

Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего образования
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт цифрового развития
Кафедра инфокоммуникаций

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

**ОТЧЕТ
по лабораторной работе №2
дисциплины
«Основы программной инженерии»**

Выполнил:
Конныжев Максим Викторович
2 курс, группа ПИЖ-б-о-21-1,
09.03.04 «Программная инженерия»,
направленность (профиль) «Разработка
и сопровождение программного
обеспечения», очная форма обучения

(подпись)

Проверил:

(подпись)

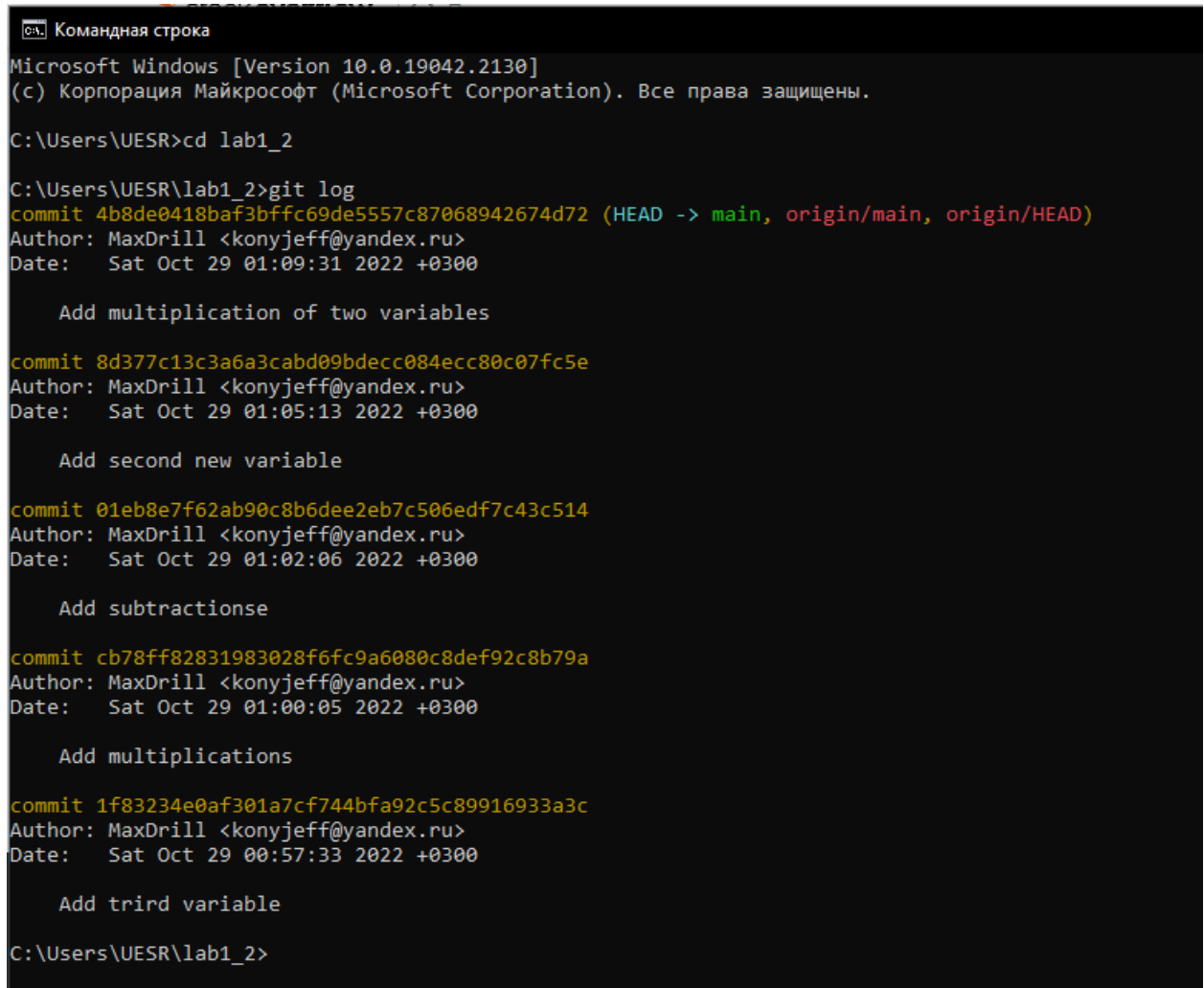
Отчет защищен с оценкой _____ Дата защиты _____

Ставрополь, 2022 г.

