



МИНОБРНАУКИ РОССИИ
федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технологический университет «СТАНКИН»
(ФГБОУ ВО «МГТУ «СТАНКИН»)

Институт цифровых
интеллектуальных систем

Кафедра
компьютерных систем управления

Дисциплина «Основы системного программного обеспечения»

Отчет по лабораторной работе № 1
Работа Git Hub

Выполнил
студент гр. АДБ-20-06:

Майоров А.М.

(дата)

(подпись)

Проверил
к.т.н., доцент

Ковалев И.А.

(дата)

(подпись)

Москва 2023 г.

Содержание

- 1. Цель**
- 2. Задачи работы**
- 3. Описание и выполнение тестовых и индивидуальных заданий**

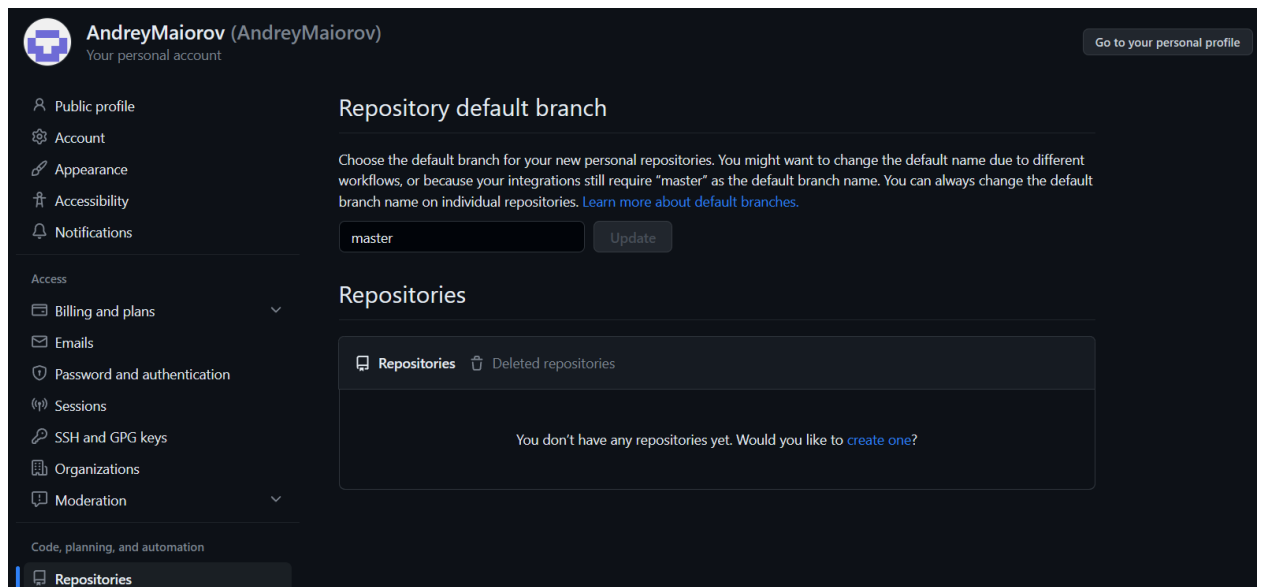
1.Цель: научиться работать с системой контроля версий Git Hub.

2. Задачи работы:

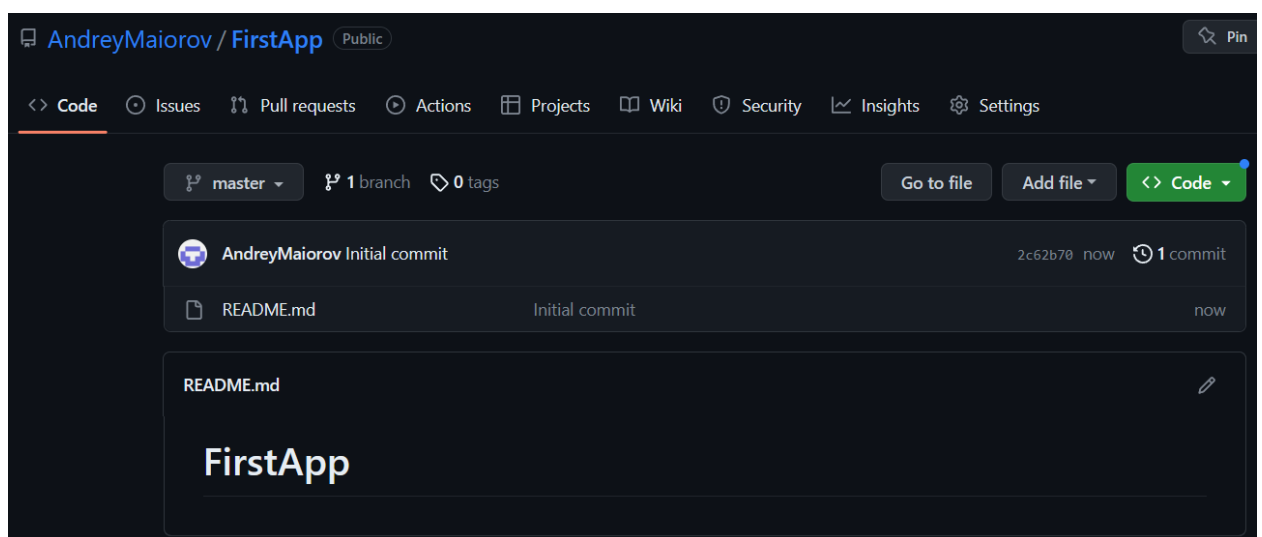
- 1) Создать учетную запись на Git Hub.
- 2) Создать репозиторий и научиться добавлять туда файлы.
- 3) Научиться отправлять изменения на сервер.

3. Описание и выполнение тестовых и индивидуальных заданий

Создадим новый репозиторий FirstApp.

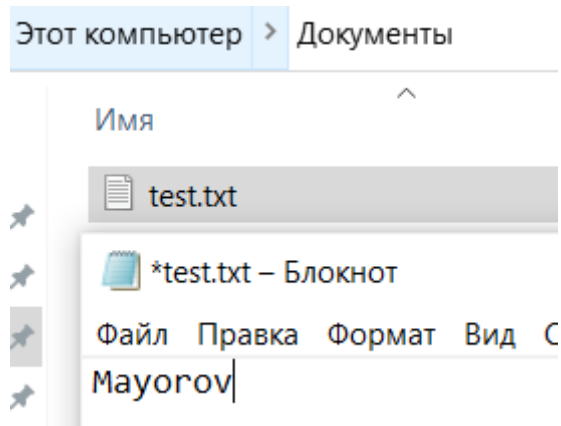


Увидим, что наш репозиторий создан.



Создадим теперь локальную папку, которая будет связана с нашим облачным git хранилищем.

В ней создадим файл test.txt с фамилией.



Настроим наше имя и адрес электронной почты. А также проинициализируем эту папку как git репозиторий.

```
C:\Windows\System32\cmd.exe
Microsoft Windows [Version 10.0.19045.2965]
(c) Корпорация Майкрософт (Microsoft Corporation). Все права защищены.

C:\Users\Admin\Documents>git config --global user.name "AndreyMaiorov"

C:\Users\Admin\Documents>git config --global user.email "andrejmajorov@gmail.com"

C:\Users\Admin\Documents>git init
Initialized empty Git repository in C:/Users/Admin/Documents/.git/

C:\Users\Admin\Documents>
```

Командная строка вернула нам сообщение, что проинициализирована пустая git директория. Теперь используя проводник windows создадим в своей папке любой текстовый файл и пропишем туда наше имя. Введем команду:

```
C:\Users\Admin\Documents\Mayorov>git status
On branch master

No commits yet

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

nothing added to commit but untracked files present (use "git add" to track)

C:\Users\Admin\Documents\Mayorov>
```

Нам отобразилось сообщение, что есть новый файл, но он не отслеживается.

Это значит, что система еще не знает, нужно следить за изменениями в файле или его можно игнорировать. Для этого нужно добавить файлы командой add.

```
C:\Users\Admin\Documents\Mayorov>git add .

C:\Users\Admin\Documents\Mayorov>git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
        new file:   test.txt

C:\Users\Admin\Documents\Mayorov>
```

Выполнив эту команду посмотрим на статус репозитория и видим, что файл готов к коммиту.

Теперь закоммитим наш файл.

Чтобы зафиксировать изменения, нам нужно хотя бы одно изменение в области подготовки.

```
C:\Users\Admin\Documents\Mayorov>git commit -m "first"
[master (root-commit) ecfc277] first
 1 file changed, 1 insertion(+)
 create mode 100644 test.txt

C:\Users\Admin\Documents\Mayorov>git status
On branch master
nothing to commit, working tree clean

C:\Users\Admin\Documents\Mayorov>
```

Коммит создан. И после вызова git status видно, что файлов для коммитов больше нет.

