

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

**Кафедра инфокоммуникаций
«Исследование основных возможностей Git и GitHub»**

**Отчет по лабораторной работе № 1.2
по дисциплине «Основы программной инженерии»**

Выполнил студент группы ПИЖ-б-о-21-1

Кучеренко С.Ю. « 24 » сентября 2022г.

Подпись студента _____

Работа защищена « » _____ 2022г.

Проверил Воронкин Р.А. _____
(подпись)

Ставрополь 2022

Цель работы: исследовать базовые возможности системы контроля версий Git и веб-сервиса для хостинга IT-проектов GitHub.

Выполнение 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).

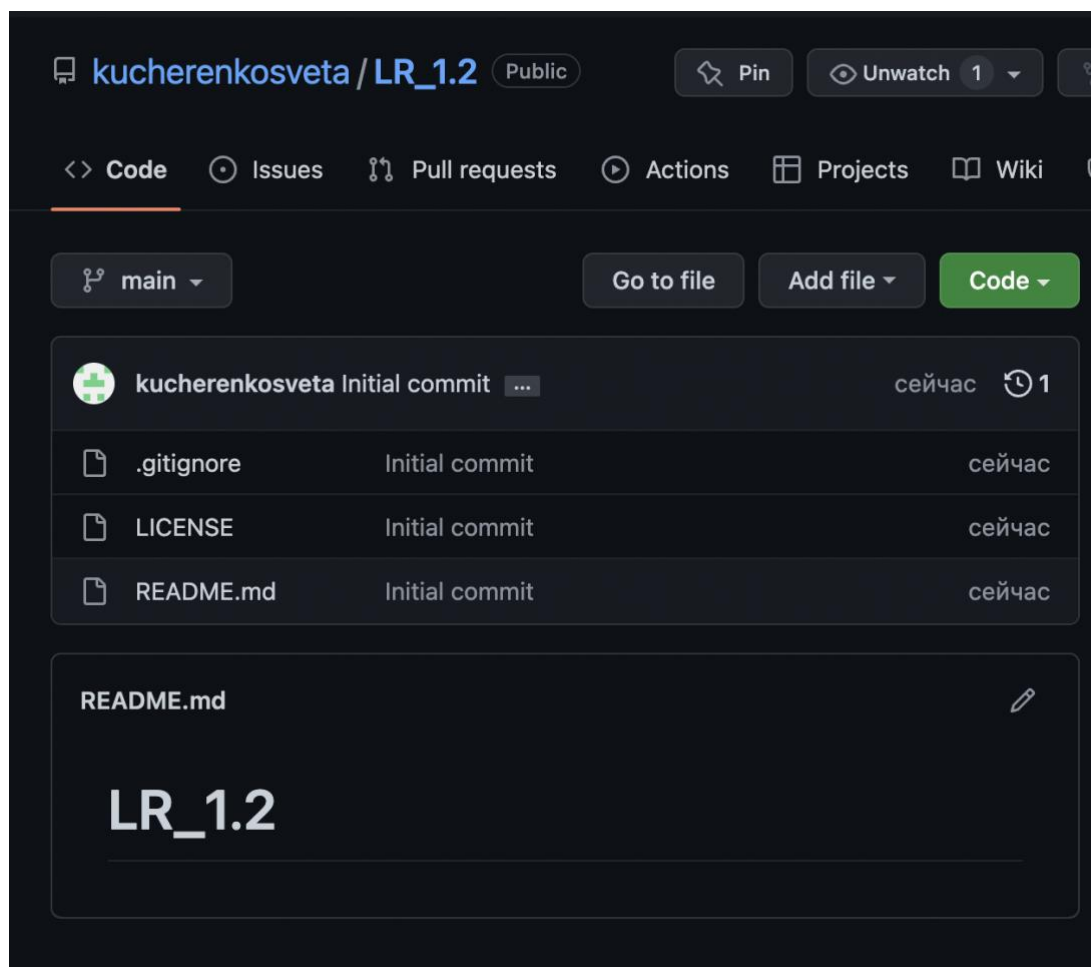


Рисунок 1 - Созданный репозиторий в GitHub

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

```
commit cd5f9a35187ffaf764a7d633bb49ee2acb45d73b (HEAD -> main)
Author: kucherenskova <kucherenko55sss55@gmail.com>
Date: Sat Oct 22 01:29:20 2022 +0300

7save

commit 20916cf0f5db440db1b88206d24f83c0bc765742
Author: kucherenskova <kucherenko55sss55@gmail.com>
Date: Sat Oct 22 01:28:23 2022 +0300

6save

commit 48ff8fd37f75640af5f32ea4c2a38899350c9099
Author: kucherenskova <kucherenko55sss55@gmail.com>
Date: Sat Oct 22 01:27:19 2022 +0300

5save

commit 1ba03a0118a04e432daf4321d1fb0cf35f346504
Author: kucherenskova <kucherenko55sss55@gmail.com>
Date: Sat Oct 22 01:26:53 2022 +0300

4save

commit 20307fb975a474cffe5cc319cb36588babc9529
Author: kucherenskova <kucherenko55sss55@gmail.com>
Date: Sat Oct 22 01:26:16 2022 +0300

3save

commit d271ae5629e3a7ca66a939886bfcd19bc3961ffa
Author: kucherenskova <kucherenko55sss55@gmail.com>
Date: Sat Oct 22 01:25:47 2022 +0300

2save

commit 87f9b62ad43a7239ca142d72b1bbcd5b75c7ff7d
Author: kucherenskova <kucherenko55sss55@gmail.com>
Date: Sat Oct 22 01:24:19 2022 +0300

1save

commit f07152972df02a0c1b3b1ba93d9d76764d398863 (origin/main, origin/HEAD)
Author: kucherenskova <112749526+kucherenskova@users.noreply.github.com>
Date: Sat Oct 22 00:04:15 2022 +0300

Update README.md

commit cede93881ffe5789e117c49fd2cf4323bee28e10
Author: kucherenskova <112749526+kucherenskova@users.noreply.github.com>
Date: Fri Oct 21 23:55:39 2022 +0300

Initial commit
~
~
~
~
~
~
~
svetik@MacBook-Air-Svetik LR_1.2 %
```

