

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ
ФЕДЕРАЦИИ**
**Федеральное государственное автономное
образовательное учреждение высшего образования
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»**

Кафедра инфокоммуникаций

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

Отчет по лабораторной работе № 1.2

по дисциплине «Основы программной инженерии»

Выполнил студент группы ПИЖ-б-о-21-1

Гребенкин Е. А.. «14» сентября 2022г.

Подпись студента _____

Работа защищена « » _____ 20__ г.

Проверила Воронкин Р.А. _____
(подпись)

Ставрополь 2022

Методика и порядок выполнения работы

https://github.com/woeshiga/OPI_lw_2.git

1. Обработка примеров лабораторной работы.

Запуск команд `git log`, `git log -p -2`, `git log --stat`, `git log --pretty=oneline`, `git log --pretty=short`, `git log --format :"%h - %s, %an , %ar"`, `git log --pretty=format:"%h %s" --graph`, `git log --since=2.weeks`, `git log -S function_name`, `git log -- path/to/file`, в каталоге клонированного проекта, показан на рисунках 1,2,3,4,5,6,7,8,9,10.

```
C:\Users\diejester\Desktop\Программирование\ОПИЖ\OPI_lw_2>git log
commit fd508946378f785656bf5b75631c410386790314 (HEAD -> main, origin/main, origin/HEAD)
Author: diejester <egor.grebenckin16@gmail.com>
Date:   Fri Oct 21 23:30:55 2022 +0300

    Writed code

commit 3b815cb27388ca939a10f8d75f0f32b4963b12cd
Author: diejester <egor.grebenckin16@gmail.com>
Date:   Fri Oct 21 23:29:39 2022 +0300

    Creating main.py file

commit b6f67172d00fb331e3479fa12ea760d36e77eab0
Author: woeshiga <63983519+woeshiga@users.noreply.github.com>
Date:   Fri Oct 21 23:23:47 2022 +0300

    Initial commit

C:\Users\diejester\Desktop\Программирование\ОПИЖ\OPI_lw_2>
```

Рисунок 1 – Результат команды `git log`

```

Администратор: C:\Windows\System32\cmd.exe - git log -p -2

C:\Users\diejester\Desktop\Программирование\ОПИЖ\ОПИ_1w_2>git log -p -2
commit fd508946378f785656bf5b75631c410386790314 (HEAD -> main, origin/main, origin/HEAD)
Author: diejester <egor.grebenckin16@gmail.com>
Date: Fri Oct 21 23:30:55 2022 +0300

    Writed code

diff --git a/main.py b/main.py
index e69de29..64f1e02 100644
--- a/main.py
+++ b/main.py
@@ -0,0 +1,35 @@
+from random import randint
+
+name = input("Enter your name: ")
+
+age = input("Enter your age: ")
+
+hobby = input("Enter your hobby: ")
+
+print(f"Your name is {name}\nYour age is {age}\nYour hobby is {hobby}")
+
+print("Nice to meet you! Let's play to my game!")
+
+score = 0
+
+while True:
+    number = randint(0, 1)
+
+    while True:
+        ...skipping...
commit fd508946378f785656bf5b75631c410386790314 (HEAD -> main, origin/main, origin/HEAD)
Author: diejester <egor.grebenckin16@gmail.com>
Date: Fri Oct 21 23:30:55 2022 +0300

    Writed code

diff --git a/main.py b/main.py
index e69de29..64f1e02 100644
--- a/main.py
+++ b/main.py
@@ -0,0 +1,35 @@
+from random import randint
+
+name = input("Enter your name: ")
+
+age = input("Enter your age: ")
+
+hobby = input("Enter your hobby: ")
+
+print(f"Your name is {name}\nYour age is {age}\nYour hobby is {hobby}")
+
+print("Nice to meet you! Let's play to my game!")

```

Рисунок 2 – Результат команды git log -p -2

```

C:\Users\diejester\Desktop\Программирование\ОПИЖ\ОПИ_lw_2>git log --stat
commit fd508946378f785656bf5b75631c410386790314 (HEAD -> main, origin/main, origin/HEAD)
Author: diejester <egor.grebenckin16@gmail.com>
Date:   Fri Oct 21 23:30:55 2022 +0300

    Writed code

main.py | 35 ++++++++++++++++++++++++++++++++++++++
1 file changed, 35 insertions(+)

commit 3b815cb27388ca939a10f8d75f0f32b4963b12cd
Author: diejester <egor.grebenckin16@gmail.com>
Date:   Fri Oct 21 23:29:39 2022 +0300

    Creating main.py file

main.py | 0
1 file changed, 0 insertions(+), 0 deletions(-)

commit b6f67172d00fb331e3479fa12ea760d36e77eab0
Author: woeshiga <63983519+woeshiga@users.noreply.github.com>
Date:   Fri Oct 21 23:23:47 2022 +0300

    Initial commit

.gitignore | 129 ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
LICENSE    | 21 ++++++++
README.md  | 1 +
3 files changed, 151 insertions(+)

C:\Users\diejester\Desktop\Программирование\ОПИЖ\ОПИ_lw_2>

