

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 №1.3
дисциплины
«Основы кроссплатформенного программирования»

Выполнил:
Наумов Никита Викторович 1 курс,
группа ИТС-б-о-22-1,
11.03.02 «Инфокоммуникационные
технологии и системы связи»,
направленность (профиль)
«Инфокоммуникационные системы и
сети», заочная форма обучения

(подпись)

Проверил:
Воронкин Р. А., доцент кафедры
инфокоммуникаций

(подпись)

Отчет защищен с оценкой _____ Дата защиты _____

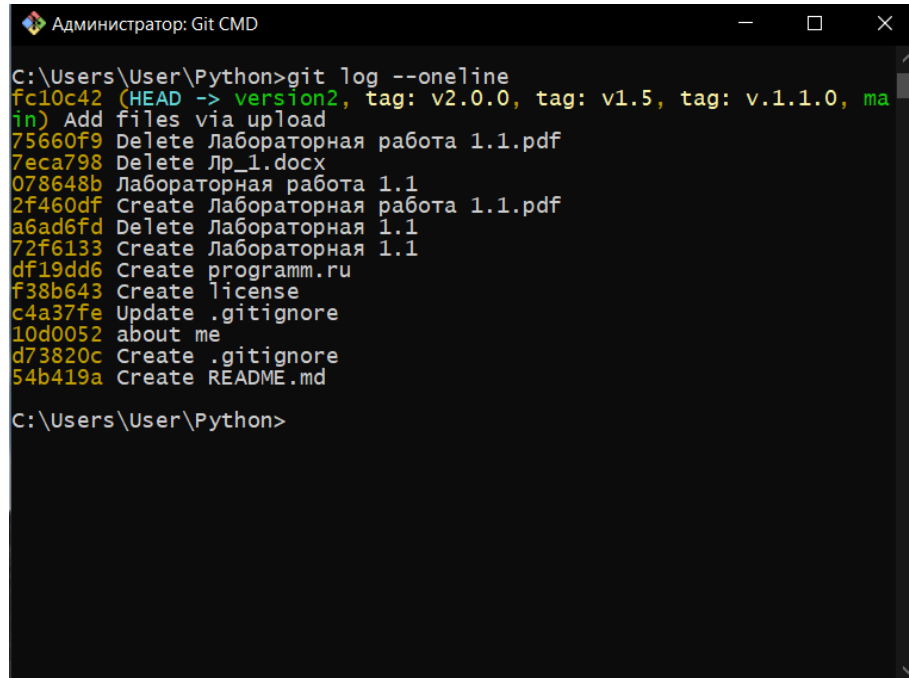
Ставрополь, 2023 г.

Тема: Основы ветвления Git.

Цель: Исследование базовых возможностей по работе с локальными и удаленными ветками Git.

Ход работы:

