

**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
Коновалова В.Н. « 10 » сентября 2022г.

Подпись студента _____

Работа защищена « » _____ 20__ г.

Проверил Воронкин Р.А. _____
(подпись)

Ставрополь 2022

Ход работы:

- 1. Изучить теоретический материал работы.
- 2. Создать общедоступный репозиторий на GitHub, в котором будет использована лицензия MIT и выбранный Вами язык программирования (выбор языка программирования будет доступен после установки флажка Add .gitignore).
- 3. Выполните клонирование созданного репозитория на рабочий компьютер.
- 4. Дополните файл .gitignore необходимыми правилами для выбранного языка

Owner *

Repository name *

VictoriaKonovalova

/ Lr_2

Great repository names are short and easy to remember. Lr_2 is available. Need inspiration? How about friendly-

Description (optional)

☒ Public

Anyone on the internet can see this repository. You choose who can commit.

☐ Private

You choose who can see and commit to this repository.

Initialize this repository with:

Skip this step if you're importing an existing repository.

☒ Add a README file

This is where you can write a long description for your project. [Learn more.](#)

Add .gitignore

Choose which files not to track from a list of templates. [Learn more.](#)

.gitignore template: None

Choose a license

A license tells others what they can and can't do with your code. [Learn more.](#)

License: None

ⓘ You are creating a public repository in your personal account.

Create repository

main

1 branch

0 tags

Go to file

Add file

Code

	VictoriaKonovalova Update readme.txt	3bc78c1 1 minute ago	🕒 5 commits
📁	ConsoleApplication_Lr1	commit	4 minutes ago
📄	.gitignore	Initial commit	8 minutes ago
📄	LICENSE	Initial commit	8 minutes ago
📄	README.md	Update README.md	2 minutes ago
📄	README_new.txt	commit	2 minutes ago
📄	readme.txt	Update readme.txt	1 minute ago

5. Проработайте примеры лабораторной работы. Отрадите вывод на консоли при выполнении команд git в отчете для лабораторной работы.

```
vika1@LAPTOP-3EVI6GNJ MINGW64 ~/OneDrive/Desktop/univer/Основы программной инженерии/Lr2 (main)
$ git log
commit 1f50d368e6e41ed931fe87ee39338a6d999e82a9 (HEAD -> main, origin/main, origin/HEAD)
Author: VictoriaKonovalova <112749149+VictoriaKonovalova@users.noreply.github.com>
Date:   Wed Nov 9 06:05:51 2022 +0300

    commit

commit 6fc363db537a4770458e0afe440c180545dffaff
Author: VictoriaKonovalova <112749149+VictoriaKonovalova@users.noreply.github.com>
Date:   Wed Nov 9 06:04:20 2022 +0300

    commit

commit 5382df234cf14432aca4a3159e30633f3785548c
Author: VictoriaKonovalova <112749149+VictoriaKonovalova@users.noreply.github.com>
Date:   Wed Nov 9 06:00:01 2022 +0300

    Initial commit
```

Рисунок 1 – Результат выполнения команды git log

```
vika1@LAPTOP-3EVI6GNJ MINGW64 ~/OneDrive/Desktop/univer/Основы программной инженерии/Lr2 (main)
$ git log -p -1
commit 1f50d368e6e41ed931fe87ee39338a6d999e82a9 (HEAD -> main, origin/main, origin/HEAD)
Author: VictoriaKonovalova <112749149+VictoriaKonovalova@users.noreply.github.com>
Date:   Wed Nov 9 06:05:51 2022 +0300

    commit

diff --git a/README_new.txt b/README_new.txt
index 9f26980..67cce05 100644
--- a/README_new.txt
+++ b/README_new.txt
@@ -1,2 @@
-Новый README файл
+Коновалова Вика
\ No newline at end of file
```

