название тега»

```
C:\Users\УчебНа\Glek\Lab-1>git tag -a v1.1.6 -m "тег для удаления"

C:\Users\УчебНа\Glek\Lab-1>git tag list

C:\Users\УчебНа\Glek\Lab-1>git tag -list
error: did you mean `--list` (with two dashes)?

C:\Users\УчебНа\Glek\Lab-1>git tag --list
list
v1.1
v1.1.1
v1.1.2
v1.1.6

C:\Users\УчебНа\Glek\Lab-1>git tag -d list
Deleted tag 'list' (was 3f85a7a)

C:\Users\УчебНа\Glek\Lab-1>git tag -d v1.1.6
Deleted tag 'v1.1.6' (was af8d9c1)
```

Рисунок 15 – Удаление тегов

4. Удаление и восстановление файлов

В git есть возможность отмены изменений файла, при помощи команды `git checkout`

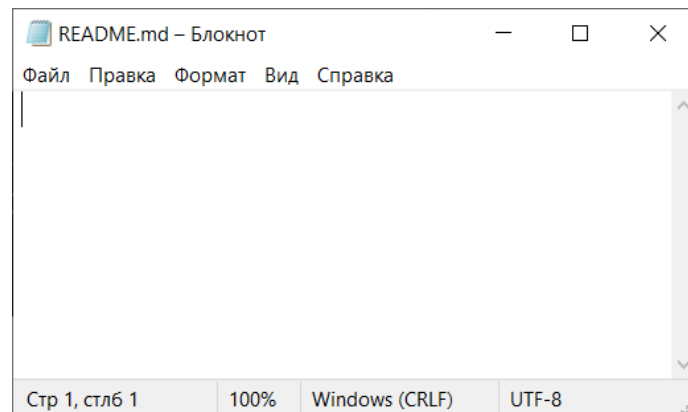


Рисунок 16.1 – Изменённый файл README

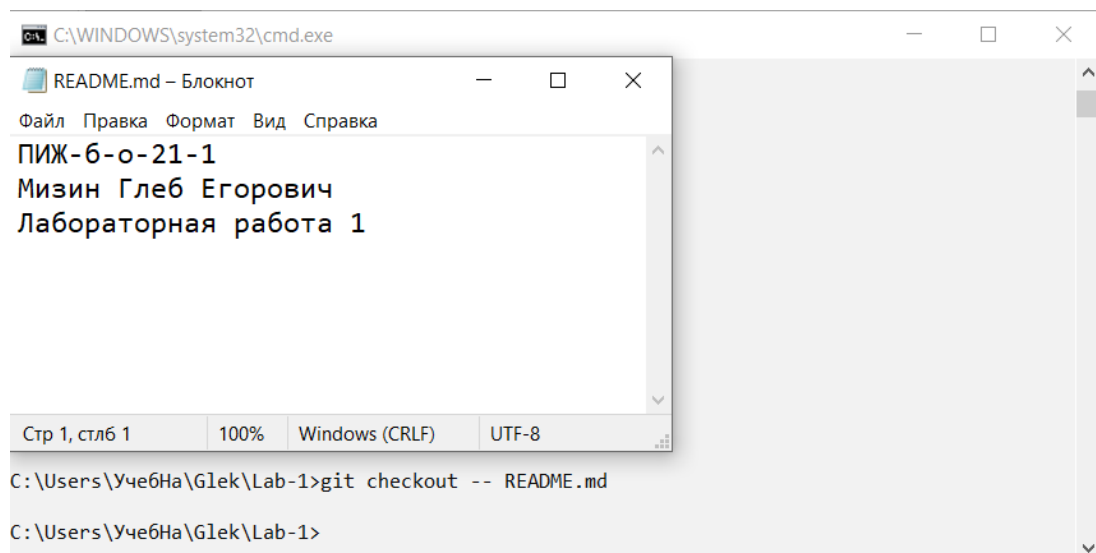


Рисунок 16.2 – Восстановленный файл README

Так же можно откатывать версию файла при помощи команды `git reset --hard`

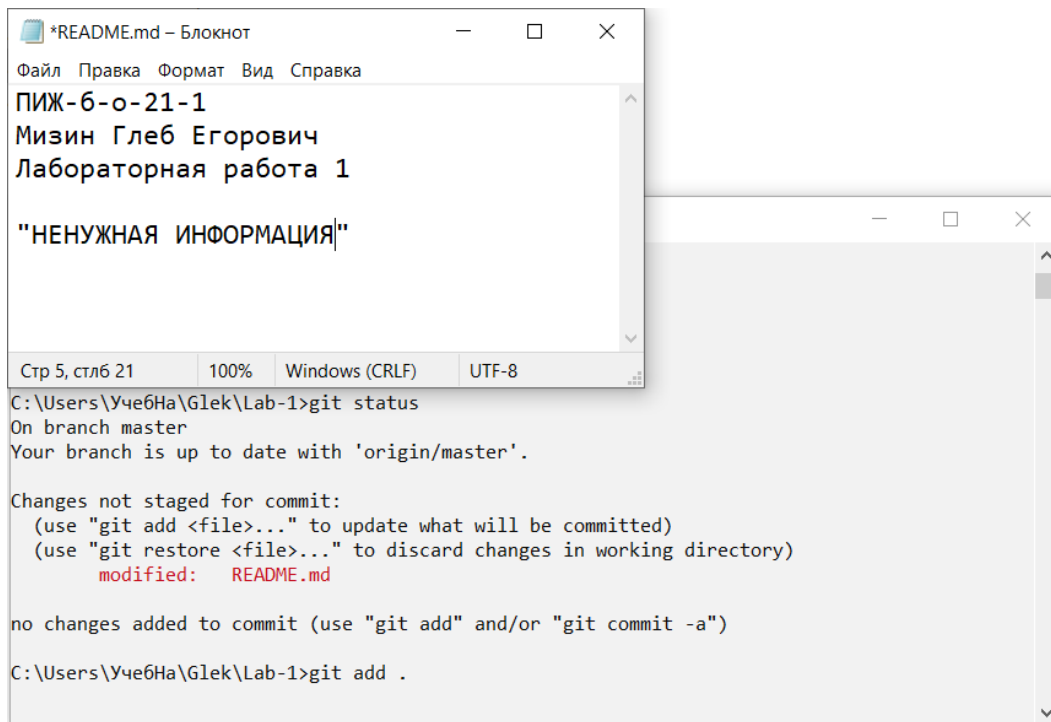


Рисунок 17 – Измененный файл README

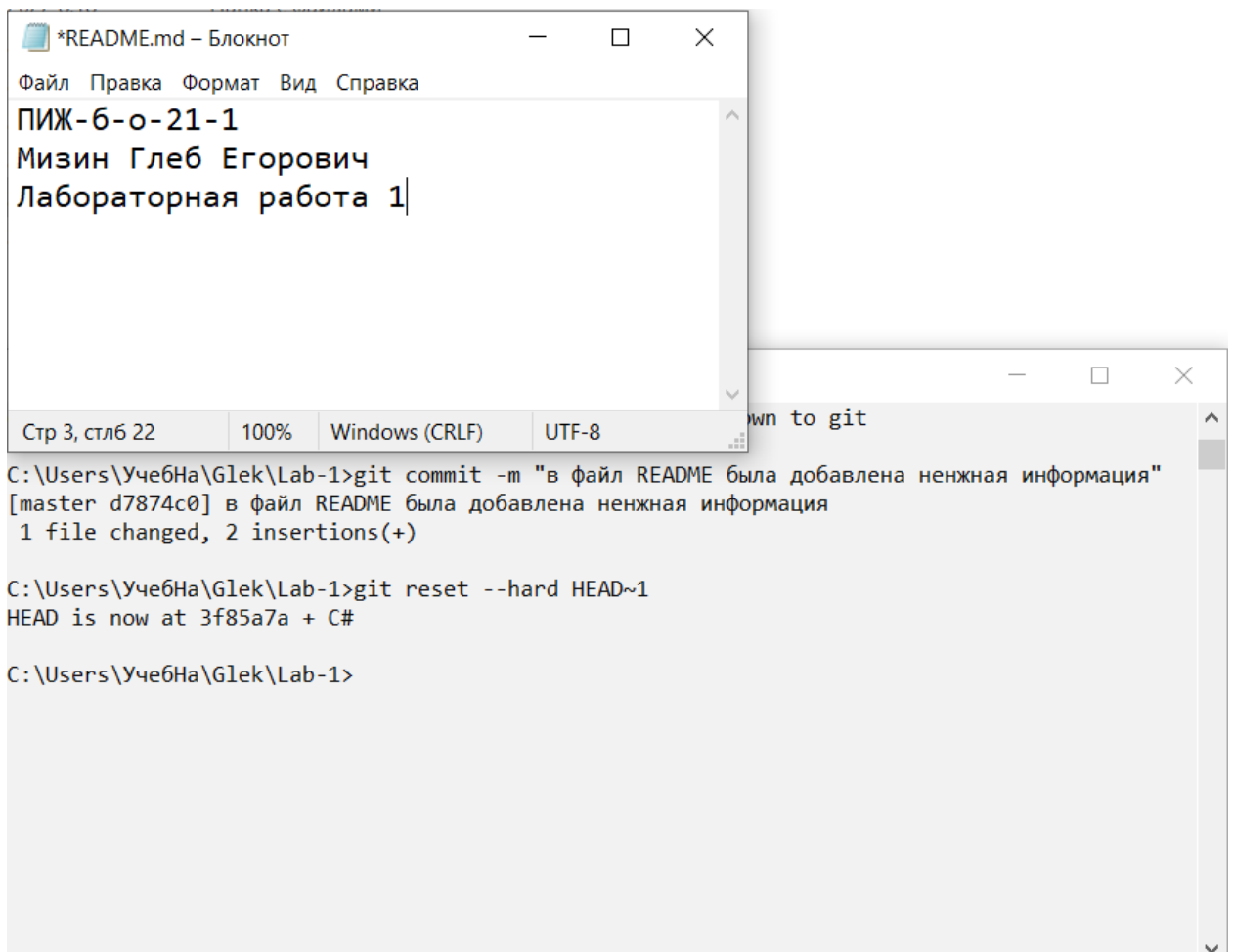


Рисунок 18 – Восстановленный файл README

5. Создание зеркала

Локальный репозиторий из GitHub можно добавить и в другие удалённые хранилища так, например, при помощи команды `git push --prune https://gitlab.com/web_studying/123.git` мы добавим его на площадку GitLab

```
C:\Users\student-09-525\Lab-1>git fetch --prune
C:\Users\student-09-525\Lab-1>git push --prune https://gitlab.com/web_studying/123.git
Enumerating objects: 177, done.
Counting objects: 100% (177/177), done.
Delta compression using up to 2 threads
Compressing objects: 100% (90/90), done.
Writing objects: 100% (177/177), 6.42 MiB | 1.26 MiB/s, done.
Total 177 (delta 67), reused 162 (delta 59), pack-reused 0
remote: Resolving deltas: 100% (67/67), done.
```

Рисунок 19 – Загрузка репозитория на GitLab

Контрольные вопросы

1. Как выполнить историю коммитов в Git? Какие существуют дополнительные опции для просмотра истории коммитов?