1. HEAD

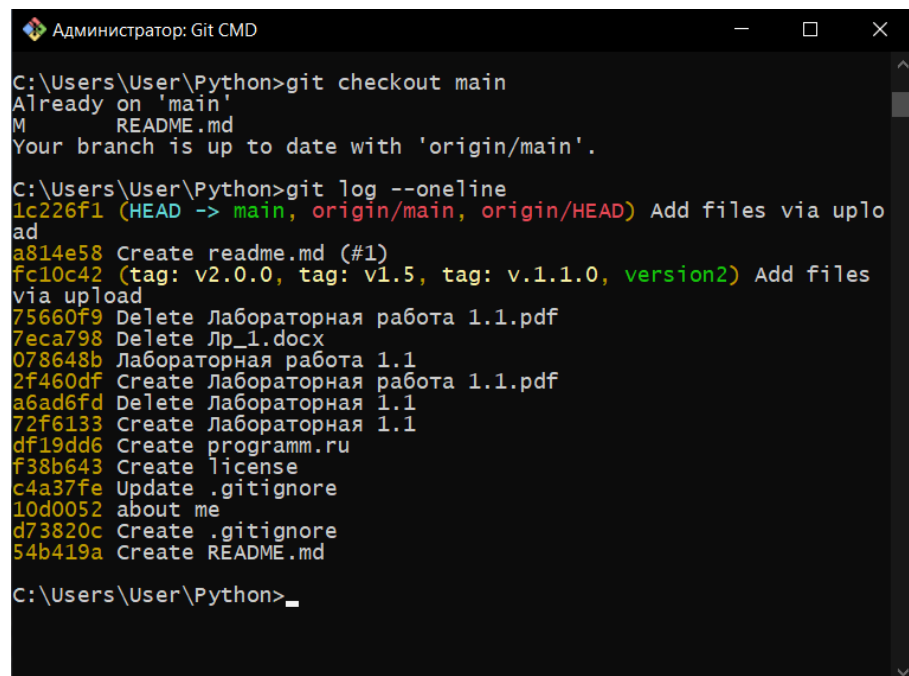


```
Администратор: Git CMD

C:\Users\User\Python>git log --oneline
fc10c42 (HEAD -> version2, tag: v2.0.0, tag: v1.5, tag: v.1.1.0, main) Add files via upload
75660f9 Delete Лабораторная работа 1.1.pdf
7eca798 Delete Лр_1.docx
078648b Лабораторная работа 1.1
2f460df Create Лабораторная работа 1.1.pdf
a6ad6fd Delete Лабораторная 1.1
72f6133 Create Лабораторная 1.1
df19dd6 Create programm.ru
f38b643 Create license
c4a37fe Update .gitignore
10d0052 about me
d73820c Create .gitignore
54b419a Create README.md

C:\Users\User\Python>
```

Рисунок 1



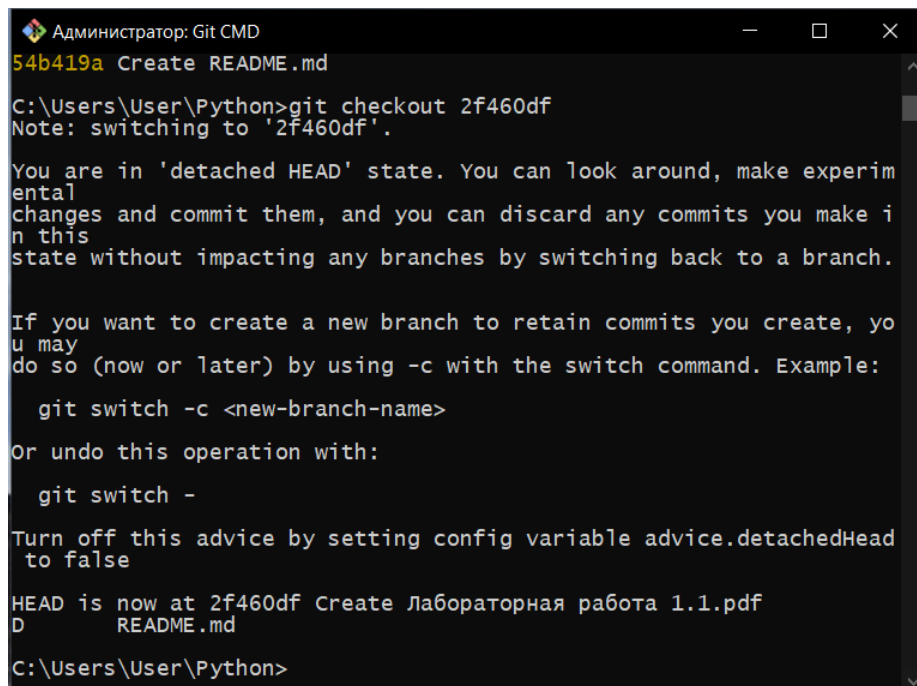
```
Администратор: Git CMD

C:\Users\User\Python>git checkout main
Already on 'main'
M   README.md
Your branch is up to date with 'origin/main'.

C:\Users\User\Python>git log --oneline
1c226f1 (HEAD -> main, origin/main, origin/HEAD) Add files via upload
a814e58 Create readme.md (#1)
fc10c42 (tag: v2.0.0, tag: v1.5, tag: v.1.1.0, version2) Add files via upload
75660f9 Delete Лабораторная работа 1.1.pdf
7eca798 Delete Лр_1.docx
078648b Лабораторная работа 1.1
2f460df Create Лабораторная работа 1.1.pdf
a6ad6fd Delete Лабораторная 1.1
72f6133 Create Лабораторная 1.1
df19dd6 Create programm.ru
f38b643 Create license
c4a37fe Update .gitignore
10d0052 about me
d73820c Create .gitignore
54b419a Create README.md

C:\Users\User\Python>
```

