



МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение  
высшего образования

«МИРЭА – Российский технологический университет»

**РТУ МИРЭА**

---

Институт Информационных технологий

Кафедра Математического обеспечения и стандартизации информационных  
технологий

## **Отчет по практической работе №1**

по дисциплине «Технологии разработки программных приложений»

**Тема практической работы:** «Системы контроля версий»

**Выполнил:**

Студент группы **ИКБО-62-23**

Кокшарова А.А.

**Проверил преподаватель:**

Золотухин С.А.

Москва 2025

## **Оглавление**

Постановка задачи.....	3
Часть 1. Основные команды Git .....	4
Часть 2. Управление репозиториями .....	11
Выполнение индивидуального задания .....	15
Вывод о проделанной работе.....	19

## Постановка задачи

Цель работы: получить навыки по работе с командной строкой и git'ом.

Персональное задание представлено в таблице 1.

*Таблица 1.*

№ Варианта	Задание
11	1) Клонировать непустой удаленный репозиторий на локальную машину 2) Создайте новую ветку и выведите список всех веток 3) Произведите 3 коммита в новой ветке в разные файлы 4) Выгрузите изменения в удаленный репозиторий 5) Добейтесь того, чтобы эти три изменения были представлены одним коммитом 6) Выведите в консоли различия между веткой master и новой веткой 7) Слейте новую ветку с master при помощи merge

## Часть 1. Основные команды Git

1. Установите и настройте клиент git на своей рабочей станции (Рисунок 1)

```
core.autocrlf=input
core.safecrlf=warn
core.quotePath=off
user.name=Ann
user.email=mopsin666@mail.ru
```

Рисунок 1 – Настройка git

2. Создайте локальный репозиторий и добавьте в него несколько файлов (Рисунок 2)

```
$ cd Git
Ann@MSI MINGW64 ~/Git (master)
$ mkdir TRPP
Ann@MSI MINGW64 ~/Git (master)
$ cd TRPP
Ann@MSI MINGW64 ~/Git/TRPP (master)
$ touch proect.html
Ann@MSI MINGW64 ~/Git/TRPP (master)
$ nano proect.html
Ann@MSI MINGW64 ~/Git/TRPP (master)
$ git init
Initialized empty Git repository in C:/Users/mopsi/Git/TRPP/.git/
Ann@MSI MINGW64 ~/Git/TRPP (master)
$ git add proect.html
Ann@MSI MINGW64 ~/Git/TRPP (master)
$ git commit -m "The first commit"
[master (root-commit) bbc37d] The first commit
1 file changed, 3 insertions(+)
 create mode 100644 proect.html
Ann@MSI MINGW64 ~/Git/TRPP (master)
$ git status
On branch master
nothing to commit, working tree clean
Ann@MSI MINGW64 ~/Git/TRPP (master)
$ |
```

Рисунок 2 – Создание локального репозитория и добавление в него файла

3. Внесение изменений в один из файлов (Рисунок 3)

```
Ann@MSI MINGW64 ~/Git/TRPP (master)
$ nano proect.html
Ann@MSI MINGW64 ~/Git/TRPP (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:   proect.html

no changes added to commit (use "git add" and/or "git commit -a")
Ann@MSI MINGW64 ~/Git/TRPP (master)
$ git add proect.html
Ann@MSI MINGW64 ~/Git/TRPP (master)
$ git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   proect.html
Ann@MSI MINGW64 ~/Git/TRPP (master)
$
```