Цель работы: исследовать базовые возможности системы контроля версий Git для работы с локальными репозиториями.

1. Просмотр истории коммитов

В git есть возможность посмотреть историю коммитов при помощи команды git log



```
Командная строка
Microsoft Windows [Version 10.0.19042.2130]
(c) Корпорация Майкрософт (Microsoft Corporation). Все права защищены.

C:\Users\UESR>cd lab1_2

C:\Users\UESR\lab1_2>git log
commit 4b8de0418baf3bffc69de5557c87068942674d72 (HEAD -> main, origin/main, origin/HEAD)
Author: MaxDrill <konyjeff@yandex.ru>
Date: Sat Oct 29 01:09:31 2022 +0300

    Add multiplication of two variables

commit 8d377c13c3a6a3cabd09bdecc084ecc80c07fc5e
Author: MaxDrill <konyjeff@yandex.ru>
Date: Sat Oct 29 01:05:13 2022 +0300

    Add second new variable

commit 01eb8e7f62ab90c8b6dee2eb7c506edf7c43c514
Author: MaxDrill <konyjeff@yandex.ru>
Date: Sat Oct 29 01:02:06 2022 +0300

    Add subtractionse

commit cb78ff82831983028f6fc9a6080c8def92c8b79a
Author: MaxDrill <konyjeff@yandex.ru>
Date: Sat Oct 29 01:00:05 2022 +0300

    Add multiplications

commit 1f83234e0af301a7cf744bfa92c5c89916933a3c
Author: MaxDrill <konyjeff@yandex.ru>
Date: Sat Oct 29 00:57:33 2022 +0300

    Add trird variable

C:\Users\UESR\lab1_2>
```

Рисунок 1 – Команда git log

Существует параметр -p, который отображает патч для каждого коммита, и параметр -2, при помощи которого можно вывести определенное количество коммитов

```
Командная строка

C:\Users\UESR\lab1_2>git log -p -2
commit 4b8de0418baf3bffc69de5557c87068942674d72 (HEAD -> main, origin/main, origin/HEAD)
Author: MaxDrill <konyjeff@yandex.ru>
Date: Sat Oct 29 01:09:31 2022 +0300

    Add multiplication of two variables

diff --git a/program.py b/program.py
index 343eb3e..74323eb 100644
--- a/program.py
+++ b/program.py
@@ -5,4 +5,5 @@ c = a + b + d
 umn = a * b * d
 razn = a - b - d
 g = c + umn + razn
-print(c, umn, razn, g)
\ No newline at end of file
+f = razn * g
+print(c, umn, razn, g, f)
\ No newline at end of file

commit 8d377c13c3a6a3cabd09bdecc084ecc80c07fc5e
Author: MaxDrill <konyjeff@yandex.ru>
Date: Sat Oct 29 01:05:13 2022 +0300

    Add second new variable

diff --git a/program.py b/program.py
index 355243a..343eb3e 100644
--- a/program.py
+++ b/program.py
@@ -4,4 +4,5 @@ d = 6
 c = a + b + d
 umn = a * b * d
 razn = a - b - d
-print(c, umn, razn)
\ No newline at end of file
+g = c + umn + razn

C:\Users\UESR\lab1_2>
```

Рисунок 2 – Команда git log -p -2

Параметр `--stat` отображает статистику измененных файлов для каждого КОММИТА.

```
C:\Users\UESR\lab1_2>git log --stat
commit 4b8de0418baf3bffc69de5557c87068942674d72 (HEAD -> main, origin/main, origin/HEAD)
Author: MaxDrill <konyjeff@yandex.ru>
Date: Sat Oct 29 01:09:31 2022 +0300

    Add multiplication of two variables

program.py | 3 ++-
1 file changed, 2 insertions(+), 1 deletion(-)

commit 8d377c13c3a6a3cabd09bdecc084ecc80c07fc5e
Author: MaxDrill <konyjeff@yandex.ru>
Date: Sat Oct 29 01:05:13 2022 +0300

    Add second new variable

program.py | 3 ++-
1 file changed, 2 insertions(+), 1 deletion(-)

commit 01eb8e7f62ab90c8b6dee2eb7c506edf7c43c514
Author: MaxDrill <konyjeff@yandex.ru>
Date: Sat Oct 29 01:02:06 2022 +0300

    Add subtractionse

program.py | 3 ++-
1 file changed, 2 insertions(+), 1 deletion(-)
```

