

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

Институт цифрового развития
Кафедра инфокоммуникаций

ОТЧЕТ
ПО ЛАБОРАТОРНОЙ РАБОТЕ №1.2
дисциплины «Основы программной инженерии»

Выполнил:
Баратов Семен Григорьевич
2 курс, группа ПИЖ-б-о-22-1,
09.03.04 «Программная инженерия»,
направленность (профиль) «Разработка
и сопровождение программного
обеспечения», очная форма обучения

(подпись)

Преподаватель:
Воронкин Р.А., канд. тех. наук, доцент,
доцент кафедры инфокоммуникаций

(подпись)

Отчет защищен с оценкой _____ Дата защиты _____

Ставрополь, 2023 г.

Тема: Исследование возможностей Git для работы с локальными репозиториями.

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

Результаты выполнения

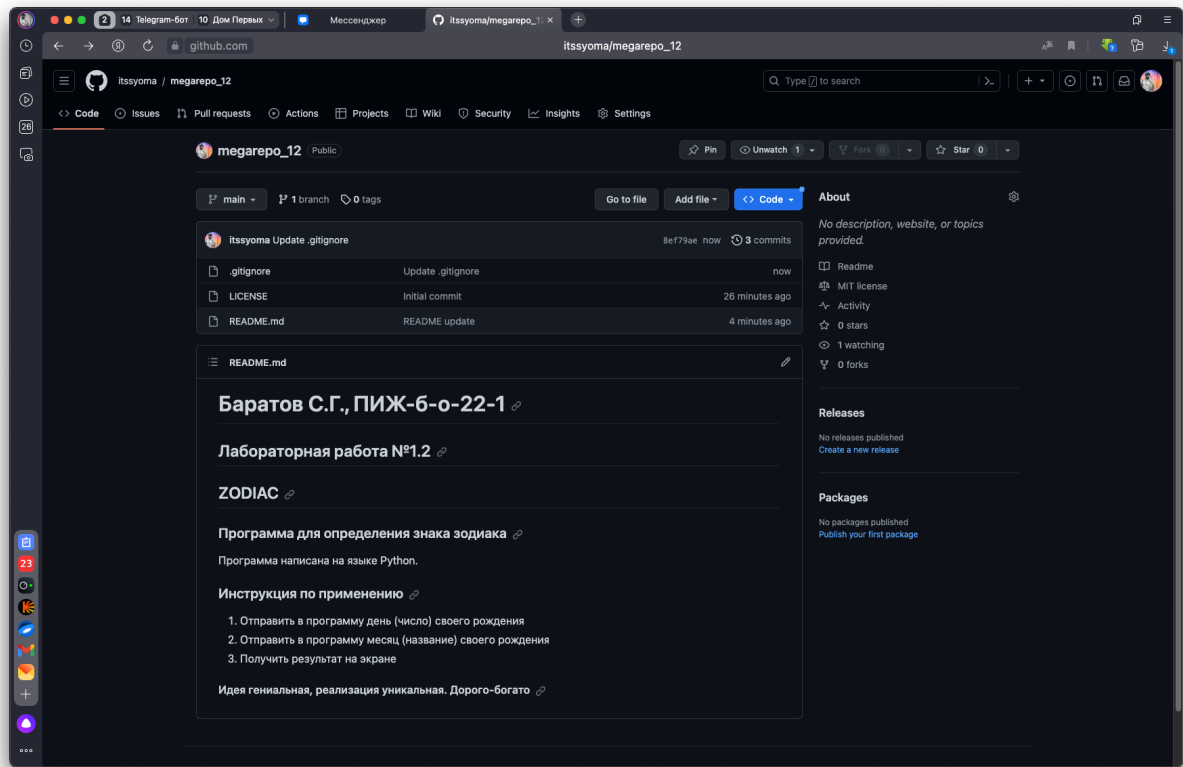


Рисунок 1 – Страница репозитория после создания и внесения изменений в файлы README.md и .gitignore