Рисунок 3 – Внесение изменений в файле proect.html

#### 4. Проиндексируйте изменения и проверьте состояние (Рисунок 4)

```
Ann@MSI MINGW64 ~/Git/TRPP (master)
$ git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   proect.html

Ann@MSI MINGW64 ~/Git/TRPP (master)
$ touch a.html

Ann@MSI MINGW64 ~/Git/TRPP (master)
$ touch b.html

Ann@MSI MINGW64 ~/Git/TRPP (master)
$ touch c.html

Ann@MSI MINGW64 ~/Git/TRPP (master)
$ git add a.html

Ann@MSI MINGW64 ~/Git/TRPP (master)
$ git add b.html

Ann@MSI MINGW64 ~/Git/TRPP (master)
$ git commit -m "Changes for a and b"
[master 5bcdfdf] Changes for a and b
3 files changed, 1 insertion(+), 1 deletion(-)
create mode 100644 a.html
create mode 100644 b.html

Ann@MSI MINGW64 ~/Git/TRPP (master)
$ git add c.html

Ann@MSI MINGW64 ~/Git/TRPP (master)
$ git commit -m "Unrelated change to c"
[master 666f393] Unrelated change to c
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 c.html

Ann@MSI MINGW64 ~/Git/TRPP (master)
$ git status
On branch master
nothing to commit, working tree clean

Ann@MSI MINGW64 ~/Git/TRPP (master)
$
```

Рисунок 4 – Индексация изменений в файле proect.html

#### 5. Сделайте коммит того, что было проиндексировано в репозиторий. Добавьте к коммиту комментарий (Рисунок 5)

```
Ann@MSI MINGW64 ~/Git/TRPP (master)
$ nano proect.html

Ann@MSI MINGW64 ~/Git/TRPP (master)
$ git add proect.html

Ann@MSI MINGW64 ~/Git/TRPP (master)
$ nano proect.html

Ann@MSI MINGW64 ~/Git/TRPP (master)
$ git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   proect.html

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:   proect.html

Ann@MSI MINGW64 ~/Git/TRPP (master)
$ git commit -m "Added standart HTML page tags"
[master 25317d9] Added standart HTML page tags
1 file changed, 7 insertions(+), 2 deletions(-)

Ann@MSI MINGW64 ~/Git/TRPP (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:   proect.html

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

Ann@MSI MINGW64 ~/Git/TRPP (master)
$ git add .
```

Рисунок 5 – Сделан коммит того, что было проиндексировано в  
репозиторий

6. Измените еще один файл. Добавьте это изменение в индекс git. Измените файл еще раз. Проверьте состояние и произведите коммит проиндексированного изменения. Теперь добавьте второе изменение в индекс, а затем проверьте состояние с помощью команды `git status`. Сделайте коммит второго изменения. (Рисунок 6)

```
Ann@MSI MINGW64 ~/Git/TRPP (master)
$ git add .

Ann@MSI MINGW64 ~/Git/TRPP (master)
$ git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   proect.html

Ann@MSI MINGW64 ~/Git/TRPP (master)
$ git commit -m "Added HTML header"
[master c39f39d] Added HTML header
1 file changed, 3 insertions(+), 1 deletion(-)

Ann@MSI MINGW64 ~/Git/TRPP (master)
$ git log
commit c39f39d06dbecd0c89ccc9173e4b964afef239bf (HEAD -> master)
Author: Ann <mopsin666@mail.ru>
Date:   Wed Feb 26 11:44:32 2025 +0300

    Added HTML header

commit 25317d9a6593c4e0d0c43725548c849f361cb28d
Author: Ann <mopsin666@mail.ru>
Date:   Wed Feb 26 11:43:54 2025 +0300

    Added standart HTML page tags

commit 666f393a19ba2a603b8a6b537392117fd7c33ade
Author: Ann <mopsin666@mail.ru>
Date:   Wed Feb 26 11:39:21 2025 +0300

    Unrelated change to c

commit 5bcafd9c06d98f638f4ee8c57cb84e822cf1625a
Author: Ann <mopsin666@mail.ru>
Date:   Wed Feb 26 11:38:58 2025 +0300

    Changes for a and b

commit bbcb37ddb2058f385a7777b7c6568b9f7c7ce5f1
Author: Ann <mopsin666@mail.ru>
Date:   Wed Feb 26 11:34:14 2025 +0300

    The first commit
```

Рисунок 6 - Изменение еще одного файла. Добавление этого изменения в индекс git

7. Просмотрите историю коммитов с помощью команды `git log`. Ознакомьтесь с параметрами команды и используйте некоторые из них для различного формата отображения истории коммитов. Верните рабочий каталог к одному из предыдущих состояний. Изучите, как создавать теги для коммитов для использования в будущем. Отмените

некоторые изменения в рабочем каталоге (до и после индексирования).

