

**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 ветвления Git»

Выполнил: студент 1 курса

группы ИВТ-б-о-21-1

Стригалов Дмитрий

Ставрополь 2022

1. Создал репозиторий с использованием лицензии MIT. Затем клонировал его на ПК:

```
c:\temp>git clone https://github.com/eMoushen/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.
```

Рисунок 1.2 Клонирование репозитория

2. Создал три файла, проиндексировал 1-й и сделал коммит “add 1.txt file”. Затем проиндексировал 2-й и 3-й и перезаписал коммит с новым комментарием “add 2.txt and 3.txt”:

1.txt	11.05.2022 12:00
2.txt	11.05.2022 11:59
3.txt	11.05.2022 12:00
LICENSE	11.05.2022 11:56
README.md	11.05.2022 11:56

Рисунок 2.1 Созданные файлы в репозитории

```
c:\temp\Lab1.3>git add 1.txt
c:\temp\Lab1.3>git commit -m "add 1.txt file"
[main f31dfcb] add 1.txt file
```

Рисунок 2.2 1-й коммит

```
c:\temp\Lab1.3>git add .
c:\temp\Lab1.3>git commit --amend -m "add 2.txt and 3.txt"
[main 5bc1e8e] add 2.txt and 3.txt
Date: Wed May 11 12:01:37 2022 +0300
Committer: student-09-525 <student-09-525@ncfu.net>
Your name and email address were configured automatically based
on your username and hostname. Please check that they are accurate.
You can suppress this message by setting them explicitly. Run the
following command and follow the instructions in your editor to edit
your configuration file:

    git config --global --edit

After doing this, you may fix the identity used for this commit with:

    git commit --amend --reset-author

3 files changed, 0 insertions(+), 0 deletions(-)
create mode 100644 1.txt
create mode 100644 2.txt
create mode 100644 3.txt
```

Рисунок 2.3 Изменение 1-го коммита

3. Создал новую ветку `my_first_branch`, перешел на нее и создал в ней новый файл `in_branch.txt`, закоммитил изменения. После этого вернулся на ветку `master`, создал и сразу перешел на ветку `new_branch`, сделал изменения в файле `1.txt` и закоммитил изменения.

```
c:\temp\Lab1.3>git checkout my_first_branch
Switched to branch 'my_first_branch'

c:\temp\Lab1.3>git branch
main
* my_first_branch
```

Рисунок 3.1 Создание и переход на новую ветку

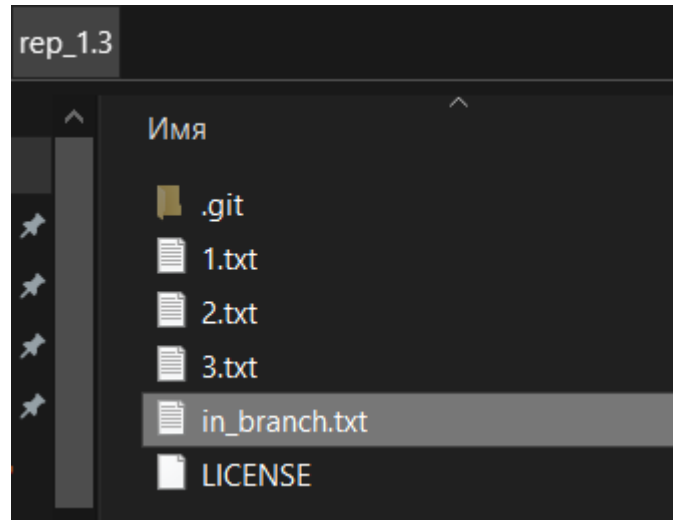


Рисунок 3.2 Добавление файла в репозиторий

```
c:\temp\Lab1.3>git add .
c:\temp\Lab1.3>git commit -m "add in_branch.txt"
[my_first_branch 29585631 add in_branch.txt
Committer: student-09-525 <student-09-525@acfu.ne
```

Рисунок 3.3 Коммит изменений

```
c:\temp\Lab1.3>git checkout -b new_branch
Switched to a new branch 'new_branch'

c:\temp\Lab1.3>git branch
main
my_first_branch
* new_branch
```