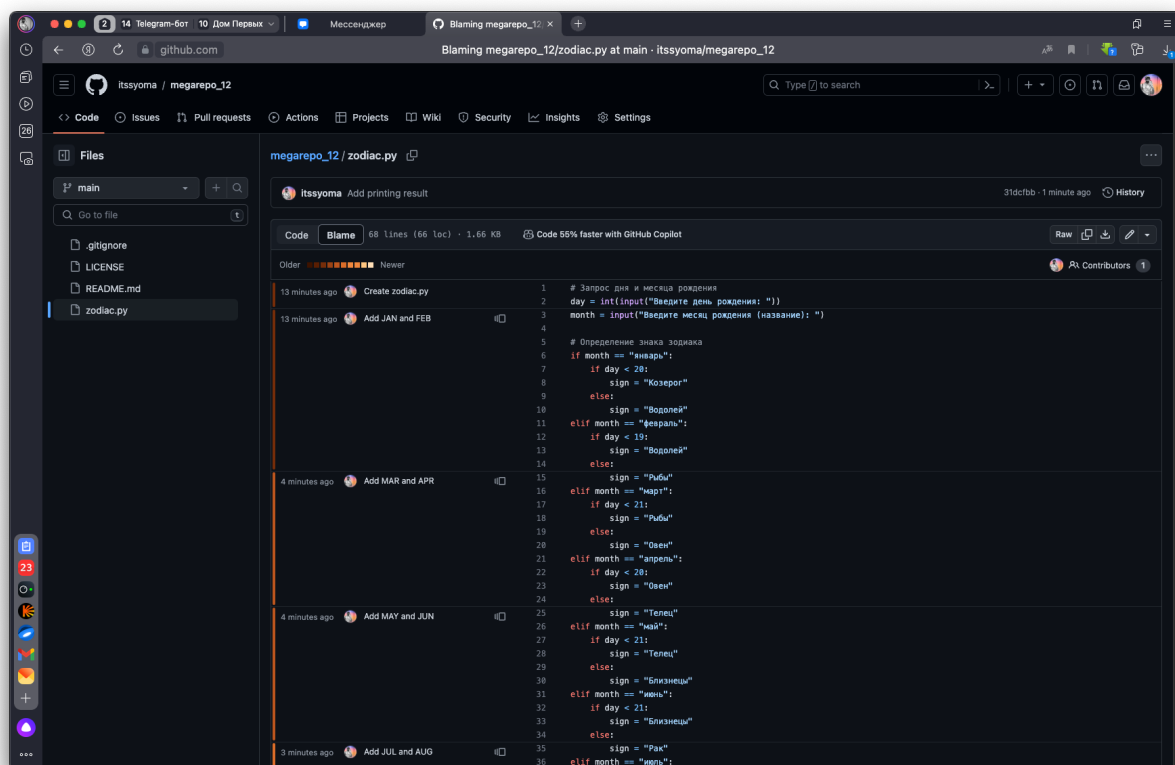


Рисунок 2 – Коммиты файла программы

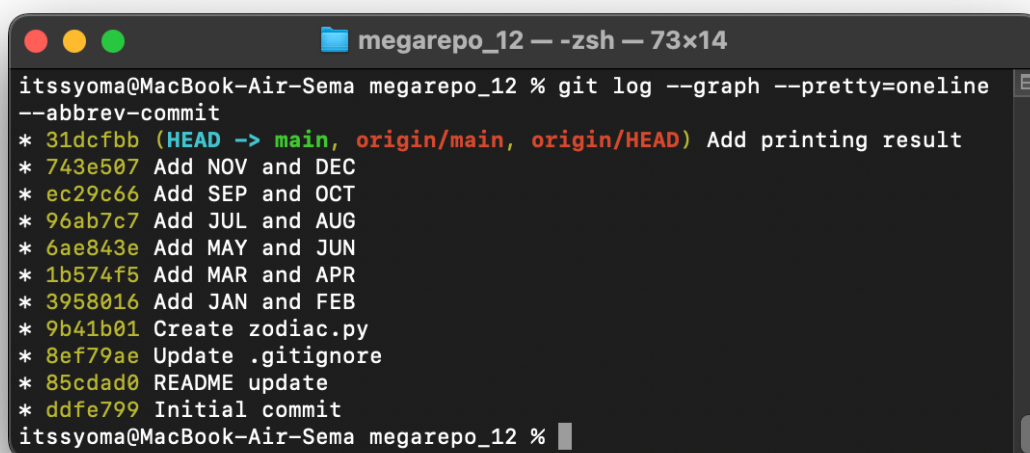


Рисунок 3 – История хранилища

```
megarepo_12 — zsh — 88x24
Last login: Wed Oct 4 10:40:04 on ttys000
[itssyoma@MacBook-Air-Sema megarepo_12 % git show HEAD
commit 31dcfbb3720de8005b74df5adaf9ccb2e8197c9c (HEAD -> main, origin/main, origin/HEAD)
Author: itssyoma <itssyoma@gmail.com>
Date: Wed Oct 4 10:46:14 2023 +0300

    Add printing result

diff --git a/zodiac.py b/zodiac.py
index a8c9e4a..029c47d 100644
--- a/zodiac.py
+++ b/zodiac.py
@@ -62,4 +62,7 @@ elif month == "декабрь":
     if day < 22:
         sign = "Стрелец"
     else:
-        sign = "Козерог"
\ No newline at end of file
+        sign = "Козерог"
+
+# Вывод знака зодиака
+print("Ваш знак зодиака: ", sign)
\ No newline at end of file
itssyoma@MacBook-Air-Sema megarepo_12 %
```

