

и 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 №2
дисциплины «Программирование на Python»
Вариант ____

Выполнил:
Репкин Александр Павлович
2 курс, группа ИВТ-б-о-22-1,
09.03.01 «Информатика и
вычислительная техника»,
направленность (профиль)
«Программное обеспечение средств
вычислительной
техники и автоматизированных
систем», очная форма обучения

(подпись)

Руководитель практики:
Воронкин Р.А., канд. техн. наук,
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.

Порядок выполнения работы:

1. Получены данные об истории изменения репозитория, используя команду git log.

```
C:\Users\yabuz\FirstRepository>git log
commit 0f39fcca8543eade019693bf15792166ae59c3d0 (HEAD -> main, origin/main, origin/HEAD)
Author: AlexRepkin <realpal30@gmail.com>
Date: Mon Sep 11 11:49:48 2023 +0300

    Added information about local repository in README file

commit 8ab7f89826f9ab2777e66565a4af95e8f0f9726f
Author: Alex <71083717+AlexRepkin@users.noreply.github.com>
Date: Mon Sep 11 11:31:29 2023 +0300

    Initial commit
```

Рисунок 1. Команда git log.

2. Просмотрен патч последнего коммита с помощью команды git log, которой также было передано --patch и число последних записей (В данном случае – одна, т.к. -1).

```
C:\Users\yabuz\FirstRepository>git log --patch -1
commit 0f39fcca8543eade019693bf15792166ae59c3d0 (HEAD -> main, origin/main, origin/HEAD)
Author: AlexRepkin <realpal30@gmail.com>
Date: Mon Sep 11 11:49:48 2023 +0300

    Added information about local repository in README file

diff --git a/README.md b/README.md
index 4b0216d..65384c6 100644
--- a/README.md
+++ b/README.md
@@ -1,2 @@
-# FirstRepository
\ No newline at end of file
+# FirstRepository
+# I love this lesson
\ No newline at end of file
```

Рисунок 2. Команда git log с дополнительными условиями.

3. Получена сокращённая статистика всех коммитов, используя команду git log --stat.

```
C:\Users\yabuz\FirstRepository>git log --stat
commit 0f39fcca8543eade019693bf15792166ae59c3d0 (HEAD -> main, origin/main, origin/HEAD)
Author: AlexRepkin <realpal30@gmail.com>
Date:   Mon Sep 11 11:49:48 2023 +0300

    Added information about local repository in README file

README.md | 3 ++-
1 file changed, 2 insertions(+), 1 deletion(-)

commit 8ab7f89826f9ab2777e66565a4af95e8f0f9726f
Author: Alex <71083717+AlexRepkin@users.noreply.github.com>
Date:   Mon Sep 11 11:31:29 2023 +0300

    Initial commit

README.md | 1 +
1 file changed, 1 insertion(+)
```

Рисунок 3. Команда git log --stat.

4. Изучена возможность изменения формата вывода данных в консоль. Для этого была использована команда `git log --pretty=*`. Сначала опробован формат вывода в одну строку. Затем опробован самостоятельно задаваемый формат вывода с помощью `format` и хешей.

```
C:\Users\yabuz\FirstRepository>git log --pretty=oneline
0f39fcca8543eade019693bf15792166ae59c3d0 (HEAD -> main, origin/main, origin/HEAD) Added information about local repository in README file
8ab7f89826f9ab2777e66565a4af95e8f0f9726f Initial commit
```

Рисунок 4. Вывод в одну строку.

```
C:\Users\yabuz\FirstRepository>git log --pretty=format:"%h - %an _ %cn : %ar , %p : %s"
0f39fcc - AlexRepkin _ AlexRepkin : 2 days ago , 8ab7f89 : Added information about local repository in README file
8ab7f89 - Alex _ GitHub : 2 days ago , : Initial commit
```

Рисунок 5. Использование format.

```
C:\Users\yabuz\FirstRepository>git log --pretty=format:"%h - %an _ %cn : %ar , %p : %s" --graph
* 0f39fcc - AlexRepkin _ AlexRepkin : 2 days ago , 8ab7f89 : Added information about local repository in README file
* 8ab7f89 - Alex _ GitHub : 2 days ago , : Initial commit
```

Рисунок 6. Использование format с --graph.