Рисунок 2 – Результат выполнения команды git log

```
svetik@MacBook-Air-Svetik LR_1.2 % git log -p -1
commit cd5f9a35187ffaf764a7d633bb49ee2acb45d73b (HEAD -> main, origin/main, origin/HEAD)
Author: kucherenskosceta <kucherenko55sss55@gmail.com>
Date: Sat Oct 22 01:29:20 2022 +0300

    7save

diff --git a/2/2.xcodeproj/project.xcworkspace/xcuserdata/svetik.xcuserdatad/UserInterfaceState.xcuserstate b/2/2.xcodeproj/project.xcworkspace/xcuserdata/svetik.xcuserdatad/UserInterfaceState.xcuserstate
index 475c952..ecbffe5 100644
Binary files a/2/2.xcodeproj/project.xcworkspace/xcuserdata/svetik.xcuserdatad/UserInterfaceState.xcuserstate and b/2/2.xcodeproj/project.xcworkspace/xcuserdata/svetik.xcuserdatad/UserInterfaceState.xcuserstate differ
diff --git a/2/2/main.cpp b/2/2/main.cpp
index fde7fec..67b5e5f 100644
--- a/2/2/main.cpp
+++ b/2/2/main.cpp
@@ -2,5 +2,6 @@
 using namespace std;
 int main() {
     cout << "Hello, World!\n";
+    cout << "LR_1.2\n";
     return 0;
 }
svetik@MacBook-Air-Svetik LR_1.2 %
```

Рисунок 3 - Результат выполнения команды git log -p -1

```
    }
    return 0;
}
svetik@MacBook-Air-Svetik LR_1.2 % git log --stat
fatal: ambiguous argument '-stat': unknown revision or path not in the working tree.
Use '--' to separate paths from revisions, like this:
'git <command> [<revision>...] -- [<file>...]'
svetik@MacBook-Air-Svetik LR_1.2 % git log --stat
commit cd5f9a35187ffaf764a7d633bb49ee2acb45d73b (HEAD -> main, origin/main, origin/HEAD)
Author: kucherenskosceta <kucherenko55sss55@gmail.com>
Date: Sat Oct 22 01:29:20 2022 +0300

    7save

2/2.xcodeproj/project.xcworkspace/xcuserdata/svetik.xcuserdatad/UserInterfaceState.xcuserstate | Bin 16150 -> 16150 bytes
2/2/main.cpp | 1 +
2 files changed, 1 insertion(+)

commit 20916cf0f5db440db1b88206d24f83c0bc765742
Author: kucherenskosceta <kucherenko55sss55@gmail.com>
Date: Sat Oct 22 01:28:23 2022 +0300

    6save

2/2.xcodeproj/project.xcworkspace/xcuserdata/svetik.xcuserdatad/UserInterfaceState.xcuserstate | Bin 16143 -> 16150 bytes
2/2/main.cpp | 3 +-
2 files changed, 2 insertions(+), 1 deletion(-)

commit 48ff8fd37f75640af5f32ea4c2a38899350c9099
Author: kucherenskosceta <kucherenko55sss55@gmail.com>
Date: Sat Oct 22 01:27:19 2022 +0300

    5save

2/2.xcodeproj/project.xcworkspace/xcuserdata/svetik.xcuserdatad/UserInterfaceState.xcuserstate | Bin 16150 -> 16143 bytes
2/2/main.cpp | 1 -
2 files changed, 1 deletion(-)

commit 1ba03a0110a04e432daf4321d1fb0cf35f346504
Author: kucherenskosceta <kucherenko55sss55@gmail.com>
Date: Sat Oct 22 01:26:53 2022 +0300

    4save

2/2.xcodeproj/project.xcworkspace/xcuserdata/svetik.xcuserdatad/UserInterfaceState.xcuserstate | Bin 15786 -> 16150 bytes
2/2/main.cpp | 2 +-
2 files changed, 1 insertion(+), 1 deletion(-)

commit 20307fb975a474cffe5cc319cb36588babc9529
Author: kucherenskosceta <kucherenko55sss55@gmail.com>
Date: Sat Oct 22 01:26:16 2022 +0300

    3save

:
Author: kucherenskosceta <kucherenko55sss55@gmail.com>
Date: Sat Oct 22 01:29:20 2022 +0300

    7save

2/2.xcodeproj/project.xcworkspace/xcuserdata/svetik.xcuserdatad/UserInterfaceState.xcuserstate | Bin 16150 -> 16150 bytes
2/2/main.cpp | 1 +
2 files changed, 1 insertion(+)
```

Рисунок 4 – Результат выполнения команды git log --stat

```
svetik@MacBook-Air-Svetik LR_1.2 % git log --pretty=oneline
cd5f9a35187ffaf764a7d633bb49ee2acb45d73b (HEAD -> main, origin/main, origin/HEAD) 7save
20916cf0f5db440db1b88206d24f83c0bc765742 6save
48ff8fd37f75640af5f32ea4c2a38899350c9099 5save
1ba03a0118a04e432daf4321d1fb0cf35f346504 4save
20307fb975a474cffe5cc319cb36588babc9529 3save
d271ae5629e3a7ca66a939886bfcd19bc3961ffa 2save
87f9b62ad43a7239ca142d72b1bbcd5b75c7ff7d 1save
f07152972df02a0c1b3b1ba93d9d76764d398863 Update README.md
cede93881ffe5789e117c49fd2cf4323bee28e10 Initial commit
svetik@MacBook-Air-Svetik LR_1.2 %
```

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

```
svetik@MacBook-Air-Svetik LR_1.2 % git log --pretty=format:"%h - %an, %ar : %s"
cd5f9a3 - kucherenkosveta, 24 minutes ago : 7save
20916cf - kucherenkosveta, 25 minutes ago : 6save
48ff8fd - kucherenkosveta, 26 minutes ago : 5save
1ba03a0 - kucherenkosveta, 26 minutes ago : 4save
20307fb - kucherenkosveta, 27 minutes ago : 3save
d271ae5 - kucherenkosveta, 27 minutes ago : 2save
87f9b62 - kucherenkosveta, 29 minutes ago : 1save
f071529 - kucherenkosveta, 2 hours ago : Update README.md
cede938 - kucherenkosveta, 2 hours ago : Initial commit
svetik@MacBook-Air-Svetik LR_1.2 %
```

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

```
svetik@MacBook-Air-Svetik LR_1.2 % git log --pretty=format:"%h %s" --graph
* cd5f9a3 7save
* 20916cf 6save
* 48ff8fd 5save
* 1ba03a0 4save
* 20307fb 3save
* d271ae5 2save
* 87f9b62 1save
* f071529 Update README.md
* cede938 Initial commit
svetik@MacBook-Air-Svetik LR_1.2 %
```