Рисунок 4 – Содержимое последнего коммита

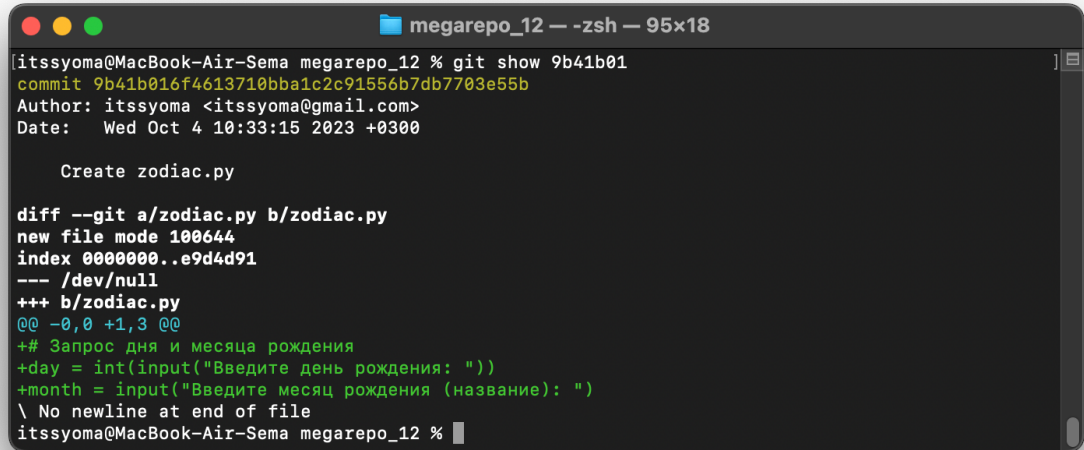
```
megarepo_12 — zsh — 105x31
[itssyoma@MacBook-Air-Sema megarepo_12 % git show HEAD~1
commit 743e507c63ae75f69d8d2d162978d4369cb62310
Author: itssyoma <itssyoma@gmail.com>
Date: Wed Oct 4 10:45:25 2023 +0300

    Add NOV and DEC

diff --git a/zodiac.py b/zodiac.py
index 7fb99fb..a8c9e4a 100644
--- a/zodiac.py
+++ b/zodiac.py
@@ -52,4 +52,14 @@ elif month == "октябрь":
     if day < 23:
         sign = "Весы"
     else:
-        sign = "Скорпион"
\ No newline at end of file
+        sign = "Скорпион"
+elif month == "ноябрь":
+    if day < 22:
+        sign = "Скорпион"
+    else:
+        sign = "Стрелец"
+elif month == "декабрь":
+...skipping...
commit 743e507c63ae75f69d8d2d162978d4369cb62310
Author: itssyoma <itssyoma@gmail.com>
Date: Wed Oct 4 10:45:25 2023 +0300

    Add NOV and DEC
```