Теперь свяжем наш локальный репозиторий с репозиторием на GitHub, для этого выполним следующее.

```
C:\Users\Admin\Documents\Mayorov>git remote add origin https://github.com/AndreyMaierov/FirstApp.git

C:\Users\Admin\Documents\Mayorov>git remote -v
origin  https://github.com/AndreyMaierov/FirstApp.git (fetch)
origin  https://github.com/AndreyMaierov/FirstApp.git (push)
```

Мы указываем, что подключаемся к репозиторию по URI, так как проект может иметь несколько репозитория одновременно нужно давать им разные имена, а также указать origin у главного репозитория.

Затем посмотрим с помощью команды `git remote -v` к какому проекту мы подключены.

`git pull origin master --allow-unrelated-histories` — команда нужная, если у нас указано несколько репозиториев и у нас образовалось запутывание веток. В нашем случае таких проблем не было. Так же можно удалить не правильный репозиторий командой `git remote rm origin`.

В нашем случае у нас есть файл и на сервере и локальном репозитории, поэтому нам вначале надо получить все данные с нашего репозитория, а потом уже их заливать на сервер нашими локальными данными.

Мы сделали изменения в нашем репозитории, теперь другие пользователи могут скачать изменения при помощи команды `pull`.

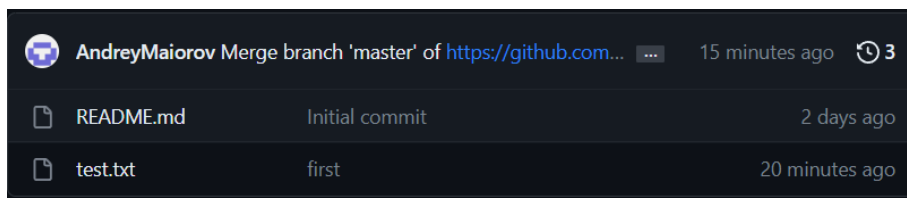
`git pull origin master`

Теперь перешлем локальный коммит на сервер. Для этого существует команда `push`. Она принимает два параметра: имя удаленного репозитория (мы назвали нашу ветку `origin`) и ветку, в которую необходимо внести изменения (`master` — это ветка по умолчанию для всех репозиториев).

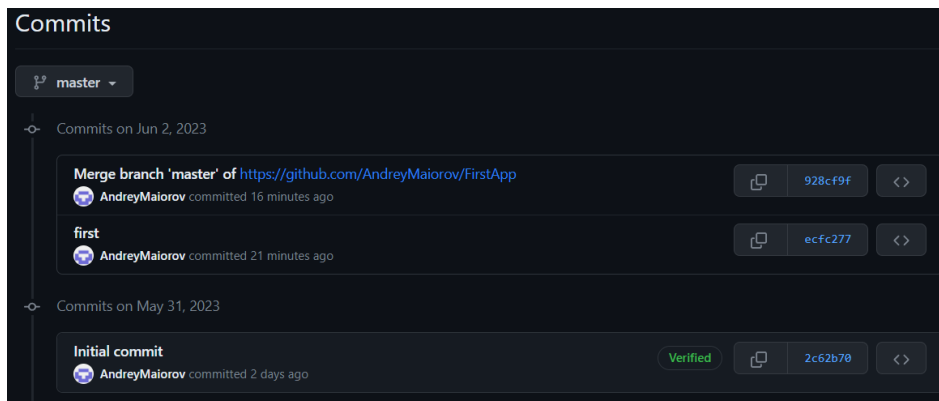
```
C:\Users\Admin\Documents\Mayorov>git push origin master
Enumerating objects: 6, done.
Counting objects: 100% (6/6), done.
Delta compression using up to 8 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (5/5), 524 bytes | 524.00 KiB/s, done.
Total 5 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/AndreyMaierov/FirstApp.git
   2c62b70..928cf9f  master -> master

C:\Users\Admin\Documents\Mayorov>
```

Видим, что изменения отобразились и на нашем сервере.



Если нажмем на «commit», то увидим все наши изменения.



Теперь создадим новую ветку.

Ветка master создается сама, а дополнительные, мы создаем по своему желанию, с помощью команды git branch.