5. Опробованы функции ограничения вывода. С помощью `-s` была получена информации о выбранной функции – `push`. С помощью команды `git log` – “путь” была получена информация об изменении конкретной директории.

```
C:\Users\yabuz\FirstRepository>git log --since=2.weeks
commit 0f39fcca8543eade019693bf15792166ae59c3d0 (HEAD -> main, origin/main, origin/HEAD)
Author: AlexRepkin <realpal30@gmail.com>
Date:   Mon Sep 11 11:49:48 2023 +0300

    Added information about local repository in README file

commit 8ab7f89826f9ab2777e66565a4af95e8f0f9726f
Author: Alex <71083717+AlexRepkin@users.noreply.github.com>
Date:   Mon Sep 11 11:31:29 2023 +0300

    Initial commit
```

Рисунок 7. Использование git log --since=2.weeks.

```

C:\Users\yabuz\Python1>git log -S push
commit 67b4740fafd4cedce660cdc763a4ccc4574d89ab (HEAD -> main, origin/main, origin/HEAD)
Author: AlexRepkin <realpal30@gmail.com>
Date: Thu Sep 21 10:14:12 2023 +0300

    Added report

commit 67d8bfc31b1c55fb3f2f4842ed8af527d6ca3f82
Author: AlexRepkin <realpal30@gmail.com>
Date: Thu Sep 21 09:31:04 2023 +0300

    Final fixes

commit c67e76b58993caa17c54a0d95b07cfd4e72b07a9
Author: AlexRepkin <realpal30@gmail.com>
Date: Thu Sep 21 09:18:54 2023 +0300

    Some fixes and adjustments.

```

Рисунок 8. Использование git log -s push.

```

C:\Users\yabuz\Python1>git log -- C:\Users\yabuz\Python1\doc
commit 67b4740fafd4cedce660cdc763a4ccc4574d89ab (HEAD -> main, origin/main, origin/HEAD)
Author: AlexRepkin <realpal30@gmail.com>
Date: Thu Sep 21 10:14:12 2023 +0300

    Added report

```

Рисунок 9. Использование git log -- C:\Users\yabuz\Python1\doc.

6. Опробованы функции отмены. Команда `commit --amend` перезаписала последний коммит, добавив к нему файлы с состоянием “подготовленный”. Открылся Блокнот, в котором предлагалось изменить название коммита. При закрытии Блокнота, коммит был перезаписан. Проверка с помощью команды `git log -S commit` показала, что за всё время был сделан только один коммит. После этого была опробована команда отмены индексации файла – `git restore --staged “путь к файлу”`.

```

C:\Users\yabuz\Python2>git commit -m "First commit"
[main 002c2c3] First commit
 4 files changed, 343 insertions(+)
 create mode 100644 "\320\237\321\200\320\276\320\263\321\200\320\260\320\274\320\274\321\213\Programs2Tasks/s.sln"
 create mode 100644 "\320\237\321\200\320\276\320\263\321\200\320\260\320\274\320\274\321\213\Programs2Tasks/s/Python1Tasks.cpp"
 create mode 100644 "\320\237\321\200\320\276\320\263\321\200\320\260\320\274\320\274\321\213\Programs2Tasks/s/Python1Tasks.vcxproj"
 create mode 100644 "\320\237\321\200\320\276\320\263\321\200\320\260\320\274\320\274\321\213\Programs2Tasks/s/Python1Tasks.vcxproj.filters"

C:\Users\yabuz\Python2>git add .

C:\Users\yabuz\Python2>git commit --amend
hint: Waiting for your editor to close the file...

```

Рисунок 10. Использована команда git commit --amend.

```

First commit

# Please enter the commit message for your changes. Lines starting
# with '#' will be ignored, and an empty message aborts the commit.
#
# Date:      Sun Sep 24 20:40:37 2023 +0300
#
# On branch main
# Your branch is ahead of 'origin/main' by 1 commit.
#   (use "git push" to publish your local commits)
#
# Changes to be committed:
#   new file:   "doc/\320\234\320\265\321\202\320\276\320\264\320\270\321\207\320\265\321\201\320\272\320\270\320\265 \321\203\320\272\320\260\320\26
#   new file:   "\320\237\321\200\320\276\320\263\321\200\320\260\320\274\320\274\321\213\Programs2Tasks/Python1Tasks.sln"
#   new file:   "\320\237\321\200\320\276\320\263\321\200\320\260\320\274\320\274\321\213\Programs2Tasks/Python1Tasks/Python1Tasks.cpp"
#   new file:   "\320\237\321\200\320\276\320\263\321\200\320\260\320\274\320\274\321\213\Programs2Tasks/Python1Tasks/Python1Tasks.vcxproj"
#   new file:   "\320\237\321\200\320\276\320\263\321\200\320\260\320\274\320\274\321\213\Programs2Tasks/Python1Tasks/Python1Tasks.vcxproj.filters"
#

