

**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

Кочкаров Умар Ахматович

Ставрополь 2022

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

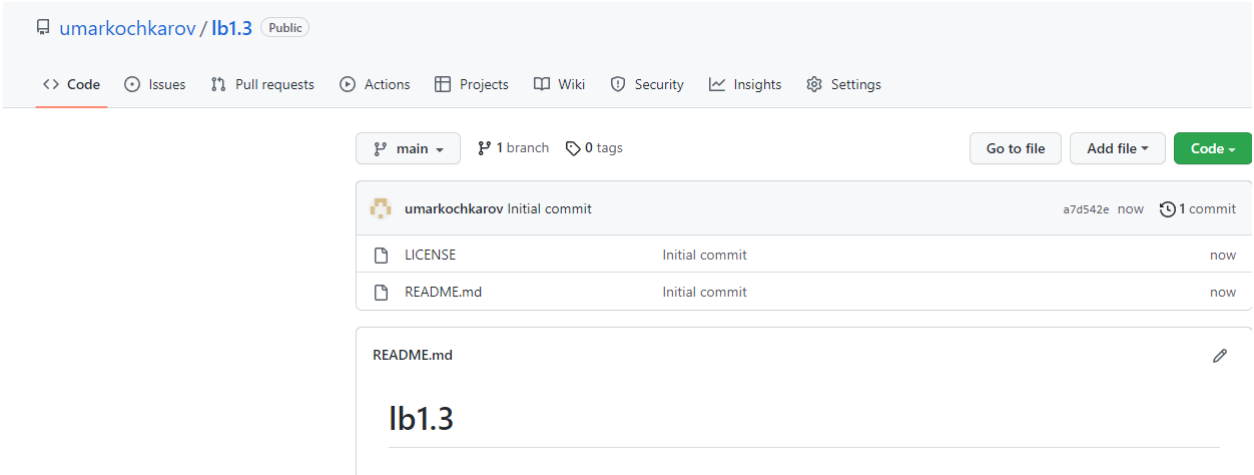


Рисунок 1.1. Репозиторий на github

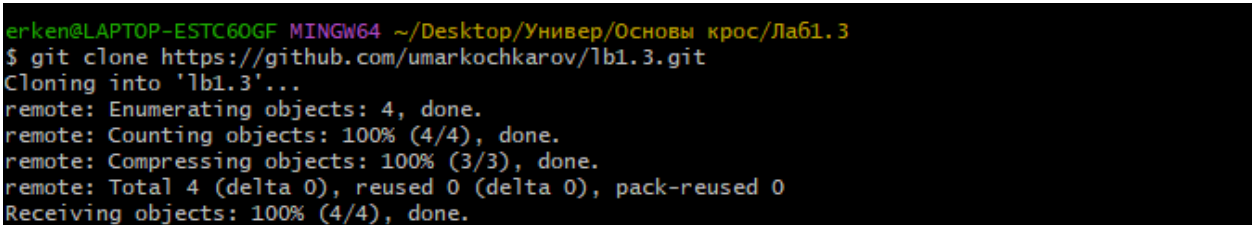


Рисунок 1.2. Клонирование репозитория

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”:

Имя	Дата изменения	Тип	Размер
.git	11.09.2022 19:20	Папка с файлами	
1	11.09.2022 19:20	Текстовый докум...	0 КБ
2	11.09.2022 19:20	Текстовый докум...	0 КБ
3	11.09.2022 19:20	Текстовый докум...	0 КБ
LICENSE	11.09.2022 19:20	Файл	2 КБ
README	11.09.2022 19:20	Исходный файл ...	1 КБ

Рисунок 2.1. Созданные файлы

```

erken@LAPTOP-ESTC60GF MINGW64 ~/Desktop/Универ/Основы крос/Лаб1.3/1b1.3 (main)
$ git add 1.txt

erken@LAPTOP-ESTC60GF MINGW64 ~/Desktop/Универ/Основы крос/Лаб1.3/1b1.3 (main)
$ git status
On branch main
Your branch is up to date with 'origin/main'.

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:   1.txt

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

erken@LAPTOP-ESTC60GF MINGW64 ~/Desktop/Универ/Основы крос/Лаб1.3/1b1.3 (main)
$ git commit -m "add 1.txt file"
[main f466a7c] add 1.txt file
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 1.txt