Рисунок 2 – Результат выполнения команды git log -p -1

```
vika1@LAPTOP-3EVI6GNJ MINGW64 ~/OneDrive/Desktop/univer/Основы программной инженерии/Lr2 (main)
$ git log --stat
commit 1f50d368e6e41ed931fe87ee39338a6d999e82a9 (HEAD -> main, origin/main, origin/HEAD)
Author: VictoriaKonovalova <112749149+VictoriaKonovalova@users.noreply.github.com>
Date:   Wed Nov 9 06:05:51 2022 +0300

    commit

    README_new.txt | 1 +
    1 file changed, 1 insertion(+)

commit 6fc363db537a4770458e0afe440c180545dffaff
Author: VictoriaKonovalova <112749149+VictoriaKonovalova@users.noreply.github.com>
Date:   Wed Nov 9 06:04:20 2022 +0300

    commit

    ConsoleApplication_Lr1/ConsoleApplication_Lr1.cpp | 16 +++
    ConsoleApplication_Lr1/ConsoleApplication_Lr1.sln | 31 +++++
    .../ConsoleApplication_Lr1.vcxproj | 147 +++++
    .../ConsoleApplication_Lr1.vcxproj.filters | 22 +++
    README.md | 4 +-
    README_new.txt | 1 +
    readme.txt | 4 +
    7 files changed, 224 insertions(+), 1 deletion(-)

commit 5382df234cf14432aca4a3159e30633f3785548c
Author: VictoriaKonovalova <112749149+VictoriaKonovalova@users.noreply.github.com>
Date:   Wed Nov 9 06:00:01 2022 +0300

    Initial commit

    .gitignore | 350 +++++
    LICENSE | 21 +++
    README.md | 1 +
    3 files changed, 372 insertions(+)
```

Рисунок 3 – Результат выполнения команды git log --stat

```
vika1@LAPTOP-3EV1GGNJ MINGW64 ~/OneDrive/Desktop/univer/Основы программной инженерии/Lr2 (main)
$ git log --pretty=oneline
1f50d368e6e41ed931fe87ee39338a6d999e82a9 (HEAD -> main, origin/main, origin/HEAD) commit
6fc363db537a4770458e0afe440c180545dffaff commit
5382df234cf14432aca4a3159e30633f3785548c Initial commit
```

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

```
vika1@LAPTOP-3EV1GGNJ MINGW64 ~/OneDrive/Desktop/univer/Основы программной инженерии/Lr2 (main)
$ git log --pretty=short
commit 1f50d368e6e41ed931fe87ee39338a6d999e82a9 (HEAD -> main, origin/main, origin/HEAD)
Author: VictoriaKonovalova <112749149+VictoriaKonovalova@users.noreply.github.com>

    commit

commit 6fc363db537a4770458e0afe440c180545dffaff
Author: VictoriaKonovalova <112749149+VictoriaKonovalova@users.noreply.github.com>

    commit

commit 5382df234cf14432aca4a3159e30633f3785548c
Author: VictoriaKonovalova <112749149+VictoriaKonovalova@users.noreply.github.com>

    Initial commit
```

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

```
vika1@LAPTOP-3EV1GGNJ MINGW64 ~/OneDrive/Desktop/univer/Основы программной инженерии/Lr2 (main)
$ git log --pretty=full
commit 1f50d368e6e41ed931fe87ee39338a6d999e82a9 (HEAD -> main, origin/main, origin/HEAD)
Author: VictoriaKonovalova <112749149+VictoriaKonovalova@users.noreply.github.com>
Commit: VictoriaKonovalova <112749149+VictoriaKonovalova@users.noreply.github.com>

    commit

commit 6fc363db537a4770458e0afe440c180545dffaff
Author: VictoriaKonovalova <112749149+VictoriaKonovalova@users.noreply.github.com>
Commit: VictoriaKonovalova <112749149+VictoriaKonovalova@users.noreply.github.com>

    commit

commit 5382df234cf14432aca4a3159e30633f3785548c
Author: VictoriaKonovalova <112749149+VictoriaKonovalova@users.noreply.github.com>
Commit: GitHub <noreply@github.com>

    Initial commit
```