Рисунок 2



```
Администратор: Git CMD
54b419a Create README.md

C:\Users\User\Python>git checkout 2f460df
Note: switching to '2f460df'.

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:

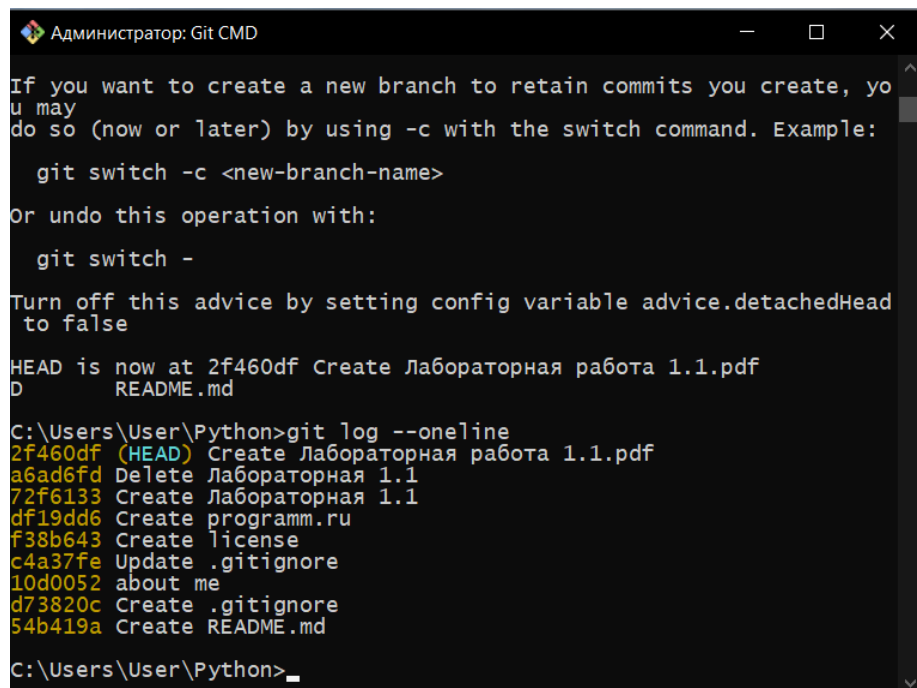
    git switch -

Turn off this advice by setting config variable advice.detachedHead to false

HEAD is now at 2f460df Create Лабораторная работа 1.1.pdf
D      README.md

C:\Users\User\Python>
```

Рисунок 2



```
Администратор: Git CMD

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 2f460df Create Лабораторная работа 1.1.pdf
D      README.md

C:\Users\User\Python>git log --oneline
2f460df (HEAD) Create Лабораторная работа 1.1.pdf
a6ad6fd Delete Лабораторная 1.1
72f6133 Create Лабораторная 1.1
df19dd6 Create programm.ru
f38b643 Create license
c4a37fe Update .gitignore
10d0052 about me
d73820c Create .gitignore
54b419a Create README.md

C:\Users\User\Python>
```