```

Рисунок 3 – Результат команды git log --stat

```
C:\Users\diejester\Desktop\Программирование\ОПИЖ\ОПИ_lw_2>git log --pretty=oneline
fd508946378f785656bf5b75631c410386790314 (HEAD -> main, origin/main, origin/HEAD) Writed code
3b815cb27388ca939a10f8d75f0f32b4963b12cd Creating main.py file
b6f67172d00fb331e3479fa12ea760d36e77eab0 Initial commit

C:\Users\diejester\Desktop\Программирование\ОПИЖ\ОПИ_lw_2>_
```

Рисунок 4 – Результат команды git log --pretty=oneline

```
C:\Users\diejester\Desktop\Программирование\ОПИЖ\ОПИ_lw_2>git log --pretty=short
commit fd508946378f785656bf5b75631c410386790314 (HEAD -> main, origin/main, origin/HEAD)
Author: diejester <egor.grebenckin16@gmail.com>

    Writed code

commit 3b815cb27388ca939a10f8d75f0f32b4963b12cd
Author: diejester <egor.grebenckin16@gmail.com>

    Creating main.py file

commit b6f67172d00fb331e3479fa12ea760d36e77eab0
Author: woeshiga <63983519+woeshiga@users.noreply.github.com>

    Initial commit

C:\Users\diejester\Desktop\Программирование\ОПИЖ\ОПИ_lw_2>_
```

Рисунок 5 – Результат команды git log --pretty=short

```
C:\Users\diejester\Desktop\Программирование\ОПИЖ\ОПИ_1w_2>git log --pretty=format:"%h - %s, %an , %ar"
fd50894 - Writed code, diejester , 12 minutes ago
3b815cb - Creating main.py file, diejester , 13 minutes ago
b6f6717 - Initial commit, woeshiga , 19 minutes ago

C:\Users\diejester\Desktop\Программирование\ОПИЖ\ОПИ_1w_2>
```

Рисунок 6 – Результат команды `git log --pretty=format:"%h - %s, %an , %ar"`

```
C:\Users\diejester\Desktop\Программирование\ОПИЖ\ОПИ_1w_2>git log --pretty=format:"%h %s" --graph
* fd50894 Writed code
* 3b815cb Creating main.py file
* b6f6717 Initial commit

C:\Users\diejester\Desktop\Программирование\ОПИЖ\ОПИ_1w_2>_
```

Рисунок 7 – Результат команды `git log --pretty=format:"%h %s" --graph`

```

C:\Users\diejester\Desktop\Программирование\ОПИЖ\ОПИ_lw_2>git log --since=2.weeks
commit fd508946378f785656bf5b75631c410386790314 (HEAD -> main, origin/main, origin/HEAD)
Author: diejester <egor.grebenckin16@gmail.com>
Date:   Fri Oct 21 23:30:55 2022 +0300

    Writed code

commit 3b815cb27388ca939a10f8d75f0f32b4963b12cd
Author: diejester <egor.grebenckin16@gmail.com>
Date:   Fri Oct 21 23:29:39 2022 +0300

    Creating main.py file

commit b6f67172d00fb331e3479fa12ea760d36e77eab0
Author: woeshiga <63983519+woeshiga@users.noreply.github.com>
Date:   Fri Oct 21 23:23:47 2022 +0300

    Initial commit

C:\Users\diejester\Desktop\Программирование\ОПИЖ\ОПИ_lw_2>

```

Рисунок 8 – Результат команды git log --since=2.weeks

```

C:\Users\diejester\Desktop\Программирование\ОПИЖ\ОПИ_lw_2>git log -S print
commit fd508946378f785656bf5b75631c410386790314 (HEAD -> main, origin/main, origin/HEAD)
Author: diejester <egor.grebenckin16@gmail.com>
Date:   Fri Oct 21 23:30:55 2022 +0300

    Writed code

C:\Users\diejester\Desktop\Программирование\ОПИЖ\ОПИ_lw_2>

```

Рисунок 9 – Результат команды git log -S function_name

```
Администратор: C:\Windows\System32\cmd.exe
Author: diejester <egor.grebenckin16@gmail.com>
Date: Fri Oct 21 23:30:55 2022 +0300

Wrote code

C:\Users\diejester\Desktop\Программирование\ОПИЖ\ОПИ_1w_2>git log --C:\Users\diejester\Desktop\Программирование\ОПИЖ\ОПИ_1w_2
fatal: unrecognized argument: --C:\Users\diejester\Desktop\Программирование\ОПИЖ\ОПИ_1w_2

C:\Users\diejester\Desktop\Программирование\ОПИЖ\ОПИ_1w_2>git log -- C:\Users\diejester\Desktop\Программирование\ОПИЖ\ОПИ_1w_2
commit fd508946378f785656bf5b75631c410386790314 (HEAD -> main, origin/main, origin/HEAD)
Author: diejester <egor.grebenckin16@gmail.com>
Date: Fri Oct 21 23:30:55 2022 +0300

    Wrote code

commit 3b815cb27388ca939a10f8d75f0f32b4963b12cd
Author: diejester <egor.grebenckin16@gmail.com>
Date: Fri Oct 21 23:29:39 2022 +0300

    Wrote code

    Creating main.py file

commit b6f67172d00fb331e3479fa12ea760d36e77eab0
Author: woeshiga <63983519+woeshiga@users.noreply.github.com>
Date: Fri Oct 21 23:23:47 2022 +0300

    Initial commit

C:\Users\diejester\Desktop\Программирование\ОПИЖ\ОПИ_1w_2>
```

Рисунок 10 – Результат команды git log -- path/to/file

Операция отмены – git commit –amend показана на рисунке 11.