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


```

svetik@MacBook-Air-Svetik LR_1.2 % git log --since=2.weeks
commit cd5f9a35187ffaf764a7d633bb49ee2acb45d73b (HEAD -> main, origin/main, origin/HEAD)
Author: kucherenkosveta <kucherenko55sss55@gmail.com>
Date: Sat Oct 22 01:29:20 2022 +0300

    7save

commit 20916cf0f5db440db1b88206d24f83c0bc765742
Author: kucherenkosveta <kucherenko55sss55@gmail.com>
Date: Sat Oct 22 01:28:23 2022 +0300

    6save

commit 48ff8fd37f75640af5f32ea4c2a38899350c9099
Author: kucherenkosveta <kucherenko55sss55@gmail.com>
Date: Sat Oct 22 01:27:19 2022 +0300

    5save

commit 1ba03a0118a04e432daf4321d1fb0cf35f346504
Author: kucherenkosveta <kucherenko55sss55@gmail.com>
Date: Sat Oct 22 01:26:53 2022 +0300

    4save

commit 20307fb975a474cffe5cc319cb36588babc9529
Author: kucherenkosveta <kucherenko55sss55@gmail.com>
Date: Sat Oct 22 01:26:16 2022 +0300

...skipping...
commit cd5f9a35187ffaf764a7d633bb49ee2acb45d73b (HEAD -> main, origin/main, origin/HEAD)
Author: kucherenkosveta <kucherenko55sss55@gmail.com>
Date: Sat Oct 22 01:29:20 2022 +0300

    7save

commit 20916cf0f5db440db1b88206d24f83c0bc765742
Author: kucherenkosveta <kucherenko55sss55@gmail.com>
Date: Sat Oct 22 01:28:23 2022 +0300

    6save

commit 48ff8fd37f75640af5f32ea4c2a38899350c9099
Author: kucherenkosveta <kucherenko55sss55@gmail.com>
Date: Sat Oct 22 01:27:19 2022 +0300

    5save

commit 1ba03a0118a04e432daf4321d1fb0cf35f346504
Author: kucherenkosveta <kucherenko55sss55@gmail.com>
Date: Sat Oct 22 01:26:53 2022 +0300

```

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

```

svetik@MacBook-Air-Svetik LR_1.2 % git log -S cout
commit cd5f9a35187ffaf764a7d633bb49ee2acb45d73b (HEAD -> main, origin/main, origin/HEAD)
Author: kucherenkosveta <kucherenko55sss55@gmail.com>
Date: Sat Oct 22 01:29:20 2022 +0300

    7save

commit 87f9b62ad43a7239ca142d72b1bbcd5b75c7ff7d
Author: kucherenkosveta <kucherenko55sss55@gmail.com>
Date: Sat Oct 22 01:24:19 2022 +0300

    1save

```

Рисунок 9 – Результат выполнения команды git log -S cout

```
svetik@MacBook-Air-Svetik LR_1.2 % git remote  
origin
```

Рисунок 10 – Результат выполнения команды git remote

```
svetik@MacBook-Air-Svetik LR_1.2 % git remote -v  
origin https://github.com/kucherenkosveta/LR_1.2.git (fetch)  
origin https://github.com/kucherenkosveta/LR_1.2.git (push)  
svetik@MacBook-Air-Svetik LR_1.2 %
```

Рисунок 11 – Результат выполнения команды git remote -v

```
svetik@MacBook-Air-Svetik LR_1.2 % git remote add pb https://github.com/kucherenkosveta/LR_1.2.git  
svetik@MacBook-Air-Svetik LR_1.2 % git remote -v  
origin https://github.com/kucherenkosveta/LR_1.2.git (fetch)  
origin https://github.com/kucherenkosveta/LR_1.2.git (push)  
pb https://github.com/kucherenkosveta/LR_1.2.git (fetch)  
pb https://github.com/kucherenkosveta/LR_1.2.git (push)  
svetik@MacBook-Air-Svetik LR_1.2 %
```

Рисунок 12 – Результат выполнения команды git remote add <shortname><url>

```
svetik@MacBook-Air-Svetik LR_1.2 % git fetch pb  
From https://github.com/kucherenkosveta/LR_1.2  
* [new branch]      main      -> pb/main
```

Рисунок 13 – Результат выполнения команды git fetch [remote-name]

```
svetik@MacBook-Air-Svetik LR_1.2 % git remote show origin  
* remote origin  
Fetch URL: https://github.com/kucherenkosveta/LR_1.2.git  
Push URL: https://github.com/kucherenkosveta/LR_1.2.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 (up to date)  
svetik@MacBook-Air-Svetik LR_1.2 %
```

Рисунок 14 – Результат выполнения команды git remote show origin

```

[svetik@MacBook-Air-Svetik LR_1.2 % git tag -a -m "my first tag"
usage: git tag [-a | -s | -u <key-id>] [-f] [-m <msg> | -F <file>]
               <tagname> [<head>]
or: git tag -d <tagname>...
or: git tag -l [-n<num>]] [--contains <commit>] [--no-contains <commit>] [--points-at <object>]
               [--format=<format>] [--merged <commit>] [--no-merged <commit>] [<pattern>...]
or: git tag -v [--format=<format>] <tagname>...

    -l, --list                list tag names
    -n[<n>]                  print <n> lines of each tag message
    -d, --delete             delete tags
    -v, --verify             verify tags

Tag creation options
    -a, --annotate           annotated tag, needs a message
    -m, --message <message> tag message
    -F, --file <file>       read message from file
    -e, --edit               force edit of tag message
    -s, --sign               annotated and GPG-signed tag
    --cleanup <mode>        how to strip spaces and #comments from message
    -u, --local-user <key-id> use another key to sign the tag
    -f, --force              replace the tag if exists
    --create-reflog          create a reflog

Tag listing options
    --column[=<style>]       show tag list in columns
    --contains <commit>      print only tags that contain the commit
    --no-contains <commit>   print only tags that don't contain the commit
    --merged <commit>        print only tags that are merged
    --no-merged <commit>     print only tags that are not merged
    --sort <key>             field name to sort on
    --points-at <object>     print only tags of the object
    --format <format>        format to use for the output
    --color[=<when>]         respect format colors
    -i, --ignore-case        sorting and filtering are case insensitive

...skipping...
usage: git tag [-a | -s | -u <key-id>] [-f] [-m <msg> | -F <file>]
               <tagname> [<head>]
or: git tag -d <tagname>...
or: git tag -l [-n<num>]] [--contains <commit>] [--no-contains <commit>] [--points-at <object>]
               [--format=<format>] [--merged <commit>] [--no-merged <commit>] [<pattern>...]
or: git tag -v [--format=<format>] <tagname>...

    -l, --list                list tag names
    -n[<n>]                  print <n> lines of each tag message
    -d, --delete             delete tags
    -v, --verify             verify tags

Tag creation options