Рисунок 3

```
Администратор: Git CMD

10d0052 about me
d73820c Create .gitignore
54b419a Create README.md

C:\Users\User\Python>git checkout 1c226f1
Previous HEAD position was 2f460df Create Лабораторная работа 1.1.p
df
HEAD is now at 1c226f1 Add files via upload
M      README.md

C:\Users\User\Python> git log --oneline
1c226f1 (HEAD, origin/main, origin/HEAD, main) Add files via upload
a814e58 Create readme.md (#1)
fc10c42 (tag: v2.0.0, tag: v1.5, tag: v.1.1.0, version2) Add files
via upload
75660f9 Delete Лабораторная работа 1.1.pdf
7eca798 Delete Лр_1.docx
078648b Лабораторная работа 1.1
2f460df Create Лабораторная работа 1.1.pdf
a6ad6fd Delete Лабораторная 1.1
72f6133 Create Лабораторная 1.1
df19dd6 Create programm.ru
f38b643 Create license
c4a37fe Update .gitignore
10d0052 about me
d73820c Create .gitignore
54b419a Create README.md

C:\Users\User\Python>
```

Рисунок 4

```
Администратор: Git CMD

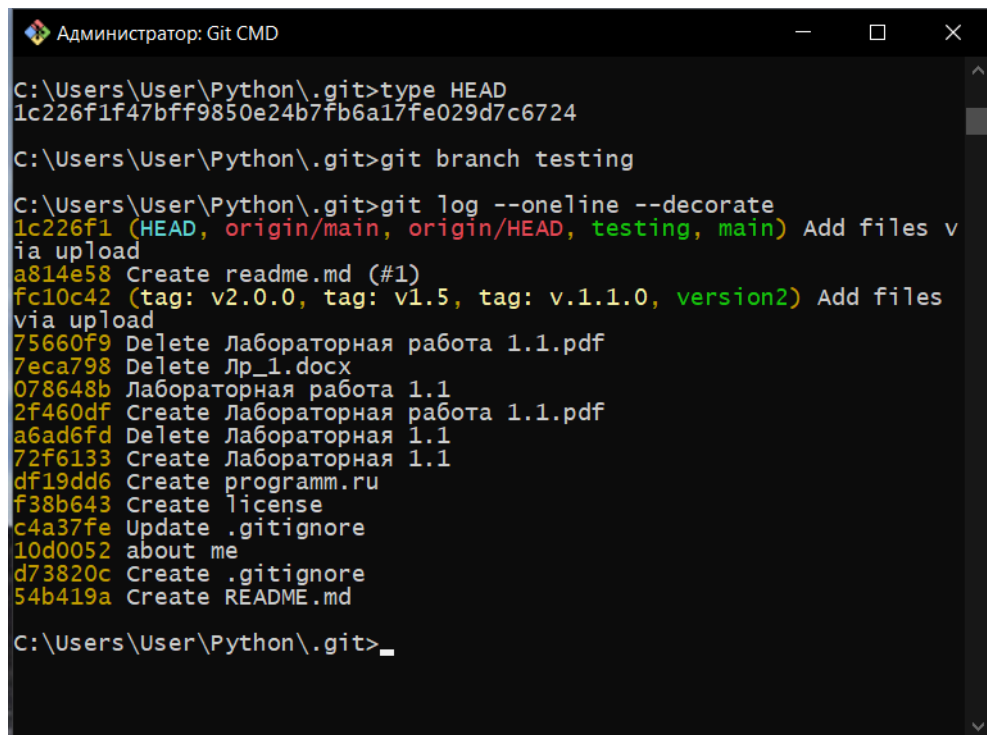
C:\Users\User\Python\.git>dir
Том в устройстве C имеет метку Windows 10
Серийный номер тома: 281B-DD32

Содержимое папки C:\Users\User\Python\.git

06.06.2023  19:33           12 288 .COMMIT_EDITMSG.swo
06.06.2023  19:25           12 288 .COMMIT_EDITMSG.swp
08.06.2023  03:23            15 COMMIT_EDITMSG
08.06.2023  02:38           435 config
06.06.2023  17:53            73 description
08.06.2023  03:10          200 FETCH_HEAD
08.06.2023  03:29           41 HEAD
06.06.2023  17:53      <DIR>      hooks
08.06.2023  03:29          665 index
06.06.2023  17:53      <DIR>      info
06.06.2023  17:58      <DIR>      logs
08.06.2023  03:10      <DIR>      objects
08.06.2023  03:10           41 ORIG_HEAD
06.06.2023  21:23          112 packed-refs
06.06.2023  17:58      <DIR>      refs
                        10 файлов      26 158 байт
                        5 папок      352 751 566 848 байт свободно

C:\Users\User\Python\.git>
```