Теперь переключимся на другую ветку. Это мы делаем с помощью команды checkout

```
C:\Users\Admin\Documents\Mayorov>git branch second

C:\Users\Admin\Documents\Mayorov>git branch
* master
  second

C:\Users\Admin\Documents\Mayorov>git checkout second
Switched to branch 'second'

C:\Users\Admin\Documents\Mayorov>_
```

Создаем в нашем локальном репозитории test2.txt и напишем там свою фамилию, затем добавляем в области подготовленных файлов, закомитим и отправим на сервер.

```
C:\Users\Admin\Documents\Mayorov>git add .

C:\Users\Admin\Documents\Mayorov>git status
On branch second
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        new file:   test2.txt
```

```

C:\Users\Admin\Documents\Mayorov>git commit -m "second"
[second 18a000a] second
1 file changed, 1 insertion(+)
create mode 100644 test2.txt

C:\Users\Admin\Documents\Mayorov>git push origin second
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 8 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 309 bytes | 103.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
remote:
remote: Create a pull request for 'second' on GitHub by visiting:
remote:   https://github.com/AndreyMaiorov/FirstApp/pull/new/second
remote:
To https://github.com/AndreyMaiorov/FirstApp.git
 * [new branch]      second -> second

```

Зайдем на Git Hub и посмотрим, что в ветке master нет нашего файла, а в ветке second он присутствует.

master ▾
Go to file
Add file ▾
<> Code ▾

Your master branch isn't protected
 Protect this branch from force pushing or deletion, or require status checks before merging. [Learn more](#)
Protect this branch Dismiss

AndreyMaiorov Merge branch 'master' of [https://github.com/...](https://github.com/AndreyMaiorov/FirstApp) ... 20 minutes ago 3

README.md	Initial commit	2 days ago
test.txt	first	24 minutes ago

AndreyMaiorov second ... 1 minute ago 4

README.md	Initial commit	2 days ago
test.txt	first	25 minutes ago
test2.txt	second	1 minute ago

Переключимся снова на ветку master. Теперь если через проводник windows зайти в папку с проектом, то файла test2.txt, там не будет видно.


```
C:\Users\Admin\Documents\Mayorov>git checkout master
Switched to branch 'master'

C:\Users\Admin\Documents\Mayorov>_
```

←

→

⌵

⬆

« Документы » Mayorov

↺

🔍 Поиск в: Mayorov

⌵

★ Быстрый доступ

🖨 Рабочий стол

⬇ Загрузки

📁 Документы

Имя

Дата изменения

Тип

Размер

.git

02.06.2023 13:44

Папка с файлами

README.md

02.06.2023 13:23

Файл "MD"

1 КБ

test.txt

02.06.2023 13:15

Текстовый докум...

1 КБ

Чтобы он появился, нужно воспользоваться merge для объединения веток.

```
C:\Users\Admin\Documents\Mayorov>git merge second
Updating 928cf9f..18a000a
Fast-forward
 test2.txt | 1 +
 1 file changed, 1 insertion(+)
 create mode 100644 test2.txt
```

Теперь удалим ветку second.

```
C:\Users\Admin\Documents\Mayorov>git branch -d second
Deleted branch second (was 18a000a).
```

Создадим ветку, в которой будут одинаковые файлы.

```
C:\Users\Admin\Documents\Mayorov>git branch newdev

C:\Users\Admin\Documents\Mayorov>git branch
* master
  newdev
```

Переключимся на неё

```
C:\Users\Admin\Documents\Mayorov>git checkout newdev
Switched to branch 'newdev'
```

Добавим в файл изменения, а, то есть имя и отчество и закомитим их.

```
C:\Users\Admin\Documents\Mayorov>git add .

C:\Users\Admin\Documents\Mayorov>git status
On branch newdev
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   test.txt

C:\Users\Admin\Documents\Mayorov>git commit -m "three"
[newdev ebdba4e] three
1 file changed, 1 insertion(+), 1 deletion(-)
```

Переключимся на ветку master и посмотрим файл, в нем только фамилия, имя и отчество в другой ветке.