Рисунок 6 – Результат выполнения команды `git log --pretty=full`

```

vika1@LAPTOP-3EV1GGNJ MINGW64 ~/OneDrive/Desktop/univer/Основы программной инженерии/Lr2 (main)
$ git log --pretty=fuller
commit 1f50d368e6e41ed931fe87ee39338a6d999e82a9 (HEAD -> main, origin/main, origin/HEAD)
Author: VictoriaKonovalova <112749149+VictoriaKonovalova@users.noreply.github.com>
AuthorDate: Wed Nov 9 06:05:51 2022 +0300
Commit: VictoriaKonovalova <112749149+VictoriaKonovalova@users.noreply.github.com>
CommitDate: Wed Nov 9 06:05:51 2022 +0300

    commit

commit 6fc363db537a4770458e0afe440c180545dffa9f
Author: VictoriaKonovalova <112749149+VictoriaKonovalova@users.noreply.github.com>
AuthorDate: Wed Nov 9 06:04:20 2022 +0300
Commit: VictoriaKonovalova <112749149+VictoriaKonovalova@users.noreply.github.com>
CommitDate: Wed Nov 9 06:04:20 2022 +0300

    commit

commit 5382df234cf14432aca4a3159e30633f3785548c
Author: VictoriaKonovalova <112749149+VictoriaKonovalova@users.noreply.github.com>
AuthorDate: Wed Nov 9 06:00:01 2022 +0300
Commit: GitHub <noreply@github.com>
CommitDate: Wed Nov 9 06:00:01 2022 +0300

    Initial commit

```

Рисунок 7 – Результат выполнения команды `git log --pretty=fuller`

```

vika1@LAPTOP-3EV1GGNJ MINGW64 ~/OneDrive/Desktop/univer/Основы программной инженерии/Lr2 (main)
$ git log --pretty=format:"%h - %an, %ar : %s"
1f50d36 - VictoriaKonovalova, 13 minutes ago : commit
6fc363d - VictoriaKonovalova, 15 minutes ago : commit
5382df2 - VictoriaKonovalova, 19 minutes ago : Initial commit

```

Рисунок 8 – Результат выполнения команды `git log --pretty=format:"%h - %an, %ar : %s"`

```

vika1@LAPTOP-3EV1GGNJ MINGW64 ~/OneDrive/Desktop/univer/Основы программной инженерии/Lr2 (main)
$ git log --pretty=format:"%h %s" --graph
* 1f50d36 commit
* 6fc363d commit
* 5382df2 Initial commit

```

Рисунок 9 – Результат выполнения команды `git log --pretty=format:"%h %s" --graph`

```

vika1@LAPTOP-3EV1GGNJ MINGW64 ~/OneDrive/Desktop/univer/Основы программной инженерии/Lr2 (main)
$ git log --since=2.weeks
commit 1f50d368e6e41ed931fe87ee39338a6d999e82a9 (HEAD -> main, origin/main, origin/HEAD)
Author: VictoriaKonovalova <112749149+VictoriaKonovalova@users.noreply.github.com>
Date: Wed Nov 9 06:05:51 2022 +0300

    commit

commit 6fc363db537a4770458e0afe440c180545dffaff
Author: VictoriaKonovalova <112749149+VictoriaKonovalova@users.noreply.github.com>
Date: Wed Nov 9 06:04:20 2022 +0300

    commit

commit 5382df234cf14432aca4a3159e30633f3785548c
Author: VictoriaKonovalova <112749149+VictoriaKonovalova@users.noreply.github.com>
Date: Wed Nov 9 06:00:01 2022 +0300

    Initial commit

```

Рисунок 10 – Результат выполнения команды git log --since=2.weeks

```

vika1@LAPTOP-3EV1GGNJ MINGW64 ~/OneDrive/Desktop/univer/Основы программной инженерии/Lr2 (main)
$ git log -S cout
commit 6fc363db537a4770458e0afe440c180545dffaff
Author: VictoriaKonovalova <112749149+VictoriaKonovalova@users.noreply.github.com>
Date: Wed Nov 9 06:04:20 2022 +0300

    commit