```

Рисунок 11. Блокнот.

```

C:\Users\yabuz\Python2>git log -S commit
commit 28100f61b0495cd99c3a5065322716b245f78205 (HEAD -> main)
Author: AlexRepkin <realpal30@gmail.com>
Date:   Sun Sep 24 20:40:37 2023 +0300

    First commit

```

Рисунок 12. Проверка в истории коммитов.

```

C:\Users\yabuz\Python2>git status
On branch main
Your branch is ahead of 'origin/main' by 1 commit.
  (use "git push" to publish your local commits)

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   README.md
        new file:   doc/Additional_txt_file.txt

C:\Users\yabuz\Python2>git restore --staged doc/Additional_txt_file.txt

C:\Users\yabuz\Python2>git status
On branch main
Your branch is ahead of 'origin/main' by 1 commit.
  (use "git push" to publish your local commits)

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   README.md

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        doc/Additional_txt_file.txt

```

Рисунок 13. Отмена индексации файла.

7. Получена информация о доступных удалённых репозиториях, используя команду `git remote`. Добавление доп. условия `-v` к этой команде вывело также адреса репозиторияев. Также получена информация об удалённом репозитории при помощи команды `git remote show`. Команды `git remote rename` и `git remote remove` позволили переименовать и удалить

удалённый репозиторий. Добавлен удалённый репозиторий с помощью команды `git remote add` (короткое имя) (ссылка).

```
C:\Users\yabuz\Python2>git remote
origin
```

Рисунок 14. Команда `git remote`.

```
C:\Users\yabuz\Python2>git remote -v
origin https://github.com/AlexRepkin/Python2.git (fetch)
origin https://github.com/AlexRepkin/Python2.git (push)
```

Рисунок 15. Команда `git remote -v`.

```
C:\Users\yabuz\Python2>git remote add smth https://github.com/AlexRepkin/Python1.git
C:\Users\yabuz\Python2>git remote -v
origin https://github.com/AlexRepkin/Python2.git (fetch)
origin https://github.com/AlexRepkin/Python2.git (push)
smth https://github.com/AlexRepkin/Python1.git (fetch)
smth https://github.com/AlexRepkin/Python1.git (push)
```

Рисунок 16. Команда `git remote add` добавила удалённый репозиторий.

```
C:\Users\yabuz\Python2>git remote show origin
* remote origin
Fetch URL: https://github.com/AlexRepkin/Python2.git
Push URL: https://github.com/AlexRepkin/Python2.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 (fast-forwardable)
```

Рисунок 17. Команда `git remote show`.

```
C:\Users\yabuz\Python2>git remote rename smth bye-bye
C:\Users\yabuz\Python2>git remote
bye-bye
origin
```

Рисунок 18. Переименовывание удалённого репозитория.

```
C:\Users\yabuz\Python2>git remote remove bye-bye
C:\Users\yabuz\Python2>git remote
origin
```

Рисунок 19. Удаление удалённого репозитория.