```

Рисунок 15 – Результат выполнения команды git tag -a -m

```

[svetik@MacBook-Air-Svetik LR_1.2 % git push origin --tags
Everything up-to-date

```

Рисунок 16 - Результат выполнения команды git push origin --tags

```

[svetik@MacBook-Air-Svetik LR_1.2 % git push origin :refs/tags/v1.4-lw
remote: warning: deleting a non-existent ref
To https://github.com/kucherenkosveta/LR_1.2.git
- [deleted]          v1.4-lw

```

Рисунок 17 – Результат выполнения команды git push origin :refs/tags/v1.4-lw

4. Выполните клонирование созданного репозитория на рабочий компьютер.

```
[svetik@MacBook-Air-Svetik LR_1.2 % git clone https://github.com/kucherenkosveta/LR_1.2.git
Cloning into 'LR_1.2'...
remote: Enumerating objects: 86, done.
remote: Counting objects: 100% (86/86), done.
remote: Compressing objects: 100% (51/51), done.
remote: Total 86 (delta 19), reused 78 (delta 18), pack-reused 0
Receiving objects: 100% (86/86), 38.02 KiB | 589.00 KiB/s, done.
Resolving deltas: 100% (19/19), done.
svetik@MacBook-Air-Svetik LR_1.2 %
```

Рисунок 18 – Клонирование репозитория

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

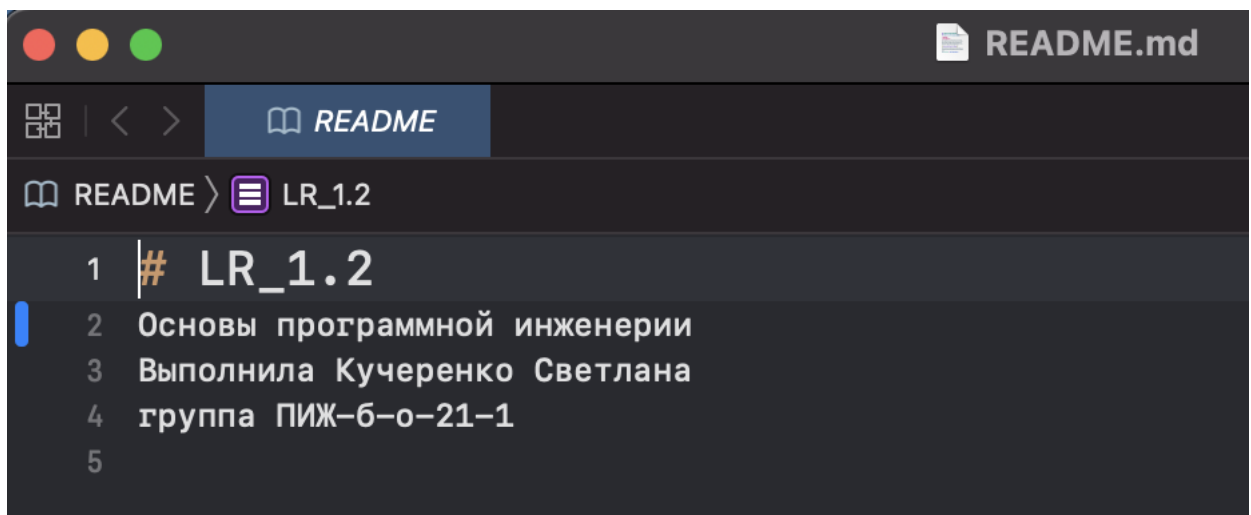


Рисунок 19 – Дополнение файла README.md

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.

```

svetik@MacBook-Air-Svetik LR_1.2 % git add .
svetik@MacBook-Air-Svetik LR_1.2 % git commit -m "1save"
[main fd8f577] 1save
 6 files changed, 15 insertions(+), 4 deletions(-)
 create mode 100644 .DS_Store
 create mode 100644 2/.DS_Store
 rewrite 2/2.xcodeproj/project.xcworkspace/xcuserdata/svetik.xcuserdatad/UserInterfaceState.xcuserstate (67%)
 create mode 160000 LR_1.2
svetik@MacBook-Air-Svetik LR_1.2 % git add .
svetik@MacBook-Air-Svetik LR_1.2 % git commit -m "2save"
[main d7013eb] 2save
 2 files changed, 1 deletion(-)
 rewrite 2/2.xcodeproj/project.xcworkspace/xcuserdata/svetik.xcuserdatad/UserInterfaceState.xcuserstate (66%)
svetik@MacBook-Air-Svetik LR_1.2 % git add .
svetik@MacBook-Air-Svetik LR_1.2 % git commit -m "3save"
[main 0ba56ed] 3save
 2 files changed, 3 insertions(+), 3 deletions(-)
svetik@MacBook-Air-Svetik LR_1.2 % git add .
svetik@MacBook-Air-Svetik LR_1.2 % git commit -m "4save"
[main 0042fab] 4save
 2 files changed, 1 insertion(+), 1 deletion(-)
svetik@MacBook-Air-Svetik LR_1.2 % git add .
svetik@MacBook-Air-Svetik LR_1.2 % git commit -m "5save"
[main 1efc353] 5save
 2 files changed, 1 insertion(+), 1 deletion(-)
svetik@MacBook-Air-Svetik LR_1.2 % git add .
svetik@MacBook-Air-Svetik LR_1.2 % git commit -m "6save"
[main d9d8201] 6save
 2 files changed, 1 insertion(+), 1 deletion(-)
svetik@MacBook-Air-Svetik LR_1.2 % git add .
svetik@MacBook-Air-Svetik LR_1.2 % git commit -m "7save"
[main 6177b19] 7save