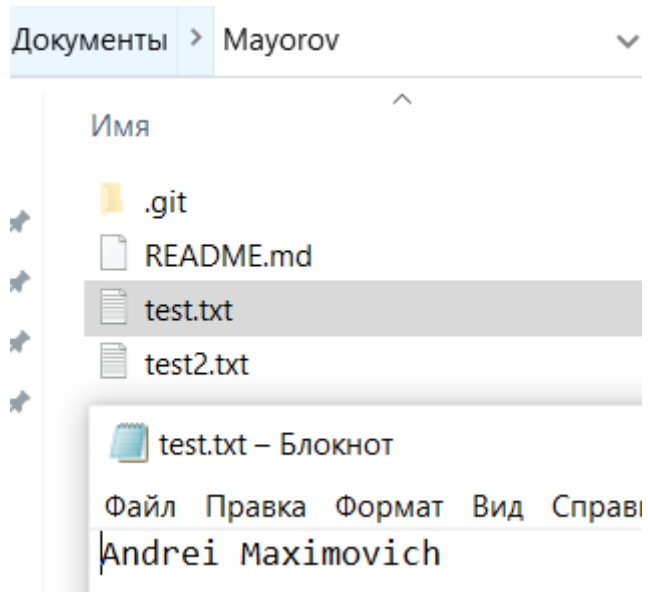
Теперь объединим ветки

```
C:\Users\Admin\Documents\Mayorov>git checkout master
Switched to branch 'master'

C:\Users\Admin\Documents\Mayorov>git checkout newdev
Switched to branch 'newdev'

C:\Users\Admin\Documents\Mayorov>git merge newdev
Already up to date.
```

Откроем файл снова и увидим изменения



Переключимся на ветку newdev и удалим в отчестве несколько букв

Зафиксируем, закоммитим

Переключимся на ветку master, добавим к отчеству несколько букв

Зафиксируем, закоммитим

Объединим ветки

```

C:\Users\Admin\Documents\Mayorov>git add .

C:\Users\Admin\Documents\Mayorov>git status
On branch newdev
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   test.txt

C:\Users\Admin\Documents\Mayorov>git commit -m "four"
[newdev ffee50a] four
1 file changed, 1 insertion(+), 1 deletion(-)

C:\Users\Admin\Documents\Mayorov>git checkout master
Switched to branch 'master'

C:\Users\Admin\Documents\Mayorov>git add .

C:\Users\Admin\Documents\Mayorov>git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   test.txt

C:\Users\Admin\Documents\Mayorov>git commit -m "five"
[master d89b58f] five
1 file changed, 1 insertion(+), 1 deletion(-)

```

Теперь ничего не получится, так как изменения есть в обеих ветках.
Разрешим конфликт.

```

C:\Users\Admin\Documents\Mayorov>git diff
diff --cc test.txt
index 08b1334,a86949c..0000000
--- a/test.txt
+++ b/test.txt
@@@ -1,1 -1,1 +1,5 @@@
- Andrei Maximovicheee
- Andrei Maxim
++<<<<<< HEAD
++Andrei Maximovicheee
++=====
++Andrei Maxim
++>>>>>> newdev

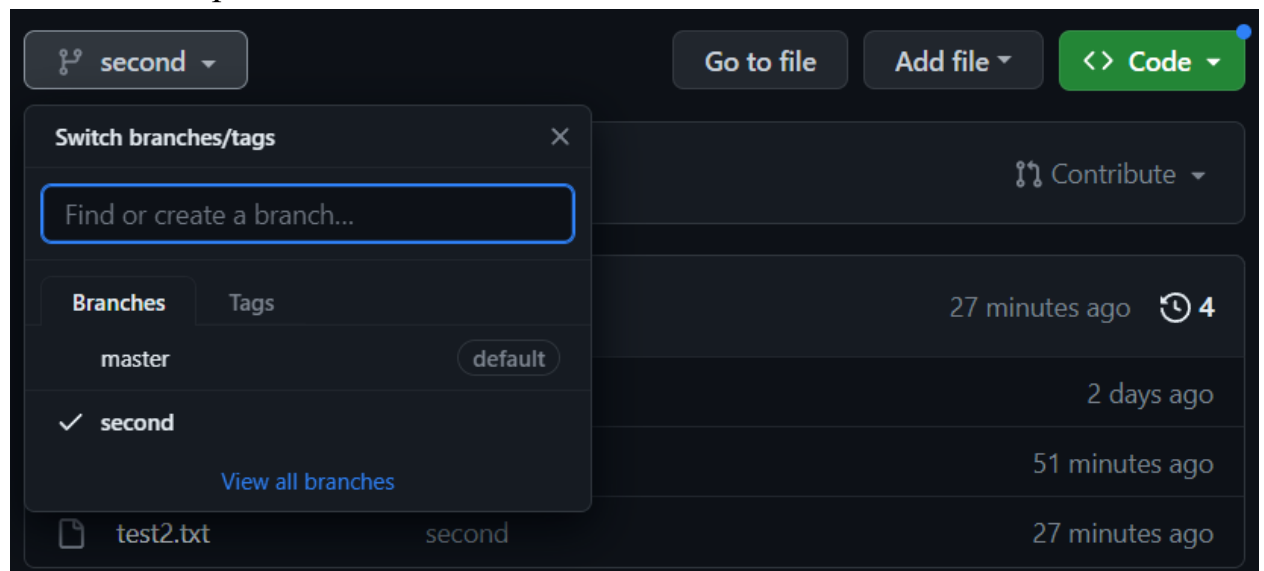
```