Отмените один из коммитов в локальной репозитории. (Рисунок 7–10)

```
Ann@MSI MINGW64 ~/Git/TRPP (master)
$ git log --pretty=oneline
c39f39d06dbecd0c89ccc9173e4b964afef239bf (HEAD -> master) Added HTML header
25317d9a6593c4e0d0c43725548c849f361cb28d Added standart HTML page tags
666f393a19ba2a603b8a6b537392117fd7c33ade Unrelated change to c
5bcafdcf06d98f638f4ee8c57cb84e822cf1625a Changes for a and b
bbcb37ddb2058f385a7777b7c6568b9f7c7ce5f1 The first commit

Ann@MSI MINGW64 ~/Git/TRPP (master)
$ git log --pretty=oneline --max-count=2
c39f39d06dbecd0c89ccc9173e4b964afef239bf (HEAD -> master) Added HTML header
25317d9a6593c4e0d0c43725548c849f361cb28d Added standart HTML page tags

Ann@MSI MINGW64 ~/Git/TRPP (master)
$ git log --pretty=oneline --since='5 minutes ago'
c39f39d06dbecd0c89ccc9173e4b964afef239bf (HEAD -> master) Added HTML header
25317d9a6593c4e0d0c43725548c849f361cb28d Added standart HTML page tags

Ann@MSI MINGW64 ~/Git/TRPP (master)
$ git log --pretty=oneline --until='5 minutes ago'
666f393a19ba2a603b8a6b537392117fd7c33ade Unrelated change to c
5bcafdcf06d98f638f4ee8c57cb84e822cf1625a Changes for a and b
bbcb37ddb2058f385a7777b7c6568b9f7c7ce5f1 The first commit

Ann@MSI MINGW64 ~/Git/TRPP (master)
$ git log --pretty=oneline --author=<Ann>
bash: syntax error near unexpected token `newline'

Ann@MSI MINGW64 ~/Git/TRPP (master)
$ git log --pretty=oneline --author=<Anna>
bash: syntax error near unexpected token `newline'

Ann@MSI MINGW64 ~/Git/TRPP (master)
$ git log --pretty=oneline --author=<Ann@MSI>
bash: syntax error near unexpected token `newline'

Ann@MSI MINGW64 ~/Git/TRPP (master)
$ git log --pretty=oneline --all
c39f39d06dbecd0c89ccc9173e4b964afef239bf (HEAD -> master) Added HTML header
25317d9a6593c4e0d0c43725548c849f361cb28d Added standart HTML page tags
666f393a19ba2a603b8a6b537392117fd7c33ade Unrelated change to c
5bcafdcf06d98f638f4ee8c57cb84e822cf1625a Changes for a and b
bbcb37ddb2058f385a7777b7c6568b9f7c7ce5f1 The first commit
```

Рисунок 7 - История коммитов

```

Ann@MSI MINGW64 ~/Git/TRPP (master)
$ git log --pretty=format:"%h %ad | %s%d [%an]" --graph --date=short
* c39f39d 2025-02-26 | Added HTML header (HEAD -> master) [Ann]
* 25317d9 2025-02-26 | Added standart HTML page tags [Ann]
* 666f393 2025-02-26 | Unrelated change to c [Ann]
* 5bcafd9 2025-02-26 | Changes for a and b [Ann]
* bbc37d 2025-02-26 | The first commit [Ann]

Ann@MSI MINGW64 ~/Git/TRPP (master)
$ git checkout bbc37d
Note: switching to 'bbc37d'.

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 bbc37d The first commit

Ann@MSI MINGW64 ~/Git/TRPP ((bbc37d...))
$ cat proect.html

Мы создали наполнение в данном файле!!!

Ann@MSI MINGW64 ~/Git/TRPP ((bbc37d...))
$ git checkout master
Previous HEAD position was bbc37d The first commit
Switched to branch 'master'

Ann@MSI MINGW64 ~/Git/TRPP (master)
$ cat proect.html
Третье изменение

<html>
<head>
</head>
<body>
<h1>Hello World!!!</h1>
</body>
</html>

Ann@MSI MINGW64 ~/Git/TRPP (master)
$ |