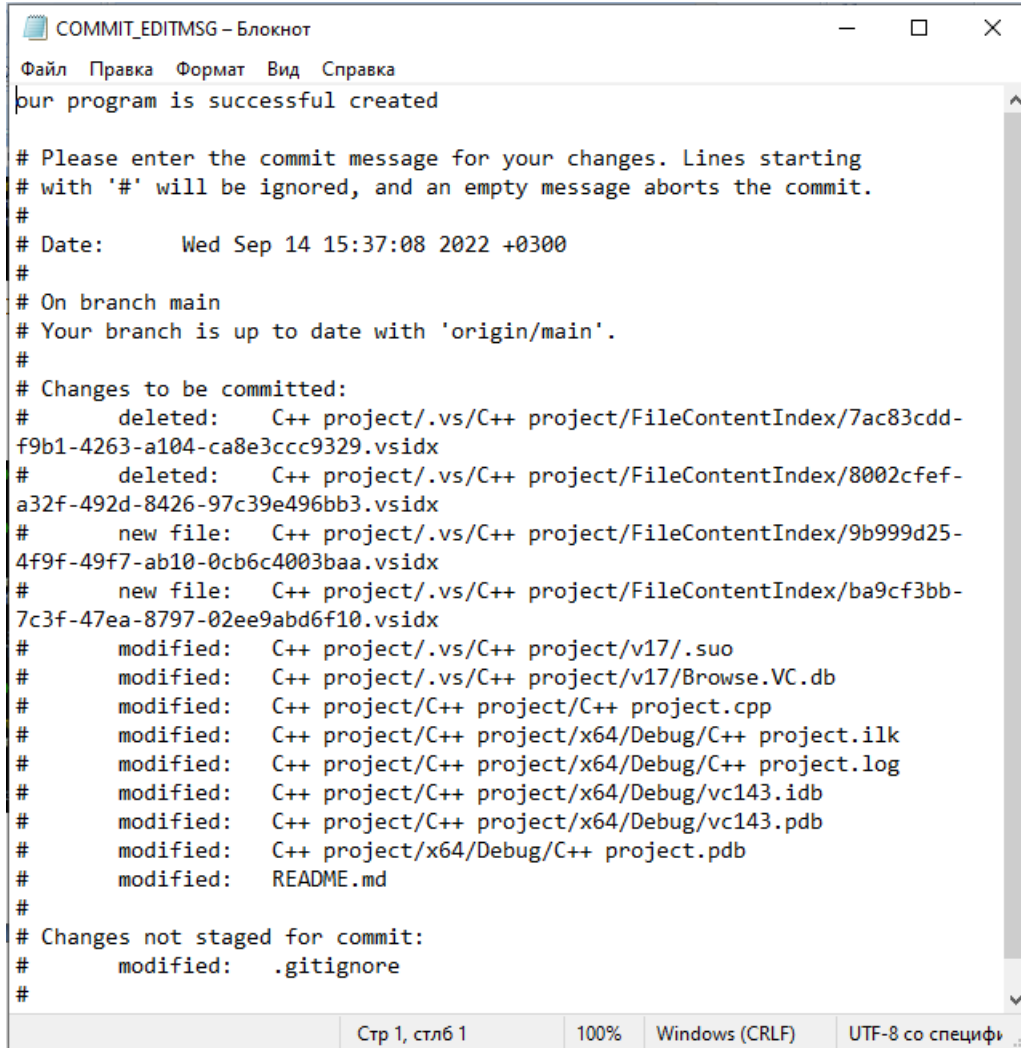
```
M@DESKTOP-UVM9NOL MINGW64 ~/desktop/OPI_LR_2 (main)
$ git add .

M@DESKTOP-UVM9NOL MINGW64 ~/desktop/OPI_LR_2 (main)
$ git commit --amend
hint: waiting for your editor to close the file... dos2unix: converting file C:/Users/M/Desktop/OPI_LR_2/.git/COMMIT_EDITMSG to
unix format...
[main 6b742a5] изменение в README файле
Date: Sat Sep 24 03:15:11 2022 +0300
1 file changed, 5 insertions(+), 1 deletion(-)

M@DESKTOP-UVM9NOL MINGW64 ~/desktop/OPI_LR_2 (main)
$ git log
commit 6b742a53272203c447517f712b64439f40141ffa (HEAD -> main)
Author: obrMaria <midsummer.madness@mail.ru>
Date: Sat Sep 24 03:15:11 2022 +0300

    изменение в README файле
```