```

Рисунок 20 – Сделанные коммиты

```

cd /Users/.../LR_1.2 % git tag -a v1.4 -m "my version 1.4"
[svetik@MacBook-Air-Svetik LR_1.2 % $ git tag -a v1.4 -m "my version 1.4"
zsh: command not found: $
[svetik@MacBook-Air-Svetik LR_1.2 % git tag -a v1.4 -m "my version 1.4"
fatal: tag 'v1.4' already exists
[svetik@MacBook-Air-Svetik LR_1.2 % git add .
[svetik@MacBook-Air-Svetik LR_1.2 % git commit -m "8save"
[main fee42e8] 8save
 1 file changed, 0 insertions(+), 0 deletions(-)
[svetik@MacBook-Air-Svetik LR_1.2 % git tag -a v1.4 -m "my version 1.5"
fatal: tag 'v1.4' already exists
[svetik@MacBook-Air-Svetik LR_1.2 % git tag -a v1.5 -m "my version 1.5"
[svetik@MacBook-Air-Svetik LR_1.2 % git tag
v1.4
v1.5
[svetik@MacBook-Air-Svetik LR_1.2 % git add .
[svetik@MacBook-Air-Svetik LR_1.2 % git commit -m "9save"
[main 3be0b21] 9save
 2 files changed, 1 insertion(+), 1 deletion(-)
[svetik@MacBook-Air-Svetik LR_1.2 % git tag -a v1.6 -m "my version 1.6"
[svetik@MacBook-Air-Svetik LR_1.2 % git tag
v1.4
v1.5
v1.6
svetik@MacBook-Air-Svetik LR_1.2 % █

```

Рисунок 21 – Добавление тегов

```
svetik@MacBook-Air-Svetik LR_1.2 % git log --graph --pretty=oneline --abbrev-commit
* 3be0b2141c93a832b09f032a3704a0ec02fcbf4d (HEAD -> main, tag: v1.6) 9save
* fee42e8e57fdcbda62b8f31058ee8e8a04fe0ab0 (tag: v1.5) 8save
* 6177b192051079fd15d9d36ad15e96afaeb64faf (tag: v1.4, origin/main, origin/HEAD) 7save
* d9d82010dce70ea1a5ecba5f0723495804a3d540 6save
* 1efc3539ae9f3e47175ac10cbe6296e022845ea3 5save
* 0042fabb4029f4dc36fca76b933c15423e551b90 4save
* 0ba56edbdbc3b02258200b693c3c0ee7b8f7b89b 3save
* d7013eb402963c8625c466f7c443c83c7674c406 2save
* fd8f577a72daf2d80689e2ae926d25e6bbb5f5480 1save
* cd5f9a35187ffaf764a7d633bb49ee2acb45d73b 7save
* 20916cf0f5db440db1b88206d24f83c0bc765742 6save
* 48ff8fd37f75640af5f32ea4c2a38899350c9099 5save
* 1ba03a0118a04e432daf4321d1fb0cf35f346504 4save
* 20307fb975a474cffea5cc319cb36588babc9529 3save
* d271ae5629e3a7ca66a939886bfcd19bc3961ffa 2save
* 87f9b62ad43a7239ca142d72b1bbcd5b75c7ff7d 1save
* f07152972df02a0c1b3b1ba93d9d76764d398863 Update README.md
* cede93881ffe5789e117c49fd2cf4323bee28e10 Initial commit
zsh: command not found: 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.

```
svetik@MacBook-Air-Svetik LR_1.2 % git show HEAD
[commit 3be0b2141c93a832b09f032a3704a0ec02fcbf4d (HEAD -> main, tag: v1.6)]
Author: kucherenkosveta <kucherenko55sss55@gmail.com>
Date: Sat Oct 22 03:03:55 2022 +0300

    9save

diff --git a/2/2.xcodeproj/project.xcworkspace/xcuserdata/svetik.xcuserdatad/UserInterfaceState.xcuserstate b/2/2.xcodeproj/project.xcworkspace/xcuserdata/svetik.xcuserdatad/UserInterfaceState.xcuserstate
index 6cf805f..2766fc5 100644
Binary files a/2/2.xcodeproj/project.xcworkspace/xcuserdata/svetik.xcuserdatad/UserInterfaceState.xcuserstate and b/2/2.xcodeproj/project.xcworkspace/xcuserdata/svetik.xcuserdatad/UserInterfaceState.xcuserstate differ
diff --git a/2/2/main.cpp b/2/2/main.cpp
index 997aea8..89668ff 100644
--- a/2/2/main.cpp
+++ b/2/2/main.cpp
@@ -6,7 +6,7 @@ using namespace std;
int main()
{
    int a;
-   cout << "Введите число с клавиатуры: ";
+   cout << "Введите циферку с клавиатуры: ";
    cin >> a;
    cin.ignore();
    cout << "Введенное вами число: "<< a << "\n";
svetik@MacBook-Air-Svetik LR_1.2 %
```

Рисунок 23 – Результат выполнения команды git show HEAD

```
svetik@MacBook-Air-Svetik LR_1.2 % git show HEAD~1
commit fee42e8e57fdcbda62b8f31058ee8e8a04fe0ab0 (tag: v1.5)
Author: kucherenkosveta <kucherenko55sss55@gmail.com>
Date: Sat Oct 22 03:01:43 2022 +0300