Рисунок 5 – Содержимое предпоследнего коммита

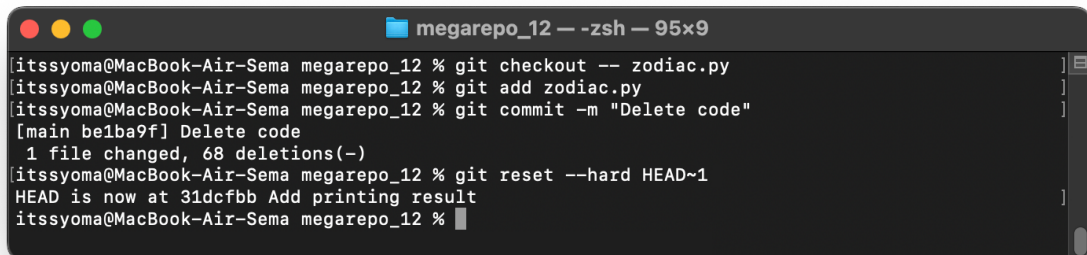


```
megarepo_12 — -zsh — 95x18
[itssyoma@MacBook-Air-Sema megarepo_12 % git show 9b41b01
commit 9b41b016f4613710bba1c2c91556b7db7703e55b
Author: itssyoma <itssyoma@gmail.com>
Date:   Wed Oct 4 10:33:15 2023 +0300

    Create zodiac.py

diff --git a/zodiac.py b/zodiac.py
new file mode 100644
index 0000000..e9d4d91
--- /dev/null
+++ b/zodiac.py
@@ -0,0 +1,3 @@
+# Запрос дня и месяца рождения
+day = int(input("Введите день рождения: "))
+month = input("Введите месяц рождения (название): ")
\ No newline at end of file
itssyoma@MacBook-Air-Sema megarepo_12 %
```

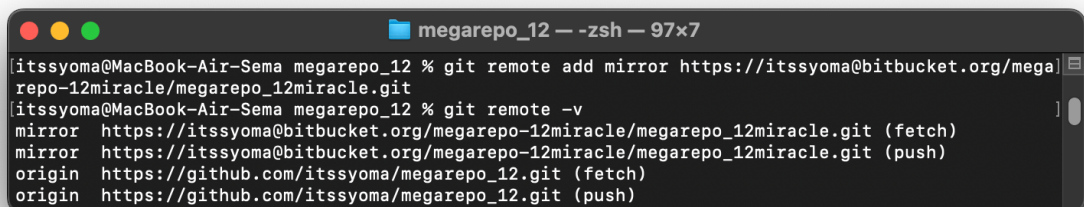
Рисунок 6 – Содержимое коммита с хэшем 9b41b01



```
megarepo_12 — -zsh — 95x9
[itssyoma@MacBook-Air-Sema megarepo_12 % git checkout -- zodiac.py
[itssyoma@MacBook-Air-Sema megarepo_12 % git add zodiac.py
[itssyoma@MacBook-Air-Sema megarepo_12 % git commit -m "Delete code"
[main be1ba9f] Delete code
1 file changed, 68 deletions(-)
[itssyoma@MacBook-Air-Sema megarepo_12 % git reset --hard HEAD~1
HEAD is now at 31dcfbb Add printing result
itssyoma@MacBook-Air-Sema megarepo_12 %
```

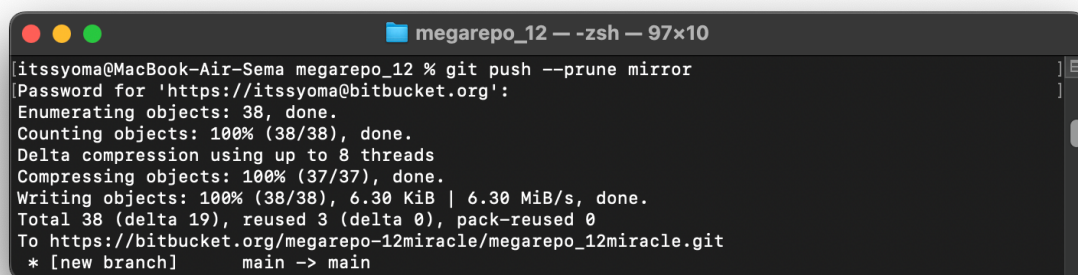
Рисунок 6 – Откат к заданным версиям.

При использовании *git checkout* все локальные изменения в файле пропали, *git* заменил их версией из последнего коммита. *git reset --hard* перевел ветку на предпоследний коммит, а последний удалил.



```
megarepo_12 — -zsh — 97x7
[itssyoma@MacBook-Air-Sema megarepo_12 % git remote add mirror https://itssyoma@bitbucket.org/mega
repo-12miracle/megarepo_12miracle.git
[itssyoma@MacBook-Air-Sema megarepo_12 % git remote -v
mirror https://itssyoma@bitbucket.org/megarepo-12miracle/megarepo_12miracle.git (fetch)
mirror https://itssyoma@bitbucket.org/megarepo-12miracle/megarepo_12miracle.git (push)
origin https://github.com/itssyoma/megarepo_12.git (fetch)
origin https://github.com/itssyoma/megarepo_12.git (push)
```

