

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ  
РОССИЙСКОЙ  
ФЕДЕРАЦИИ**

**Федеральное государственное автономное  
образовательное учреждение высшего образования  
«СЕВЕРОКАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»**

**Кафедра инфокоммуникаций**

**Институт цифрового развития**

**ОТЧЁТ**

**по лабораторной работе №1.3**

Дисциплина: «Основы кроссплатформенного программирования»

Тема: «Основы языка Python»

Выполнил: студент 1 курса

группы ИВТ-б-о-21-1

Назаров Никита Юрьевич

Ставрополь 2022

## Выполнение работы.

```
Командная строка
Microsoft Windows [Version 10.0.19044.1706]
(c) Корпорация Майкрософт (Microsoft Corporation). Все права защищены.

C:\Users\Y>cd /d H:\cross\git\nazarov

H:\cross\git\nazarov>git clone https://github.com/NikitaNazarov179/lab1-3.git
Cloning into 'lab1-3'...
remote: Enumerating objects: 4, done.
remote: Counting objects: 100% (4/4), done.
remote: Compressing objects: 100% (3/3), done.
remote: Total 4 (delta 0), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (4/4), done.

H:\cross\git\nazarov>git clone https://github.com/NikitaNazarov179/lab1-3.git_
```

Рисунок 1 – клонирование репозитория




 1.txt	30.05.2022 14:17	Текстовый докум...	0 КБ
 2.txt	30.05.2022 14:20	Текстовый докум...	0 КБ
 3.txt	30.05.2022 14:20	Текстовый докум...	0 КБ

Рисунок 2 – создание текстовых файлов

```
Командная строка
H:\cross\git\nazarov\lab1-3>git add .

H:\cross\git\nazarov\lab1-3>git commit -m "add 1.txt file"
[main a4d48bd] add 1.txt file
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 1.txt

H:\cross\git\nazarov\lab1-3>git push
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 12 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 328 bytes | 328.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/NikitaNazarov179/lab1-3.git
8197d54..a4d48bd main -> main

H:\cross\git\nazarov\lab1-3>git add .

H:\cross\git\nazarov\lab1-3>git commit -m "add 2.txt and 3.txt"
[main 50a2707] add 2.txt and 3.txt
2 files changed, 0 insertions(+), 0 deletions(-)
create mode 100644 2.txt
create mode 100644 3.txt

H:\cross\git\nazarov\lab1-3>git push
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Delta compression using up to 12 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (2/2), 330 bytes | 330.00 KiB/s, done.
Total 2 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/NikitaNazarov179/lab1-3.git
a4d48bd..50a2707 main -> main

H:\cross\git\nazarov\lab1-3>
```

Рисунок 3 – коммит и пуш файлов

	NikitaNazarov179 add 1.txt and 2.txt	9f9e5d4 14 seconds ago	🕒 4 commits
	1.txt	add 1.txt and 2.txt	14 seconds ago
	2.txt	add 1.txt and 2.txt	14 seconds ago
	3.txt	add 1.txt and 2.txt	14 seconds ago
	LICENSE	Initial commit	32 minutes ago
	README.md	Initial commit	32 minutes ago

Рисунок 4 – результат

```
H:\cross\git\nazarov\lab1-3>git branch my_first_branch
H:\cross\git\nazarov\lab1-3>git checkout my_first_branch
Switched to branch 'my_first_branch'
```

Рисунок 5 – создание новой ветки и переход на нее






	1.txt	30.05.2022 14:46	Текстовый докум...	1 КБ
	2.txt	30.05.2022 14:46	Текстовый докум...	1 КБ
	3.txt	30.05.2022 14:46	Текстовый докум...	1 КБ
	in_branch.txt	30.05.2022 14:50	Текстовый докум...	0 КБ
	LICENSE	30.05.2022 14:15	Файл	2 КБ

Рисунок 6 – добавление файла в локальный репозиторий

```
H:\cross\git\nazarov\lab1-3>git add .
H:\cross\git\nazarov\lab1-3>git commit -m "add in_branch.txt"
[my_first_branch db32e79] add in_branch.txt
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 in_branch.txt
```

Рисунок 7 – коммит изменений

```
H:\cross\git\nazarov\lab1-3>git checkout main
Switched to branch 'main'
Your branch is up to date with 'origin/main'.
```

Рисунок 8 – переход на ветку main

```
H:\cross\git\nazarov\lab1-3>git checkout -b new_branch
Switched to a new branch 'new_branch'
```