Рисунок 5



```
Администратор: Git CMD
C:\Users\User\Python\.git>type HEAD
1c226f1f47bfff9850e24b7fb6a17fe029d7c6724

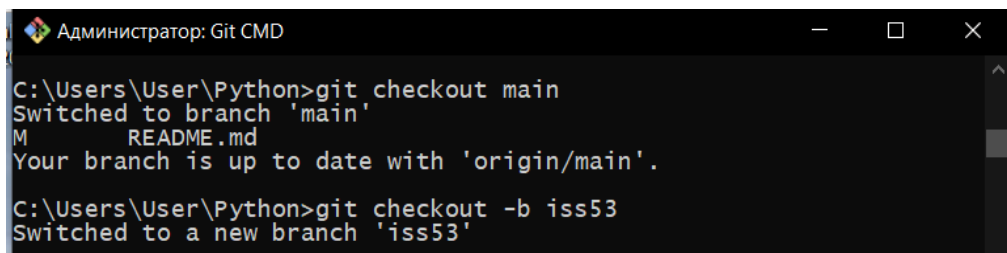
C:\Users\User\Python\.git>git branch testing

C:\Users\User\Python\.git>git log --oneline --decorate
1c226f1 (HEAD, origin/main, origin/HEAD, testing, main) Add files v
ia upload
a814e58 Create readme.md (#1)
fc10c42 (tag: v2.0.0, tag: v1.1.0, version2) Add files
via upload
75660f9 Delete Лабораторная работа 1.1.pdf
7eca798 Delete Лр_1.docx
078648b Лабораторная работа 1.1
2f460df Create Лабораторная работа 1.1.pdf
a6ad6fd Delete Лабораторная 1.1
72f6133 Create Лабораторная 1.1
df19dd6 Create programm.ru
f38b643 Create license
c4a37fe Update .gitignore
10d0052 about me
d73820c Create .gitignore
54b419a Create README.md

C:\Users\User\Python\.git>
```

Рисунок 6

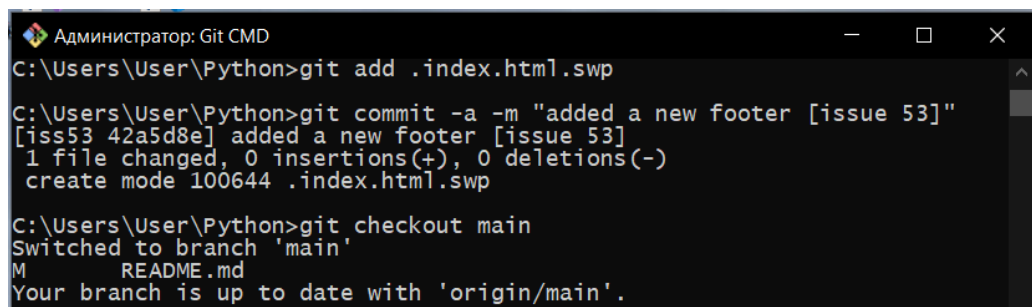
2. Создание новой ветки.



```
Администратор: Git CMD
C:\Users\User\Python>git checkout main
Switched to branch 'main'
M      README.md
Your branch is up to date with 'origin/main'.

C:\Users\User\Python>git checkout -b iss53
Switched to a new branch 'iss53'
```

Рисунок 7



```
Администратор: Git CMD
C:\Users\User\Python>git add .index.html.swp

C:\Users\User\Python>git commit -a -m "added a new footer [issue 53]"
[iss53 42a5d8e] added a new footer [issue 53]
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 .index.html.swp

C:\Users\User\Python>git checkout main
Switched to branch 'main'
M      README.md
Your branch is up to date with 'origin/main'.
```

