

**МИНИСТЕРСТВО ЦИФРОВОГО РАЗВИТИЯ, СВЯЗИ И МАССОВЫХ
КОММУНИКАЦИЙ РОССИЙСКОЙ ФЕДЕРАЦИИ**

**Ордена Трудового Красного Знамени федеральное государственное бюджетное
образовательное учреждение высшего образования**

«Московский технический университет связи и информатики»

Кафедра “Математическая кибернетика и информационные технологии”

ОТЧЕТ
по лабораторной работе №9
по дисциплине «Введение в информационные технологии»

Тема: «Практическая работа с Git»

Выполнил: студент группы БВТ2505
Коротков Артём Сергеевич

Проверил: Павликов. А.Е.

Москва, 2025

Цель работы:

Ознакомиться с основными возможностями системы управления версиями Git и научиться выполнять основные операции с помощью этой системы.

Задание:

Задание 1:

Установить Git на свой компьютер:

1. Скачайте установочный пакет Git введя в командной строке команду: apt-get install git
2. Запустите установочный пакет и следуйте инструкциям по установке Git на свой компьютер.

Задание 2:

Создать новый репозиторий на GitHub или GitLab:

1. Зайти на свой аккаунт на GitHub или GitLab.
2. Нажать кнопку "Create repository" (Создать репозиторий).
3. Заполнить имя репозитория, описание и выбрать опции создания (например, добавить README файл).
4. Нажать "Create repository" (Создать репозиторий).

Задание 3:

Клонировать репозиторий на локальный компьютер:

1. Скопировать URL репозитория с GitHub или GitLab.
2. Открыть командную строку (или Git Bash) на локальном компьютере.
3. Ввести команду git clone чтобы склонировать репозиторий на локальный компьютер.

Задание 4:

Создать новый файл в репозитории:

1. Перейти в каталог, в который был склонирован репозиторий.
2. Создать новый файл (например, touch new_file.txt для Unix-подобных систем или echo.>new_file.txt для Windows)

Задание 5:

Добавить новый файл в индекс и произвести коммит изменения:

1. В командной строке выполнить команду git add new_file.txt чтобы добавить файл в индекс.
2. Выполнить команду git commit -m "Добавлен новый файл" чтобы закоммитить добавленный файл с комментарием "Добавлен новый файл".

Задание 6:

Клонировать удаленный репозиторий:

1. Найти удаленный репозиторий на GitHub, GitLab или другой платформе.

2. На локальном компьютере выполнить команду `git clone` .
3. Убедиться, что репозиторий успешно склонирован на локальный компьютер.

Задание 7:

Добавить удаленный репозиторий:

1. В командной строке выполнить команду `git remote add origin` для добавления удаленного репозитория в локальный.

Задание 8:

Отправить изменения на удаленный репозиторий:

1. Выполнить команду `git push origin` для отправки изменений из локального репозитория на удаленный.
2. Проверить, что изменения успешно отправлены на удаленный репозиторий.

Задание 9:

Работа с ветками на удаленном репозитории:

1. Создать новую ветку локально с помощью команды `git checkout -b` .
2. Запушить новую ветку на удаленный репозиторий с помощью команды `git push origin` .
3. Удалить ветку из удаленного репозитория с помощью команды `git push origin --delete` .

Задание 10:

Получение изменений из удаленного репозитория:

1. Выполнить команду `git pull origin` для получения изменений из удаленного репозитория на локальный.
2. Убедиться, что изменения успешно применены к локальному репозиторию.