Рисунок 11 – Результат команды git commit –amend



```
COMMIT_EDITMSG - Блокнот
Файл  Правка  Формат  Вид  Справка
our program is successful created

# Please enter the commit message for your changes. Lines starting
# with '#' will be ignored, and an empty message aborts the commit.
#
# Date:      Wed Sep 14 15:37:08 2022 +0300
#
# On branch main
# Your branch is up to date with 'origin/main'.
#
# Changes to be committed:
#   deleted:    C++ project/.vs/C++ project/FileContentIndex/7ac83cdd-
f9b1-4263-a104-ca8e3ccc9329.vsidx
#   deleted:    C++ project/.vs/C++ project/FileContentIndex/8002cfef-
a32f-492d-8426-97c39e496bb3.vsidx
#   new file:   C++ project/.vs/C++ project/FileContentIndex/9b999d25-
4f9f-49f7-ab10-0cb6c4003baa.vsidx
#   new file:   C++ project/.vs/C++ project/FileContentIndex/ba9cf3bb-
7c3f-47ea-8797-02ee9abd6f10.vsidx
#   modified:   C++ project/.vs/C++ project/v17/.suo
#   modified:   C++ project/.vs/C++ project/v17/Browse.VC.db
#   modified:   C++ project/C++ project/C++ project.cpp
#   modified:   C++ project/C++ project/x64/Debug/C++ project.ilc
#   modified:   C++ project/C++ project/x64/Debug/C++ project.log
#   modified:   C++ project/C++ project/x64/Debug/vc143.idb
#   modified:   C++ project/C++ project/x64/Debug/vc143.pdb
#   modified:   C++ project/x64/Debug/C++ project.pdb
#   modified:   README.md
#
# Changes not staged for commit:
#   modified:   .gitignore
#
```

Стр 1, столб 1 100% Windows (CRLF) UTF-8 со спецификацией

Рисунок 12 – Результаты сохранения

Отмена индексации файла показана на рисунке 13:

```

diejester@DESKTOP-OMK2MPO MINGW64 ~/Desktop/Программирование/ОПИЖ/ОПИ_1w_2 (main)
$ git add .

diejester@DESKTOP-OMK2MPO MINGW64 ~/Desktop/Программирование/ОПИЖ/ОПИ_1w_2 (main)
$ git status
On branch main
Your branch is ahead of 'origin/main' by 1 commit.
(use "git push" to publish your local commits)

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        new file:   hello.txt

diejester@DESKTOP-OMK2MPO MINGW64 ~/Desktop/Программирование/ОПИЖ/ОПИ_1w_2 (main)
$ git reset HEAD hello.txt

diejester@DESKTOP-OMK2MPO MINGW64 ~/Desktop/Программирование/ОПИЖ/ОПИ_1w_2 (main)
$ git status
On branch main
Your branch is ahead of 'origin/main' by 1 commit.
(use "git push" to publish your local commits)

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        hello.txt

nothing added to commit but untracked files present (use "git add" to track)
diejester@DESKTOP-OMK2MPO MINGW64 ~/Desktop/Программирование/ОПИЖ/ОПИ_1w_2 (main)
$ |

```

Рисунок 13—git reset HEAD <file>

Работа с удалёнными репозиториями. Команды для просмотра удалённых репозиторий `git remote`, `git remote -v`, показаны на рисунках 14,15, добавления `git remote add <shortname> <url>` на рисунке 16, просмотра `git remote show origin` показано на рисунке 17, удаления и переименования `git remote rename`, `git remote remove` показано на рисунках 18,19.

```

$ git remote
origin

diejester@DESKTOP-OMK2MPO MINGW64 ~/Desktop/Программирование/ОПИЖ/OPI_lw_2 (main)
$ git remote -v
origin https://github.com/woeshiga/OPI_lw_2.git (fetch)
origin https://github.com/woeshiga/OPI_lw_2.git (push)

diejester@DESKTOP-OMK2MPO MINGW64 ~/Desktop/Программирование/ОПИЖ/OPI_lw_2 (main)
$

```

Рисунок 14 – Результат команды git remote

```

diejester@DESKTOP-OMK2MPO MINGW64 ~/Desktop/Программирование/ОПИЖ/OPI_lw_2 (main)
$ git remote -v
origin https://github.com/woeshiga/OPI_lw_2.git (fetch)
origin https://github.com/woeshiga/OPI_lw_2.git (push)

diejester@DESKTOP-OMK2MPO MINGW64 ~/Desktop/Программирование/ОПИЖ/OPI_lw_2 (main)
$

```

Рисунок 15– Результат команды git remote -v

```

diejester@DESKTOP-OMK2MPO MINGW64 ~/Desktop/Программирование/ОПИЖ/OPI_lw_2 (main)
$ git remote add new_r https://github.com/woeshiga/OPI_lw_2.git

diejester@DESKTOP-OMK2MPO MINGW64 ~/Desktop/Программирование/ОПИЖ/OPI_lw_2 (main)
$ git remote -v
new_r https://github.com/woeshiga/OPI_lw_2.git (fetch)
new_r https://github.com/woeshiga/OPI_lw_2.git (push)
origin https://github.com/woeshiga/OPI_lw_2.git (fetch)
origin https://github.com/woeshiga/OPI_lw_2.git (push)

diejester@DESKTOP-OMK2MPO MINGW64 ~/Desktop/Программирование/ОПИЖ/OPI_lw_2 (main)
$

```

Рисунок 16– Результат команды git remote add <shortname> <url>

```

diejester@DESKTOP-OMK2MPO MINGW64 ~/Desktop/Программирование/ОПИЖ/OPI_lw_2 (main)
$ git remote show origin
* remote origin
  Fetch URL: https://github.com/woeshiga/OPI_lw_2.git
  Push URL: https://github.com/woeshiga/OPI_lw_2.git
  HEAD branch: main
  Remote branch:
    main tracked
  Local branch configured for 'git pull':
    main merges with remote main
  Local ref configured for 'git push':
    main pushes to main (fast-forwardable)

diejester@DESKTOP-OMK2MPO MINGW64 ~/Desktop/Программирование/ОПИЖ/OPI_lw_2 (main)
$