Рисунок 8

```
Администратор: Git CMD
Your branch is up to date with 'origin/testing'.

C:\Users\User\Python>git checkout main
Switched to branch 'main'
D      README.md
Your branch is up to date with 'origin/main'.

C:\Users\User\Python>git checkout -b hotfix
Switched to a new branch 'hotfix'

C:\Users\User\Python>vim index.html

C:\Users\User\Python>git checkout hotfix
Already on 'hotfix'
D      README.md

C:\Users\User\Python>vim index.html
```

Рисунок 9

```
Администратор: Git CMD

C:\Users\User\Python>git commit -a -m "fixed the broken
email address"
[main f497c74] fixed the broken email address
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 git

C:\Users\User\Python>git checkout main
Switched to branch 'main'
Your branch is ahead of 'origin/main' by 1 commit.
(use "git push" to publish your local commits)

C:\Users\User\Python>git merge hotfix
Merge made by the 'ort' strategy.
 README.md | 2 --
1 file changed, 2 deletions(-)
delete mode 100644 README.md

C:\Users\User\Python>
```

Рисунок 10

```
Администратор: Git CMD

C:\Users\User\Python>git branch -d hotfix
Deleted branch hotfix (was 2a68aa7).

C:\Users\User\Python>git checkout iss53
Switched to branch 'iss53'
Your branch is up to date with 'origin/iss53'.

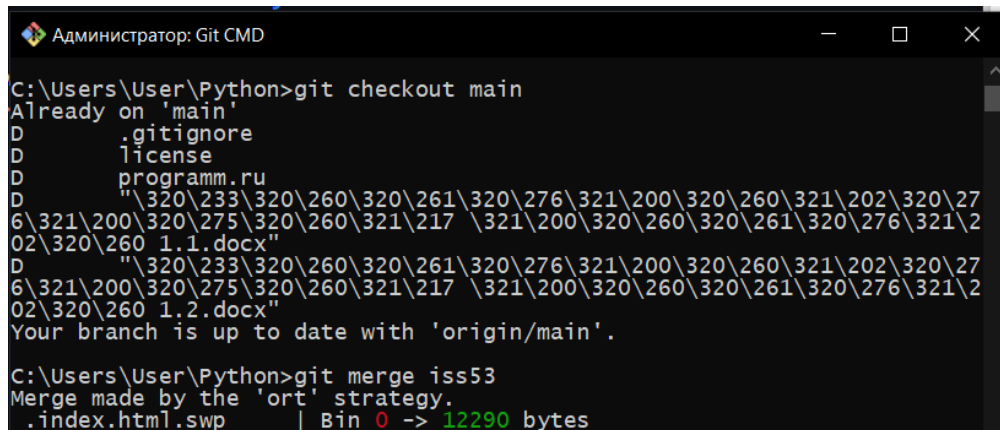
C:\Users\User\Python>vim index.html

C:\Users\User\Python>git commit -a -m "finished the new
footer [issue 53]"
[iss53 349e3dc] finished the new footer [issue 53]
1 file changed, 0 insertions(+), 0 deletions(-)

C:\Users\User\Python>
```

Рисунок 11

3. Основы слияния.