```

Рисунок 2.2. Первый коммит

```

erken@LAPTOP-ESTC60GF MINGW64 ~/Desktop/Универ/Основы крос/Лаб1.3/1b1.3 (main)
$ git add .

erken@LAPTOP-ESTC60GF MINGW64 ~/Desktop/Универ/Основы крос/Лаб1.3/1b1.3 (main)
$ 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)
    new file:   2.txt
    new file:   3.txt

erken@LAPTOP-ESTC60GF MINGW64 ~/Desktop/Универ/Основы крос/Лаб1.3/1b1.3 (main)
$ git commit --amend -m "add 2.txt and 3.txt"
[main 2459040] add 2.txt and 3.txt
Date: Sun Sep 11 19:22:16 2022 +0300
3 files changed, 0 insertions(+), 0 deletions(-)
create mode 100644 1.txt
create mode 100644 2.txt
create mode 100644 3.txt

```

Рисунок 2.2. 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:

```

erken@LAPTOP-ESTC60GF MINGW64 ~/Desktop/Универ/Основы крос/Лаб1.3/1b1.3 (main)
$ git branch my_first_branch

erken@LAPTOP-ESTC60GF MINGW64 ~/Desktop/Универ/Основы крос/Лаб1.3/1b1.3 (main)
$ git checkout my_first_branch
Switched to branch 'my_first_branch'

erken@LAPTOP-ESTC60GF MINGW64 ~/Desktop/Универ/Основы крос/Лаб1.3/1b1.3 (my_firs
t_branch)
$ git branch
  main
* my_first_branch

```

Рисунок 3.1 Создание и переход на новую ветку

Имя	Дата изменения	Тип	Размер
.git	11.09.2022 19:26	Папка с файлами	
1	11.09.2022 19:20	Текстовый докум...	0 КБ
2	11.09.2022 19:20	Текстовый докум...	0 КБ
3	11.09.2022 19:20	Текстовый докум...	0 КБ
in_branch	11.09.2022 19:27	Текстовый докум...	0 КБ
LICENSE	11.09.2022 19:20	Файл	2 КБ
README	11.09.2022 19:20	Исходный файл ...	1 КБ

Рисунок 3.2 Добавление файла в репозиторий

```
erken@LAPTOP-ESTC60GF MINGW64 ~/Desktop/Универ/Основы крос/Лаб1.3/lb1.3 (my_firs
t_branch)
$ git add .

erken@LAPTOP-ESTC60GF MINGW64 ~/Desktop/Универ/Основы крос/Лаб1.3/lb1.3 (my_firs
t_branch)
$ git commit -m "add in_branch.txt"
[my_first_branch ea2ac7c] add in_branch.txt
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 in_branch.txt
```

Рисунок 3.3 Коммит изменений

```
erken@LAPTOP-ESTC60GF MINGW64 ~/Desktop/Универ/Основы крос/Лаб1.3/lb1.3 (my_firs
t_branch)
$ 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)

erken@LAPTOP-ESTC60GF MINGW64 ~/Desktop/Универ/Основы крос/Лаб1.3/lb1.3 (main)
$ git checkout -b new_branch
Switched to a new branch 'new_branch'

erken@LAPTOP-ESTC60GF MINGW64 ~/Desktop/Универ/Основы крос/Лаб1.3/lb1.3 (new_bra
nch)
$ git branch
  main
  my_first_branch
* new_branch
```

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

```
erken@LAPTOP-ESTC60GF MINGW64 ~/Desktop/Универ/Основы крос/Лаб1.3/lb1.3 (new_bra
nch)
$ git commit -am "fix 1.txt"
[new_branch e8e1520] fix 1.txt
1 file changed, 1 insertion(+)
```