```

Рисунок 17– Результат команды git remote show origin

```

diejester@DESKTOP-OMK2MPO MINGW64 ~/Desktop/Программирование/ОПИЖ/OPI_1w_2 (main
)
$ git remote rename new_r now_old_r

diejester@DESKTOP-OMK2MPO MINGW64 ~/Desktop/Программирование/ОПИЖ/OPI_1w_2 (main
)
$ git remote
now_old_r
origin

diejester@DESKTOP-OMK2MPO MINGW64 ~/Desktop/Программирование/ОПИЖ/OPI_1w_2 (main
)

```

Рисунок 18– Результат команды git remote rename

```

diejester@DESKTOP-OMK2MPO MINGW64 ~/Desktop/Программирование/ОПИЖ/OPI_1w_2 (main
)
$ git remote remove now_old_r

diejester@DESKTOP-OMK2MPO MINGW64 ~/Desktop/Программирование/ОПИЖ/OPI_1w_2 (main
)
$ git remote
origin

diejester@DESKTOP-OMK2MPO MINGW64 ~/Desktop/Программирование/ОПИЖ/OPI_1w_2 (main
)
$ |

```

Рисунок 19– Результат команды Git remove

Работа с тегами. Команда для просмотра списка тегов `git tag` показана на рисунке 20, для создания и просмотра `git tag -a` и `git show` на рисунках 21,21,23, для обмена тегами `git push origin -tags` на рисунке 24, для удаления тега `git tag -d` на рисунке 25.

```

diejester@DESKTOP-OMK2MPO MINGW64 ~/Desktop/Программирование/ОПИЖ/OPI_1w_2 (main
)
$ git tag

diejester@DESKTOP-OMK2MPO MINGW64 ~/Desktop/Программирование/ОПИЖ/OPI_1w_2 (main
)

```

Рисунок 20 – Результат команды git tag

```

diejester@DESKTOP-OMK2MPO MINGW64 ~/Desktop/Программирование/ОПИЖ/ОПИ_lw_2 (main)
$ git tag -a v0.1b -m "Early beta version"

diejester@DESKTOP-OMK2MPO MINGW64 ~/Desktop/Программирование/ОПИЖ/ОПИ_lw_2 (main)
$ git tag
v0.1b

diejester@DESKTOP-OMK2MPO MINGW64 ~/Desktop/Программирование/ОПИЖ/ОПИ_lw_2 (main)
$ git show v0.1b
tag v0.1b
Tagger: diejester <egor.grebenckin16@gmail.com>
Date: Sat Oct 22 01:04:11 2022 +0300

Early beta version

commit b763Fcb88e1728bf2baa29f9037afa1926fb54aa (HEAD -> main, tag: v0.1b)
Author: diejester <egor.grebenckin16@gmail.com>
Date: Sat Oct 22 00:39:21 2022 +0300

    Something wrong...

diff --git a/main.py b/main.py
index 64f1e02..6c4086e 100644
--- a/main.py
+++ b/main.py
@@ -1,4 +1,8 @@
-from random import randint
+"""
+developer: woeshiga
+"""
+from random import randint

name = input("Enter your name: ")

diejester@DESKTOP-OMK2MPO MINGW64 ~/Desktop/Программирование/ОПИЖ/ОПИ_lw_2 (main)

```

Рисунок 21 –Создание и просмотр тегов с помощью команд
git tag –a и git show

Если не указать сообщение, то Git запустит редактор, чтобы вы смогли его ввести.

```

diejester@DESKTOP-OMK2MPO MINGW64 ~/Desktop/Программирование/ОПИЖ/ОПИ_lw_2 (main)
$ git tag -a new_tag

diejester@DESKTOP-OMK2MPO MINGW64 ~/Desktop/Программирование/ОПИЖ/ОПИ_lw_2 (main)
$
$ git tag
new_tag
v0.1b

diejester@DESKTOP-OMK2MPO MINGW64 ~/Desktop/Программирование/ОПИЖ/ОПИ_lw_2 (main)
$