    8save

diff --git a/2/2.xcodeproj/project.xcworkspace/xcuserdata/svetik.xcuserdatad/UserInterfaceState.xcuserstate b/2/2.xcodeproj/project.xcworkspace/xcuserdata/svetik.xcuserdatad/UserInterfaceState.xcuserstate
index aa48dc1..6cf805f 100644
Binary files a/2/2.xcodeproj/project.xcworkspace/xcuserdata/svetik.xcuserdatad/UserInterfaceState.xcuserstate and b/2/2.xcodeproj/project.xcworkspace/xcuserdata/svetik.xcuserdatad/UserInterfaceState.xcuserstate differ
svetik@MacBook-Air-Svetik LR_1.2 %
```

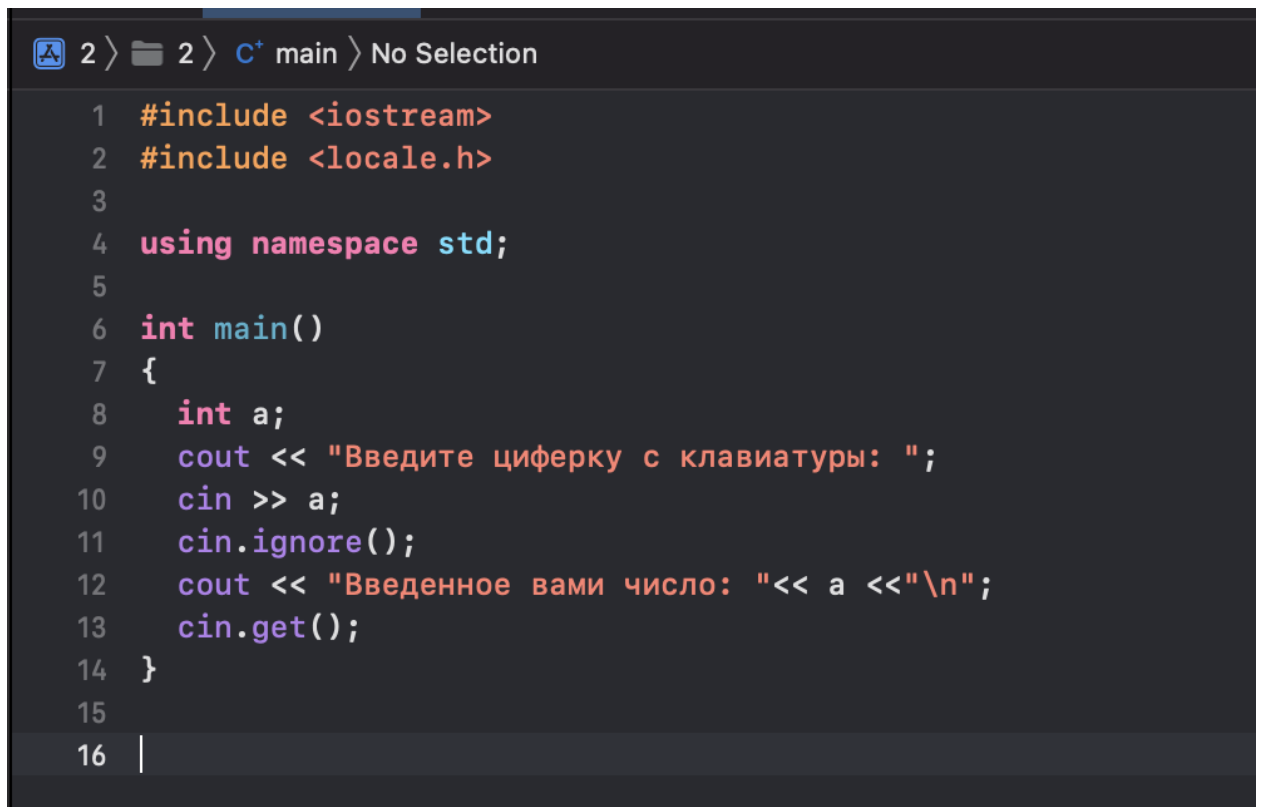
Рисунок 24 – Результат выполнения команды git show HEAD~1

```
svetik@MacBook-Air-Svetik LR_1.2 % git show 51a78d
fatal: ambiguous argument '51a78d': unknown revision or path not in the working tree.
Use '--' to separate paths from revisions, like this:
'git <command> [<revision>...] -- [<file>...]'
svetik@MacBook-Air-Svetik LR_1.2 %
```

Рисунок 25 – Результат выполнения команды git show 51a78d

10. Освойте возможность отката к заданной версии.

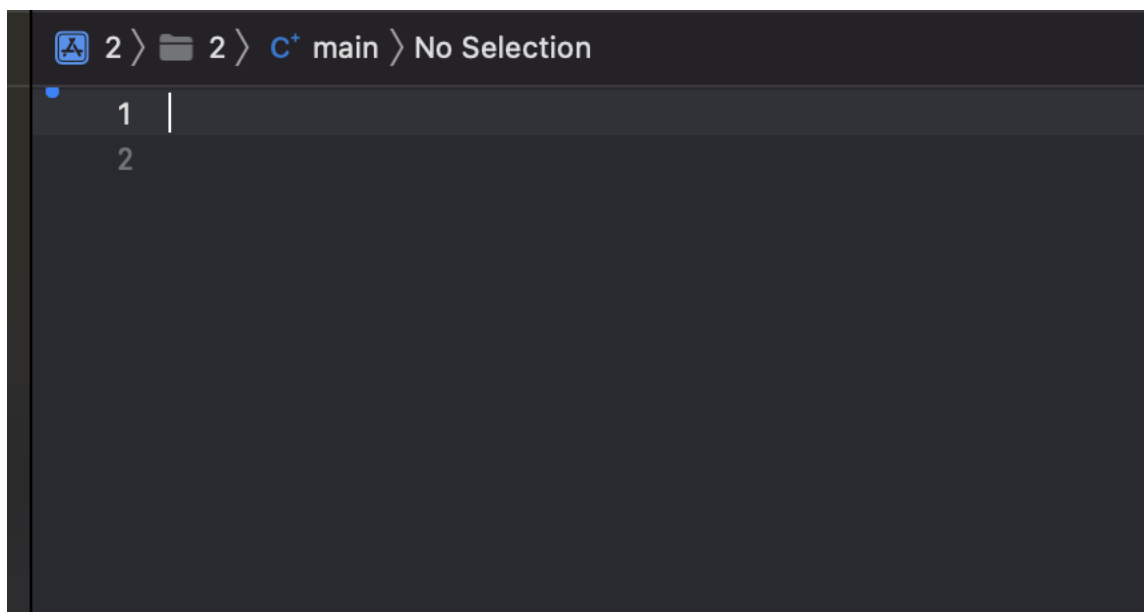
10.1 Удалите весь код из одного из файлов программы репозитория, например main.cpp, и сохраните этот файл.



A screenshot of a code editor window. The title bar shows a file icon, the path '2 > 2 > C+ main >', and 'No Selection'. The editor contains C++ code with line numbers 1 through 16. The code includes `<iostream>` and `<locale.h>`, uses the `std` namespace, and defines a `main` function that prompts the user to enter a digit and prints the entered number.