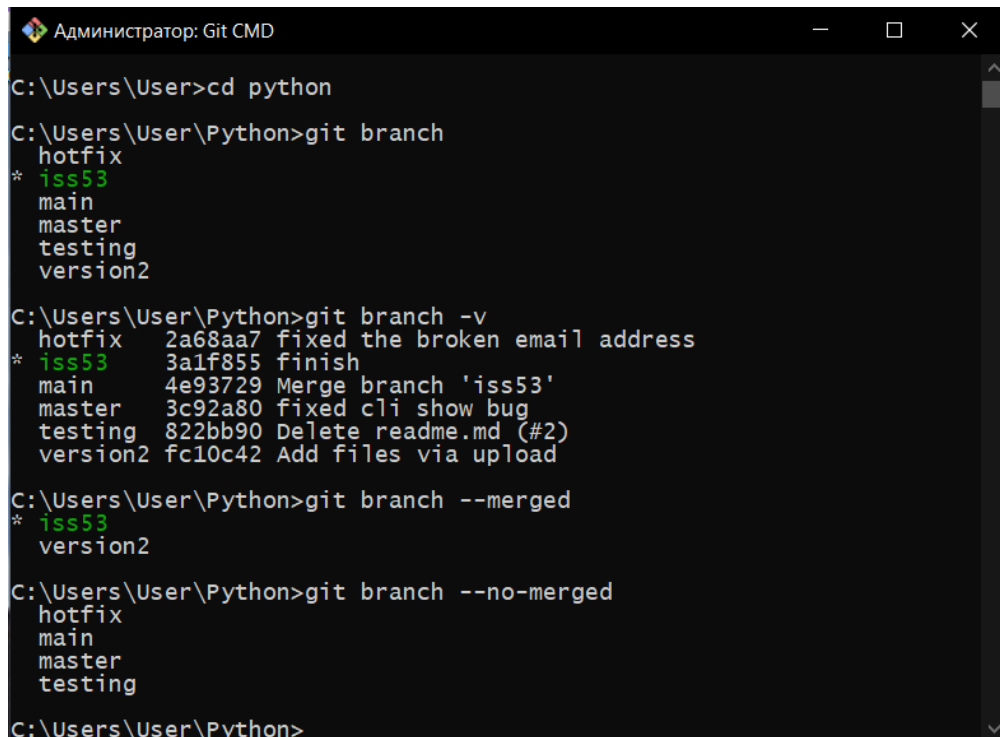


```
Администратор: Git CMD
C:\Users\User\Python>git checkout main
Already on 'main'
D      .gitignore
D      license
D      programm.ru
D      "\320\233\320\260\320\261\320\276\321\200\320\260\321\202\320\27
6\321\200\320\275\320\260\321\217 \321\200\320\260\320\261\320\276\321\2
02\320\260 1.1.docx"
D      "\320\233\320\260\320\261\320\276\321\200\320\260\321\202\320\27
6\321\200\320\275\320\260\321\217 \321\200\320\260\320\261\320\276\321\2
02\320\260 1.2.docx"
Your branch is up to date with 'origin/main'.

C:\Users\User\Python>git merge iss53
Merge made by the 'ort' strategy.
.index.html.swp | Bin 0 -> 12290 bytes
```

Рисунок 12

4. Управление ветками



```
Администратор: Git CMD
C:\Users\User>cd python
C:\Users\User\Python>git branch
hotfix
* iss53
main
master
testing
version2

C:\Users\User\Python>git branch -v
hotfix      2a68aa7 fixed the broken email address
* iss53     3a1f855 finish
main        4e93729 Merge branch 'iss53'
master      3c92a80 fixed cli show bug
testing     822bb90 Delete readme.md (#2)
version2    fc10c42 Add files via upload

C:\Users\User\Python>git branch --merged
* iss53
version2

C:\Users\User\Python>git branch --no-merged
hotfix
main
master
testing

C:\Users\User\Python>
```

Рисунок 13

```
Администратор: Git CMD
C:\Users\User\Python>git branch -vv
hotfix    2a68aa7 [origin/hotfix] fixed the broken email address
iss53     3a1f855 [origin/iss53] finish
* main    4e93729 [origin/main] Merge branch 'iss53'
master    3c92a80 [paul/master] fixed cli show bug
testing   822bb90 [origin/testing] Delete readme.md (#2)
version2  fc10c42 Add files via upload

C:\Users\User\Python>git fetch --all
Fetching origin
remote: Enumerating objects: 6, done.
remote: Counting objects: 100% (6/6), done.
remote: Compressing objects: 100% (5/5), done.
remote: Total 5 (delta 2), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (5/5), 1.76 KiB | 8.00 KiB/s, done.
From https://github.com/Diana-372/Python
  4e93729..58fd3a3  main       -> origin/main
  2a68aa7..58fd3a3  hotfix     -> origin/hotfix
  3a1f855..58fd3a3  iss53     -> origin/iss53
+ 3c92a80..58fd3a3  master    -> origin/master (forced update)
+ 822bb90..e7e9163  testing   -> origin/testing (forced update)
  fc10c42..58fd3a3  version2  -> origin/version2
Fetching paul
C:\Users\User\Python>
```