```

Рисунок 8 – Возврат рабочего каталога к одному из предыдущих состояний



```

Ann@MSI MINGW64 ~/Git/TRPP (master)
$ git checkout master
Already on 'master'

Ann@MSI MINGW64 ~/Git/TRPP (master)
$ nano proect.html

Ann@MSI MINGW64 ~/Git/TRPP (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:   proect.html

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

Ann@MSI MINGW64 ~/Git/TRPP (master)
$ git checkout proect.html
Updated 1 path from the index

Ann@MSI MINGW64 ~/Git/TRPP (master)
$ git status
On branch master
nothing to commit, working tree clean

Ann@MSI MINGW64 ~/Git/TRPP (master)
$ cat proect.html
Третье изменение

<html>
<head>
</head>
<body>
<h1>Hello World!!!</h1>
</body>
</html>

Ann@MSI MINGW64 ~/Git/TRPP (master)
$ |

```

Рисунок 9 – Отмена некоторых изменений в рабочем каталоге

```

Ann@MSI MINGW64 ~/Git/TRPP (master)
$ git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    modified:   proect.html

Ann@MSI MINGW64 ~/Git/TRPP (master)
$ git reset HEAD proect.html
Unstaged changes after reset:
M    proect.html

Ann@MSI MINGW64 ~/Git/TRPP (master)
$ git checkout proect.html
Updated 1 path from the index

Ann@MSI MINGW64 ~/Git/TRPP (master)
$ git status
On branch master
nothing to commit, working tree clean

Ann@MSI MINGW64 ~/Git/TRPP (master)
$ git add proect.html

Ann@MSI MINGW64 ~/Git/TRPP (master)
$ git commit -m "0000И нам не нужен этот коммит"
On branch master
nothing to commit, working tree clean

Ann@MSI MINGW64 ~/Git/TRPP (master)
$ git config --global core.editor nano

Ann@MSI MINGW64 ~/Git/TRPP (master)
$ git revert HEAD
[master 104f8b4] Revert "Added HTML header"
 1 file changed, 1 insertion(+), 3 deletions(-)

Ann@MSI MINGW64 ~/Git/TRPP (master)
$ git revert HEAD --no-edit
[master a7b5c0f] Reapply "Added HTML header"
  Date: Wed Feb 26 15:14:08 2025 +0300
  1 file changed, 3 insertions(+), 1 deletion(-)

Ann@MSI MINGW64 ~/Git/TRPP (master)
$ git log --pretty=format:"%h %ad | %s%d [%an]" --graph --date=short
* a7b5c0f 2025-02-26 | Reapply "Added HTML header" (HEAD -> master) [Ann]
* 104f8b4 2025-02-26 | Revert "Added HTML header" [Ann]
* c39f39d 2025-02-26 | Added HTML header [Ann]
* 25317d9 2025-02-26 | Added standart HTML page tags [Ann]
* 666f393 2025-02-26 | Unrelated change to c [Ann]
* 5bcafdf 2025-02-26 | Changes for a and b [Ann]
* bbc37d 2025-02-26 | The first commit [Ann]

Ann@MSI MINGW64 ~/Git/TRPP (master)
$