8. Рассмотрена возможность работы с тегами. Команда `git tag -a v(версия) -m "Название версии"` позволило создать свой собственный тег. Его существование было подтверждено командами `git tag` и `git show`. Команда `git push v0.0` отправляет тег на сервер. Опробована возможность удаления тегов, для этого был создан тег `v0.1`, который был удалён локально с помощью команды `git tag -d v0.1` и с сервера с помощью команды `git push origin --delete v0.1`.

```
C:\Users\yabuz\Python2>git tag -a v0.0 -m "First ever version"

C:\Users\yabuz\Python2>git tag
v0.0

C:\Users\yabuz\Python2>git show v0.0
tag v0.0
Tagger: AlexRepkin <realpal30@gmail.com>
Date: Mon Sep 25 08:28:50 2023 +0300

First ever version

commit cd11a1f119acdc3a7c9a91c6754d030efb4bf41 (HEAD -> main, tag: v0.0)
Author: AlexRepkin <realpal30@gmail.com>
Date: Sun Sep 24 21:19:50 2023 +0300

    Cancelling is done.

diff --git a/README.md b/README.md
index 227470b..3989b35 100644
--- a/README.md
+++ b/README.md
@@ -1,2 +1,2 @@
 # Python2
-Лабораторная работа №2 Репкина Александра. ИВТ-6-о-22-1 2 курс. Дисциплина - Программирование на Python
+Лабораторная работа №2 Репкина Александра. ИВТ-6-о-22-1 2 курс. Дисциплина - Программирование на Python.
```

Рисунок 20. Создание собственного тега и просмотр существующих.

```
C:\Users\yabuz\Python2>git push origin v0.0
Enumerating objects: 17, done.
Counting objects: 100% (17/17), done.
Delta compression using up to 16 threads
Compressing objects: 100% (14/14), done.
Writing objects: 100% (15/15), 424.09 KiB | 8.83 MiB/s, done.
Total 15 (delta 2), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (2/2), completed with 1 local object.
To https://github.com/AlexRepkin/Python2.git
 * [new tag]          v0.0 -> v0.0
```

Рисунок 21. Отправка на сервер созданного тега.


```

C:\Users\yabuz\Python2>git tag -a v0.1 -m "Delete this tag"

C:\Users\yabuz\Python2>git push origin v0.1
Enumerating objects: 1, done.
Counting objects: 100% (1/1), done.
Writing objects: 100% (1/1), 167 bytes | 55.00 KiB/s, done.
Total 1 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/AlexRepkin/Python2.git
 * [new tag]          v0.1 -> v0.1

C:\Users\yabuz\Python2>git tag -d v0.1
Deleted tag 'v0.1' (was 18779b7)

C:\Users\yabuz\Python2>git push origin --delete v0.1
To https://github.com/AlexRepkin/Python2.git
 - [deleted]          v0.1

```

Рисунок 22. Удаление тега с локального репозитория и сервера.

9. Использована команда `git log --graph --pretty=oneline --abbrev-commit`.

```

C:\Users\yabuz\Python2>git log --graph --pretty=oneline --abbrev-commit
* 43b8a4c (HEAD -> main, tag: v1.0, origin/main, origin/HEAD) Final project with random player
* f4f8145 (tag: v0.2) Added characters for players.
* b74e269 (tag: v0.1) No symbols abilities here yet
* 96437f0 Start of the project
* cd11a1f (tag: v0.0) Cancelling is done.
* 28100f6 First commit
* 2f966ee Initial commit

```

Рисунок 23. Команда `git log --graph --pretty=oneline --abbrev-commit`.

10. Использована команда `git show` с разными значениями: HEAD, HEAD~1, b34a0e.

```

C:\Users\yabuz\Python2>git show HEAD
commit 43b8a4cd9bbc9739d1602eb087dcc093cc39151b (HEAD -> main, tag: v1.0, origin/main, origin/HEAD)
Author: AlexRepkin <realpal30@gmail.com>
Date:   Wed Sep 27 07:49:12 2023 +0300

    Final project with random player

diff --git "a/\320\237\321\200\320\276\320\263\321\200\320\260\320\274\320\274\321\213\Python2_Tasks_On_C++/Python2_Tasks_On_C++/Python2_Tasks_On_C++.cpp" "b/\320\237\321\200\320\276\320\263\321\200\320\260\320\274\320\274\321\213\Python2_Tasks_On_C++/Python2_Tasks_On_C++/Python2_Tasks_On_C++.cpp"
index e505e9e..0e9d41f 100644
--- "a/\320\237\321\200\320\276\320\263\321\200\320\260\320\274\320\274\321\213\Python2_Tasks_On_C++/Python2_Tasks_On_C++/Python2_Tasks_On_C++.cpp"
+++ "b/\320\237\321\200\320\276\320\263\321\200\320\260\320\274\320\274\321\213\Python2_Tasks_On_C++/Python2_Tasks_On_C++/Python2_Tasks_On_C++.cpp"
@@ -22,6 +22,7 @@ bool checking() {

    bool playing() {
        cout << "-----\n| | | \n-----\n| | | \n-----\n| | | \n-----\n";
+       srand((unsigned)time(0));
        player = rand() % 2;
        int space;
        char current;