```

Рисунок 11 – Результат выполнения команды git log -S cout

Операции отмены.

```

vika1@LAPTOP-3EV1GGNJ MINGW64 ~/OneDrive/Desktop/univer/Основы программной инженерии/Lr2 (main)
$ git commit --amend
hint: Waiting for your editor to close the file... unix2dos: converting file c:/Users/vika1/OneDrive/Desktop/univer/Основы программной инженерии/Lr2/.git/COMMIT_EDITMSG to unix format...
dos2unix: converting file c:/Users/vika1/OneDrive/Desktop/univer/Основы программной инженерии/Lr2/.git/COMMIT_EDITMSG to unix format...
[main 74512ef] Merge branch 'main' of https://github.com/VictoriaKonovalova/Lr2
Date: Wed Nov 9 06:38:42 2022 +0300

```

Рисунок 12 – Результат выполнения команды git commit --amend.

Отмена индексации файла

```

vika1@LAPTOP-3EV1GGNJ MINGW64 ~/OneDrive/Desktop/univer/Основы программной инженерии/Lr2 (main)
$ git restore --stage readme.txt

vika1@LAPTOP-3EV1GGNJ MINGW64 ~/OneDrive/Desktop/univer/Основы программной инженерии/Lr2 (main)
$ git status
On branch main
Your branch is ahead of 'origin/main' by 2 commits.
(use "git push" to publish your local commits)

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

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:   readme.txt

```

Рисунок 13 – Результат выполнения команды git restore --staged <file>...

Работа с удалёнными репозиториями. Просмотр удалённых репозиторияев.

```
vika1@LAPTOP-3EV1GGNJ MINGW64 ~/OneDrive/Desktop/univer/Основы программной инженерии/Lr2 (main)
$ git remote
origin
```

Рисунок 14 – Результат выполнения команды git remote

```
vika1@LAPTOP-3EV1GGNJ MINGW64 ~/OneDrive/Desktop/univer/Основы программной инженерии/Lr2 (main)
$ git remote
origin

vika1@LAPTOP-3EV1GGNJ MINGW64 ~/OneDrive/Desktop/univer/Основы программной инженерии/Lr2 (main)
$ git remote -v
origin https://github.com/VictoriaKonovalova/Lr2.git (fetch)
origin https://github.com/VictoriaKonovalova/Lr2.git (push)
```

Рисунок 15 – Результат выполнения команды git remote -v

Добавление удалённых репозиторияев.

```
vika1@LAPTOP-3EV1GGNJ MINGW64 ~/OneDrive/Desktop/univer/Основы программной инженерии
$ git remote add pb https://github.com/VictoriaKonovalova/opi2

vika1@LAPTOP-3EV1GGNJ MINGW64 ~/OneDrive/Desktop/univer/Основы программной инженерии
/Lr2 (main)
$ git remote -v
origin https://github.com/VictoriaKonovalova/Lr2.git (fetch)
origin https://github.com/VictoriaKonovalova/Lr2.git (push)
pb https://github.com/VictoriaKonovalova/opi2 (fetch)
pb https://github.com/VictoriaKonovalova/opi2 (push)
```

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

Получение изменений из удалённого репозитория — Fetch и Pull.

```
vika1@LAPTOP-3EV1GGNJ MINGW64 ~/OneDrive/Desktop/univer/Основы программной инженерии
/Lr2 (main)
$ git fetch pb
remote: Enumerating objects: 8, done.
remote: Counting objects: 100% (8/8), done.
remote: Compressing objects: 100% (5/5), done.
remote: Total 8 (delta 1), reused 4 (delta 1), pack-reused 0
Unpacking objects: 100% (8/8), 1.68 KiB | 42.00 KiB/s, done.
From https://github.com/VictoriaKonovalova/opi2
* [new branch]      main      -> pb/main
```

Рисунок 17 – Результат выполнения команды git fetch [remote-name]

Просмотр удаленного репозитория

```

vika1@LAPTOP-3EV1GGNJ MINGW64 ~/OneDrive/Desktop/univer/Основы программной инженерии
/Lr2 (main)
$ git remote show pb
* remote pb
Fetch URL: https://github.com/VictoriaKonovalova/opi2
Push URL: https://github.com/VictoriaKonovalova/opi2
HEAD branch: main
Remote branch:
  main tracked
Local ref configured for 'git push':
  main pushes to main (local out of date)

```

Рисунок 18 – Результат выполнения команды git remote show origin

Удаление и переименование удалённых репозитория

```

vika1@LAPTOP-3EV1GGNJ MINGW64 ~/OneDrive/Desktop/univer/Основы программной инженерии
/Lr2 (main)
$ git remote rename pb paul
Renaming remote references: 100% (1/1), done.