```

Рисунок 10 – Отмена одного из коммитов в локальном репозитории

## Часть 2. Управление репозиториями

Создание и вставка в git SSH-ключа для авторизации (Рисунок 11–13)

```
Ann@MSI MINGW64 ~ (master)
$ cd ~/.ssh

Ann@MSI MINGW64 ~/.ssh (master)
$ ssh-keygen -t rsa -b 4096 -C "mopsin666@mail.ru"
Generating public/private rsa key pair.
Enter file in which to save the key (/c/Users/mopsi/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /c/Users/mopsi/.ssh/id_rsa
Your public key has been saved in /c/Users/mopsi/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:1+8BsVn7RsHU5giATfYkIjMGWM/86y7C+wmAXtvedLs mopsin666@mail.ru
The key's randomart image is:
+---[RSA 4096]---+
|    oo.= .+.=.. .. |
| . = +.o.+o o |
|    + . o.o+ |
| . . * .... |
|.. .. S * . . |
|.. .. o o o |
| . o.. ... o o |
|    +.+o. .. o |
|    .+o+oE. . |
+-----[SHA256]-----+

Ann@MSI MINGW64 ~/.ssh (master)
$ ssh-agent
bash: ssh-agent: command not found

Ann@MSI MINGW64 ~/.ssh (master)
$ ssh-agent
SSH_AUTH_SOCK=/tmp/ssh-1ZTaGnzYVsy1/agent.1077; export SSH_AUTH_SOCK;
SSH_AGENT_PID=1078; export SSH_AGENT_PID;
echo Agent pid 1078;

Ann@MSI MINGW64 ~/.ssh (master)
$ eval "$(ssh-agent -s)"
Agent pid 1084
```

Рисунок 11 – Создание SSH-ключа

```
Ann@MSI MINGW64 ~/.ssh (master)
$ ssh-add ~/.ssh/id_rsa
Identity added: /c/Users/mopsi/.ssh/id_rsa (mopsin666@mail.ru)

Ann@MSI MINGW64 ~/.ssh (master)
$ ssh-add ~/.ssh/id_rsa.pub
Error loading key "/c/Users/mopsi/.ssh/id_rsa.pub": error in libcrypto

Ann@MSI MINGW64 ~/.ssh (master)
$ cat ~/.ssh/id_rsa.pub
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQACyTEFjNTH7g4/ILQ7yyGpgXGAVUT/DInGx8bvrROSsL
PhGNhv9GYxSEcny3Wx7IjqdDoEPoMcGzPCzYXt95mHmyygT0AvGcEshz2nS/LkpD7M72H1Gn+niLQZuy
EOYfh6C0g67MnRjJenuZVASQ6ELrWNw98gf86jXgMjZ20kzDi7QI3CVjbzxu55/jfYJ83HNBLgzroemQ
vkVvV6Q/b7ntZJnbS09zQ0AJFGln/6fyfki/a5J/EMosQeFivJ8Kv0AgNUVHIJvG4CTAfLPao0fb6t1A
d3w8cBjEp4JKfi9xQf1VYFv8PdC9t/m1963ySNo4knN+1Y7XNnguYMAftCRU4irRGiUi4pokUnjPj5Rd
f8ejjckvtQs1sqLurQJujeEV8qYjIX6td3k1H1lC31N8r7p7IqSw3sIijyIt9+nd/HOPY5Lh8yncQrYd
cqj5hc1vKCUhK2FDHTZrzN9/OUjO9d311DzL+m2/BzXgRgZ3UKVy501P56vvg2auYH2ZCc1Zx52fb8Y0
BhaaRrEZ28xqBQtrc9BC/tI1etF1utEHmYICK1CF5a6hm6450dXNCnBEYa/sxtItjNTifQPZo3dutENZ
KV9tvDSK5B71vt6jJHOU3Pp7s3amvBN8bdg2nRYi9NS9JazBPbS2s5dNZivEQPkrCBLL49+iwYS1AW53
Bw== mopsin666@mail.ru

Ann@MSI MINGW64 ~/.ssh (master)
$
```