```

Рисунок 22 – Результат команды git tag –a <tagname>

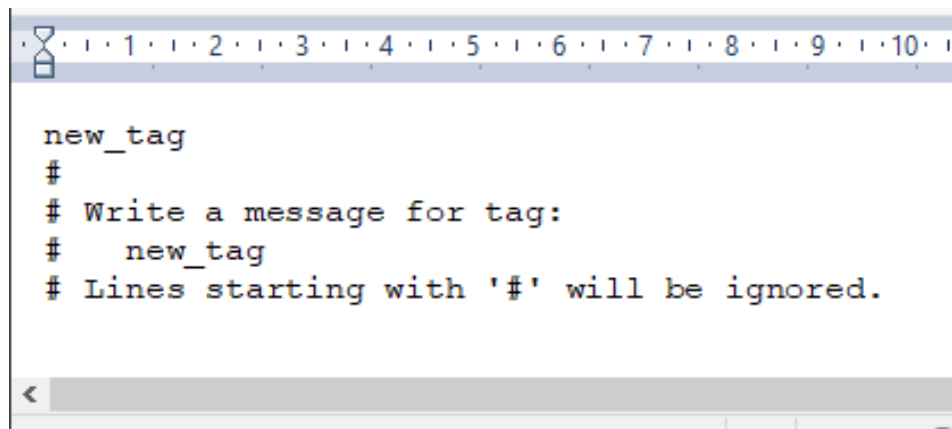


Рисунок 23 – ввод сообщения для тега в редактор

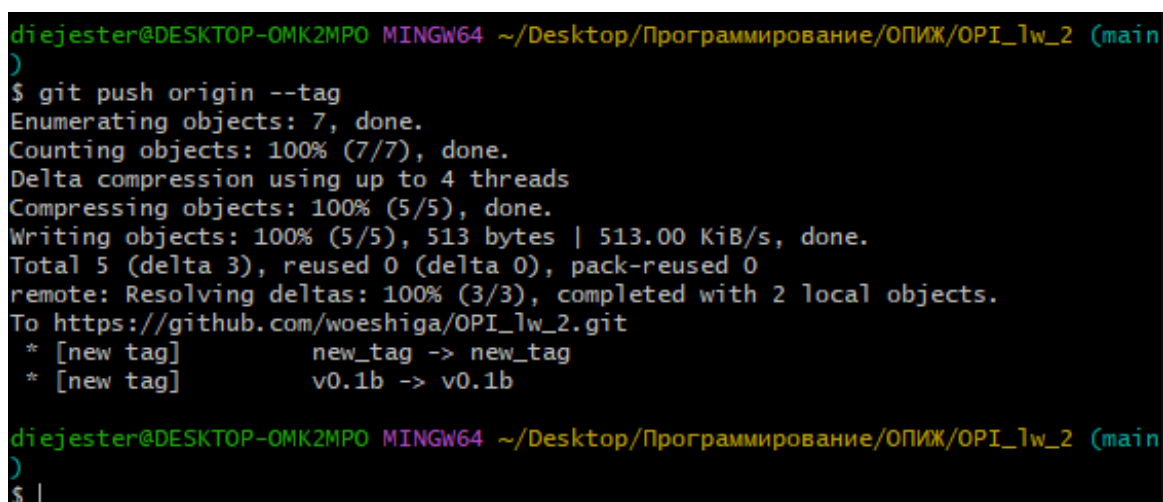


Рисунок 24 – Результат команды `git push origin --tags`

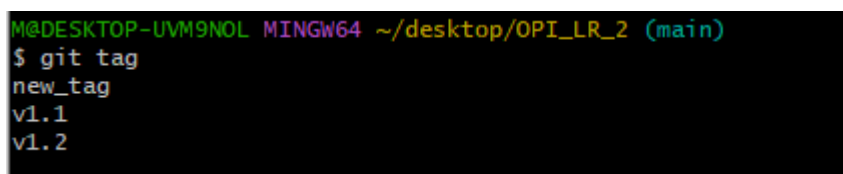


Рисунок 25 – Результат команды `Git tag -d`

Результаты команд `git log --graph --pretty=oneline` для просмотра истории хранилищ, показаны на рисунке 26.

```

diejester@DESKTOP-OMK2MPO MINGW64 ~/Desktop/Программирование/ОПИЖ/OPI_1w_2 (main)
$ git log --graph --pretty=oneline
* b763fcb88e1728bf2baa29f9037afa1926fb54aa (HEAD -> main, tag: v0.1b, tag: new_tag) Something wrong...
* fd508946378f785656bf5b75631c410386790314 (origin/main, origin/HEAD) Writed code
* 3b815cb27388ca939a10f8d75f0f32b4963b12cd Creating main.py file
* b6f67172d00fb331e3479fa12ea760d36e77eab0 Initial commit

diejester@DESKTOP-OMK2MPO MINGW64 ~/Desktop/Программирование/ОПИЖ/OPI_1w_2 (main)
$ |

```

Рисунок 26– результат команды git log --graph --pretty=oneline

Результаты команд git show <ref> , показаны на рисунках 27,28,29, где <ref>

- HEAD: последний коммит;

```

diejester@DESKTOP-OMK2MPO MINGW64 ~/Desktop/Программирование/ОПИЖ/OPI_1w_2 (main)
$ git show HEAD
commit b763fcb88e1728bf2baa29f9037afa1926fb54aa (HEAD -> main, tag: v0.1b, tag: new_tag)
Author: diejester <egor.grebenckin16@gmail.com>
Date: Sat Oct 22 00:39:21 2022 +0300

    Something wrong...

diff --git a/main.py b/main.py
index 64f1e02..6c4086e 100644
--- a/main.py
+++ b/main.py
@@ -1,4 +1,8 @@
-from random import randint
+
+developer: woeshiga
+
+from random import randint

name = input("Enter your name: ")

diejester@DESKTOP-OMK2MPO MINGW64 ~/Desktop/Программирование/ОПИЖ/OPI_1w_2 (main)
$ |