Скриншоты выполнения:

```
MINGW64:/c/Users/tvink/labs
tvink@DESKTOP-EQE6D2K MINGW64 ~
$ git clone https://github.com/Frutin/labs
Cloning into 'labs'...
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.

tvink@DESKTOP-EQE6D2K MINGW64 ~
$ cd labs

tvink@DESKTOP-EQE6D2K MINGW64 ~/labs (main)
$ echo>new_file.txt

tvink@DESKTOP-EQE6D2K MINGW64 ~/labs (main)
$ git add new_file.txt
warning: in the working copy of 'new_file.txt', LF will be replaced by CRLF the
next time Git touches it

tvink@DESKTOP-EQE6D2K MINGW64 ~/labs (main)
$ git commit -m "Добавлен новый файл"
[main 4c516d4] Добавлен новый файл
 1 file changed, 1 insertion(+)
 create mode 100644 new_file.txt
```

```
MINGW64:/c/Users/tvink/rendercv
tvink@DESKTOP-EQE6D2K MINGW64 ~
$ git clone https://github.com/rendercv/rendercv
Cloning into 'rendercv'...
remote: Enumerating objects: 14131, done.
remote: Counting objects: 100% (2172/2172), done.
remote: Compressing objects: 100% (730/730), done.
remote: Total 14131 (delta 1819), reused 1442 (delta 1442), pack-reused 11959 (f
rom 4)
Receiving objects: 100% (14131/14131), 243.43 MiB | 8.56 MiB/s, done.
Resolving deltas: 100% (8815/8815), done.

tvink@DESKTOP-EQE6D2K MINGW64 ~
$ cd rendercv
bash: cd: rendercv: No such file or directory

tvink@DESKTOP-EQE6D2K MINGW64 ~
$ ^C

tvink@DESKTOP-EQE6D2K MINGW64 ~
$ cd ~/rendercv

tvink@DESKTOP-EQE6D2K MINGW64 ~/rendercv (main)
$ ls
Dockerfile README.md examples/ mkdocs.yaml schema.json src/ uv.lock
LICENSE docs/ justfile pyproject.toml scripts/ tests/
```

```
tvink@DESKTOP-EQE6D2K MINGW64 ~/rendercv (main)
$ cd ~/labs

tvink@DESKTOP-EQE6D2K MINGW64 ~/labs (main)
$ ls
README.md new_file.txt
```

```
tvink@DESKTOP-EQE6D2K MINGW64 ~/labs (main)
$ git remote add origin https://github.com/FrutIn/labs
error: remote origin already exists.

tvink@DESKTOP-EQE6D2K MINGW64 ~/labs (main)
$ git remote -v
origin  https://github.com/FrutIn/labs (fetch)
origin  https://github.com/FrutIn/labs (push)

tvink@DESKTOP-EQE6D2K MINGW64 ~/labs (main)
$ git push origin main
info: please complete authentication in your browser...
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), 306 bytes | 306.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/FrutIn/labs
  ece3741..4c516d4  main -> main

tvink@DESKTOP-EQE6D2K MINGW64 ~/labs (main)
$ git checkout -b new_branch
Switched to a new branch 'new_branch'

tvink@DESKTOP-EQE6D2K MINGW64 ~/labs (new_branch)
$ git push origin new_branch
Total 0 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
remote:
remote: Create a pull request for 'new_branch' on GitHub by visiting:
remote:     https://github.com/FrutIn/labs/pull/new/new_branch
remote:
To https://github.com/FrutIn/labs
 * [new branch]      new_branch -> new_branch

tvink@DESKTOP-EQE6D2K MINGW64 ~/labs (new_branch)
$ git push origin --delete new_branch
To https://github.com/FrutIn/labs
 - [deleted]      new_branch

tvink@DESKTOP-EQE6D2K MINGW64 ~/labs (new_branch)
$ git pull origin new_branch
fatal: couldn't find remote ref new_branch

tvink@DESKTOP-EQE6D2K MINGW64 ~/labs (new_branch)
$ ls
README.md  new_file.txt

tvink@DESKTOP-EQE6D2K MINGW64 ~/labs (new_branch)
$ git pull origin main
From https://github.com/FrutIn/labs
 * branch            main      -> FETCH_HEAD
Already up to date.
```

Заключение

В ходе выполнения лабораторной работы были успешно решены следующие задачи:

1. Ознакомился с основными возможностями системы управления версиями Git.
2. Научился выполнять основные операции с помощью этой системы.