Рисунок 12 – Создание SSH-ключа

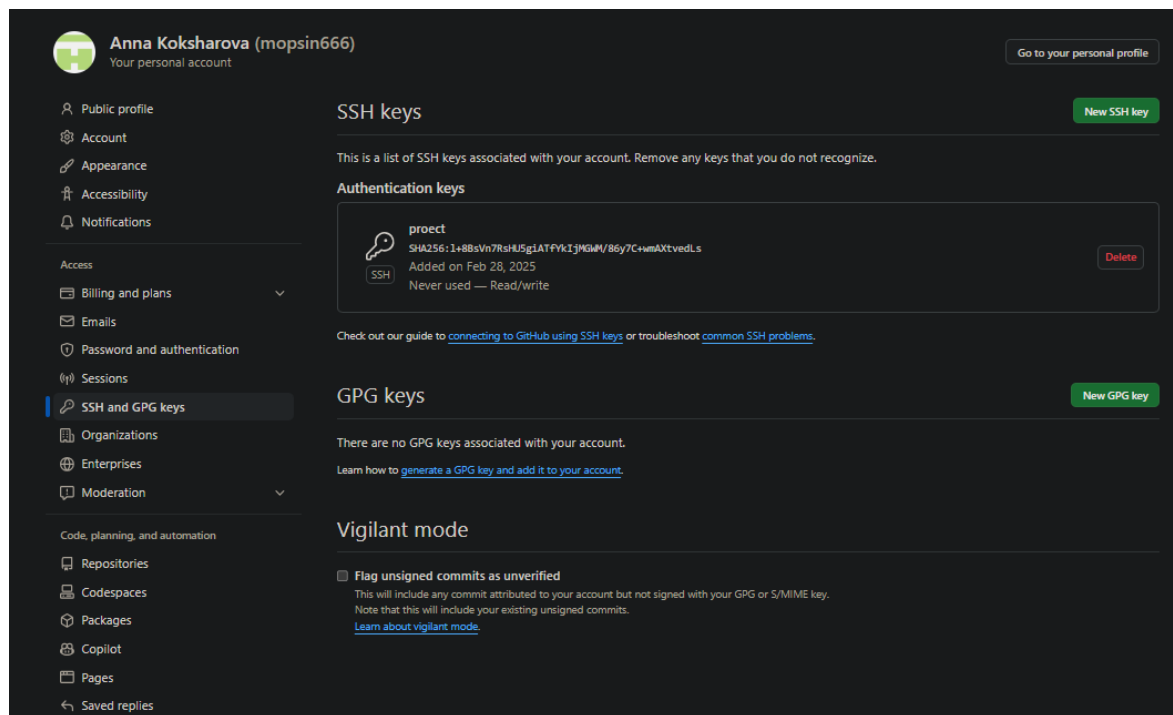


Рисунок 13 – Вставка ключа в Git

Создание локального репозитория, добавление в него файлов и изменение этих файлов (Рисунок 14)

```
Ann@MSI MINGW64 ~/.ssh (master)
$ cd C:

Ann@MSI MINGW64 /c
$ cd Users/mopsi/Git

Ann@MSI MINGW64 ~/Git (master)
$ mkdir TRPP_2

Ann@MSI MINGW64 ~/Git (master)
$ cd TRPP_2

Ann@MSI MINGW64 ~/Git/TRPP_2 (master)
$ touch readme.txt

Ann@MSI MINGW64 ~/Git/TRPP_2 (master)
$ touch File_1.html

Ann@MSI MINGW64 ~/Git/TRPP_2 (master)
$ git init
Initialized empty Git repository in C:/Users/mopsi/Git/TRPP_2/.git/

Ann@MSI MINGW64 ~/Git/TRPP_2 (master)
$ git add readme.txt

Ann@MSI MINGW64 ~/Git/TRPP_2 (master)
$ git add File_1.html

Ann@MSI MINGW64 ~/Git/TRPP_2 (master)
$ git remote add project git@github.com:mopsin666/proect.git

Ann@MSI MINGW64 ~/Git/TRPP_2 (master)
$ |
```

Рисунок 14 – Создание локального репозитория, добавление в него файлов и изменение этих файлов