```

Рисунок 24. Команда `git show HEAD`.


```

C:\Users\yabuz\Python2>git show HEAD~1
commit f4f81450d9738406aedcae6f8489b309e46d8cbc (tag: v0.2)
Author: AlexRepin <realpal30@gmail.com>
Date:   Wed Sep 27 07:46:17 2023 +0300

    Added characters for players.

diff --git "a/\320\237\321\200\320\276\320\263\321\200\320\260\320\274\320\274\321\213\Python2_Tasks_On_C++/Python2_Tasks_On_C++/Python2_Tasks_On_C++.cpp" "b/\320\237\321\200\320\276\320\263\321\200\320\260\320\274\320\274\321\213\Python2_Tasks_On_C++/Python2_Tasks_On_C++/Python2_Tasks_On_C++.cpp"
index a79763d..e505e9e 100644
--- "a/\320\237\321\200\320\276\320\263\321\200\320\260\320\274\320\274\321\213\Python2_Tasks_On_C++/Python2_Tasks_On_C++/Python2_Tasks_On_C++.cpp"
+++ "b/\320\237\321\200\320\276\320\263\321\200\320\260\320\274\320\274\321\213\Python2_Tasks_On_C++/Python2_Tasks_On_C++/Python2_Tasks_On_C++.cpp"
@@ -8,12 +8,12 @@ bool continuing, player;

char matrix[9] = { ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ', ' ' };

-//void symbols() {
-//    cout << "Please, choose symbol for the first player.\nCharacter - ";
-//    cin >> first;
-//    cout << "Please, choose symbol for the second player.\nCharacter - ";
-//    cin >> second;
-//}
+void symbols() {
+    cout << "Please, choose symbol for the first player.\nCharacter - ";
+    cin >> first;
+    cout << "Please, choose symbol for the second player.\nCharacter - ";
+    ...skipping...
commit f4f81450d9738406aedcae6f8489b309e46d8cbc (tag: v0.2)
Author: AlexRepin <realpal30@gmail.com>

```

Рисунок 25. Команда git show HEAD~1.

```

C:\Users\yabuz\Python2>git show b34a0e
fatal: ambiguous argument 'b34a0e': unknown revision or path not in the working tree.
Use '--' to separate paths from revisions, like this:
'git <command> [<revision>...] -- [<file>...]'

```

Рисунок 26. Команда git show b34a0e.

11. Удалён весь код из файла Python2_Tasks_On_C++.cpp. После удаления использована команда git checkout -- Python2_Tasks_On_C++.cpp, что привело к удалению не сохранённых в указанном файле изменений (Код вернулся). После этого испробована команда git reset --hard HEAD~1, что привело к идентичному результату.

```

C:\Users\yabuz\Python2\Программы\Python2_Tasks_On_C++\Python2_Tasks_On_C++>git checkout -- Python2_Tasks_On_C++.cpp
C:\Users\yabuz\Python2\Программы\Python2_Tasks_On_C++\Python2_Tasks_On_C++>

```

Рисунок 27. Команда git checkout -- Python2_Tasks_On_C++.cpp.

```

C:\Users\yabuz\Python2\Программы\Python2_Tasks_On_C++\Python2_Tasks_On_C++>git reset --hard HEAD~1
HEAD is now at b74e269 No symbols abilities here yet

```

Рисунок 28. Команда git reset --hard HEAD~1.

12. Создан аккаунт на сервисе BitBucket. После этого был создан новый репозиторий. Активирована команда git push --prune <https://bitbucket.org/python2ncfu/python2/src/origin/>, после предоставления доступа BitBucket, произведено копирование файлов в ново созданный

репозиторий на BitBucket с локального репозитория от GitHub. Полученный результат: <https://bitbucket.org/python2ncfu/python2/src/main/>.

