

**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:  
Грובה Софья Кирилловна,  
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.

## ВЫПОЛНЕНИЕ:

Знакомство с возможностями просмотра журнала хранилища и отката изменений

```
D:\Пользователь\Desktop\ОПИ\lab2\lab2\laba2>git log --graph --pretty=oneline --abbrev-commit
* 5904337 (HEAD -> main, origin/main, origin/HEAD) 7
* 73d116f 6
* 1adca4e 5
* 68b96ff 4
* 5b707ea 3
* 3f84ea7 2
* e856a2f 1
* 5edaace Initial commit

D:\Пользователь\Desktop\ОПИ\lab2\lab2\laba2>
```

Рисунок 1.1 – Просмотр хранилища с помощью команды «git log».

## Просмотр отдельных коммитов

```
D:\Пользователь\Desktop\ОПИ\lab2\lab2\laba2>git show HEAD
commit 5904337630cff77b99c4802fa5457ef600786f22 (HEAD -> main, origin/main, origin/HEAD)
Author: SofyaGrobova <sofya.grobova@yandex.ru>
Date:   Fri Dec 17 15:58:32 2021 +0300

    7

diff --git a/laba2/main.cpp b/laba2/main.cpp
index d992b62..70e01b7 100644
--- a/laba2/main.cpp
+++ b/laba2/main.cpp
@@ -4,6 +4,6 @@ using namespace std;

int main()
{
-    cout << "Hellsssd!" << endl;
+    cout << "Hello world!" << endl;
    return 0;
}

D:\Пользователь\Desktop\ОПИ\lab2\lab2\laba2>
```

Рисунок 1.2 – Вывод последнего коммита с помощью команды «git show HEAD»

```

D:\Пользователь\Desktop\ОПИ\lab2\lab2\laba2>git show HEAD~1
commit 73d116f5f75572779c8c1e8da69f0c70dff24ceb
Author: SofyaGrobova <sofya.grobova@yandex.ru>
Date:   Fri Dec 17 15:58:12 2021 +0300

    6

diff --git a/laba2/main.cpp b/laba2/main.cpp
index 733fee1..d992b62 100644
--- a/laba2/main.cpp
+++ b/laba2/main.cpp
@@ -4,6 +4,6 @@ using namespace std;

int main()
{
-    cout << "Hellffmgld!" << endl;
+    cout << "Hellsssd!" << endl;
    return 0;
}

D:\Пользователь\Desktop\ОПИ\lab2\lab2\laba2>

```

Рисунок 1.3 – Вывод предпоследнего коммита с помощью добавления флага «~1»

```

D:\Пользователь\Desktop\ОПИ\lab2\lab2\laba2>git show 73d116f5f75572779c8c1e8da69f0c70dff24ceb
commit 73d116f5f75572779c8c1e8da69f0c70dff24ceb
Author: SofyaGrobova <sofya.grobova@yandex.ru>
Date:   Fri Dec 17 15:58:12 2021 +0300

    6

diff --git a/laba2/main.cpp b/laba2/main.cpp
index 733fee1..d992b62 100644
--- a/laba2/main.cpp
+++ b/laba2/main.cpp
@@ -4,6 +4,6 @@ using namespace std;

int main()
{
-    cout << "Hellffmgld!" << endl;
+    cout << "Hellsssd!" << endl;
    return 0;
}

D:\Пользователь\Desktop\ОПИ\lab2\lab2\laba2>

```

Рисунок 1.4 – Вывод коммита по значению хэша

Отмена изменений в локальном файле с помощью удаленного репозитория

```

D:\Пользователь\Desktop\ОПИ\lab2\lab2\laba2>git checkout main.cpp
Updated 1 path from the index

D:\Пользователь\Desktop\ОПИ\lab2\lab2\laba2>git commit -m "tt"
On branch main
Your branch is up to date with 'origin/main'.

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:   main.cpp

no changes added to commit (use "git add" and/or "git commit -a")

D:\Пользователь\Desktop\ОПИ\lab2\lab2\laba2>