```
1 #include <iostream>
2 #include <locale.h>
3
4 using namespace std;
5
6 int main()
7 {
8     int a;
9     cout << "Введите циферку с клавиатуры: ";
10    cin >> a;
11    cin.ignore();
12    cout << "Введенное вами число: " << a << "\n";
13    cin.get();
14 }
15
16 |
```

Рисунок 26 – Файл до изменений



A screenshot of the same code editor window after changes. The title bar is identical. The editor shows line 1 with a cursor at the end of the line, and line 2 is visible below it. The rest of the code is not visible in this view.

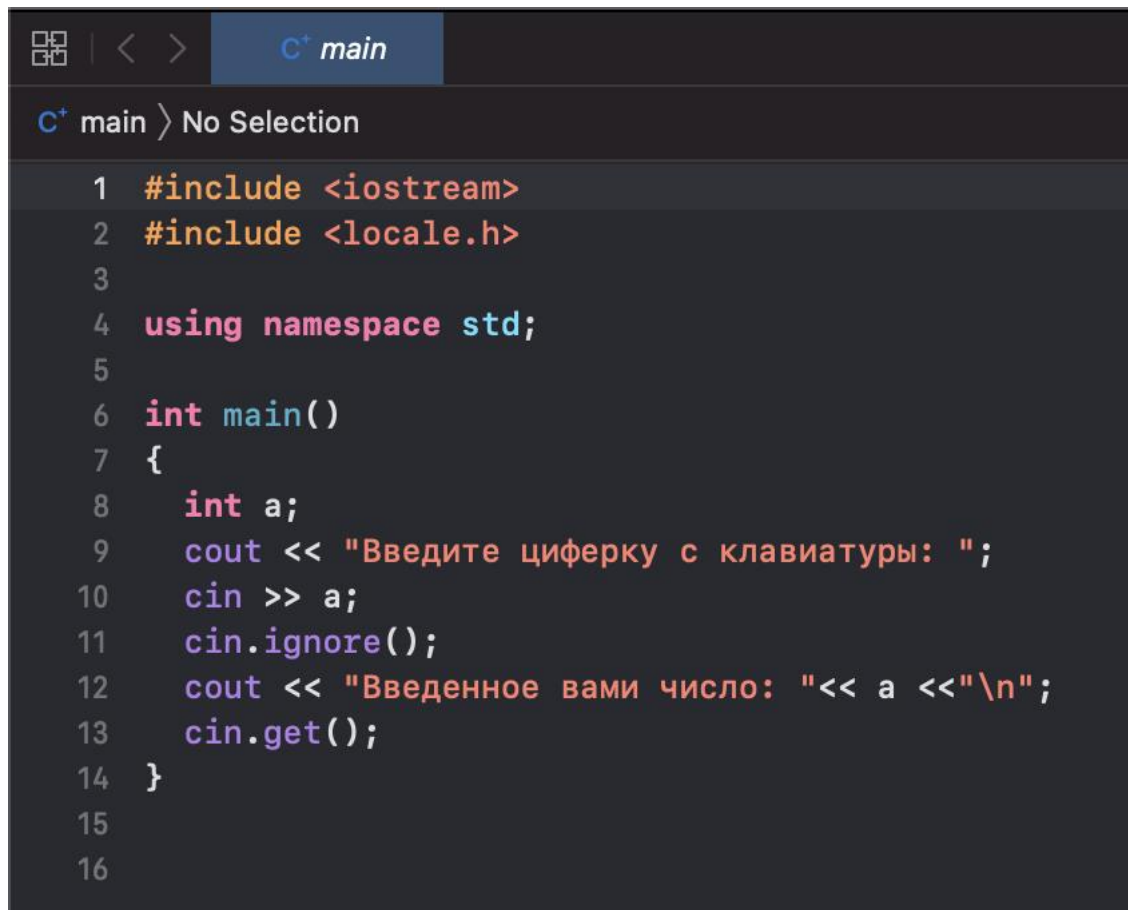
```
1 |
2
```

Рисунок 26 – Файл после изменений

10.2 Удалите все несохраненные изменения в файле командой: `git checkout --`, например `git checkout -- main.cpp`.


```
[svetik@MacBook-Air-Svetik LR_1.2 % git checkout -- 2/2/main.cpp
[svetik@MacBook-Air-Svetik LR_1.2 % git commit
On branch main
Your branch is ahead of 'origin/main' by 2 commits.
```

Рисунок 27 – Выполнение команды git checkout



```
C++ main
C++ main > No Selection

1 #include <iostream>
2 #include <locale.h>
3
4 using namespace std;
5
6 int main()
7 {
8     int a;
9     cout << "Введите циферку с клавиатуры: ";
10    cin >> a;
11    cin.ignore();
12    cout << "Введенное вами число: " << a << "\n";
13    cin.get();
14 }
15
16
```

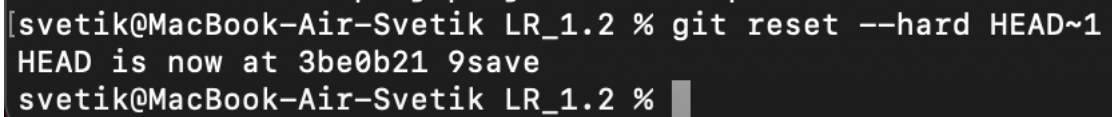
Рисунок 28 – Файл после изменений

Повторите пункт 10.1 и сделайте коммит.

```
[svetik@MacBook-Air-Svetik LR_1.2 % git add .
[svetik@MacBook-Air-Svetik LR_1.2 % git commit -m "10.1"
[main 307ee1b] 10.1
1 file changed, 0 insertions(+), 0 deletions(-)
rewrite 2/2.xcodeproj/project.xcworkspace/xcuserdata/svetik.xcuserdatad/UserInterfaceState.xcuserstate (72%)
svetik@MacBook-Air-Svetik LR_1.2 %
```

Рисунок 29 – Сделанный коммит

10.4 Откатить состояние хранилища к предыдущей версии командой: `git reset --hard HEAD~1`. Сделайте выводы об изменении содержимого выбранного Вами файла программы после выполнения пунктов 10.1– 10.4. Отрадите эти выводы в отчете.