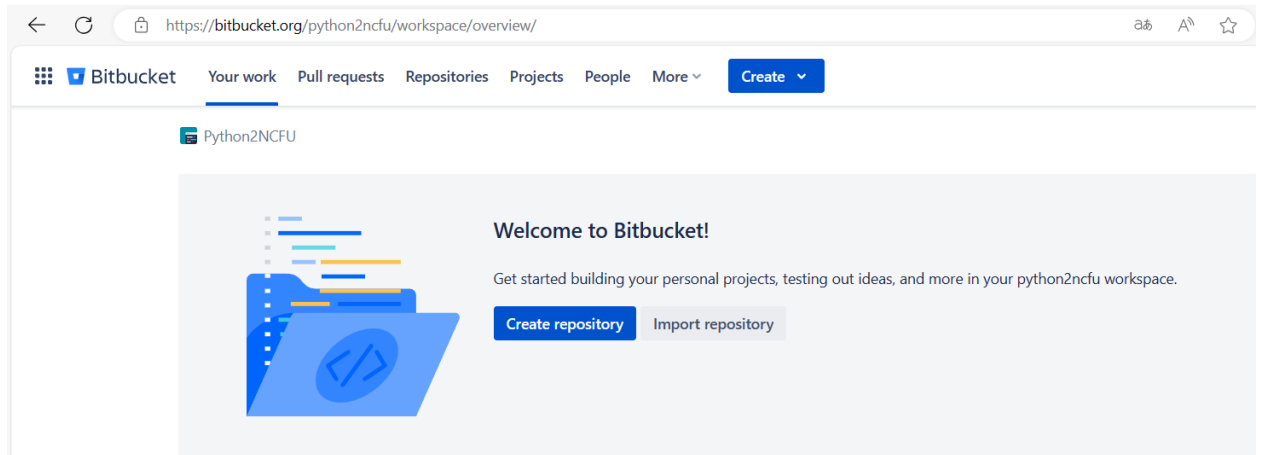


Рисунок 29. Новое рабочее пространство на BitBucket.

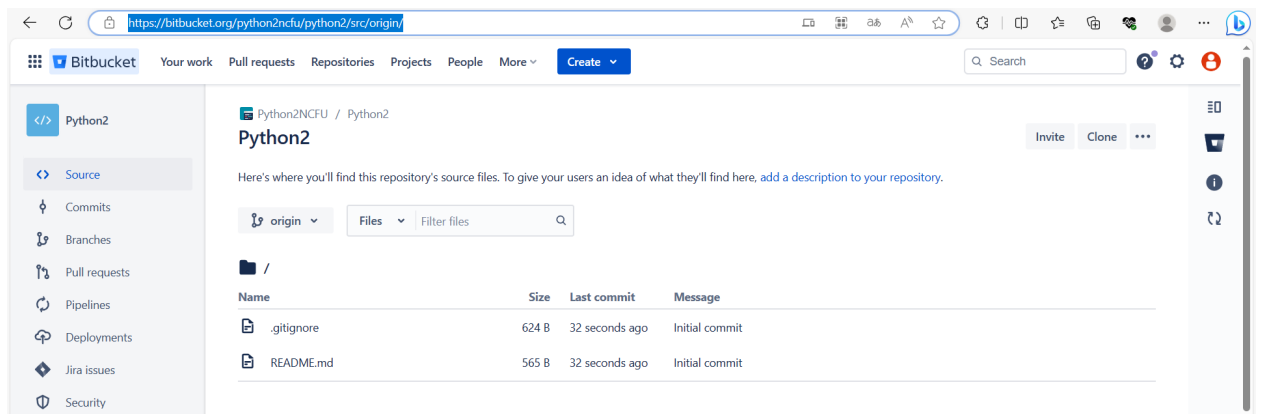


Рисунок 30. Новый репозиторий на BitBucket.