```

Рисунок 19 – Результат выполнения команды git remote rename

```

vika1@LAPTOP-3EV1GGNJ MINGW64 ~/OneDrive/Desktop/univer/Основы программной инженерии
/Lr2 (main)
$ git remote remove paul

vika1@LAPTOP-3EV1GGNJ MINGW64 ~/OneDrive/Desktop/univer/Основы программной инженерии
/Lr2 (main)
$ git remote
origin

```

Рисунок 20 – Результат выполнения команды git remote remove

Работа с тегами

Просмотр списка тегов

```

vika1@LAPTOP-3EV1GGNJ MINGW64 ~/OneDrive/Desktop/univer/Основы программной инженерии
/Lr2 (main)
$ git tag

vika1@LAPTOP-3EV1GGNJ MINGW64 ~/OneDrive/Desktop/univer/Основы программной инженерии
/Lr2 (main)
$ git tag -l

```

Рисунок 21 – Результат выполнения команды git tag и git tag -l

Создание тегов

Аннотированные теги

V1

```
vika1@LAPTOP-3EV1GGNJ MINGW64 ~/OneDrive/Desktop/univer/Основы программной инженерии/Lr2 (main)
$ git tag -a v1.4 -m "my version 1.4"

vika1@LAPTOP-3EV1GGNJ MINGW64 ~/OneDrive/Desktop/univer/Основы программной инженерии/Lr2 (main)
$ git tag
v1.4
```

Рисунок 22 – Результат выполнения команды git tag -a -m

```
vika1@LAPTOP-3EV1GGNJ MINGW64 ~/OneDrive/Desktop/univer/Основы программной инженерии/Lr2 (main)
$ git show v1.4
tag v1.4
Tagger: VictoriaKonovalova <112749149+VictoriaKonovalova@users.noreply.github.com>
Date: Sat Nov 12 01:07:19 2022 +0300

my version 1.4

commit 334cdd4b0124c15ad081ee5cdf0ed7b0de9f3a37 (HEAD -> main, tag: v1.4, origin/main, origin/HEAD)
Author: VictoriaKonovalova <112749149+VictoriaKonovalova@users.noreply.github.com>
Date: Wed Nov 9 06:48:26 2022 +0300

    Committ

diff --git a/README_new.txt b/README_new.txt
index 15b6619..dde8f1d 100644
--- a/README_new.txt
+++ b/README_new.txt
@@ -1,2 +1,3 @@
-новый RAEADME файл
+Коновалова Вика
+Lr2
diff --git a/readme.txt b/readme.txt
index ca92402..957b33c 100644
--- a/readme.txt
+++ b/readme.txt
@@ -1 +1,2 @@
-Лабораторна работа №2
+Lr2
```

Рисунок 23 – Результат выполнения команды git show v1.4

Обмен тегами

```
vika1@LAPTOP-3EV1GGNJ MINGW64 ~/OneDrive/Desktop/univer/Основы программной инженерии/Lr2 (main)
$ git push origin v1.4
Enumerating objects: 1, done.
Counting objects: 100% (1/1), done.
Writing objects: 100% (1/1), 186 bytes | 186.00 KiB/s, done.
Total 1 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/VictoriaKonovalova/Lr2.git
 * [new tag]          v1.4 -> v1.4
```

Рисунок 24 – Результат выполнения команды `git push origin <tagname>`

Удаление тегов

```
vika1@LAPTOP-3EV1GGNJ MINGW64 ~/OneDrive/Desktop/univer/Основы программной инженерии/Lr2 (main)
$ git push origin :refs/tags/v1.4-lw
remote: warning: deleting a non-existent ref
To https://github.com/VictoriaKonovalova/Lr2.git
 - [deleted]          v1.4-lw
```