```
[svetik@MacBook-Air-Svetik LR_1.2 % git reset --hard HEAD~1  
HEAD is now at 3be0b21 9save  
svetik@MacBook-Air-Svetik LR_1.2 %
```

Рисунок 30 – Результат `git reset --hard HEAD~1`

Контрольные вопросы и ответы на них:

Вопросы для защиты работы.

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

Наиболее простой и в то же время мощный инструмент для этого — команда `git log`. По умолчанию, без аргументов, `git log` выводит список коммитов созданных в данном репозитории в обратном хронологическом порядке. То есть самые последние коммиты показываются первыми.

Одна из опций, когда вы хотите увидеть сокращенную статистику для каждого коммита, вы можете использовать опцию `–stat`.

Вторая опция (одна 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; используйте параметр `-2` для вывода только двух записей (пример команды `git log -p -2`).

Третья действительно полезная опция это `--pretty`. Она меняет формат вывода. Существует несколько встроенных вариантов отображения. Опция `oneline` выводит каждый коммит в одну строку, что может быть очень удоб-

```
C:\Users\adamk\OneDrive\Рабочий стол\Моя папка\Учеба\Универ\1 курс 2 семестр\ОКП\ЛР 1.2\rep_1.2>git commit -am "Delete CPP content"
[main 3dde801] Delete CPP content
1 file changed, 1 insertion(+), 18 deletions(-)
C:\Users\adamk\OneDrive\Рабочий стол\Моя папка\Учеба\Универ\1 курс 2 семестр\ОКП\ЛР 1.2\rep_1.2>_
```

ным 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. Особенно это может быть полезным, когда вы хотите сгенерировать вывод для автоматического анализа — так как вы указываете формат явно, он не будет изменен даже после обновления Git.

Для опции `git log --pretty=format` существуют различного рода опции для изменения формата отображения.

2. Как ограничить вывод при просмотре истории коммитов?

Для ограничения может использоваться функция `git log <n>`, где `n` число записей.

Также, существуют опции для ограничения вывода по времени, такие как `--since` и `--until`, они являются очень удобными. Например, следующая команда покажет список коммитов, сделанных за последние две недели:

```
git log --since=2.weeks
```

Это команда 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`.

Также вы можете фильтровать список коммитов по заданным параметрам. Опция `--author` дает возможность фильтровать по автору коммита, а опция `--grep` (показывает только коммиты, сообщение которых содержит указанную строку) искать по ключевым словам в сообщении коммита. Функция `-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?

Внести изменения можно с помощью команды `git commit --amend`

Эта команда берёт индекс и применяет его к последнему коммиту. Если после последнего коммита не было никаких проиндексированных изменений (например, вы запустили приведённую команду сразу после предыдущего коммита), то состояние проекта будет абсолютно таким же и всё, что мы изменим, это комментарий к коммиту.

Для того, чтобы внести необходимые изменения - нам нужно проиндексировать их и выполнить команду `git commit --amend`.

```
git          commit          -m          'initial          commit'
git          add              forgotten_file
git          commit          --amend
```

Эффект от выполнения этой команды такой, как будто мы не выполнили

предыдущий коммит, а еще раз выполнили команду `git add` и выполнили коммит.

4. Как отменить индексацию файла в Git?

Например, вы изменили два файла и хотите добавить их в разные коммиты, но случайно выполнили команду `git add *` и добавили в индекс оба. Как исключить из индекса один из них? Команда `git status` напомним вам:

Прямо под текстом «Changes to be committed» говорится: используйте `git reset HEAD <file>` для исключения из индекса.

5. Как отменить изменения в файле?
С помощью команды `git checkout -- <file>`.

6. Что такое удаленный репозиторий Git?
Удалённый репозиторий это своего рода наше облако, в которое мы

сохраняем те или иные изменения в нашей программе/коде/файлах.

7. Как выполнить просмотр удаленных репозитория данного ло-

кального репозитория?

Для того, чтобы просмотреть список настроенных удалённых репозитория, необходимо запустить команду `git remote`.

Также можно указать ключ `-v`, чтобы просмотреть адреса для чтения и записи, привязанные к репозиторию. Пример: `git remote -v`

8. Как добавить удаленный репозиторий для данного локального репозитория?

Для того, чтобы добавить удалённый репозиторий и присвоить ему имя (shortname), просто выполните команду `git remote add <shortname> <url>`.

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

Если необходимо получить изменения, которые есть у Пола, но нету у вас, вы можете выполнить команду `git fetch <Название репозитория>`. Важно отметить, что команда `git fetch` забирает данные в ваш локальный репозиторий, но не сливает их с какими-либо вашими наработками и не модифицирует то, над чем вы работаете в данный момент. Вам необходимо вручную слить эти данные с вашими, когда вы будете готовы.

Если ветка настроена на отслеживание удалённой ветки, то вы можете использовать команду `git pull` чтобы автоматически получить изменения из удалённой ветки и слить их со своей текущей. Выполнение `git pull`, как правило, извлекает (`fetch`) данные с сервера, с которого вы изначально клонировали, и автоматически пытается слить (`merge`) их с кодом, над которым вы в данный момент работаете.

Чтобы отправить изменения на удалённый репозиторий необходимо отправить их в удалённый репозиторий. Команда для этого действия простая: `git push <remote-name> <branch-name>`.

10. Как выполнить просмотр удаленного репозитория?

Для просмотра удалённого репозитория, можно использовать команду `git remote show <remote>`.

11. Каково назначение тэгов Git?

Теги - это ссылки указывающие на определённые версии кода/написанной программы. Они удобны чтобы в случае чего вернуться к нужному моменту. Также при помощи тегов можно помечать важные моменты.

12. Как осуществляется работа с тэгами Git?

Просмотреть наличие тегов можно с помощью команды: `git tag`. А назначить (указать, добавить тег) можно с помощью команды `git tag -`

`a v1.4(версия изначальная) -m "Название"`.

С помощью команды `git show` вы можете посмотреть данные тега вместе

с коммитом: `git show v1.4`.

Отправка тегов, по умолчанию, команда `git push` не отправляет теги на

удалённые сервера. После создания теги нужно отправлять явно на удалённый сервер. Процесс аналогичен отправке веток — достаточно выполнить

команду `git push origin <tagname>`. Для отправки всех тегов можно использовать команду `git push origin tags`.

Для удаления тега в локальном репозитории достаточно выполнить команду `git tag -d <tagname>`. Например, удалить созданный ранее легко- весный тег можно следующим образом: `git tag -d v1.4-lw`

Для удаления тега из внешнего репозитория используется команда `git push origin --delete <tagname>`.

Если вы хотите получить версии файлов, на которые указывает тег, то вы можете сделать `git checkout` для тега пример: `git checkout -b version2 v2.0.0`.

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

`Git fetch --prune` команда получения всех изменений с репозитория GitHub.

В команде `git push --prune` удаляет удаленные ветки, у которых нет локального аналога.