Рисунок 9 - создание и переход на новую ветку

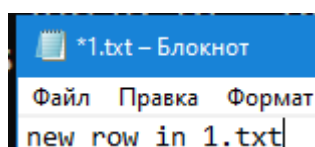


Рисунок 10 – изменения в файле 1.txt

```
H:\cross\git\nazarov\lab1-3>git commit -am "fix 1.txt"
[new_branch 318846b] fix 1.txt
1 file changed, 1 insertion(+), 1 deletion(-)
```

Рисунок 11 – коммит изменений

```
H:\cross\git\nazarov\lab1-3>git checkout main
Switched to branch 'main'
Your branch is up to date with 'origin/main'.

H:\cross\git\nazarov\lab1-3>git merge my_first_branch
Updating 9f9e5d4..db32e79
Fast-forward
 in_branch.txt | 0
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 in_branch.txt

H:\cross\git\nazarov\lab1-3>git branch
* main
  my_first_branch
  new_branch

H:\cross\git\nazarov\lab1-3>git merge new_branch
Merge made by the 'ort' strategy.
1.txt | 2 +-
1 file changed, 1 insertion(+), 1 deletion(-)

H:\cross\git\nazarov\lab1-3>git branch -d my_first_branch
Deleted branch my_first_branch (was db32e79).

H:\cross\git\nazarov\lab1-3>git branch -d new_branch
Deleted branch new_branch (was 318846b).
```

Рисунок 12 – слияние и удаление веток

```
H:\cross\git\nazarov\lab1-3>git branch branch_1

H:\cross\git\nazarov\lab1-3>git branch branch_2

H:\cross\git\nazarov\lab1-3>git branch
branch_1
branch_2
* main
```

Рисунок 13 – создание 2 новых веток

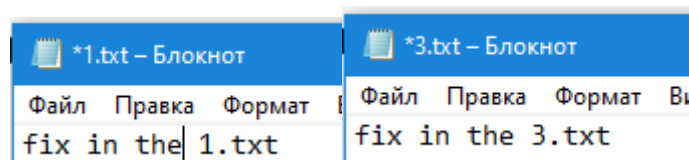


Рисунок 14 – изменения в текстовых файлах

```
H:\cross\git\nazarov\lab1-3>git commit -am "fix 1.txt & 3.txt in branch_1"
[branch_1 470d170] fix 1.txt & 3.txt in branch_1
2 files changed, 2 insertions(+), 2 deletions(-)
```

Рисунок 15 – коммит изменений

```
H:\cross\git\nazarov\lab1-3>git checkout branch_2
Switched to branch 'branch_2'
```

Рисунок 16 – переход на другую ветку

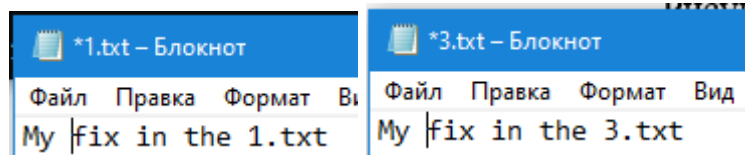


Рисунок 17 – изменения в файлах 1 и 3

```
H:\cross\git\nazarov\lab1-3>git commit -am "fix 1.txt & 3.txt in branch_2"
[branch_2 24858f8] fix 1.txt & 3.txt in branch_2
2 files changed, 2 insertions(+), 2 deletions(-)
```

Рисунок 18 – коммит изменений

```
H:\cross\git\nazarov\lab1-3>git checkout branch_1
Switched to branch 'branch_1'

H:\cross\git\nazarov\lab1-3>git merge branch_2
Auto-merging 1.txt
CONFLICT (content): Merge conflict in 1.txt
Auto-merging 3.txt
CONFLICT (content): Merge conflict in 3.txt
Automatic merge failed; fix conflicts and then commit the result.
```

Рисунок 19 – попытка слияния 2 веток

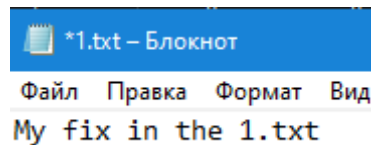


Рисунок 20 – ручное исправление 1 файла

```
H:\cross\git\nazarov\lab1-3>git add 1.txt

H:\cross\git\nazarov\lab1-3>git status
On branch branch_1
You have unmerged paths.
  (fix conflicts and run "git commit")
  (use "git merge --abort" to abort the merge)

Changes to be committed:
  modified:   1.txt

Unmerged paths:
  (use "git add <file>..." to mark resolution)
  both modified: 3.txt
```