Рисунок 25 – Результат выполнения команды `git push origin :refs/tags/v1.4-lw`

Переход на тег

```
vika1@LAPTOP-3EV1GGNJ MINGW64 ~/OneDrive/Desktop/univer/Основы программной инженерии/Lr2 (main)
$ git checkout v1.4
Note: switching to 'v1.4'.

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 334cdd4 Committ
```

Рисунок 26 – Результат выполнения команды `git checkout v.1.0`

6. Добавьте в файл `README.md` информацию о дисциплине, группе и ФИО студента, выполняющего лабораторную работу.

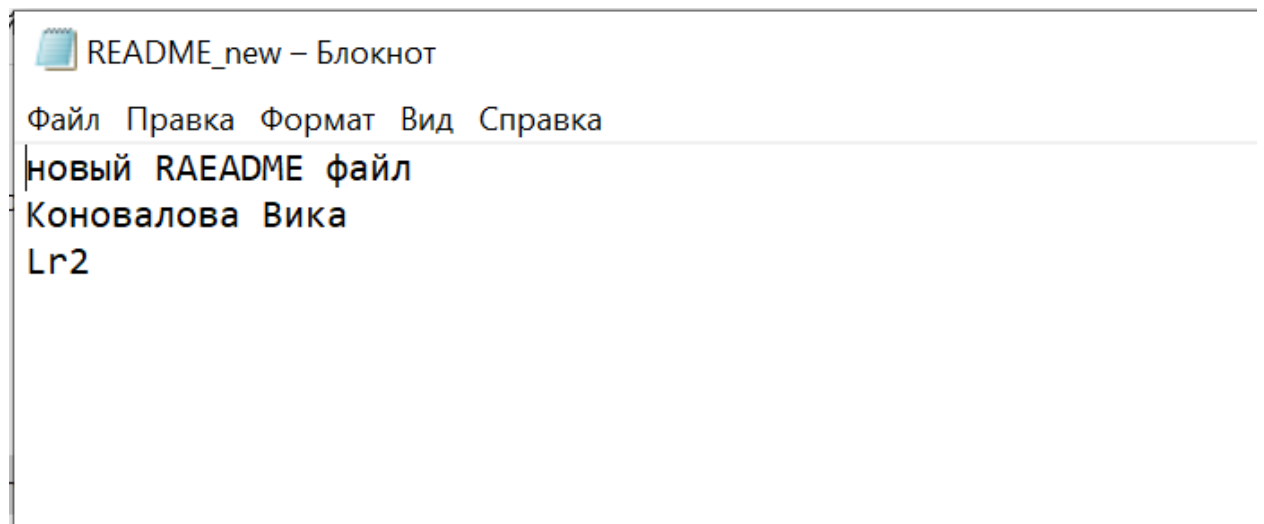


Рисунок 29 – Дополнение файла README.md

7. Напишите небольшую программу на выбранном Вами языке программирования. Фиксируйте изменения при написании программы в локальном репозитории. Должно быть сделано не менее 7 коммитов, отмеченных не менее 3 тэгами.









 VictoriaKonovalova Update Lr_2.cpp		4ad476b 5 minutes ago	 21 commits
 Lr_2	Update Lr_2.cpp	5 minutes ago	
 .gitignore	Initial commit	3 days ago	
 LICENSE	Initial commit	3 days ago	
 README.md	Update README.md	3 days ago	
 README_new.txt	Committ	3 days ago	
 readme.txt	Committ	3 days ago	