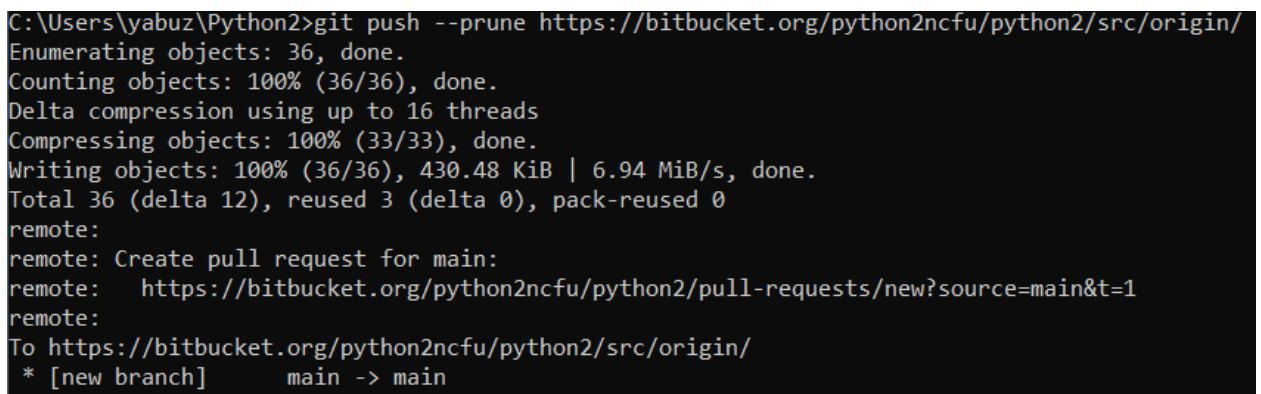


Рисунок 31. Отправка файлов на удалённый репозиторий BitBucket.

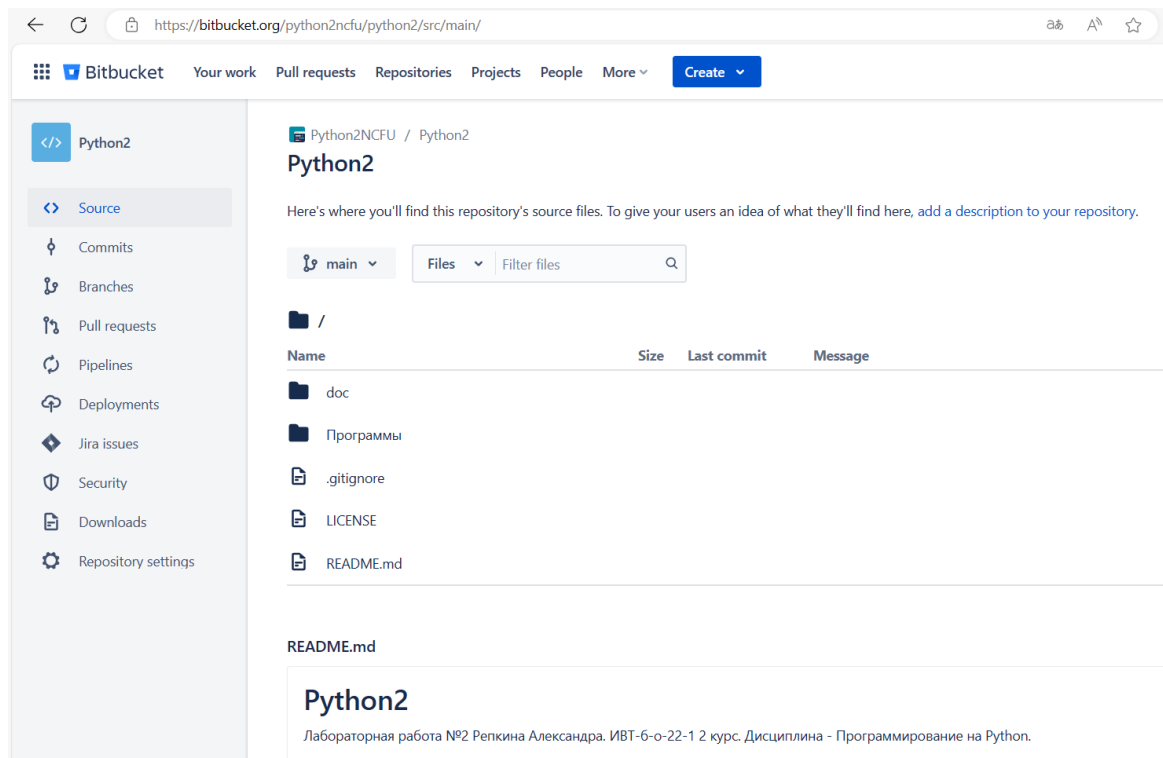


Рисунок 32. Результат клонирования.

13. Ответы на вопросы.