```

Рисунок 27– результат команды git show HEAD

- HEAD~1 предпоследний коммит (и т. д.);

```

diejester@DESKTOP-OMK2MPO MINGW64 ~/Desktop/Программирование/ОПИЖ/ОПИ_1w_2 (main)
$ git show HEAD~1
commit fd508946378f785656bf5b75631c410386790314 (origin/main, origin/HEAD)
Author: diejester <egor.grebenckin16@gmail.com>
Date: Fri Oct 21 23:30:55 2022 +0300

    Writed code

diff --git a/main.py b/main.py
index e69de29..64f1e02 100644
--- a/main.py
+++ b/main.py
@@ -0,0 +1,35 @@
+from random import randint
+
+name = input("Enter your name: ")
+
+age = input("Enter your age: ")
+
+hobby = input("Enter your hobby: ")
+
+print(f"Your name is {name}\nYour age is {age}\nYour hobby is {hobby}")
+
+print("Nice to meet you! Let's play to my game!")
+
+score = 0
+
+while True:
+    number = randint(0, 1)
+
+    while True:
+        user_number = int(input("Zero or One? (0/1): "))
+        if user_number in [0, 1]:
+            break
+
+    if user_number == number:
+        print("Yes!))")
+        score += 1
+    else:
+        print("Nooo((")
+
+    answer = input("One more round?;) (y - yes/n- no)")
+
+    if answer == "n":
+        print(f"Thank you for the game! Your score: {score}")
+        break
+
+input()
diejester@DESKTOP-OMK2MPO MINGW64 ~/Desktop/Программирование/ОПИЖ/ОПИ_1w_2 (main)

```

Рисунок 28— результат команды git show HEAD~1

– fd50894637: коммит с указанным хэшем

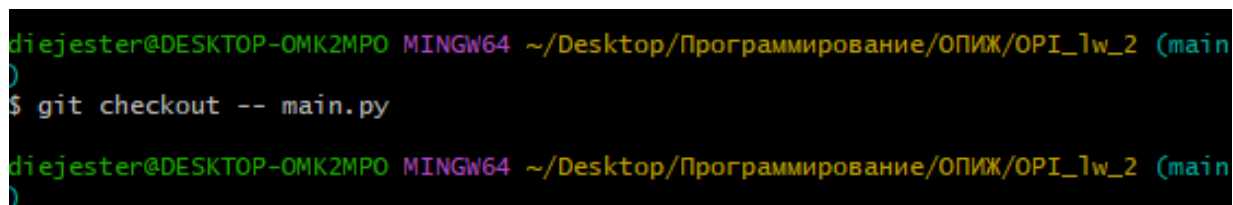
```
$ git show fd50894637|
commit fd508946378f785656bf5b75631c410386790314 (origin/main, origin/HEAD)
Author: diejester <egor.grebenckin16@gmail.com>
Date:   Fri Oct 21 23:30:55 2022 +0300

    Writed code

diff --git a/main.py b/main.py
index e69de29..64f1e02 100644
--- a/main.py
+++ b/main.py
@@ -0,0 +1,35 @@
+from random import randint
+
+name = input("Enter your name: ")
+
+age = input("Enter your age: ")
+
+hobby = input("Enter your hobby: ")
+
+print(f"Your name is {name}\nYour age is {age}\nYour hobby is {hobby}")
+
+print("Nice to meet you! Let's play to my game!")
+
+score = 0
+
+while True:
+    number = randint(0, 1)
+
+    while True:
+        user_number = int(input("Zero or One? (0/1): "))
+        if user_number in [0, 1]:
+            break
+
+    if user_number == number:
+        print("Yes!))")
+        score += 1
+    else:
+        print("Nooo((")
+
+    answer = input("One more round?;) (y - yes/n- no)")
+
+    if answer == "n":
+        print(f"Thank you for the game! Your score: {score}")
+        break
+
+input()
```

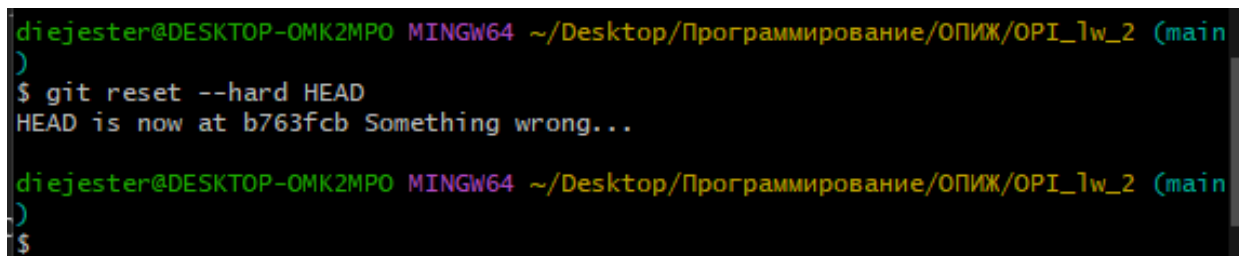
Рисунок 29– результат команды git show fd50894637

Результаты команд для просмотра содержимого коммитов, и команды `git checkout -- <имя_файла>` и `git reset --hard HEAD` для отката к заданной версии, показаны на рисунках.



```
diejester@DESKTOP-OMK2MPO MINGW64 ~/Desktop/Программирование/ОПИЖ/ОПИ_1w_2 (main)
$ git checkout -- main.py
diejester@DESKTOP-OMK2MPO MINGW64 ~/Desktop/Программирование/ОПИЖ/ОПИ_1w_2 (main)
```

Рисунок 30– результат команды `git checkout -- <имя_файла>`



```
diejester@DESKTOP-OMK2MPO MINGW64 ~/Desktop/Программирование/ОПИЖ/ОПИ_1w_2 (main)
$ git reset --hard HEAD
HEAD is now at b763fcb Something wrong...
diejester@DESKTOP-OMK2MPO MINGW64 ~/Desktop/Программирование/ОПИЖ/ОПИ_1w_2 (main)
$
```

Рисунок 31– результат команды `git reset --hard HEAD~1`

Ответы на вопросы:

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

Одним из основных и наиболее мощных инструментов для этого является команда `git log`.