Рисунок 30 – Коммиты и теги в Github

8. Просмотреть историю (журнал) хранилища командой `git log` . Например, с помощью команды `git log --graph --pretty=oneline --abbrev-commit` .
Добавить скриншот консоли с выводом в отчет по лабораторной работе.

```

vika1@LAPTOP-3EV1GGNJ MINGW64 ~/OneDrive/Desktop/univer/Основы программной инженерии
/Lr2 (main)
$ git log --graph --pretty=oneline --abbrev-commit
* 334cdd4 (HEAD -> main, tag: v1.5, tag: v1.4-lw, tag: v1.4, origin/main, origin/HEA
D) Committ
* 044205e Merge branch 'main' of https://github.com/VictoriaKonovalova/Lr2
|
| * 4b857eb Merge branch 'main' of https://github.com/VictoriaKonovalova/Lr2
|
| * 74512ef Merge branch 'main' of https://github.com/VictoriaKonovalova/Lr2
|
|
| * 3bc78c1 Update readme.txt
| * e5d2994 Update README.md
| * 88132a8 Commit
| * fa6a677 commit
|/
* 1f50d36 commit
* 6fc363d commit
* 5382df2 Initial commit

```

Рисунок 31 – Результат выполнения команды `git log --graph --pretty=oneline --abbrev-commit`.

9. Просмотреть содержимое коммитов командой `git show`, где :
 HEAD : последний коммит; HEAD~1 : предпоследний коммит (и т. д.); b34a0e : коммит с указанным хэшем. Отобразите результаты работы этих команд в отчете.

```

$ git show HEAD
commit 334cdd4b0124c15ad081ee5cdf0ed7b0de9f3a37 (HEAD -> main, tag: v1.5, tag: v1.4-lw, tag: v1.4, origin/main, origin/HEAD)
Author: VictoriaKonovalova <112749149+VictoriaKonovalova@users.noreply.github.com>
Date: Wed Nov 9 06:48:26 2022 +0300

    Committ

diff --git a/README_new.txt b/README_new.txt
index 15b6619..dde8f1d 100644
--- a/README_new.txt
+++ b/README_new.txt
@@ -1,2 +1,3 @@
-новый README файл
+Коновалова Вика
+Lr2
diff --git a/readme.txt b/readme.txt
index ca92402..957b33c 100644
--- a/readme.txt
+++ b/readme.txt
@@ -1 +1,2 @@
-Лабораторна работа №2
+Lr2

```

Рисунок 32 – Результат выполнения команды `git show HEAD`

```

vika1@LAPTOP-3EV1GGNJ MINGW64 ~/OneDrive/Desktop/univer/Основы программной инженерии/Lr2 (main)
$ git show HEAD~1
commit 044205e7e82e01da2e172c3a52b48b9096c1ef6c
Merge: 74512ef 4b857eb
Author: VictoriaKonovalova <112749149+VictoriaKonovalova@users.noreply.github.com>
Date:   Wed Nov 9 06:44:18 2022 +0300

Merge branch 'main' of https://github.com/VictoriaKonovalova/Lr2

```

Рисунок 33 – Результат выполнения команды git show HEAD~1

```

vika1@LAPTOP-3EV1GGNJ MINGW64 ~/OneDrive/Desktop/univer/Основы про
$ git show 334cdd
commit 334cdd4b0124c15ad081ee5cdf0ed7b0de9f3a37 (HEAD -> main, tag:
1w, tag: v1.4, origin/main, origin/HEAD)
Author: VictoriaKonovalova <112749149+VictoriaKonovalova@users.nor
Date:   Wed Nov 9 06:48:26 2022 +0300

Committ

diff --git a/README_new.txt b/README_new.txt
index 15b6619..dde8f1d 100644
--- a/README_new.txt
+++ b/README_new.txt
@@ -1,2 +1,3 @@
  новый RAЕADME файл
  Коновалова Вика
+Lr2
diff --git a/readme.txt b/readme.txt
index ca92402..957b33c 100644
--- a/readme.txt
+++ b/readme.txt
@@ -1 +1,2 @@
  Лабораторна работа № 2
+Lr2

```

Рисунок 34– Результат выполнения команды git show (с предпоследним коммитом 334cdd)

Вопросы для защиты работы

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

Для просмотра истории коммитов в git предусмотрена команда git log.

