

## Вариативная самостоятельная работа

### Задание 2.1

#### Использование системы контроля версий Git

В данной работе будет рассмотрено использование командной строки в Git Bash.

Для создания репозитория используется команда `git init`.

```
User@LAPTOP-L6343K26 MINGW32 ~/OneDrive/Рабочий стол/ptact (master)
$ git init
Reinitialized existing Git repository in C:/Users/User/OneDrive/Рабочий стол/ptact/.git/
```

Чтобы убедиться в том, что создание репозитория прошло успешно, используется `ls-la`

```
User@LAPTOP-L6343K26 MINGW32 ~/OneDrive/Рабочий стол/ptact (master)
$ ls
total 29
drwxr-xr-x 1 User 197121 0 Sep 27 20:22 ./
drwxr-xr-x 1 User 197121 0 Sep 27 20:22 ../
drwxr-xr-x 1 User 197121 0 Sep 27 20:23 .git/
-rw-r--r-- 1 User 197121 66 Sep 27 20:22 .gitattributes
User@LAPTOP-L6343K26 MINGW32 ~/OneDrive/Рабочий стол/ptact (master)
```

С помощью команды `git status` можно посмотреть статус репозитория

```
User@LAPTOP-L6343K26 MINGW32 ~/OneDrive/Рабочий стол/ptact (master)
$ git status
On branch master
nothing to commit, working tree clean
```

Используя команду `touch` создается текстовый файл, который можно будет найти в проводнике

```
User@LAPTOP-L6343K26 MINGW32 ~/OneDrive/Рабочий стол/ptact (master)
$ touch readme.txt
```

Используя команду `nano` можно добавлять или изменять текст в созданном файле



Далее, чтобы новый файл отслеживался, используется команда `git add`

```
User@LAPTOP-L6343K26 MINGW32 ~/OneDrive/Рабочий стол/pract (master)
$ git status
On branch master
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)

User@LAPTOP-L6343K26 MINGW32 ~/OneDrive/Рабочий стол/pract (master)
$ git add readme.txt
warning: in the working copy of 'readme.txt', LF will be replaced
by CRLF the next time Git touches it
```

После того, как файл отслеживается, мы можем закоммитить с помощью команды `commit -m`

```
User@LAPTOP-L6343K26 MINGW32 ~/OneDrive/Рабочий стол/pract (master)
$ git commit -m "First file"
[master 5b2cf0d] First file
1 file changed, 1 insertion(+)
create mode 100644 readme.txt
```

Чтобы изменить содержание файл снова используем команду `nano`

```
User@LAPTOP-L6343K26 MINGW32 ~/OneDrive/Рабочий стол/pract (master)
$ nano readme.txt

User@LAPTOP-L6343K26 MINGW32 ~/OneDrive/Рабочий стол/pract (master)
$ git status
On branch 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

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

Чтобы отслеживать файл с изменениями используем `add`

Далее создадим две копии текстового файла и добавим их с помощью команды `git add "*.txt"`. После проверим статус.

```
User@LAPTOP-L6343K26 MINGW32 ~/OneDrive/Рабочий стол/pract
(master)
$ git add "*.txt"

User@LAPTOP-L6343K26 MINGW32 ~/OneDrive/Рабочий стол/pract
(master)
$ git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   readme.txt
        new file:   readmeone.txt
        new file:   readmetwo.txt

User@LAPTOP-L6343K26 MINGW32 ~/OneDrive/Рабочий стол/pract
(master)
$ git commit -m"All txt files"
[master ccd5928] All txt files
3 files changed, 1 insertion(+)
create mode 100644 readmeone.txt
create mode 100644 readmetwo.txt
```

Копии появились, теперь можем закоммитить их

```
User@LAPTOP-L6343K26 MINGW32 ~/OneDrive/Рабочий стол/pract
(master)
$ git commit -m"All txt files"
[master ccd5928] All txt files
3 files changed, 1 insertion(+)
create mode 100644 readmeone.txt
create mode 100644 readmetwo.txt
```

Для просмотра истории коммитов используется команда git log. Сначала будут показаны последние изменения.

```
$ git log
commit ccd59287ff84c33b860130a382a225b0d0394efe (HEAD -> ma
ster)
Author: quimoriz <ekaterinasprt@yandex.ru>
Date:   Tue Sep 27 17:14:43 2022 +0300

    All txt files

commit 5b2cf0d3cd64bec2cfbeb274bb933dece2db8f39
Author: quimoriz <ekaterinasprt@yandex.ru>
Date:   Tue Sep 27 17:03:55 2022 +0300

    First file

commit 4d07313941e1655aac58dbff9470ea1d6ccb3e8a
Author: quimoriz <ekaterinasprt@yandex.ru>
Date:   Tue Sep 27 16:57:54 2022 +0300
```

Для того, чтобы выгрузить репозиторий на удаленный репозиторий используем команду remote add origin + вставляем ссылку на GitHub. Для загрузки файлов на репозиторий используем git push -u origin master.

```
User@LAPTOP-L6343K26 MINGW32 ~/OneDrive/Рабочий стол/pract (master)
$ git remote add origin https://github.com/EkTarkhanova/Psem2.git
error: remote origin already exists.