Рисунок 3.5 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:

```
erken@LAPTOP-ESTC60GF MINGW64 ~/Desktop/Универ/Основы крос/Лаб1.3/1b1.3 (new_branch)
$ 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)

erken@LAPTOP-ESTC60GF MINGW64 ~/Desktop/Универ/Основы крос/Лаб1.3/1b1.3 (main)
$ git merge my_first_branch
Updating 2459040..ea2ac7c
Fast-forward
 in_branch.txt | 0
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 in_branch.txt

erken@LAPTOP-ESTC60GF MINGW64 ~/Desktop/Универ/Основы крос/Лаб1.3/1b1.3 (main)
$ git branch
* main
  my_first_branch
  new_branch

erken@LAPTOP-ESTC60GF MINGW64 ~/Desktop/Универ/Основы крос/Лаб1.3/1b1.3 (main)
$ git merge new_branch
Merge made by the 'ort' strategy.
 1.txt | 1 +
 1 file changed, 1 insertion(+)

erken@LAPTOP-ESTC60GF MINGW64 ~/Desktop/Универ/Основы крос/Лаб1.3/1b1.3 (main)
$ git branch -d my_first_branch
Deleted branch my_first_branch (was ea2ac7c).

erken@LAPTOP-ESTC60GF MINGW64 ~/Desktop/Универ/Основы крос/Лаб1.3/1b1.3 (main)
$ git branch -d new_branch
Deleted branch new_branch (was e8e1520).
```

Рисунок 4.1 Слияние и удаление 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:

```
erken@LAPTOP-ESTC60GF MINGW64 ~/Desktop/Универ/Основы крос/Лаб1.3/1b1.3 (main)
$ git branch branch_1

erken@LAPTOP-ESTC60GF MINGW64 ~/Desktop/Универ/Основы крос/Лаб1.3/1b1.3 (main)
$ git branch branch_2

erken@LAPTOP-ESTC60GF MINGW64 ~/Desktop/Универ/Основы крос/Лаб1.3/1b1.3 (main)
$ git branch
  branch_1
  branch_2
* main
```

Рисунок 5.1 Создание веток branch\_1 и branch\_2

```
erken@LAPTOP-ESTC60GF MINGW64 ~/Desktop/Универ/Основы крос/Лаб1.3/1b1.3 (main)
$ git checkout branch_1
Switched to branch 'branch_1'
```

Рисунок 5.2 Переход на ветку branch\_1



Рисунок 5.3 Изменения в 1.txt и 3.txt

```
erken@LAPTOP-ESTC60GF MINGW64 ~/Desktop/Универ/Основы крос/Лаб1.3/1b1.3 (branch_1)
$ git add .

erken@LAPTOP-ESTC60GF MINGW64 ~/Desktop/Универ/Основы крос/Лаб1.3/1b1.3 (branch_1)
$ git commit -m "fix 1.txt and 3.txt"
[branch_1 1656bd6] fix 1.txt and 3.txt
2 files changed, 2 insertions(+), 1 deletion(-)
```

Рисунок 5.4 Коммит изменений

```
erken@LAPTOP-ESTC60GF MINGW64 ~/Desktop/Универ/Основы крос/Лаб1.3/1b1.3 (branch_1)
$ git checkout branch_2
Switched to branch 'branch_2'
```

Рисунок 5.5 Переход на ветку branch\_2

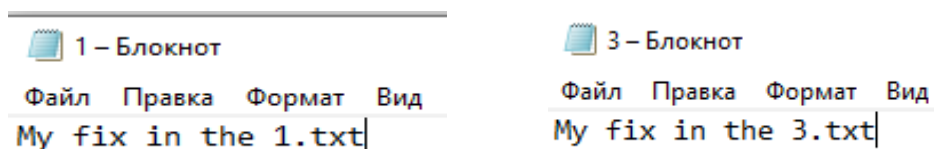


Рисунок 5.6 Изменения в 1.txt и 3.txt

```
erken@LAPTOP-ESTC60GF MINGW64 ~/Desktop/Универ/Основы крос/Лаб1.3/1b1.3 (branch_2)
$ git add .

erken@LAPTOP-ESTC60GF MINGW64 ~/Desktop/Универ/Основы крос/Лаб1.3/1b1.3 (branch_2)
$ git commit -m "fix 1.txt and 3.txt in branch_2"
[branch_2 56cbbf2] fix 1.txt and 3.txt in branch_2
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:

```
erken@LAPTOP-ESTC60GF MINGW64 ~/Desktop/Универ/Основы крос/Лаб1.3/1b1.3 (branch_2)
$ git checkout branch_1
Switched to branch 'branch_1'

erken@LAPTOP-ESTC60GF MINGW64 ~/Desktop/Универ/Основы крос/Лаб1.3/1b1.3 (branch_1)
$ 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 6.1 Слияние веток