## Индексация изменений и коммиты к ним (Рисунок 15-16)

```
Ann@MSI MINGW64 ~/Git/TRPP_2 (master)
$ git checkout -b main
Switched to a new branch 'main'

Ann@MSI MINGW64 ~/Git/TRPP_2 (main)
$ git status
On branch main

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
        new file:   File_1.html
        new file:   readme.txt

Ann@MSI MINGW64 ~/Git/TRPP_2 (main)
$ nano File_1.html

Ann@MSI MINGW64 ~/Git/TRPP_2 (main)
$ git add File_1.html

Ann@MSI MINGW64 ~/Git/TRPP_2 (main)
$ git status
On branch main

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
        new file:   File_1.html
        new file:   readme.txt

Ann@MSI MINGW64 ~/Git/TRPP_2 (main)
$ nano readme.txt

Ann@MSI MINGW64 ~/Git/TRPP_2 (main)
$ git add readme.txt
```

Рисунок 15 – Индексация изменений и коммиты к ним

```

Ann@MSI MINGW64 ~/Git/TRPP_2 (main)
$ git status
On branch main

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
        new file:   File_1.html
        new file:   readme.txt

Ann@MSI MINGW64 ~/Git/TRPP_2 (main)
$ nano readme.txt

Ann@MSI MINGW64 ~/Git/TRPP_2 (main)
$ git add readme.txt

Ann@MSI MINGW64 ~/Git/TRPP_2 (main)
$ git status
On branch main

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
        new file:   File_1.html
        new file:   readme.txt

Ann@MSI MINGW64 ~/Git/TRPP_2 (main)
$ git commit -m "Изменение двух файлов"
[main (root-commit) 22983e1] Изменение двух файлов
 2 files changed, 3 insertions(+)
 create mode 100644 File_1.html
 create mode 100644 readme.txt

Ann@MSI MINGW64 ~/Git/TRPP_2 (main)
$ git status
On branch main
nothing to commit, working tree clean

```

Рисунок 16 – Индексация изменений и коммиты к ним

Создание ветки (Рисунок 17)

```

Ann@MSI MINGW64 ~/Git/TRPP_2 (main)
$ git log
commit 22983e19f8fe0c2367663148b9dc4fa80ca922b6 (HEAD -> main)
Author: Ann <mopsin666@mail.ru>
Date:   Fri Feb 28 14:42:04 2025 +0300

    Изменение двух файлов

Ann@MSI MINGW64 ~/Git/TRPP_2 (main)
$ git merge main
Already up to date.

Ann@MSI MINGW64 ~/Git/TRPP_2 (main)
$

```

Рисунок 17 – Создание ветки

## Выполнение индивидуального задания

Клонирование непустого репозитория и создание ветки (Рисунок 18-19)

```
Ann@MSI MINGW64 ~/Git (master)
$ git clone https://github.com/mopsin666/proect
Cloning into 'proect'...
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
Receiving objects: 100% (3/3), done.

Ann@MSI MINGW64 ~/Git (master)
$ |
```

Рисунок 18 – Клонирование непустого репозитория Git

```
Ann@MSI MINGW64 ~/Git/proect (main)
$ git branch main_2

Ann@MSI MINGW64 ~/Git/proect (main)
$ git branch
* main
  main_2
```

Рисунок 19 – Создание ветки и вывод списка доступных веток

Создание и изменение файлов с коммитами (Рисунок 20-21)

```

$ git checkout main_2
Switched to branch 'main_2'

Ann@MSI MINGW64 ~/Git/proect (main_2)
$ touch first.html

Ann@MSI MINGW64 ~/Git/proect (main_2)
$ touch second.html

Ann@MSI MINGW64 ~/Git/proect (main_2)
$ nano readme.txt

Ann@MSI MINGW64 ~/Git/proect (main_2)
$ git commit -m "Изменение файла readme.txt"
On branch main_2
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)
        first.html
        second.html

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

Ann@MSI MINGW64 ~/Git/proect (main_2)
$ git add .

Ann@MSI MINGW64 ~/Git/proect (main_2)
$ git commit -m "Изменение файла readme.txt"
[main_2 f1408c0] Изменение файла readme.txt
3 files changed, 1 insertion(+)
create mode 100644 first.html
create mode 100644 second.html

Ann@MSI MINGW64 ~/Git/proect (main_2)
$ nano first.html

Ann@MSI MINGW64 ~/Git/proect (main_2)
$ git commit -m "Добавление head"
On branch main_2
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:   first.html

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