При помощи команды `git log`. Опции для просмотра коммитов:

| Опция | Описание |
|------------------------------|---|
| <code>-p</code> | Показывает патч для каждого коммита. |
| <code>--stat</code> | Показывает статистику измененных файлов для каждого коммита. |
| <code>--shortstat</code> | Отображает только строку с количеством изменений/вставок/удалений для команды <code>--stat</code> . |
| <code>--name-only</code> | Показывает список измененных файлов после информации о коммите. |
| <code>--name-status</code> | Показывает список файлов, которые добавлены/изменены/удалены. |
| <code>--abbrev-commit</code> | Показывает только несколько символов SHA-1 чек-суммы вместо всех 40. |
| <code>--relative-date</code> | Отображает дату в относительном формате (например, «2 weeks ago») вместо стандартного формата даты. |
| <code>--graph</code> | Отображает ASCII граф с ветвлениями и историей слияний. |
| <code>--pretty</code> | Показывает коммиты в альтернативном формате. Возможные варианты опций: <code>oneline</code> , <code>short</code> , <code>full</code> , <code>fuller</code> и <code>format</code> (с помощью последней можно указать свой формат). |
| <code>--oneline</code> | Сокращение для одновременного использования опций <code>--pretty=oneline</code> <code>--abbrev-commit</code> . |

2. Как ограничить вывод при просмотре истории коммитов?

В нижеприведенной таблице Вы можете увидеть эти и другие распространенные опции.

| Опция | Описание |
|--------------------------------|---|
| <code>-(n)</code> | Показывает только последние n коммитов. |
| <code>--since, --after</code> | Показывает только те коммиты, которые были сделаны после указанной даты. |
| <code>--until, --before</code> | Показывает только те коммиты, которые были сделаны до указанной даты. |
| <code>--author</code> | Показывает только те коммиты, в которых запись author совпадает с указанной строкой. |