User@LAPTOP-L6343K26 MINGW32 ~/OneDrive/Рабочий стол/pract (master)
$ git push -u origin master

Enumerating objects: 10, done.
Counting objects: 100% (10/10), done.
Delta compression using up to 4 threads
Compressing objects: 100% (6/6), done.
Writing objects: 100% (10/10), 863 bytes | 20.00 KiB/s, done.
Total 10 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/EkTarkhanova/Pract.git
 * [new branch]      master -> master
branch 'master' set up to track 'origin/master'.
```

Чтобы получить файлы обратно используется команда `git pull origin master`

```
User@LAPTOP-L6343K26 MINGW32 ~/OneDrive/Рабочий стол/pract (master)
$ git pull origin master
remote: Enumerating objects: 5, done.
remote: Counting objects: 100% (5/5), done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 3 (delta 1), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (3/3), 663 bytes | 17.00 KiB/s, done.
From https://github.com/EkTarkhanova/Pract
 * branch            master      -> FETCH_HEAD
   ccd5928..503faa2  master      -> origin/master
Updating ccd5928..503faa2
Fast-forward
 readme.txt | 1 +
 1 file changed, 1 insertion(+)
```

Команду `git diff HEAD` используем для того, чтобы проверить отличия закоммиченных файлов.

```
User@LAPTOP-L6343K26 MINGW32 ~/OneDrive/Рабочий стол/pract (master)
$ git diff HEAD
diff --git a/readme.txt b/readme.txt
index 442b37c..cd5a89e 100644
--- a/readme.txt
+++ b/readme.txt
@@ -1,3 +1,3 @@
-You know...?
+OMG! You know...?
 I know...
 I don't know...
```

Чтобы добавить каталог в репозиторий используется команда `folder`. С помощью `touch folder/bye.txt` создаем в нем текстовый файл.



```
User@LAPTOP-L6343K26 MINGW32 ~/OneDrive/Рабочий стол/pract (master)
$ mkdir folder

User@LAPTOP-L6343K26 MINGW32 ~/OneDrive/Рабочий стол/pract (master)
$ touch folder/bye.txt

User@LAPTOP-L6343K26 MINGW32 ~/OneDrive/Рабочий стол/pract (master)
$ 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 director
)

    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")
```

После проверки видим что каталог не отслеживается, поэтому используем команду git add

Команда git diff --stages позволяет посмотреть изменения

Команда git reset позволяет отменить изменения

Команда git checkout --readme.txt позволяет вернуть файл в исходное состояние

```
User@LAPTOP-L6343K26 MINGW32 ~/OneDrive/Рабочий стол/pract (master)
$ git reset folder/bye.txt
Unstaged changes after reset:
M   readme.txt

User@LAPTOP-L6343K26 MINGW32 ~/OneDrive/Рабочий стол/pract (master)
$ git diff --staged

User@LAPTOP-L6343K26 MINGW32 ~/OneDrive/Рабочий стол/pract (master)
$ git checkout -- readme.txt

User@LAPTOP-L6343K26 MINGW32 ~/OneDrive/Рабочий стол/pract (master)
$ git status
On branch master
Your branch is up to date with 'origin/master'.

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

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

Создаем ветку используя git branch clean\_up. Команда git branch показывает на какой ветке мы находимся. Чтобы перейти на другую ветку используется git checkout.

```
User@LAPTOP-L6343K26 MINGW32 ~/OneDrive/Рабочий стол/pract (master)
$ git branch clean_up

User@LAPTOP-L6343K26 MINGW32 ~/OneDrive/Рабочий стол/pract (master)
$ git branch
  clean_up
* master
```

Команда `rm -r folder` удаляет каталог и файлы на ветке

```
User@LAPTOP-L6343K26 MINGW32 ~/OneDrive/Рабочий стол/pract (clean_up)
$ git rm readmetwo.txt
rm 'readmetwo.txt'

User@LAPTOP-L6343K26 MINGW32 ~/OneDrive/Рабочий стол/pract (clean_up)
$ git status
On branch clean_up
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        deleted:    readmeone.txt
        deleted:    readmetwo.txt
```

Чтобы удалить ветку `master`, можно выполнить слияние с `clean_up`, используя `git merge clean_up`

```
$ git commit -m "Deleted folder and files"
[clean_up 3cf67cb] Deleted folder and files
 2 files changed, 0 insertions(+), 0 deletions(-)
 delete mode 100644 readmeone.txt
 delete mode 100644 readmetwo.txt

User@LAPTOP-L6343K26 MINGW32 ~/OneDrive/Рабочий стол/pract (clean_up)
$ git checkout master
Switched to branch 'master'
Your branch is up to date with 'origin/master'.

User@LAPTOP-L6343K26 MINGW32 ~/OneDrive/Рабочий стол/pract (master)
$ git merge clean_up
Updating 503faa2..3cf67cb
Fast-forward
 readmeone.txt | 0
 readmetwo.txt | 0
 2 files changed, 0 insertions(+), 0 deletions(-)
 delete mode 100644 readmeone.txt
 delete mode 100644 readmetwo.txt
```

Ветвь `clean_up` удаляем с помощью `git branch -d clean_up`

Выгружаем все на GitHub

```
$ git push
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Delta compression using up to 4 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (2/2), 231 bytes | 115.00 KiB/s, done.
Total 2 (delta 1), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To https://github.com/EkTarkhanova/Pract.git
 503faa2..3cf67cb master -> master
```