

Задание 2.1

1. Инициализация git-репозитория

```
[14:32:54] mxschardt:Herzen $ git init
hint: Using 'master' as the name for the initial branch. This default branch name
hint: is subject to change. To configure the initial branch name to use in all
hint: of your new repositories, which will suppress this warning, call:
hint:
hint:   git config --global init.defaultBranch <name>
hint:
hint: Names commonly chosen instead of 'master' are 'main', 'trunk' and
hint: 'development'. The just-created branch can be renamed via this command:
hint:
hint:   git branch -m <name>
Initialized empty Git repository in /home/mxschardt/Herzen/.git/
```

2. Создание файла и просмотр статуса репозитория

```
[14:34:04] mxschardt:Herzen git:(master) $ touch first_file.txt
[14:34:17] mxschardt:Herzen git:(master*) $ git status
On branch master

No commits yet

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

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

3. Добавление файла в отслеживание

```
[14:34:50] mxschardt:Herzen git:(master*) $ git add first_file.txt
[14:34:55] mxschardt:Herzen git:(master*) $ git status
On branch master

No commits yet

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

4. Создание коммита

```
[14:35:21] mxschardt:Herzen git:(master*) $ git commit -m "Initial Commit"
[master (root-commit) 20a97dd] Initial Commit
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 first_file.txt
```

5. Создание файлов одного расширения и добавление их в репозиторий

```
[14:35:29] mxschardt:Herzen git:(master) $ touch 1.txt
[14:35:48] mxschardt:Herzen git:(master*) $ touch 2.txt
[14:35:50] mxschardt:Herzen git:(master*) $ touch 3.txt
[14:35:53] mxschardt:Herzen git:(master*) $ git add '*.txt'
[14:38:56] mxschardt:Herzen git:(master*) $ git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:   1.txt
    new file:   2.txt
    new file:   3.txt
```

6. Просмотр истории коммитов

```
[14:55:00] mxschardt:Herzen git:(new*) $ git log --summary
commit efbff01c3ab4956d55c8f01d9e73930ad701ca38 (HEAD -> new, origin/main, main)
Author: Maxim Schardt <mxschardt@gmail.com>
Date:   Mon Sep 26 14:44:43 2022 +0300

    new files

    create mode 100644 1.txt
    create mode 100644 2.txt
    create mode 100644 3.txt

commit 20a97ddf55a877f2ced40cf6c6e24135c8bc846
Author: Maxim Schardt <mxschardt@gmail.com>
Date:   Mon Sep 26 14:35:29 2022 +0300

    Initial Commit

    create mode 100644 first_file.txt
```

7. Создание ГитХаб репозитория

```
[14:43:36] mxschardt:Herzen git:(master*) $ git branch -m main
[14:43:42] mxschardt:Herzen git:(main*) $ git remote add origin git@github.com:mxschardt/praxis2022.git
[14:44:02] mxschardt:Herzen git:(main*) $ git push -u origin main
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Writing objects: 100% (3/3), 223 bytes | 223.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To github.com:mxschardt/praxis2022.git
 * [new branch]      main -> main
Branch 'main' set up to track remote branch 'main' from 'origin'.
```

8. Просмотр изменений

```
[14:52:25] mxschardt:Herzen git:(new*) $ git diff HEAD
diff --git a/first_file.txt b/first_file.txt
index e69de29..ecf7959 100644
--- a/first_file.txt
+++ b/first_file.txt
@@ -0,0 +1 @@
+text!
```

9. Создание и переход на новую ветку

```
[14:51:57] mxschardt:Herzen git:(main*) $ git branch new
[14:52:06] mxschardt:Herzen git:(main*) $ git branch
* main
  new
[14:52:09] mxschardt:Herzen git:(main*) $ git checkout new
M   first_file.txt
Switched to branch 'new'
[14:52:16] mxschardt:Herzen git:(new*) $ git branch
  main
* new
[14:52:21] mxschardt:Herzen git:(new*) $ git status
On branch new
```

10. Удаление файлов и merge

```
[20:55:18] mxschardt:praxis2022 git:(test) $ git rm -r .
rm '1.txt'
rm '2.txt'
rm '3.txt'
rm 'first_file.txt'
[20:55:24] mxschardt:praxis2022 git:(test*) $ git status
On branch test
Your branch is up to date with 'origin/main'.

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        deleted:    1.txt
        deleted:    2.txt
        deleted:    3.txt
        deleted:    first_file.txt

[20:55:30] mxschardt:praxis2022 git:(test*) $ git merge main
Already up to date.
```