

## Задание 2.1

На примере основных действий, продемонстрированных на вебинаре по Git, освоить работу (и продемонстрировать выполнение показанных на вебинаре действий) с одним из визуальных клиентов для работы с Git на выбор

Выбранный клиент - **Командная строка**

В процессе отчета продемонстрируем показанные на вебинаре действия:

Для инициализации системы контроля версии git используется команда

**git init**

```
C:\Users\Admin\Desktop\git1>git init
Initialized empty Git repository in C:/Users/Admin/Desktop/git1/.git/
```

Для вывода статуса git используется

**git status**

```
C:\Users\Admin\Desktop\git1>git status
On branch master

No commits yet

nothing to commit (create/copy files and use "git add" to track)
```

Создали текстовый файл при помощи команды

**copy con readme.txt**

**<любой текст>**

```
C:\Users\Admin\Desktop\git1>copy con readme.txt
12345
```

Файл был создан, но пока не отслеживается системой, проверяем

**git status**

```
C:\Users\Admin\Desktop\git1>git status
On branch master

No commits yet

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

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

Чтобы начать отслеживать, нужно ввести команду

**git add <имя файла>**

```
C:\Users\Admin\Desktop\git1>git add readme.txt
```

Теперь отслеживается

```
C:\Users\Admin\Desktop\git1>git status
On branch master

No commits yet

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

Чтобы его зафиксировать, введем команду

**git commit -m "<комментарий>"**

```
C:\Users\Admin\Desktop\git1>git commit -m "First file"
[master (root-commit) 7286b93] First file
 1 file changed, 2 insertions(+)
 create mode 100644 readme.txt
```

Добавили бы все файлы с расширением .txt при помощи

**git add '\*.txt'**

```
C:\Users\Admin\Desktop\git1>git add '*.txt'
```

Команда показывает информацию о всех зафиксированных файлах

**git log**

```
C:\Users\Admin\Desktop\git1>git log
commit 7286b93150d09c8f80bea79b278e0ff92b7dd4b6 (HEAD -> master)
Author: iveremchuk <ilya.doma@mail.ru>
Date:   Tue Sep 27 19:34:53 2022 +0300

    First file
```

Более подробно, если бы были еще файлы благодаря

**git log --summary**

```
C:\Users\Admin\Desktop\git1>git log --summary
commit 7286b93150d09c8f80bea79b278e0ff92b7dd4b6 (HEAD -> master)
Author: iveremchuk <ilya.doma@mail.ru>
Date:   Tue Sep 27 19:34:53 2022 +0300

    First file

    create mode 100644 readme.txt
```

Добавляем удаленный репозиторий

**git remote add origin <ссылка на репозиторий>**

```
C:\Users\Admin\Desktop\git1>git remote add origin https://git.herzen.spb.ru/212084/upt_veremchuk_2022.git
```

Выгружаем репозиторий, предварительно входя в систему git lab, командой

**git push -u origin master**

```
C:\Users\Admin\Desktop\git1>git push -u origin master
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Writing objects: 100% (3/3), 217 bytes | 108.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To https://git.herzen.spb.ru/212084/upt_veremchuk_2022.git
 * [new branch]      master -> master
branch 'master' set up to track 'origin/master'.
```

Изменим файл в браузере

Edit file

Write

Preview changes

Y master

readme.txt

1

12345

2

54321

3

Вытянуть изменения удаленного репозитория при помощи

**git pull origin master**

```
C:\Users\Admin\Desktop\git1>git pull origin master
remote: Enumerating objects: 5, done.
remote: Counting objects: 100% (5/5), done.
remote: Total 3 (delta 0), reused 0 (delta 0)
Unpacking objects: 100% (3/3), 285 bytes | 35.00 KiB/s, done.
From https://git.herzen.spb.ru/212084/upt_veremchuk_2022
 * branch            master      -> FETCH_HEAD
  7286b93..da49893  master      -> origin/master
Updating 7286b93..da49893
Fast-forward
 readme.txt | 2 + -
 1 file changed, 1 insertion(+), 1 deletion(-)
```

Далее показывает какие изменения произошли в файле

**git diff HEAD**

```
C:\Users\Admin\Desktop\git1>git diff HEAD
diff --git a/readme.txt b/readme.txt
index 64b2311..fb77942 100644
--- a/readme.txt
+++ b/readme.txt
@@ -1,2 +1,2 @@
-12345
-54321
+12345!
+54321!
```

Создадим каталог и файл и увидим что они еще не отслеживаются

**mkdir <название папки>**

**git status**

```
C:\Users\Admin\Desktop\git1>mkdir folder
```

```
C:\Users\Admin\Desktop\git1>git status
On branch master
Your branch is up to date with 'origin/master'.

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:   readme.txt

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

no changes added to commit (use "git add" and/or "git commit -a")
C:\Users\Admin\Desktop\git1>
```

Фиксируем

```
C:\Users\Admin\Desktop\git1>git status
On branch master
Your branch is up to date with 'origin/master'.

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

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:   readme.txt
```

Изменения которые были сделаны до фиксации можно увидеть командой

**git dif --staged**

```
C:\Users\Admin\Desktop\git1>git dif --staged
git: 'dif' is not a git command. See 'git --help'.

The most similar commands are
    diff
    config
    difftool
    init
```

Если хотим вернуть изменения пишем

**git checkout – <имя файла>**

```
C:\Users\Admin\Desktop\git1>git checkout -- readme.txt
```

12345

54321

Если мы хотим создать ветку в репозитории, пишем

**git branch <имя ветки>**

```
C:\Users\Admin\Desktop\git1>git branch clean_up
```

Команда показывает в какой ветке мы работаем

**git branch**

```
C:\Users\Admin\Desktop\git1>git branch
clean_up
* master
```

Чтобы перейти на другую ветку напишем

**git checkout <название ветки>**

```
C:\Users\Admin\Desktop\git1>git checkout clean_up
Switched to branch 'clean_up'
```

Чтобы удалить файлы введем

**rm -r <имя файла>**

```
C:\Users\Admin\Desktop\git1>rm -r folder
```