Исправим ошибку вручную, удалив разделители и выбрав нужный нам вариант. Сохраняем файл, фиксируем изменения, коммитим, отправляем на сервер. И удаляем ветку newdev.

```
C:\Users\Admin\Documents\Mayorov>git commit -m "test"
On branch master
nothing to commit, working tree clean

C:\Users\Admin\Documents\Mayorov>git push origin master
Enumerating objects: 14, done.
Counting objects: 100% (14/14), done.
Delta compression using up to 8 threads
Compressing objects: 100% (8/8), done.
Writing objects: 100% (12/12), 1.11 KiB | 566.00 KiB/s, done.
Total 12 (delta 1), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (1/1), done.
To https://github.com/AndreyMaiorov/FirstApp.git
   928cf9f..1bfed3f  master -> master
```

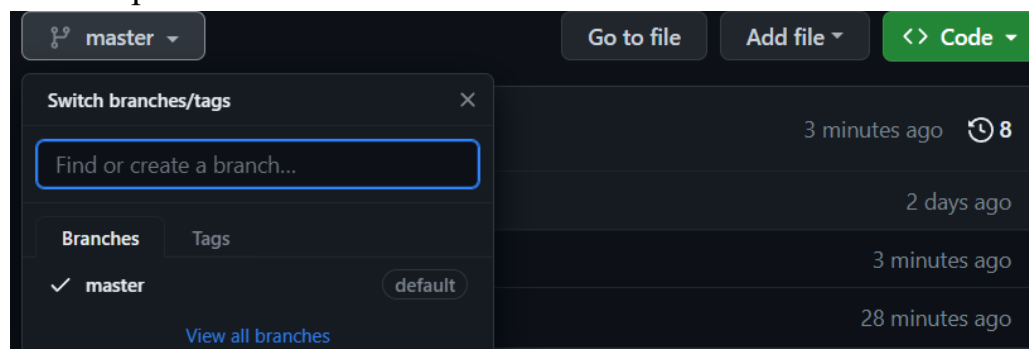
Если посмотрим на Git Hub, то наши ветки там остались.



Удалим их.

```
C:\Users\Admin\Documents\Mayorov>git push origin --delete second
To https://github.com/AndreyMaiorov/FirstApp.git
- [deleted]          second
```

Посмотрим изменения.