Рисунок 3.4 Переход на ветку main и создание + переход на новую ветку

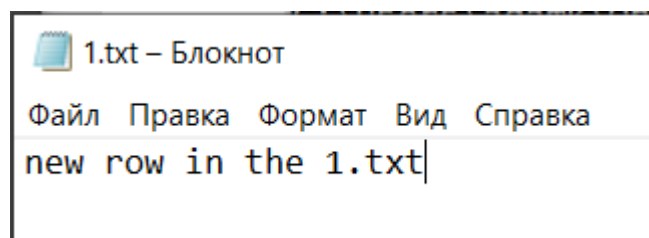


Рисунок 3.5 Изменения в файле 1.txt

```
c:\temp\Lab1.3>git commit -am "fix 1.txt"
[master 90582a5] fix 1.txt
Committer: student-09-525 <student-09-525@ncfu.net>
Your name and email address were configured automatically based
on your username and hostname. Please check that they are accurate.
You can suppress this message by setting them explicitly. Run the
following command and follow the instructions in your editor to edit
your configuration file:

    git config --global --edit

After doing this, you may fix the identity used for this commit with:

    git commit --amend --reset-author

1 file changed, 1 insertion(+)
```

Рисунок 3.6 Коммит изменений

4. Перешел на ветку main и слил ветки main и my_first_branch, после этого слил ветки main и new_branch. Удалил все ветки кроме main:

```
c:\temp\Lab1.3>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:\temp\Lab1.3>git merge my_first_branch
Updating 5bc1e8e..2958563
Fast-forward
 in_branch.txt | 0
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 in_branch.txt

c:\temp\Lab1.3>git branch
* main
  my_first_branch
  new_branch

c:\temp\Lab1.3>git merge new_branch
Merge made by the 'recursive' strategy.
 1.txt | 1 +
 1 file changed, 1 insertion(+)
```

```
c:\temp\Lab1.3>git branch -d my_first_branch
Deleted branch my_first_branch (was 2958563).

c:\temp\Lab1.3>git branch -d new_branch
Deleted branch new_branch (was 90582a5).
```

Рисунок 4.1 Слияние и удаления веток

5. Создал ветки branch_1 и branch_2. Перешел на ветку branch_1 и изменил файлы 1.txt, 3.txt, после чего закоммитил изменения. Перешел на ветку branch_2, изменил файлы 1.txt, 3.txt, после чего закоммитил изменения. После этого слил изменения ветки branch_2 в ветку branch_1:

```
c:\temp\Lab1.3>git branch branch_1
c:\temp\Lab1.3>git branch branch_2
c:\temp\Lab1.3>git branch
branch_1
branch_2
* main
```

Рисунок 5.1 Создание веток branch_1 и branch_2

```
c:\temp\Lab1.3>git checkout branch_1
Switched to branch 'branch_1'
```

Рисунок 5.2 Переход на ветку branch_1



 1.txt – Блокнот	 3.txt – Блокнот
Файл Правка Формат В	Файл Правка Формат
fix in the 1.txt	fix in the 3.txt

Рисунок 5.3 Изменения в файлах 1.txt и 3.txt

```
2 files changed, 2 insertions(+), 1 deletion(-)
```

Рисунок 5.4 Коммит изменений

```
c:\temp\Lab1.3>git checkout branch_2
Switched to branch 'branch_2'
```

Рисунок 5.5 Переход на ветку branch_2



 1.txt – Блокнот	 3.txt – Блокнот
Файл Правка Формат Вид	Файл Правка Формат Вид
My fix in the 1.txt	My fix in the 3.txt

Рисунок 5.6 Изменения в файлах 1.txt и 3.txt

```
git commit -m "Merge branch_2 into branch_1"
2 files changed, 2 insertions(+), 1 deletion(-)
```

Рисунок 5.7 Коммит изменений

6. Выполнил слияние новых веток и решил конфликт файла 1.txt в ручном режиме, а 3.txt в автоматическом инструментом vimdiff, после этого отправил ветку branch_1 в GitHub:

```
c:\temp\Lab1.3>git checkout branch_1
Switched to branch 'branch_1'

c:\temp\Lab1.3>git merge branch_2
Auto-merging 3.txt
CONFLICT (content): Merge conflict in 3.txt
Auto-merging 1.txt
CONFLICT (content): Merge conflict in 1.txt
Automatic merge failed; fix conflicts and then commit the result.
```

