

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ
ФЕДЕРАЦИИ**

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ**

ВЫСШЕГО ОБРАЗОВАНИЯ

**«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ» ИНСТИТУТ
ЦИФРОВОГО РАЗВИТИЯ**

**Отчет о лабораторной работе №1.2 по дисциплине основы программной
инженерии**

Выполнила:
Лобанова Елизавета
Евгеньевна,
2 курс, группа ПИЖ-б-о-20-1

Проверил:
Доцент кафедры прикладной
математики и
компьютерной безопасности,
Воронкин Р.А.

Отчет защищен с оценкой _____ Дата защиты _____

Ставрополь, 2021 г.

Ход работы

```
C:\Users\Acer\OneDrive\Рабочий стол\lr_opi\lr_21>git log
commit 2976325d7d1412c62cd674413574a08b5250bf52 (HEAD -> main, origin/main, origin/HEAD)
Author: lilobanova <lilobanova10@gmail.com>
Date: Wed Dec 8 14:57:32 2021 +0300

    main

commit 64f6a8c4a91900ef5ca01b2906040e3f8c2ea8cc
Author: lilobanova <lilobanova10@gmail.com>
Date: Wed Dec 8 14:56:14 2021 +0300

    s_print

commit c163dbd55399d1fb93162bf660d871fa2efd9bad
Author: lilobanova <lilobanova10@gmail.com>
Date: Wed Dec 8 14:55:14 2021 +0300

    else

commit c5ab792cd2571997082f19f188344c0d1012943c
Author: lilobanova <lilobanova10@gmail.com>
Date: Wed Dec 8 14:54:01 2021 +0300

    s_delete_key

commit 784d1c9c4fe9beacd5fea7c12d1915f8fc749b94
Author: lilobanova <lilobanova10@gmail.com>
Date: Wed Dec 8 14:52:35 2021 +0300

    s_push
...skipping...
commit 2976325d7d1412c62cd674413574a08b5250bf52 (HEAD -> main, origin/main, origin/HEAD)
Author: lilobanova <lilobanova10@gmail.com>
Date: Wed Dec 8 14:57:32 2021 +0300

    main

commit 64f6a8c4a91900ef5ca01b2906040e3f8c2ea8cc
Author: lilobanova <lilobanova10@gmail.com>
```

Результат работы команды git log

[illegible]

Результат работы команды `git log -p -2`

```

C:\Users\Acer\OneDrive\Рабочий стол\lr_opi\lr_2l>git log --stat
commit 2976325d7d1412c62cd674413574a08b5250bf52 (HEAD -> main, origin/main, origin/HEAD)
Author: lilobanova <lilobanova10@gmail.com>
Date: Wed Dec 8 14:57:32 2021 +0300

    main

opi2/main.cpp | 15 ++++++
1 file changed, 15 insertions(+)

commit 64f6a8c4a91900ef5ca01b2906040e3f8c2ea8cc
Author: lilobanova <lilobanova10@gmail.com>
Date: Wed Dec 8 14:56:14 2021 +0300

    s_print

opi2/main.cpp | 8 +++++
1 file changed, 8 insertions(+)

commit c163dbd55399d1fb93162bf660d871fa2efd9bad
Author: lilobanova <lilobanova10@gmail.com>
Date: Wed Dec 8 14:55:14 2021 +0300

    else

opi2/main.cpp | 11 ++++++
1 file changed, 11 insertions(+)

commit c5ab792cd2571997082f19f188344c0d1012943c
Author: lilobanova <lilobanova10@gmail.com>
Date: Wed Dec 8 14:54:01 2021 +0300

    s_delete_key

opi2/main.cpp | 14 ++++++
1 file changed, 14 insertions(+)

...skipping...
commit 2976325d7d1412c62cd674413574a08b5250bf52 (HEAD -> main, origin/main, origin/HEAD)
Author: lilobanova <lilobanova10@gmail.com>
Date: Wed Dec 8 14:57:32 2021 +0300

    main

opi2/main.cpp | 15 ++++++
1 file changed, 15 insertions(+)

commit 64f6a8c4a91900ef5ca01b2906040e3f8c2ea8cc
Author: lilobanova <lilobanova10@gmail.com>
Date: Wed Dec 8 14:56:14 2021 +0300

    s_print

opi2/main.cpp | 8 +++++
1 file changed, 8 insertions(+)

commit c163dbd55399d1fb93162bf660d871fa2efd9bad