Рисунок 3 – Команда `git log --stat`

Команда `--pretty` меняет формат вывода и имеет опции: `oneline`, `short`, `full` и `fuller`. При этом можно использовать опцию `format` которая позволяет вывести конкретную информацию

```
C:\Users\UESR\lab1_2>git log --pretty=oneline
4b8de0418baf3bffc69de5557c87068942674d72 (HEAD -> main, origin/main, origin/HEAD) Add multiplication of two variables
8d377c13c3a6a3cabd09bdecc084ecc80c07fc5e Add second new variable
01eb8e7f62ab90c8b6dee2eb7c506edf7c43c514 Add subtractionse
cb78ff82831983028f6fc9a6080c8def92c8b79a Add multiplications
1f83234e0af301a7cf744bfa92c5c89916933a3c Add trird variable
bfff5da0f042402fffb6b9eb49855454fba3e6e919 Add more 2 information in readme
3a25d3580f934dc558a67344fef54c1d24568cfb Add more inform in readme
1000d21037a2841c86cc155df0b5f549508e953d Create programmi
4bc65fff1a56d18411e756ad94db5fe235fe5bc3 Add information in readme
63a8fe7974f3c31c84f1e90b524b728e2c3c5fa6 Initial commit

C:\Users\UESR\lab1_2>
```

Рисунок 4 – Команда git log --pretty=oneline

```
C:\Users\UESR\lab1_2>git log --pretty=format:"%h - %an, %ar : %s"
4b8de04 - MaxDrill, 4 weeks ago : Add multiplication of two variables
8d377c1 - MaxDrill, 4 weeks ago : Add second new variable
01eb8e7 - MaxDrill, 4 weeks ago : Add subtractionse
cb78ff8 - MaxDrill, 4 weeks ago : Add multiplications
1f83234 - MaxDrill, 4 weeks ago : Add trird variable
bff5da0 - MaxDrill, 7 weeks ago : Add more 2 information in readme
3a25d35 - MaxDrill, 7 weeks ago : Add more inform in readme
1000d21 - MaxDrill, 7 weeks ago : Create programmi
4bc65ff - MaxDrill, 7 weeks ago : Add information in readme
63a8fe7 - KonyzhevMax, 7 weeks ago : Initial commit

C:\Users\UESR\lab1_2>
```

Рисунок 5 – Команда git log --pretty=format:"%h - %an, %ar : %s"

Командой git log --since можно отобразить коммиты, сделанные в конкретном промежутке времени.

```
C:\Users\UESR\lab1_2>git log --since=5.weeks
commit 4b8de0418baf3bffc69de5557c87068942674d72 (HEAD -> main, origin/main, origin/HEAD)
Author: MaxDrill <konyjeff@yandex.ru>
Date: Sat Oct 29 01:09:31 2022 +0300

    Add multiplication of two variables

commit 8d377c13c3a6a3cabd09bdecc084ecc80c07fc5e
Author: MaxDrill <konyjeff@yandex.ru>
Date: Sat Oct 29 01:05:13 2022 +0300

    Add second new variable

commit 01eb8e7f62ab90c8b6dee2eb7c506edf7c43c514
Author: MaxDrill <konyjeff@yandex.ru>
Date: Sat Oct 29 01:02:06 2022 +0300

    Add subtractionse
```