```

Рисунок 20 – Создание и изменение файлов с коммитами

```

Ann@MSI MINGW64 ~/Git/proect (main_2)
$ git add .

Ann@MSI MINGW64 ~/Git/proect (main_2)
$ git commit -m "Добавление head"
[main_2 8469733] Добавление head
1 file changed, 4 insertions(+)

Ann@MSI MINGW64 ~/Git/proect (main_2)
$ nano second.html

Ann@MSI MINGW64 ~/Git/proect (main_2)
$ git add .

Ann@MSI MINGW64 ~/Git/proect (main_2)
$ git commit -m "Добавление body"
[main_2 107c445] Добавление body
1 file changed, 4 insertions(+)

Ann@MSI MINGW64 ~/Git/proect (main_2)

```

Рисунок 21 – Создание и изменение файлов с коммитами

Выгрузка в удаленный репозиторий (Рисунок 22)



```

Ann@MSI MINGW64 ~/Git/proect (main_2)
$ git push origin main_2
Enumerating objects: 12, done.
Counting objects: 100% (12/12), done.
Delta compression using up to 12 threads
Compressing objects: 100% (7/7), done.
Writing objects: 100% (10/10), 934 bytes | 934.00 KiB/s, done.
Total 10 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (1/1), done.
remote:
remote: Create a pull request for 'main_2' on GitHub by visiting:
remote:   https://github.com/mopsin666/proect/pull/new/main_2
remote:
To https://github.com/mopsin666/proect
 * [new branch]      main_2 -> main_2

Ann@MSI MINGW64 ~/Git/proect (main_2)
$

```

Рисунок 22 – Выгрузка в удаленный репозиторий

Три изменения в одном коммите (Рисунок 23)

```

Ann@MSI MINGW64 ~/Git/proect (main_2)
$ git rebase -i HEAD~3
[detached HEAD 9038bf5] Изменение файла readme.txt
Date: Fri Mar 14 20:45:59 2025 +0300
3 files changed, 9 insertions(+)
create mode 100644 first.html
create mode 100644 second.html
Successfully rebased and updated refs/heads/main_2.

```

Рисунок 23 – Три изменения в одном коммите

Вывод различия между ветками main и main\_2(Рисунок 24)

```

Ann@MSI MINGW64 ~/Git/proect (main_2)
$ git diff main..main_2
diff --git a/first.html b/first.html
new file mode 100644
index 0000000..8120068
--- /dev/null
+++ b/first.html
@@ -0,0 +1,4 @@
+<html>
+<head>
+</head>
+</html>
diff --git a/readme.txt b/readme.txt
index 9ddb149..557dd40 100644
--- a/readme.txt
+++ b/readme.txt
@@ -1 +1,2 @@
Непустой файл
+Первое изменение ура!!
diff --git a/second.html b/second.html
new file mode 100644
index 0000000..201f283
--- /dev/null
+++ b/second.html
@@ -0,0 +1,4 @@
+<html>
+<body>
+</body>
+</html>

```

Рисунок 24 – Различие между ветками main и main\_2

Сливание двух веток (Рисунок 25)

```
Ann@MSI MINGW64 ~/Git/proect (main_2)
$ git checkout main
Switched to branch 'main'
Your branch is up to date with 'origin/main'.

Ann@MSI MINGW64 ~/Git/proect (main)
$ git merge main_2
Updating 1cb9d95..9038bf5
Fast-forward
 first.html | 4 ++++
 readme.txt | 1 +
 second.html | 4 ++++
 3 files changed, 9 insertions(+)
 create mode 100644 first.html
 create mode 100644 second.html
```

Рисунок 25 – Сливание двух веток

## **Вывод о проделанной работе**

В ходе работы были выполнены основные команды с Git. Получены навыки по работе с командной строкой. Был создан собственный локальный и удаленный git репозиторий, а также они были связаны между собой.