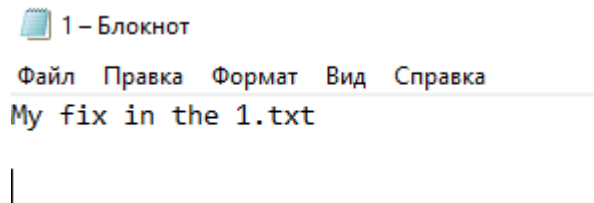


Рисунок 6.2 Решение конфликта в ручном режиме

```
erken@LAPTOP-ESTC60GF MINGW64 ~/Desktop/Универ/Основы крос/Лаб1.3/lb1.3 (branch_1|MERGING)
$ git add 1.txt

erken@LAPTOP-ESTC60GF MINGW64 ~/Desktop/Универ/Основы крос/Лаб1.3/lb1.3 (branch_1|MERGING)
$ 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



Рисунок 6.4 Правка в Meld

```

erken@LAPTOP-ESTC60GF MINGW64 ~/Desktop/Универ/Основы крос/Лаб1.3/1b1.3 (branch_1|MERGING)
$ git push --set-upstream origin branch_1
Enumerating objects: 15, done.
Counting objects: 100% (15/15), done.
Delta compression using up to 8 threads
Compressing objects: 100% (10/10), done.
Writing objects: 100% (14/14), 1.18 KiB | 301.00 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/umarkochkarov/1b1.3/pull/new/branch_1
remote:
To https://github.com/umarkochkarov/1b1.3.git
 * [new branch]      branch_1 -> branch_1
branch 'branch_1' set up to track 'origin/branch_1'.

```

Рисунок 6.5 Пуш ветки на уд.репозиторий

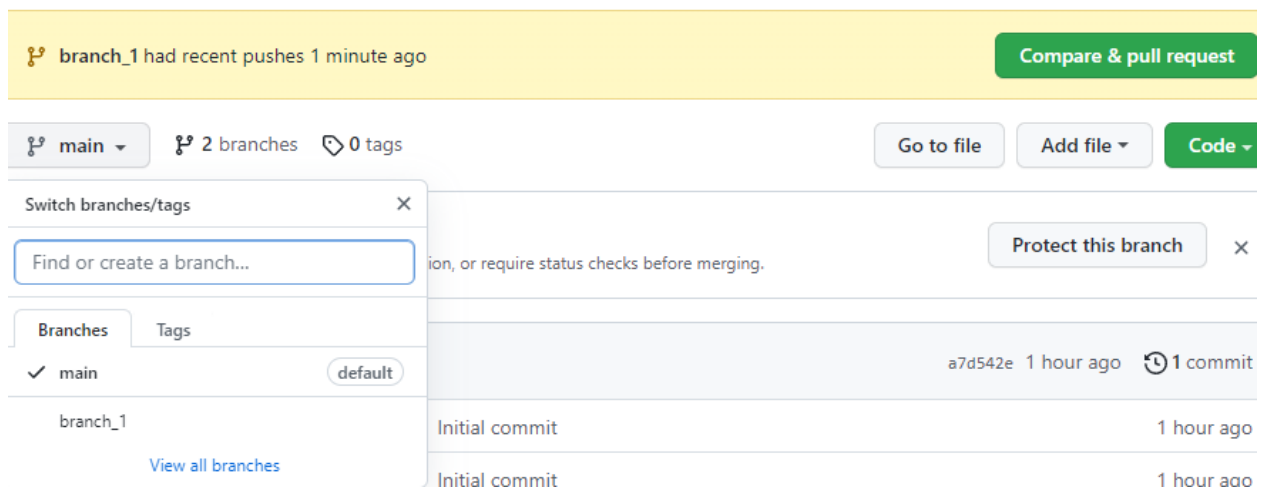


Рисунок 6.6 Изменения на уд. Репозитории

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”:

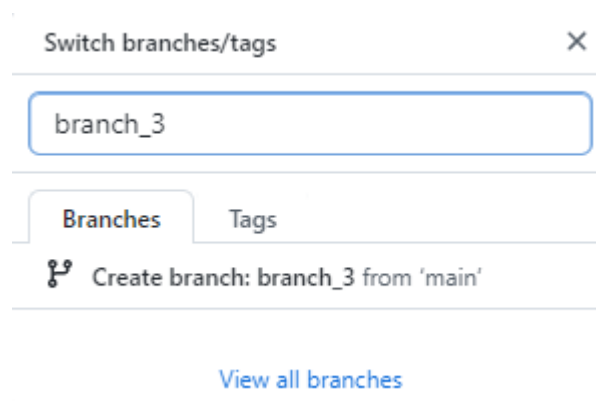


Рисунок 7.1 Создание уд. ветки branch\_3 на github

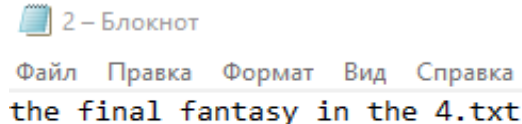