Рисунок 6 – Команда git log --since=15.days

Фильтр «Path to file» показывает коммиты файла, если указать к нему путь

```
C:\Users\UESR\lab1_2>git log -- C:\Users\UESR\lab1_2\README.md
commit bff5da0f042402ffb6b9eb49855454fba3e6e919
Author: MaxDrill <konyjeff@yandex.ru>
Date: Sat Oct 8 00:16:04 2022 +0300

    Add more 2 information in readme

commit 3a25d3580f934dc558a67344fef54c1d24568cfb
Author: MaxDrill <konyjeff@yandex.ru>
Date: Sat Oct 8 00:13:16 2022 +0300

    Add more inform in readme

commit 4bc65fff1a56d18411e756ad94db5fe235fe5bc3
Author: MaxDrill <konyjeff@yandex.ru>
Date: Fri Oct 7 23:42:13 2022 +0300

    Add information in readme
```

Рисунок 7 – Команда git log – C:\Users\UESR\lab1_2\README.md

2. Просмотр содержимого коммитов

Используя команду git show head можно смотреть содержимое
КОММИТОВ

```

C:\Users\UESR\lab1_2>git show head
commit 4b8de0418baf3bffc69de5557c87068942674d72 (HEAD -> main, origin/main, origin/HEAD)
Author: MaxDrill <konyjeff@yandex.ru>
Date: Sat Oct 29 01:09:31 2022 +0300

    Add multiplication of two variables

diff --git a/program.py b/program.py
index 343eb3e..74323eb 100644
--- a/program.py
+++ b/program.py
@@ -5,4 +5,5 @@ c = a + b + d
    umn = a * b * d
    razn = a - b - d
    g = c + umn + razn
-print(c, umn, razn, g)
\ No newline at end of file
+f = razn * g
+print(c, umn, razn, g, f)
\ No newline at end of file

C:\Users\UESR\lab1_2>

```

Рисунок 8 – Последний коммит

```

C:\Users\UESR\lab1_2>git show 4bc65
commit 4bc65fff1a56d18411e756ad94db5fe235fe5bc3
Author: MaxDrill <konyjeff@yandex.ru>
Date: Fri Oct 7 23:42:13 2022 +0300

    Add information in readme

diff --git a/README.md b/README.md
index b280c97..92f0e70 100644
--- a/README.md
+++ b/README.md
@@ -1,2 @@
-# lab1_1
\ No newline at end of file
+# lab1_1
+Коньжев Максим ПИЖ-6-о-21-1
\ No newline at end of file

C:\Users\UESR\lab1_2>

```

Рисунок 9 – Поиск коммита по хешу

3. Теги

В git есть возможность создавать теги, легковесные и аннотированные, отличие заключается в том, что первый вид, является просто ссылкой на коммит а второй хранится в базе git как полноценный объект.

```
C:\Users\UESR\lab1_2>git tag -a 11.2 -m "tag 11.2"
```

Рисунок 10 – Создание аннотированного тега

```
C:\Users\UESR\lab1_2>git tag -a 11.2 -m "tag 11.2"

C:\Users\UESR\lab1_2>git show 11.2
tag 11.2
Tagger: MaxDrill <konyjeff@yandex.ru>
Date:   Thu Nov 24 22:27:33 2022 +0300

tag 11.2

commit 4b8de0418baf3bffc69de5557c87068942674d72 (HEAD -> main, tag: 11.2, origin/main, origin/HEAD)
Author: MaxDrill <konyjeff@yandex.ru>
Date:   Sat Oct 29 01:09:31 2022 +0300

    Add multiplication of two variables

diff --git a/program.py b/program.py
index 343eb3e..74323eb 100644
--- a/program.py
+++ b/program.py
@@ -5,4 +5,5 @@ c = a + b + d
 umn = a * b * d
 razn = a - b - d
 g = c + umn + razn
-print(c, umn, razn, g)
\ No newline at end of file
+f = razn * g
+print(c, umn, razn, g, f)
\ No newline at end of file

C:\Users\UESR\lab1_2>
```

Рисунок 11 – Просмотр тега

Так как теги не отправляются на удалённый сервер по умолчанию следует отправлять теги явно, с использованием команды `git push origins`


```
C:\Users\UESR\lab1_2>git push origin --tags
Enumerating objects: 1, done.
Counting objects: 100% (1/1), done.
Writing objects: 100% (1/1), 157 bytes | 157.00 KiB/s, done.
Total 1 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/MaxDrill/lab1_2.git
 * [new tag]          11.2 -> 11.2
C:\Users\UESR\lab1_2>
```