Также Git позволяет вернуть выбранный файл к состоянию на момент определенного коммита. Это делается командой checkout. Для того, чтобы просмотреть все наши комиты используем команду git log

```
commit 1bfed3ff34a3a8d52c96651f168955741af22ad8 (HEAD -> master, origin/master)
Merge: d89b58f ffee50a
Author: AndreyMaiorov <andrejmajorov@gmail.com>
Date:   Fri Jun 2 14:07:20 2023 +0300

    six

commit d89b58f3683c99fce440016f2f16758947ae319b
Author: AndreyMaiorov <andrejmajorov@gmail.com>
Date:   Fri Jun 2 14:03:33 2023 +0300

    five

commit ffee50ada5fb6db0cc9f71f262f06ae3b3628bc6
Author: AndreyMaiorov <andrejmajorov@gmail.com>
Date:   Fri Jun 2 14:01:40 2023 +0300

    four

commit ebdba4e8410a6c77fae30e402f66f72496069653
Author: AndreyMaiorov <andrejmajorov@gmail.com>
Date:   Fri Jun 2 13:56:36 2023 +0300

    three

commit 18a000aa5c036f085e37b773a4b2a71166706938
Author: AndreyMaiorov <andrejmajorov@gmail.com>
Date:   Fri Jun 2 13:42:24 2023 +0300

    second

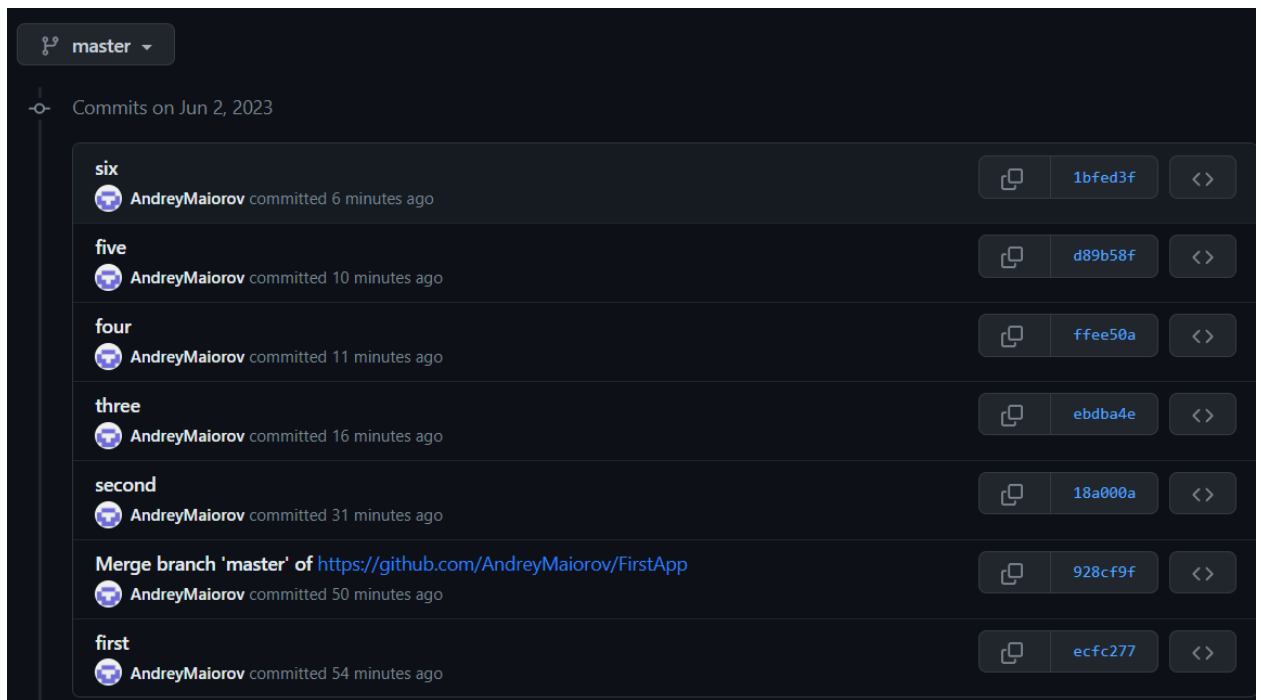
commit 928cf9f645fa4a9b6c2dd96d1f870f9148a2d803
Merge: ecfc277 2c62b70
Author: AndreyMaiorov <andrejmajorov@gmail.com>
Date:   Fri Jun 2 13:23:03 2023 +0300

    Merge branch 'master' of https://github.com/AndreyMaiorov/FirstApp

commit ecfc277fa563465a81df389c20bfd7f69888f2e4
Author: AndreyMaiorov <andrejmajorov@gmail.com>
Date:   Fri Jun 2 13:18:43 2023 +0300

    first
```

Или можно использовать github.



Выберем любой коммит, на который хотим откатиться, указав его первые несколько символов.

```
C:\Users\Admin\Documents\Mayorov>git checkout 928cf
Note: switching to '928cf'.

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 928cf9f Merge branch 'master' of https://github.com/AndreyMaiorov/FirstApp
```

Если посмотреть файлы теперь, то видно, что они поменялись – мы вернулись назад.

```
C:\Users\Admin\Documents\Mayorov>git branch
* (HEAD detached at 928cf9f)
  master

C:\Users\Admin\Documents\Mayorov>
```

Чтобы это произошло создастся псевдо-ветка начинающаяся на этом коммите. Дальнейший правильный вариант заключается в том, что нужно продолжить работать на новой ветке, а затем смириваться.