Опции:

Одним из самых полезных аргументов является `-p` или `--patch`, который показывает разницу (выводит патч), внесенную в каждый коммит. Так же вы можете ограничить количество записей в выводе команды;

Так же есть возможность использовать серию опций для обобщения. Например, если вы хотите увидеть сокращенную статистику для каждого коммита, вы можете использовать опцию `--stat`.

Следующей действительно полезной опцией является `--pretty`. Эта опция меняет формат вывода.

Существует несколько встроенных вариантов отображения. Опция `oneline` выводит каждый коммит в одну строку, что может быть очень удобным если вы просматриваете большое количество коммитов.

К тому же, опции `short`, `full` и `fuller` делают вывод приблизительно в том же формате, но с меньшим или большим количеством информации соответственно.

Наиболее интересной опцией является `format`, которая позволяет указать формат для вывода информации.

Полезные опции для `git log --pretty=format` отображает наиболее полезные опции для изменения формата.

Опции `oneline` и `format` являются особенно полезными с опцией `--graph` команды `log`. С этой опцией вы сможете увидеть небольшой граф в формате ASCII, который показывает текущую ветку и историю слияний.

2. Как ограничить вывод при просмотре истории коммитов?

Опция	Описание
<code>-(n)</code>	Показывает только последние <code>n</code> 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 <code>author</code> совпадает с указанной строкой.
<code>--committer</code>	Показывает только те коммиты, в которых запись <code>committer</code> 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?

Если вы хотите переделать коммит - внесите необходимые изменения, добавьте их в индекс и сделайте коммит ещё раз, указав параметр `--amend`: Эта команда использует область подготовки (индекс) для внесения правок в коммит.

```
$ git commit --amend
```

4. Как отменить индексацию файла в Git?

Используйте `git reset HEAD ...` для исключения из индекса.

5. Как отменить изменения в файле?

`Git checkout --` опасная команда. Все локальные изменения в файле пропадут — Git просто заменит его версией из последнего коммита.

6. Что такое удаленный 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?

Для того, чтобы просмотреть список настроенных удалённых репозиториях, вы можете запустить команду `git remote`. Она выведет названия доступных удалённых репозиториях.

8. Как добавить удаленный репозиторий для данного локального репозитория?

Для того, чтобы добавить удалённый репозиторий и присвоить ему имя (short name), просто выполните команду `git remote add`

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

Когда вы хотите поделиться своими наработками, вам необходимо отправить их в удалённый репозиторий. Команда для этого действия простая: `git push`

10. Как выполнить просмотр удаленного репозитория?

Если хотите получить побольше информации об одном из удалённых репозиториях, вы можете использовать команду `git remote show<remote>`.

11. Каково назначение тэгов Git?

Как и большинство СКВ, Git имеет возможность пометить определённые моменты в истории как важные. Такие пометки в Git называются тегами.

12. Как осуществляется работа с тэгами Git?

Просмотреть список имеющихся тегов в Git можно очень просто. Достаточно набрать команду `git tag` (параметры `-l` и `--list` опциональны): Данная команда перечисляет теги в алфавитном порядке; порядок их отображения не имеет существенного значения. Так же можно выполнить поиск тега по шаблону.

Создание тегов Git использует два основных типа тегов: легковесные и аннотированные. Легковесный тег — это что-то очень похожее на ветку, которая не изменяется — просто указатель на определённый коммит. А вот аннотированные теги хранятся в базе данных Git как полноценные объекты. Обычно рекомендуется создавать аннотированные теги, чтобы иметь всю перечисленную информацию; но если вы хотите сделать временную метку или по какой-то причине не хотите сохранять остальную информацию, то для этого годятся и легковесные.

Создание аннотированного тега в Git.

Самый простой способ — это указать - а при выполнении команды `tag`: Опция `-m` задаёт сообщение, которое будет храниться вместе с тегом. Если не указать сообщение, то Git запустит редактор, чтобы вы смогли его ввести. С помощью команды `git show` вы можете посмотреть данные тега вместе с коммитом.

Обмен тегами. Процесс аналогичен отправке веток — достаточно выполнить команду `git push origin <tag name>`. Если у вас много тегов, и вам хотелось бы отправить все за один раз, то можно использовать опцию `--tags` для команды `git push`. В таком случае все ваши теги отправятся на удалённый сервер (если только их уже там нет).

Удаление тегов.

Для удаления тега в локальной репозитории достаточно выполнить команду `git tag -d`.

Переход на тег.

Если вы хотите получить версии файлов, на которые указывает тег, то вы можете сделать `git checkout` для тега.

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

`--prune`

Если вы хотите получить все данные и одновременно удалить удаленные данные, добавьте флаг `--prune` в команде `git fetch`

Удалите удаленные ветки, у которых нет локального аналога. Например, удаленная ветка tmp будет удалена, если локальная ветка с таким же именем больше не существует. Это также учитывает refspecs, например `git push -- prune remote refs/heads/*:refs/tmp/*`, убедитесь, что удаленный объект refs/tmp/foобудет удален, если refs/heads/foo он не существует.