Рисунок 7 – Добавление удаленного репозитория



```
megarepo_12 - zsh - 97x10
itssyoma@MacBook-Air-Sema megarepo_12 % git push --prune mirror
Password for 'https://itssyoma@bitbucket.org':
Enumerating objects: 38, done.
Counting objects: 100% (38/38), done.
Delta compression using up to 8 threads
Compressing objects: 100% (37/37), done.
Writing objects: 100% (38/38), 6.30 KiB | 6.30 MiB/s, done.
Total 38 (delta 19), reused 3 (delta 0), pack-reused 0
To https://bitbucket.org/megarepo-12miracle/megarepo_12miracle.git
 * [new branch]      main -> main
```

Рисунок 8 – Загрузка файлов на удаленный репозиторий

Ответы на контрольные вопросы

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

Для выполнения истории коммитов в Git используется команда "git log". Она показывает список всех коммитов в репозитории в обратном хронологическом порядке. Дополнительные опции для просмотра истории коммитов включают:

- "--oneline": выводит каждый коммит в одной строке, содержащей хеш коммита и его сообщение;
- "--graph": отображает графическое представление ветвлений и слияний коммитов;
- "--author": фильтрует коммиты по автору;
- "--since" и "--until": фильтруют коммиты по временному диапазону и т.д.

2. Как ограничить вывод при просмотре истории коммитов?

Чтобы ограничить вывод при просмотре истории коммитов, можно использовать опцию "-n" с указанием числа коммитов, которые нужно показать. Например, "git log -5" покажет только последние 5 коммитов.

3. Как внести изменения в уже сделанный коммит?

Чтобы внести изменения в уже сделанный коммит, необходимо использовать команду "git commit --amend". Эта команда позволяет изменить сообщение коммита или добавить новые изменения к предыдущему коммиту.

4. Как отменить индексацию файла в Git?

Чтобы отменить индексацию файла в Git, используется команда "git reset HEAD <file>". Эта команда удаляет файл из индекса, но оставляет его изменения в рабочей директории.

5. Как отменить изменения в файле?

Чтобы отменить изменения в файле, используется команда "git checkout -- <file>". Эта команда откатывает файл к последнему коммиту, отменяя все несохраненные изменения.

6. Что такое удаленный репозиторий Git?

Удаленный репозиторий Git - это репозиторий, который расположен на удаленном сервере и доступен для совместной работы и обмена кодом с другими разработчиками.

7. Как выполнить просмотр удаленных репозиториях данного локального репозитория?

Для просмотра удаленных репозиториях данного локального репозитория используется команда "git remote -v". Она показывает список всех удаленных репозиториях и их URL-адресов.

8. Как добавить удаленный репозиторий для данного локального репозитория?

Чтобы добавить удаленный репозиторий для данного локального репозитория, используется команда "git remote add <имя> <URL>". Например, "git remote add origin https://github.com/user/repo.git" добавит удаленный репозиторий с именем "origin".

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

Для отправки изменений на удаленный репозиторий используется команда "git push <имя_удаленного_репо> <ветка>". Например, "git push

origin master" отправит изменения на удаленный репозиторий "origin" в ветку "master". Для получения изменений с удаленного репозитория используется команда "git pull <имя_удаленного_репо> <ветка>".

10. Как выполнить просмотр удаленного репозитория?

Чтобы выполнить просмотр удаленного репозитория, можно воспользоваться командой "git remote show <имя_удаленного_репо>". Например, "git remote show origin" покажет информацию о удаленном репозитории "origin", включая URL-адрес, ветки и другие детали.

11. Каково назначение тэгов Git?

Тэги Git используются для пометки определенных коммитов или моментов в истории репозитория. Они позволяют быстро идентифицировать и отслеживать определенные версии кода или важные точки разработки.

12. Как осуществляется работа с тэгами Git?

Работа с тэгами Git осуществляется с помощью команды "git tag". Например, "git tag v1.0" создаст тэг с именем "v1.0" для текущего коммита. Тэги могут быть аннотированными (содержащими сообщение) или легковесными (без сообщения). Тэги также могут быть отправлены на удаленный репозиторий с помощью команды "git push --tags".

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

Флаг "--prune" в командах "git fetch" и "git push" используется для удаления удаленных веток или тэгов, которые больше не существуют на удаленном репозитории. Например, "git fetch --prune" удалит все локальные ветки, которых нет на удаленном репозитории.