Рассмотрим ситуацию, если мы закрепим файлы, но еще не комитили и хотим убрать их из области закрепления.

```
C:\Users\Admin\Documents\Mayorov>git status
HEAD detached at 928cf9f
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   test.txt

no changes added to commit (use "git add" and/or "git commit -a")

C:\Users\Admin\Documents\Mayorov>git add .

C:\Users\Admin\Documents\Mayorov>git status
HEAD detached at 928cf9f
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   test.txt

C:\Users\Admin\Documents\Mayorov>git reset HEAD
Unstaged changes after reset:
M       test.txt

C:\Users\Admin\Documents\Mayorov>git status
HEAD detached at 928cf9f
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   test.txt

no changes added to commit (use "git add" and/or "git commit -a")
```

Теперь снова добавим в область закрепления и закомитим, но еще не отправляя на сервер, можно вызвать ту же команду и тогда все, что осталось незакомиченным будет удалено.

git

```

C:\Users\Admin\Documents\Mayorov>git add .

C:\Users\Admin\Documents\Mayorov>git status
HEAD detached at 928cf9f
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   test.txt

C:\Users\Admin\Documents\Mayorov>git commit -m "ten"
[detached HEAD b4fe346] ten
 1 file changed, 1 insertion(+), 1 deletion(-)

C:\Users\Admin\Documents\Mayorov>git reset HEAD

C:\Users\Admin\Documents\Mayorov>git status
HEAD detached from 928cf9f
nothing to commit, working tree clean

C:\Users\Admin\Documents\Mayorov>

```

Если хотим изменить комментарий можно вызвать команду: `git commit – amend`

Откроется консоле в редакторе Vi.

```

C:\Windows\System32\cmd.exe - git commit --amend

```

```

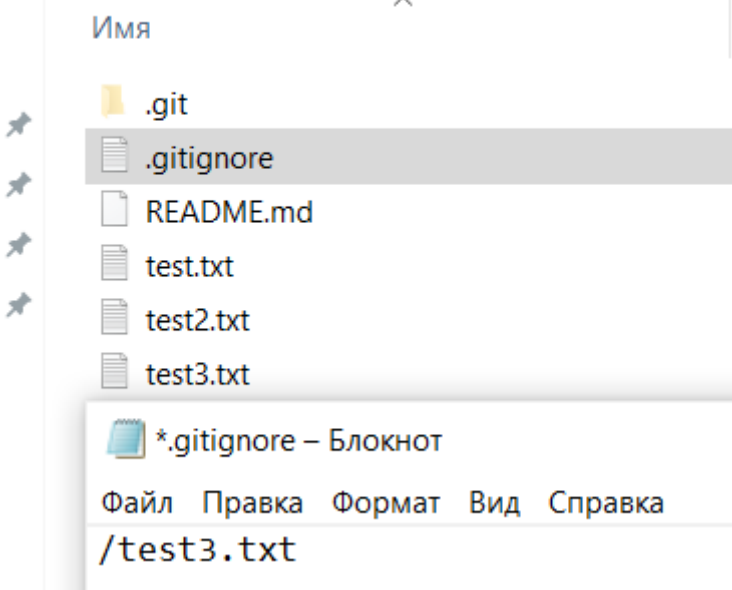
six
# Please enter the commit message for your changes. Lines starting
# with '#' will be ignored, and an empty message aborts the commit.
#
# Date:      Fri Jun 2 14:07:20 2023 +0300
#
# On branch master
# Changes to be committed:
#       modified:   test.txt
#

```

В большинстве проектов есть файлы или целые директории, в которые мы не хотим (и, скорее всего, не захотим) коммитить. Создадим файл `.gitignore` и сохраним его в директорию проекта.

И перечислим файлы которые хотим игнорировать.

Документы > Mayorov



```
C:\Users\Admin\Documents\Mayorov>git add .

C:\Users\Admin\Documents\Mayorov>git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        new file:   .gitignore

C:\Users\Admin\Documents\Mayorov>git commit -m "last"
[master 1e740a2] last
 1 file changed, 1 insertion(+)
 create mode 100644 .gitignore

C:\Users\Admin\Documents\Mayorov>git push origin master
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 8 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 346 bytes | 346.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/AndreyMaigorov/FirstApp.git
 1bfed3f..1e740a2  master -> master
```

Посмотрим на изменения.

	AndreyMaiorov last	...	now	 9
	.gitignore	last		now
	README.md	Initial commit		2 days ago
	test.txt	six		29 minutes ago
	test2.txt	second		54 minutes ago