```

Рисунок 1.5 – Отмена изменений в файле с помощью команды «git checkout»

#### Отмена последнего коммита

```
D:\Пользователь\Desktop\ОПИ\lab2\lab2\laba2>git reset --hard HEAD~1
HEAD is now at 73d116f 6

D:\Пользователь\Desktop\ОПИ\lab2\lab2\laba2>git status
On branch main
Your branch is behind 'origin/main' by 1 commit, and can be fast-forwarded.
(use "git pull" to update your local branch)

nothing to commit, working tree clean

D:\Пользователь\Desktop\ОПИ\lab2\lab2\laba2>
```

Рисунок 1.6 – Отмена последнего коммита с помощью «git reset —hard»

Благодаря команде «git checkout» можно восстановить данные, которые были удалены из файла. Для этого Git загружает данные в этот файл из последнего отправленного в удаленный репозиторий коммита. Команда «git reset —hard» позволяет даже отменять коммиты сделанных изменений, но является довольно опасным инструментом, так как удаленный коммит не будет сохранен в истории.

#### Просмотр тэгов

```
D:\Пользователь\Desktop\ОПИ\lab2\lab2\laba2>git tag
v1.1
v1.11
v1.12
```

Рисунок 1.7 – Просмотр всех тэгов в репозитории

```

D:\Пользователь\Desktop\ОПИ\lab2\lab2\laba2>git show v1.11
tag v1.11
Tagger: SofyaGrobova <sofya.grobova@yandex.ru>
Date:   Fri Dec 17 16:22:28 2021 +0300

11

commit e2328a53730508003934e8b40f20a31107755ed8 (tag: v1.11)
Author: SofyaGrobova <sofya.grobova@yandex.ru>
Date:   Fri Dec 17 16:22:10 2021 +0300

    11

diff --git a/laba2/main.cpp b/laba2/main.cpp
index d992b62..b0ac6a7 100644
--- a/laba2/main.cpp
+++ b/laba2/main.cpp
@@ -4,6 +4,6 @@ using namespace std;

int main()
{
-   cout << "Hellsssd!" << endl;
+   cout << "ff!" << endl;
    return 0;
}

D:\Пользователь\Desktop\ОПИ\lab2\lab2\laba2>

```

Рисунок 1.8 – Просмотр конкретного тэга с помощью команды «git show»

## 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`. По умолчанию (без аргументов) `git log` перечисляет коммиты, сделанные в репозитории в обратном к хронологическому порядку — последние коммиты находятся вверху. Данная команда перечисляет коммиты с их SHA-1 контрольными суммами, именем и электронной почтой автора, датой создания и сообщением коммита.

Команда `git log` имеет очень большое количество опций для поиска коммитов по разным критериям. Одним из самых полезных аргументов является `-p` или `—patch`, который показывает разницу (выводит патч), внесенную в каждый коммит. Если вы хотите увидеть сокращенную статистику для каждого коммита, вы можете использовать опцию `—stat`. Опция `—pretty` 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. 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.

## 2. Как ограничить вывод при просмотре истории коммитов?

Вы можете ограничить количество записей в выводе команды; используйте параметр -2 для вывода только двух записей, опции для ограничения вывода по времени: —since и —until, следующим действительно полезным фильтром является опция -S, которая принимает аргумент в виде строки и показывает только те коммиты, в которых изменение в коде повлекло за собой добавление или удаление этой строки. Последней полезной опцией, которую принимает команда git log как фильтр, является путь. Если вы укажете каталог или имя файла, вы ограничите вывод только теми коммитами, в которых были изменения этих файлов.

## 3. Как внести изменения в уже сделанный коммит?

git commit –amend.

## 4. Как отменить индексацию файла в Git?

git reset HEAD <file>.

## 5. Как отменить изменения в файле?

git checkout — <file>.