Рисунок 6.1 Слияние веток


 1.txt – Блокнот
Файл Правка Формат В
My fix in the 1.txt

Рисунок 6.2 Решение конфликта в ручном режиме

```
c:\temp\Lab1.3>git add 1.txt
c:\temp\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 6.3 Добавление 1.txt в индекс, так как конфликт решен

```
c:\temp\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
No files need merging
```

Рисунок 6.4 Команда mergetool

[illegible]

Рисунок 6.5 Команда :diffg LO

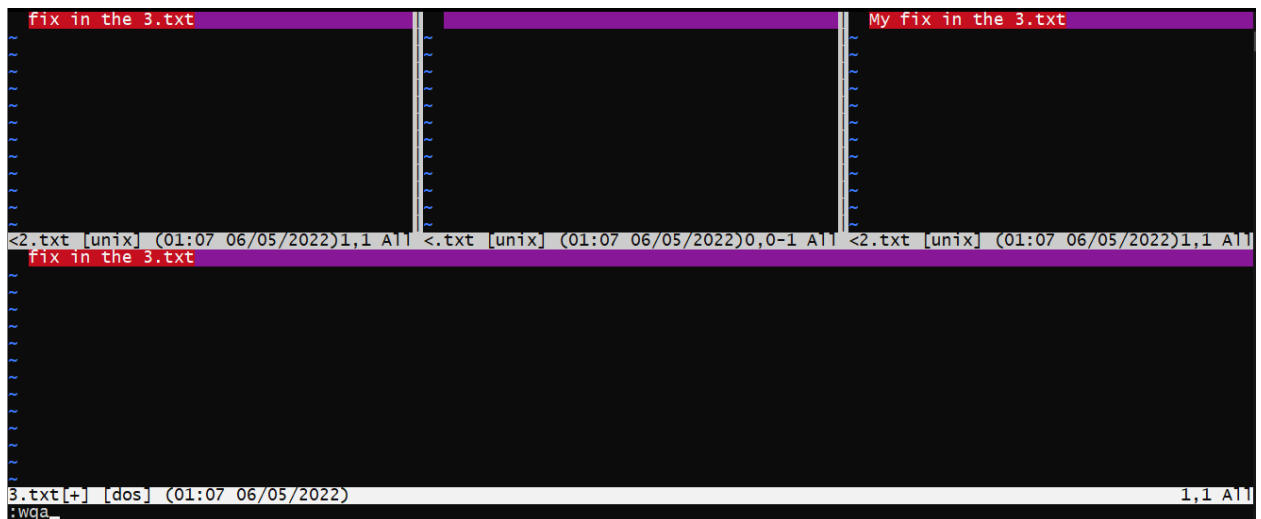


Рисунок 6.6 Команда :wga