```

Результат работы команды git log --stat

```

C:\Users\Acer\OneDrive\Рабочий стол\lr_opi\lr_2l>git log --pretty=oneline
2976325d7d1412c62cd674413574a08b5250bf52 (HEAD -> main, origin/main, origin/HEAD) main
64f6a8c4a91900ef5ca01b2906040e3f8c2ea8cc s_print
c163dbd55399d1fb93162bf660d871fa2efd9bad else
c5ab792cd2571997082f19f188344c0d1012943c s_delete_key
784d1c9c4fe9beacd5fea7c12d1915f8fc749b94 s_push
332161e88b1a8e165dd1c6ba1d0a9583159e4d41 Структура
3c058982e29c418d0c806bcdff3856eaff7ac547e Библиотеки
6fb8c6a5984143adb8df497b5401787f3d955a63 Создание файла
989b32a9edef9d5c3147b8568e39617dbcb9e778 Initial commit

```

Результат работы команды git log --pretty=format:"%h - %an, %ar : %s"

```

C:\Users\Acer\OneDrive\Рабочий стол\lr_opi\lr_2l>git log --pretty=format:"%h - %an, %ar : %s"
2976325 - lilobanova, 57 minutes ago : main
64f6a8c - lilobanova, 58 minutes ago : s_print
c163dbd - lilobanova, 59 minutes ago : else
c5ab792 - lilobanova, 60 minutes ago : s_delete_key
784d1c9 - lilobanova, 62 minutes ago : s_push
332161e - lilobanova, 63 minutes ago : Структура
3c05898 - lilobanova, 65 minutes ago : Библиотеки
6fb8c6a - lilobanova, 66 minutes ago : Создание файла
989b32a - lilobanova, 82 minutes ago : Initial commit

```

Результат работы команды git log --pretty=format:"%h - %an, %ar : %s"

```
C:\Users\Acer\OneDrive\Рабочий стол\lr_opi\lr_2l>git log --pretty=format:"%h %s" --graph
* 2976325 main | 14 ++++++
* 64f6a8c s_print14 insertions(+)
* c163dbd else
* c5ab792 s_delete_key65dd1c6ba1d0a9583159e4d41
* 784d1c9 s_pushva <lilobanova10@gmail.com>
* 332161e Структура4:51:40 2021 +0300
* 3c05898 Библиотеки
* 6fb8c6a Создание файла
* 989b32a Initial commit
```

Результат работы команды git log --pretty=format:"%h %s" --graph

```
C:\Users\Acer\OneDrive\Рабочий стол\lr_opi\lr_2l>git log --since=1.days
commit 24eac6c55f3c691322f83d375e9767d8c2c984c7 (HEAD -> main, origin/main, origin/HEAD)
Author: lilobanova <90153650+lilobanova@users.noreply.github.com>
Date: Fri Dec 10 01:14:20 2021 +0300

Update README.md
```

Результат работы команды git log --since=1.days

```
C:\Users\Acer\OneDrive\Рабочий стол\lr_opi\lr_2l>git log -s lr_2l
commit 24eac6c55f3c691322f83d375e9767d8c2c984c7 (HEAD -> main, origin/main, origin/HEAD)
Author: lilobanova <90153650+lilobanova@users.noreply.github.com>
Date: Fri Dec 10 01:14:20 2021 +0300

Update README.md

commit 989b32a9edef9d5c3147b8568e39617dbcb9e778
Author: lilobanova <90153650+lilobanova@users.noreply.github.com>
Date: Wed Dec 8 14:32:00 2021 +0300

Initial commit
```