Рисунок 21 – добавление файла в индекс

```

H:\cross\git\nazarov\lab1-3>git mergetool

This message is displayed because 'merge.tool' is not configured.
See 'git mergetool --tool-help' or 'git help config' for more details.
'git mergetool' will now attempt to use one of the following tools:
tortoisemerge emerge vimdiff nvimdiff
Merging:
3.txt

Normal merge conflict for '3.txt':
  {local}: modified file
  {remote}: modified file
Hit return to start merge resolution tool (vimdiff): _

```

Рисунок 22 – команда git mergetool

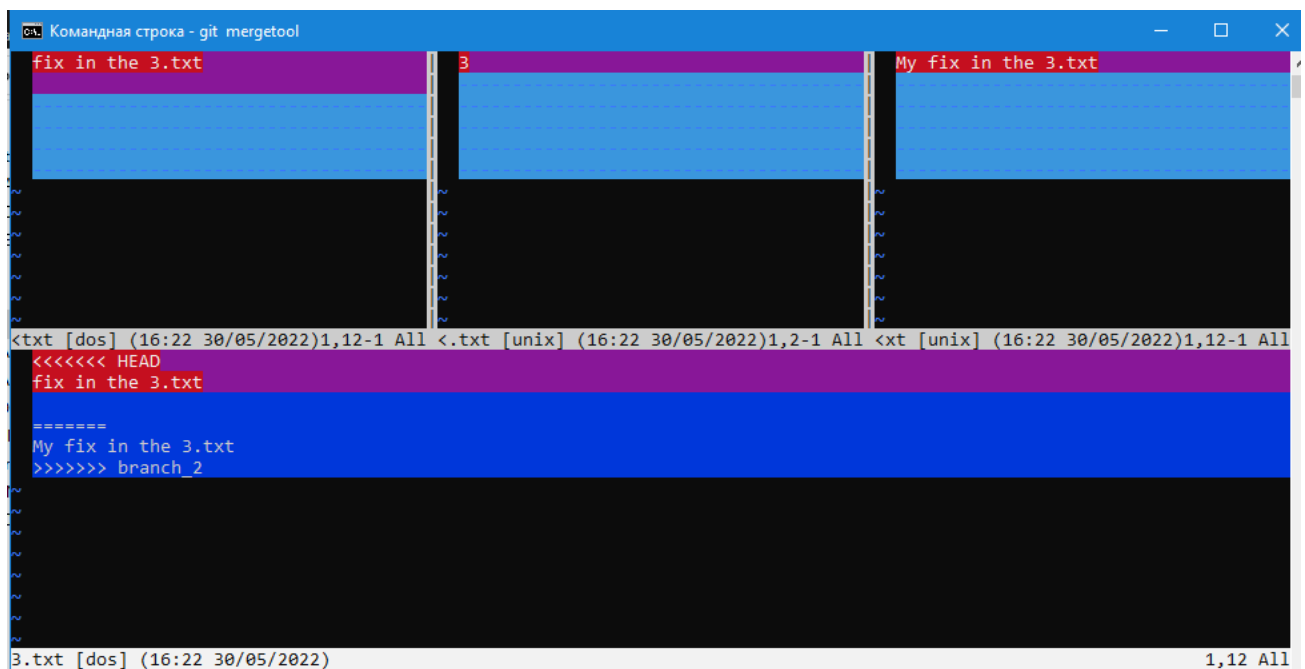


Рисунок 23 - решение конфликта при помощи mergetool

```

H:\cross\git\nazarov\lab1-3>git push --set-upstream origin branch_1
Enumerating objects: 19, done.
Counting objects: 100% (19/19), done.
Delta compression using up to 12 threads
Compressing objects: 100% (10/10), done.
Writing objects: 100% (16/16), 1.60 KiB | 820.00 KiB/s, done.
Total 16 (delta 4), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (4/4), completed with 1 local object.
remote:
remote: Create a pull request for 'branch_1' on GitHub by visiting:
remote:   https://github.com/NikitaNazarov179/lab1-3/pull/new/branch_1
remote:
To https://github.com/NikitaNazarov179/lab1-3.git
 * [new branch]      branch_1 -> branch_1
branch 'branch_1' set up to track 'origin/branch_1'.