```
c:\temp\Lab1.3>git push --set-upstream origin branch_1
info: please complete authentication in your browser...
Enumerating objects: 15, done.
Counting objects: 100% (15/15), done.
Delta compression using up to 2 threads
Compressing objects: 100% (10/10), done.
Writing objects: 100% (14/14), 1.17 KiB | 400.00 KiB/s, done.
Total 14 (delta 5), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (5/5), done.
remote:
remote: Create a pull request for 'branch_1' on GitHub by visiting:
remote:   https://github.com/eMoushen/Lab1.3/pull/new/branch_1
remote:
To https://github.com/eMoushen/Lab1.3.git
 * [new branch]      branch_1 -> branch_1
Branch 'branch_1' set up to track remote branch 'branch_1' from 'origin'.
```

Рисунок 6.7 git push

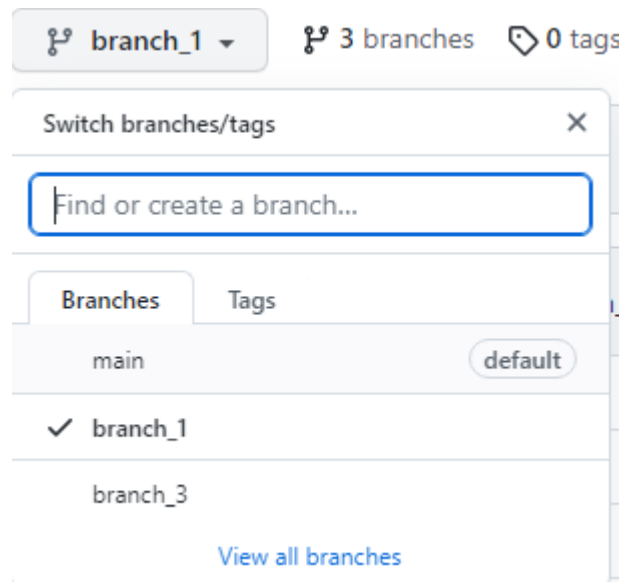


Рисунок 6.8 Изменения на уд. репозитории

7. Создал средствами GitHub удаленную ветку branch_3, создал в лок. репозитории ветку отслеживания удаленной ветки branch_3. Перешел на ветку branch_3 и добавил в файл 2.txt строку “the final fantasy in the 4.txt file”:

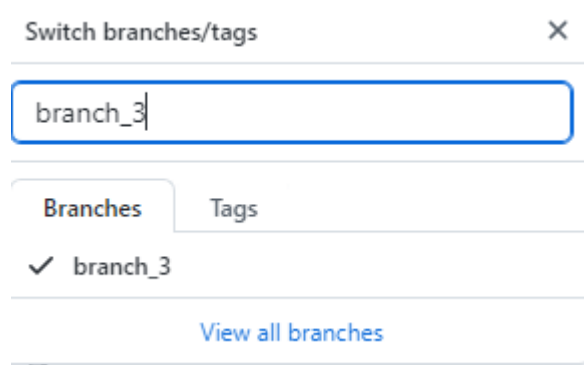


Рисунок 7.1 Создание уд. ветки branch_3

```
c:\temp\Lab1.3>git fetch --all
Fetching origin
From https://github.com/eMoushen/Lab1.3
* [new branch]      branch_3    -> origin/branch_3
```

Рисунок 7.2 Загрузка содержимого из уд. репозитория перед отслеживанием

```
c:\temp\Lab1.3>git checkout --track origin/branch_3
Switched to a new branch 'branch_3'
M       1.txt
M       3.txt
Branch 'branch_3' set up to track remote branch 'branch_3' from 'origin'.
```

Рисунок 7.3 Создание ветки отслеживания branch_3

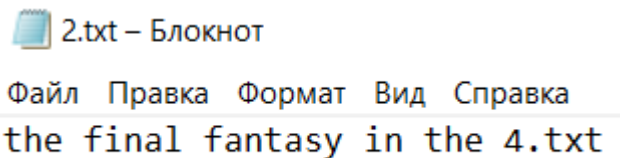


Рисунок 7.4 Добавление в 2.txt строку “the final fantasy in the 4.txt file” на ветке branch_3

8. Выполнил перемещение ветки main на ветку branch_2 и отправил изменения веток main и branch_2 на GitHub:

```
c:\temp\Lab1.3>git merge branch_2
Updating 588d89d..6cc24e9
Fast-forward
 1.txt | 2 +
 3.txt | 1 +
 2 files changed, 2 insertions(+), 1 deletion(-)
```

Рисунок 8.1 Результат выполнения команд

```
c:\temp\Lab1.3>git push
Enumerating objects: 7, done.
Counting objects: 100% (7/7), done.
Delta compression using up to 2 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (4/4), 429 bytes | 429.00 KiB/s, done.
Total 4 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/eMoushen/Lab1.3.git
 6530a7e..6cc24e9  main -> main
```


Рисунок 8.2 Пуш веток

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`.

4) Как узнать текущую ветку?

При помощи команды `git branch`.

5) Как переключаться между ветками?

При помощи команды `git checkout <название_ветки>`.

6) Что такое удаленная ветка?

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

7) Что такое ветка отслеживания?

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

8) Как создать ветку отслеживания?

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

9) Как отправить изменения из локальной ветки в удаленную ветку?

Командой `git push origin <название_ветки>`.

10) 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).

11) 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>`.