Рисунок 12 – Отправка тега на удаленный сервер

4. Удаление и восстановление файлов

В git есть возможность отмены изменений файла, при помощи команды `git checkout`

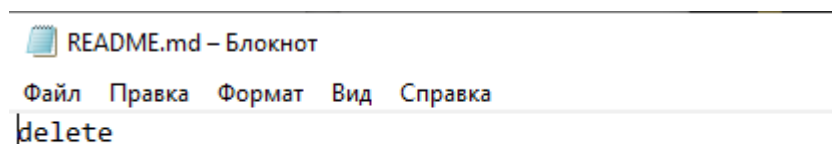


Рисунок 13 – Изменённый файл README

```
C:\Users\UESR\lab1_2>git checkout -- README.md
C:\Users\UESR\lab1_2>
```

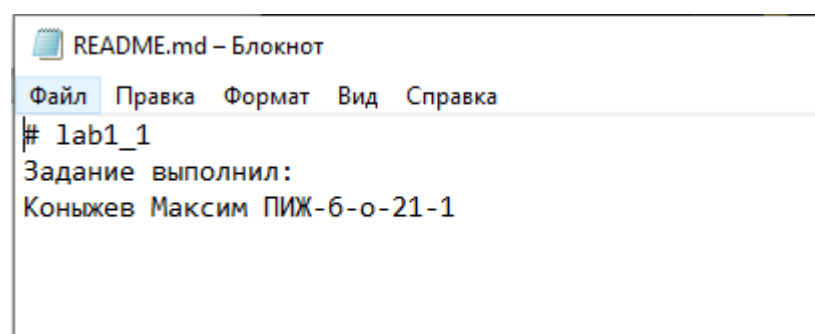


Рисунок 14 – Восстановленный файл README

Контрольные вопросы

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

При помощи команды `git log`. Опции для просмотра коммитов:

| Опция | Описание |
|------------------------------|---|
| <code>-p</code> | Показывает патч для каждого коммита. |
| <code>--stat</code> | Показывает статистику измененных файлов для каждого коммита. |
| <code>--shortstat</code> | Отображает только строку с количеством изменений/вставок/удалений для команды <code>--stat</code> . |
| <code>--name-only</code> | Показывает список измененных файлов после информации о коммите. |
| <code>--name-status</code> | Показывает список файлов, которые добавлены/изменены/удалены. |
| <code>--abbrev-commit</code> | Показывает только несколько символов SHA-1 чек-суммы вместо всех 40. |
| <code>--relative-date</code> | Отображает дату в относительном формате (например, «2 weeks ago») вместо стандартного формата даты. |
| <code>--graph</code> | Отображает ASCII граф с ветвлениями и историей слияний. |
| <code>--pretty</code> | Показывает коммиты в альтернативном формате. Возможные варианты опций: <code>oneline</code> , <code>short</code> , <code>full</code> , <code>fuller</code> и <code>format</code> (с помощью последней можно указать свой формат). |
| <code>--oneline</code> | Сокращение для одновременного использования опций <code>--pretty=oneline --abbrev-commit</code> . |

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?

Командой `git commit –amend`

4. Как отменить индексацию файла в Git?

При помощи команды `git reset HEAD`

5. Как отменить изменения в файле?

Командой `git checkout -- «имя файла»`

6. Что такое удалённый 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?

Команда `git remote` выводит названия доступных удалённых репозиториях.

8. Как добавить удаленный репозиторий для данного локального репозитория?

Выполнить команду `git remote add`

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

Командой `git fetch «Имя проекта»` можно получить недостающие данные проекта.

Командой `git push origin master` можно выполнить отправку изменений на удаленный репозиторий.

10. Как выполнить просмотр удаленного репозитория?

Команда `git remote show <remote>` отобразит URL репозитория а так же информацию о ветках

11. Каково назначение тэгов Git?

Тег – это ссылка, указывающая на определенную точку в истории Git.

12. Как осуществляется работа с тэгами Git?

`git tag «название тега»` создаст легковесный тег, добавление параметра `-a` позволит создать аннотированный тег.

13. Самостоятельно изучите назначение флага `--prune` в командах `git fetch` и

`git push`. Каково назначение этого флага?

`git fetch --prune` для удаления устаревших веток