```
erken@LAPTOP-ESTC60GF MINGW64 ~/Desktop/Универ/Основы крос/Лаб1.3/lb1.3 (branch_1|MERGING)
$ git fetch --all
Fetching origin
From https://github.com/umarkochkarov/lb1.3
* [new branch]      branch_3 -> origin/branch_3
```

Рисунок 7.2 Загрузка содержимого из уд. репозитория перед отслеживанием

```
erken@LAPTOP-ESTC60GF MINGW64 ~/Desktop/Универ/Основы крос/Лаб1.3/lb1.3 (branch_1)
$ git checkout --track origin/branch_3
Switched to a new branch 'branch_3'
branch 'branch_3' set up to track 'origin/branch_3'.
```

Рисунок 7.3 Создание ветки отслеживания branch\_3



2 – Блокнот

Файл Правка Формат Вид Справка

the final fantasy in the 4.txt

Рисунок 7.4 Добавление в 2.txt строки “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:

```
erken@LAPTOP-ESTC60GF MINGW64 ~/Desktop/Универ/Основы крос/Лаб1.3/lb1.3 (branch_2)
$ git checkout branch_2
Already on 'branch_2'

erken@LAPTOP-ESTC60GF MINGW64 ~/Desktop/Универ/Основы крос/Лаб1.3/lb1.3 (branch_2)
$ git rebase main
Current branch branch_2 is up to date.

erken@LAPTOP-ESTC60GF MINGW64 ~/Desktop/Универ/Основы крос/Лаб1.3/lb1.3 (branch_2)
$ git checkout main
Switched to branch 'main'
Your branch is ahead of 'origin/main' by 4 commits.
(use "git push" to publish your local commits)

erken@LAPTOP-ESTC60GF MINGW64 ~/Desktop/Универ/Основы крос/Лаб1.3/lb1.3 (main)
$ git merge branch_2
Updating ba85c1b..77c8307
Fast-forward
 1.txt | 2 +-
 2.txt | 1 +
 3.txt | 1 +
 3 files changed, 3 insertions(+), 1 deletion(-)
```

Рисунок 8.1 Результат выполнения команд

```

erken@LAPTOP-ESTC60GF MINGW64 ~/Desktop/Универ/Основы крос/Лаб1.3/lb1.3 (main)
$ git push
Enumerating objects: 11, done.
Counting objects: 100% (10/10), done.
Delta compression using up to 8 threads
Compressing objects: 100% (4/4), done.
Writing objects: 100% (7/7), 659 bytes | 219.00 KiB/s, done.
Total 7 (delta 1), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (1/1), done.
To https://github.com/umarkochkarov/lb1.3.git
   a7d542e..77c8307  main -> main

erken@LAPTOP-ESTC60GF MINGW64 ~/Desktop/Универ/Основы крос/Лаб1.3/lb1.3 (main)
$ git checkout main
Already on 'main'
Your branch is up to date with 'origin/main'.

erken@LAPTOP-ESTC60GF MINGW64 ~/Desktop/Универ/Основы крос/Лаб1.3/lb1.3 (main)
$ git push
Everything up-to-date

```

Рисунок 8.2 Пуш веток

Ссылка на репозиторий Git Hub: <https://github.com/umarkochkarov/lb1.3.git>

**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 локальной 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 удаления удаленной ветки существует команда `git push origin -d <название_ветки>`.