Одним из самых полезных аргументов является -p или --patch, который показывает разницу (выводит патч), внесенную в каждый коммит. Также вы можете ограничить количество записей в выводе команды; используйте параметр -2 для вывода только двух записей. Если вы хотите увидеть сокращенную статистику для каждого коммита, вы можете использовать

опцию `--stat`. Полезные опции для `git log --pretty=format` отображает наиболее полезные опции для изменения формата (`%H` Хеш коммита, `%h` Сокращенный хеш коммита, `%T` Хеш дерева, `%t` Сокращенный хеш дерева, `%P` Хеш родителей, `%p` Сокращенный хеш родителей, `%an` Имя автора, `%ae` Электронная почта автора, `%ad` Дата автора (формат даты можно задать опцией `--date=option`), `%ar` Относительная дата автора, `%cn` Имя коммитера, `%ce` Электронная почта коммитера, `%cd` Дата коммитера, `%cr` Относительная дата коммитера, `%s` Содержание). Команда `--shortstat` отображает только строку с количеством изменений/вставок/удалений для команды `--stat`; `--name-only` показывает список измененных файлов после информации о коммите; `--name-status` показывает список файлов, которые добавлены/изменены/удалены; `--abbrev-commit` показывает только несколько символов SHA-1 чек-суммы вместо всех 40; `--relative-date` отображает дату в относительном формате (например, «2 weeks ago») вместо стандартного формата даты; `--graph` отображает ASCII граф с ветвлениями и историей слияний; `--pretty` показывает коммиты в альтернативном формате. Возможные варианты опций: `oneline`, `short`, `full`, `fuller` и `format` (с помощью последней можно указать свой формат); `-oneline` сокращение для одновременного использования опций `--pretty=oneline` `--abbrev-commit`.

2. Как ограничить вывод при просмотре истории коммитов?

Можно использовать `git log -<n>`, где `n` — это любое натуральное число и представляет собой `n` последних коммитов. Команда `--since` работает с большим количеством форматов — вы можете указать определенную дату вида `2008-01-15` или же относительную дату, например `2 years 1 day 3 minutes ago`. Команды `--until`, `--before` показывают только те коммиты, которые были сделаны до указанной даты; `--author` показывает только те коммиты, в которых запись `author` совпадает с указанной строкой; `--committer`. Показывает только те коммиты, в которых запись `committer` совпадает с указанной строкой; `--grep`

показывает только коммиты, сообщение которых содержит указанную строку;
-S 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`.

4. Как отменить индексацию файла в Git?

Для отмены индексации файла можно и команды `git restore --staged <file>...`

5. Как отменить изменения в файле?

Для этого можно воспользоваться командой `"git checkout -- <file>..."`

6. Что такое удаленный репозиторий Git?

Удалённые репозитории - это версии проекта, сохраненные в интернете или ещё где-то в сети.

7. Как выполнить просмотр удаленных репозиториях данного локального репозитория?

Для осуществления этой функции можно воспользоваться командой `git remote`, которая выводит названия удаленных репозиториях, где `origin` – имя по умолчанию, которое Git даёт серверу, с которого производилось клонирование. Ключ `-v` поможет вывести адреса для чтения и записи.

8. Как добавить удаленный репозиторий для данного локального репозитория?

Для этого необходимо воспользоваться командой `git remote add <shortname> <url>`, где `<shortname>` – имя, присваиваемое удаленному репозиторию, а `<url>` – его адресс.

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

Для получения данных можно прибегнуть к команде `git fetch <remote-name>`. Также можно использовать команду `git pull` чтобы автоматически получить изменения из удалённой ветки и слить их со своей текущей.

Для отправки изменений воспользуемся `git push <remote-name> <branch-name>`, где `<remote-name>` – сервер, `<branch-name>` – ветка.

10. Как выполнить просмотр удаленного репозитория?

Для этого нам понадобится команда `git remote show <remote>`.

11. Каково назначение тэгов Git?

Тэги дают возможность пометать определённые моменты в истории как важные

12. Как осуществляется работа с тэгами Git?

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. Они имеют контрольную сумму, содержат имя автора, его e-mail и дату создания, имеют комментарий

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

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.