| <code>--committer</code> | Показывает только те коммиты, в которых запись committer совпадает с указанной строкой. |
| <code>--grep</code> | Показывает только коммиты, сообщение которых содержит указанную строку. |
| <code>-S</code> | Показывает только коммиты, в которых изменение в коде повлекло за собой добавление или удаление указанной строки. |

3. Как внести изменения в уже сделанный коммит?

Командой `git commit --amend`

4. Как отменить индексацию файла в Git?

При помощи команды `git reset HEAD`

5. Как отменить изменения в файле?

Командой `git checkout -- «имя файла»`

6. Что такое удалённый репозиторий Git

Удалённые репозитории представляют собой версии вашего проекта, сохранённые в интернете или ещё где-то в сети.

7. Как выполнить просмотр удаленных репозиториях данного локального репозитория?

Команда `git remote` выводит названия доступных удалённых репозиториях.

8. Как добавить удаленный репозиторий для данного локального репозитория?

Выполнить команду `git remote add`

9. Как выполнить отправку/получение изменений с удаленного репозитория?

Командой `git fetch «Имя проекта»` можно получить недостающие данные проекта

Командой `git push origin master` можно выполнить отправку изменений на удаленный репозиторий

10. Как выполнить просмотр удаленного репозитория?

Команда `git remote show <remote>` отобразит URL репозитория а так же информацию о ветках

11. Каково назначение тэгов Git?

Тег – это ссылка, указывающая на определенную точку в истории Git.

12. Как осуществляется работа с тэгами Git?

`git tag «название тега»` создаст легковесный тег, добавление параметра `-a` позволит создать аннотированный тег.

13. Самостоятельно изучите назначение флага `--prune` в командах `git fetch` и `git push`. Каково назначение этого флага?

`git fetch --prune` для удаления устаревших веток