```

Рисунок 24 – пуш ветки на уд.репозиторий

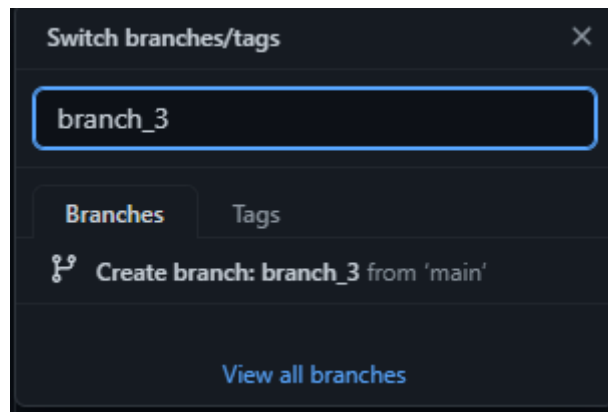


Рисунок 25 – создание ветки branch\_3 на самом сайте github

```
H:\cross\git\nazarov\lab1-3>git fetch --all
Fetching origin
From https://github.com/NikitaNazarov179/lab1-3
* [new branch]      branch_3    -> origin/branch_3
```

Рисунок 26 – загрузка содержимого из удаленного репозитория

```
H:\cross\git\nazarov\lab1-3>git checkout --track origin/branch_3
Switched to a new branch 'branch_3'
branch 'branch_3' set up to track 'origin/branch_3'.
```

Рисунок 27 – создание ветки отслеживания

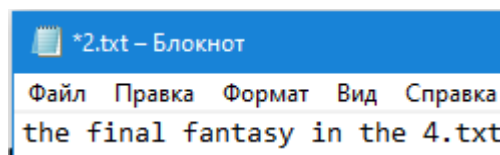


Рисунок 28 – редактирование файла 2.txt

```
H:\cross\git\nazarov\lab1-3>git rebase main
Current branch branch_2 is up to date.

H:\cross\git\nazarov\lab1-3>git checkout main
Switched to branch 'main'
Your branch is ahead of 'origin/main' by 3 commits.
(use "git push" to publish your local commits)
```

Рисунок 29 – перемещение ветки main на ветку branch\_2

```
H:\cross\git\nazarov\lab1-3>git merge branch_2
Updating ef1a2ac..5dab57a
Fast-forward
 1.txt | 2 +-
 2.txt | 2 +-

```

Рисунок 30 – слияние веток

```
H:\cross\git\nazarov\lab1-3>git push
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 12 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 303 bytes | 303.00 KiB/s, done.
Total 3 (delta 1), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To https://github.com/NikitaNazarov179/lab1-3.git
  9f9e5d4..5dab57a  main -> main
```

Рисунок 31 – пуш веток на уд. репозиторий

## **Ответы на контрольные вопросы:**

### **1) Что такое ветка?**

Это последовательность коммитов, отклоняющаяся от основной линии разработки.

### **2) Что такое HEAD?**

Это указатель, задача которого ссылаться на определённый коммит в репозитории, если точнее: на коммит, который станет родителем для следующего коммита.

### **3) Способы создания веток?**

Новую ветку можно создать командой `git branch <название_ветки>` или на удаленном репозитории `git hub`. Как узнать текущую ветку?

При помощи команды `git branch`.

### **4) Как переключаться между ветками?**

При помощи команды `git checkout <название_ветки>`.

### **5) Что такое удаленная ветка?**

Это ветка, находящаяся на удаленном репозитории. Или ссылка на состояние ветки на удаленном репозитории.

### **6) Что такое ветка отслеживания?**

Это ветка в локальном репозитории, которая напрямую связана с удаленной веткой на удаленном репозитории.

### **7) Как создать ветку отслеживания?**

Командой `git checkout --track origin/<название_ветки>`.

### **8) Как отправить изменения из локальной ветки в удаленную ветку?**

Командой `git push origin <название_ветки>`.



**9) В чем отличие команд `git fetch` и `git pull`?**

`Git pull` – это сочетание команд `git fetch` (получение изменений с удаленного репозитория) и `git merge` (объединение веток).

**10) Как удалить локальную и удаленную ветки?**

Используя команду `git branch -d <название_ветки>`. Для удаление удаленной ветки существует команда `git push origin -d <название_ветки>`.