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

Ответ: для просмотра истории репозитория используется команда `git log`. Для облегчения просмотра были также созданы дополнительные опции, добавляющие или убирающие дополнительную информацию: `-p(--patch)` показывает изменения, внесённые в данной версии; `--stat` показывает статистику изменённых файлов в коммите; `--shortstat` укороченная версия `--stat`, выводящая только число внесённых изменений; `--name-only` выводит только имена изменившихся файлов; `--abbrev-commit` выводит только несколько символов хеш-суммы, вместо всех 40; `--relative-date` отображает даты в ином от изначального формате; `--graph` изменяет отображение выводимых коммитов, добавляя ветвление; `--pretty` изменяет отображение коммита, делая его более лёгким для чтения; `--oneline` выводит всю информацию о коммите в одну линию.

2) Как ограничить вывод при просмотре истории коммитов?

Ответ: для ограничения вывода существует несколько дополнительных опций к команде `git log`. Среди них: -“цифра” выводит два последних сделанных коммита; `--since` или `--until` указывают на временной промежуток, например `--since=2.weeks` выведет данные за последние две недели, а `--until=2.weeks` выведет данные, предшествующие последним двум неделям. Также существует `--author`, показывающий только коммиты, в которых содержатся файлы, созданные конкретным автором, и `--committer`, показывающий только коммиты, которые были сделаны определённым пользователем. `--grep` выводит коммиты, содержащие в своём описании определённое сообщение. `-S` выводит коммиты, в которых была затронута определённая функция.

3) Как внести изменения в уже сделанный коммит?

Ответ: для перезаписи сделанного коммита есть дополнительная опция для команды `git commit`, это `--amend`. Если необходимо добавить файл в уже сделанный коммит, то нужно использовать команду `git commit --amend` сразу после команды `git add`.

4) Как отменить индексацию файла в Git?

Ответ: для отмены индексации файла создана команда `git restore --staged` “путь к файлу”.

5) Как отменить изменения в файле?

Ответ: для отмены изменений в файле существует команда `git checkout` – “имя файла”.

6) Что такое удалённый репозиторий Git?

Ответ: Удалённые репозитории - версии проекта, сохранённые в интернете или где-то в сети. У пользователя может быть несколько удалённых репозиториях, каждый из которых может быть доступен для чтения или для чтения-записи.

7) Как выполнить просмотр удалённых репозиториях данного локального репозитория?

Ответ: для просмотра удалённых репозиториев можно выполнить команду `git remote (-v`, если нужно также получить ссылку на данный репозиторий).

8) Как добавить удалённый репозиторий для данного локального репозитория?

Ответ: для добавления удалённых репозиториев существует команда `git remote add` “кодовое слово” “ссылка на удалённый репозиторий”.

9) Как выполнить отправку\получение изменений с удалённого репозитория?

Ответ: для получения изменений из удалённого репозитория можно использовать команду `git fetch` “кодовое слово” или более усовершенствованную команду - `git pull`. Для отправки изменений можно использовать команду `git push` “кодовое слово” “название ветки”.

10) Как выполнить просмотр удалённого репозитория?

Ответ: для просмотра удалённого репозитория можно использовать команду `git remote show` “имя репозитория”.

11) Каково назначение тегов Git?

Ответ: теги существуют для отметки определённых моментов в истории (К примеру - выпуск новой версии).

12) Как осуществляется работа с тегами Git?

Ответ: большинство команд, связанных с тегами, начинаются с `git tag`. Команда `git tag` покажет список существующих в репозитории тегов. Команда `git tag -a` “кодовое название” -m “Описание” создаст новый тег на основе последнего выполненного коммита. Для отправки тегов на удалённый репозиторий существует команда `git push origin` “имя тега или, если нужно отправить все теги, --tags”. Для удаления тега существует команда `git tag -d` “название тега”. Для получения версий файлов, на которые ссылается тег, можно использовать команду `git checkout` “название тега”.

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

Ответ: `git push --prune` удаляет файлы в удалённом репозитории, если они отсутствуют на локальном репозитории. `git fetch --prune` удаляет файлы в локальном репозитории, если они отсутствуют на удалённом репозитории.

Вывод: в ходе выполнения лабораторной работы были исследованы базовые возможности системы контроля версий Git для работы с локальными репозиториями.