Результат работы команды git log -s lr_2l

```
main
# Please enter the commit message for your changes. Lines starting
# with '#' will be ignored, and an empty message aborts the commit.
#
# Date: Wed Dec 8 14:57:32 2021 +0300
#
# On branch main
# Your branch is up to date with 'origin/main'.
#
# Changes to be committed:
#   modified:   opi2/main.cpp
#
~
~
~
```

Результат работы команды

```

Found a swap file by the name "C:/Users/Acer/OneDrive/Рабочий стол/lr_opi/lr_21/.git/.COMMIT_EDITMSG.swp"
  owned by: Acer    dated: Wed Dec 08 16:11:11 2021
  file name: ~Acer/OneDrive/Рабочий стол/lr_opi/lr_21/.git/COMMIT_EDITMSG
  modified: YES
  user name: Acer    host name: LAPTOP-E0VHJEC2
  process ID: 1977
While opening file "C:/Users/Acer/OneDrive/Рабочий стол/lr_opi/lr_21/.git/COMMIT_EDITMSG"
  dated: Wed Dec 08 16:19:02 2021
  NEWER than swap file!

(1) Another program may be editing the same file.  If this is the case,
    be careful not to end up with two different instances of the same
    file when making changes.  Quit, or continue with caution.
(2) An edit session for this file crashed.
    If this is the case, use ":recover" or "vim -r C:/Users/Acer/OneDrive/Рабочий стол/lr_opi/lr_21/.git/COMMIT_EDITMSG"
    to recover the changes (see ":help recovery").
    If you did this already, delete the swap file "C:/Users/Acer/OneDrive/Рабочий стол/lr_opi/lr_21/.git/.COMMIT_EDITMSG
    .swp"
    to avoid this message.

swap file "C:/Users/Acer/OneDrive/Рабочий стол/lr_opi/lr_21/.git/.COMMIT_EDITMSG.swp" already exists!
[O]pen Read-Only, [E]dit anyway, [R]ecover, [D]elete it, [Q]uit, [A]bort:

```

Результат работы команды

```

C:\Users\Acer\OneDrive\Рабочий стол\lr_opi\lr_21\lr_21>git remote -v
origin https://github.com/lilobanova/lr_21.git (fetch)
origin https://github.com/lilobanova/lr_21.git (push)

```

Результат работы команды git remote -v

```

C:\Users\Acer\OneDrive\Рабочий стол\lr_opi\lr_21\lr_21>git tag -a v1.4 -m "my version 1.4"
C:\Users\Acer\OneDrive\Рабочий стол\lr_opi\lr_21\lr_21>git tag
v1.4

```

Результат работы команды git tag

```

tag v1.4
Tagger: lilobanova <lilobanova10@gmail.com>
Date:   Wed Dec 8 16:33:47 2021 +0300

my version 1.4

commit 2976325d7d1412c62cd674413574a08b5250bf52 (HEAD -> main, tag: v1.4, origin/main, origin/HEAD)
Author: lilobanova <lilobanova10@gmail.com>
Date:   Wed Dec 8 14:57:32 2021 +0300

    main

diff --git a/opi2/main.cpp b/opi2/main.cpp
index 3829fbd..07589b4 100644
--- a/opi2/main.cpp
+++ b/opi2/main.cpp
@@ -55,3 +55,18 @@ void s_print(comp *top) { //
    q = q->next; //<EF><EE><F1><EB><E5> <F2><EE><E3><EE> <EA><E0><EA> <E2><FB><E2><E5><EB><E8> <EF><E5><F0><E
<E5><E4><E2><E8><E3><E0><E5><EC> <q> <ED><E0> <F1><EB><E5><E4><F3><FE><F9><E8><E9> <FD><EB><E5><EC><E5><ED><F2><FF><F7><E
<E5><E9><EA><F3>

}

+int main() {
+    comp *top = NULL; //<E2> <ED><E0><F7><E0><EB><E5> <EF><F0><EE><E3><F0><E0><EC><EC><FB> <F3> <ED><E0><F1> <ED><E
<E5><F2> <EE><F7><E5><F0><E5><E4><E8> <F1><EE><EE><F2><E2><E5><F2><F1><F2><E2><E5><ED><ED><EE> <E2><E5><F0><F8><E8><ED>
<FB> <ED><E5><F2> <E4><E0><E5><EC> <E5><E9> <E7><ED><E0><F7><E5><ED><E8><E5> <NULL>
+    //<C4><E0><EB><FC><F8><E5> <ED><E0><F7><E8><ED><E0><E5><EC> <E4><EE><E1><E0><E2><EB><FF><F2><FC> <F6><E8><F4><F
<F0><FB> <EE><F2> <1> <E4><EE> <5> <E2> <ED><E0><F8> <F1><F2><E5><EA>
+    s_push(&top, 54);