Рисунок 14

```
Администратор: Git CMD
C:\Users\User\Python>git branch -vv
hotfix    2a68aa7 [origin/hotfix: behind 12] fixed the broken email address
iss53     3a1f855 [origin/iss53: behind 14] finish
* main    4e93729 [origin/main: behind 1] Merge branch 'iss53'
master    3c92a80 [paul/master] fixed cli show bug
testing   822bb90 [origin/testing: ahead 1, behind 19] Delete readme.md (#2)
version2  fc10c42 Add files via upload
```

Рисунок 15

Контрольные вопросы:

1. Ветка в Git это параллельная версия проекта, которая может быть использована для разработки новых фич, исправления багов или проведения экспериментов, не влияющих на основную версию проекта.
2. HEAD в Git это указатель на текущую ветку или коммит, на котором находится пользователь или репозиторий.
3. Способы создания веток в Git:
 - Создание новой ветки с помощью команды `git branch <имя_ветки>`
 - Создание и переключение на новую ветку с помощью команды `git checkout -b <имя_ветки>`
4. Текущую ветку можно узнать с помощью команды `git branch`, которая выведет список всех существующих веток, при этом звездочка (*) будет указывать на текущую ветку.
5. Для переключения между ветками можно использовать команду `git checkout <имя_ветки>`, где <имя_ветки> - это имя ветки, на которую нужно переключиться.
6. Удаленная ветка в Git это ветка, которая находится на удаленном сервере (например, на Bitbucket), а не локально на компьютере пользователя.
7. Ветка отслеживания (tracking branch) в Git это локальная ветка, которая связана с удаленной веткой и служит для отслеживания изменений в удаленной ветке и синхронизации их с локальной веткой.
8. Создать ветку отслеживания можно с помощью команды `git checkout -b <имя_ветки> <имя_удаленной_ветки>`.
9. Чтобы отправить изменения из локальной ветки в удаленную ветку, нужно выполнить команду `git push <имя_удаленного_репозитория> <имя_локальной_ветки>:<имя_удаленной_ветки>`.
10. Основное отличие между командами `git fetch` и `git pull` заключается в том, что `git fetch` загружает изменения из удаленного репозитория, но не объединяет их с локальной веткой, в то время как `git pull` загружает изменения и сразу же объединяет их с локальной веткой.
11. Удалить локальную ветку можно с помощью команды `git branch -d <имя_ветки>`, а удаленную ветку - с помощью команды `git push <имя_удаленного_репозитория> --delete <имя_удаленной_ветки>`.

12. Git-flow это модель ветвления в Git, которая предлагает определенный набор стандартов и практик, которые помогают организовать процесс совместной разработки. Она состоит из двух основных веток: master и develop, а также из дополнительных веток типа feature, release и hotfix.

13. Основные типы веток в git-flow:

- master - главная ветка, в которой находится последняя стабильная версия проекта;
- develop - ветка, в которой происходит разработка новых фич;
- feature - ветка, создаваемая для разработки новой функциональности;
- release - ветка, создаваемая для подготовки выпуска новой версии проекта;
- hotfix - ветка, создаваемая для исправления критических ошибок в готовой версии проекта.

14. Инструменты для работы с ветками Git, предоставляемые средствами с GUI могут включать в себя: возможность создания, удаления и переключения между ветками с помощью графического интерфейса, просмотр списка всех доступных веток, отслеживание изменений и синхронизация с удаленным репозиторию. Некоторые популярные GUI-клиенты для работы с Git включают в себя GitHub Desktop, GitKraken, Sourcetree и другие.