```

Результат работы команды

```

C:\Users\Acer\OneDrive\Рабочий стол\lr_opi\lr_21\lr_21>git push origin v1.5
Enumerating objects: 1, done.
Counting objects: 100% (1/1), done.
Writing objects: 100% (1/1), 160 bytes | 80.00 KiB/s, done.
Total 1 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/lilobanova/lr_21.git
 * [new tag]         v1.5 -> v1.5

```

Результат работы команды git push origin v1.5

```

C:\Users\Acer\OneDrive\Рабочий стол\lr_opi\lr_21\lr_21>git push origin --tags
Enumerating objects: 1, done.
Counting objects: 100% (1/1), done.
Writing objects: 100% (1/1), 161 bytes | 161.00 KiB/s, done.
Total 1 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/lilobanova/lr_21.git
 * [new tag]         v1.4 -> v1.4

```

Результат работы команды git push origin --tags

```
C:\Users\Acer\OneDrive\Рабочий стол\lr_opi\lr_2l\lr_2l>git tag -d v1.4  
Deleted tag 'v1.4' (was 3ed847d)
```

Результат работы команды git tag -d v1.4

```
C:\Users\Acer\OneDrive\Рабочий стол\lr_opi\lr_2l\lr_2l>git push origin :refs/tags/v1.4  
To https://github.com/lilobanova/lr_2l.git  
- [deleted] v1.4
```

Результат работы команды git push origin :refs/tags/v1.4

```
C:\Users\Acer\OneDrive\Рабочий стол\lr_opi\lr_2l\lr_2l>git checkout v2.0.0  
Note: switching to 'v2.0.0'.  
  
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 2976325 main
```

Результат работы команды git checkout v2.0.0

```
C:\Users\Acer\OneDrive\Рабочий стол\lr_opi\lr_2l\lr_2l>git checkout v2.0-beta-0.1  
HEAD is now at 2976325 main
```

Результат работы команды git checkout v2.0-beta-0.1

```
C:\Users\Acer\OneDrive\Рабочий стол\lr_opi\lr_2l\lr_2l>git checkout -b version2 v2.0.0  
Switched to a new branch 'version2'
```

Результат работы команды git checkout -b version2 v2.0.0

Создание зеркала https://bitbucket.org/lilobanova/lrr_2l/src/main/

Here's where you'll find this repository's source files. To give your users an idea of what they'll find here, [add a description to your repository](#).

 main ▾

Files ▾

Filter files 

📁 /

Name	Size	Last commit	Message
📁 opi2		yesterday	main
📄 .gitignore	270 B	yesterday	Initial commit
📄 LICENSE	1.04 KB	yesterday	Initial commit
📄 README.md	108 B	2 hours ago	Update README.md
📄 cd	0 B	yesterday	Создание файла

Результат создания зеркала

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 `gitlog`.

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 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?

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. Если вы ничего не меняли с момента последнего коммита (например, команда запущена сразу после предыдущего коммита), то снимок состояния останется в точности таким же, а всё что вы сможете изменить — это ваше сообщение к коммит.

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 (shortname), 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`.

11. Каково назначение тэгов Git?

Как и большинство СКВ, Git имеет возможность помечать 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 `-a` 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`. 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 локальном 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 удалённые данные, добавьте флаг `--prune` в команду `git fetch`. Удалите удалённые ветки, у которых нет локального аналога.

Например, удалённая 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 удалённый объект `refs/tmp/foo` будет удален, если `refs/heads/foo` он не существует.