

# MODUL GIT COMMANDS

## 1. Pengenalan Git



Gambar 1. Git

**Git** adalah sebuah sistem kontrol versi yang digunakan untuk mengelola perubahan pada kode sumber atau berkas-berkas proyek. Dengan Git, kita dapat melacak perubahan, berkolaborasi dengan tim, dan mengelola proyek perangkat lunak dengan lebih efisien.

## 2. Fungsi Git

Git memiliki beberapa **fungsi utama**, antara lain:

- a. **Melacak Perubahan:** Git dapat melacak setiap perubahan yang terjadi pada berkas-berkas dalam proyek Anda.
- b. **Kolaborasi Tim:** Git memungkinkan tim bekerja sama pada proyek yang sama, memungkinkan mereka untuk menggabungkan perubahan dengan mudah.
- c. **Mengelola Versi:** Git memungkinkan Anda untuk mengelola berbagai versi perangkat lunak dan mengembalikan proyek ke versi sebelumnya jika diperlukan.
- d. **Cabang (Branching):** Anda dapat membuat cabang untuk mengembangkan fitur baru atau memperbaiki bug tanpa mempengaruhi versi utama proyek.
- e. **Distribusi:** Git memungkinkan proyek didistribusikan ke berbagai tempat secara efisien.

## 3. Perintah di Git

Berikut adalah beberapa perintah yang sering digunakan dalam Git:

- a. **git init:** Memulai repositori Git baru.
- b. **git clone:** Mengunduh repositori Git yang ada ke komputer Anda.
- c. **git add:** Menambahkan perubahan ke indeks (staging area) untuk siap di-commit.
- d. **git commit:** Merekam perubahan yang telah ditambahkan ke indeks.
- e. **git pull:** Mengambil perubahan terbaru dari repositori jarak jauh (remote).
- f. **git push:** Mengirim perubahan yang telah dicommit ke repositori jarak jauh.
- g. **git branch:** Menampilkan daftar cabang (branches) yang ada.
- h. **git checkout:** Beralih antara cabang atau commit yang berbeda.
- i. **git merge:** Menggabungkan perubahan dari cabang lain ke cabang saat ini.
- j. **git status:** Menampilkan status perubahan dalam repositori.
- k. **git log:** Menampilkan riwayat commit dalam repositori.

#### 4. Perbedaan Git, Github, dan Gitlab

##### a. Git

adalah sistem kontrol versi yang dikembangkan oleh Linus Torvalds. Ini adalah perangkat lunak open-source yang digunakan untuk melacak perubahan pada kode sumber atau berkas-berkas dalam proyek perangkat lunak.



*Gambar 2. Git*

- Git berfungsi secara lokal pada komputer pengembang, yang berarti Anda dapat menggunakan Git untuk mengelola versi kode sumber pada komputer Anda sendiri tanpa perlu terhubung ke internet atau server jarak jauh.
- Git membantu pengembang dalam melacak perubahan, menciptakan cabang untuk pengembangan fitur atau perbaikan bug, dan menggabungkan perubahan dari berbagai kontributor.

##### b. GitHub

adalah platform kolaborasi dan hosting repositori Git yang populer.



*Gambar 3. Github*

- GitHub menyediakan layanan hosting gratis untuk repositori Git dan menyediakan antarmuka web yang memudahkan kolaborasi tim dan berbagi kode sumber.
- Pengembang dapat mengunggah repositori Git mereka ke GitHub, berkolaborasi dengan anggota tim, mengelola masalah (issues), dan menjalankan alur kerja (workflow) pengembangan perangkat lunak yang terstruktur.
- GitHub juga memiliki berbagai fitur tambahan seperti wiki, proyek, dan alat-alat kolaborasi yang memudahkan pengembangan perangkat lunak dalam skala besar.

##### c. GitLab

adalah platform serupa dengan GitHub yang juga digunakan untuk kolaborasi dan hosting repositori Git. Namun, GitLab memiliki versi self-hosted yang memungkinkan Anda menjalankannya di server sendiri atau cloud Anda sendiri.



*Gambar 4. GitLab*

- GitLab menyediakan sejumlah fitur seperti manajemen siklus hidup, pelaporan masalah (issue tracking), alat pengujian otomatis (CI/CD), dan banyak lagi.
- GitLab tersedia dalam versi komunitas (GitLab Community Edition) yang gratis dan versi berbayar dengan fitur tambahan (GitLab Enterprise Edition).

## 5. Contoh Membuat file dan Upload File Menggunakan Git

### a. Download dan Install Git

Git dapat didownload pada link berikut:

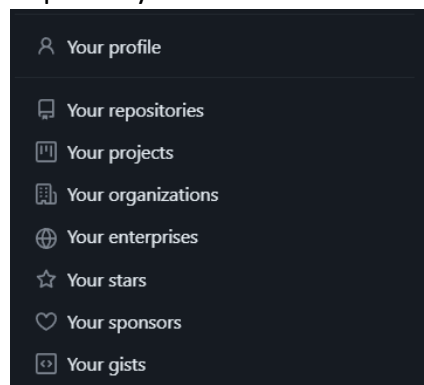
[Git - Downloads \(git-scm.com\)](https://git-scm.com/downloads)

### b. Buatlah akun Github

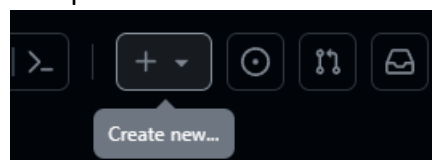
Silahkan buat akun Github di link berikut:

<https://github.com/>

Setelah akun terbuat, cobalah membuat Repositori Baru di bagian “Your Repository”



Lalu pilih “Create new”



Lalu isikan sesuai dengan project yang dibuat

## Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

*Required fields are marked with an asterisk (\*).*

Repository template

No template ▾

Start your repository with a template repository's contents.

Owner \*

nadeap ▾

/

Repository name \*


belajarDL

⚠ The repository belajarDL already exists on this account.


Great repository names are short and memorable. Need inspiration? How about **turbo-tribble** ?

Description (optional)

isikan bebas sesuai deskripsi project

☒  **Public**

Anyone on the internet can see this repository. You choose who can commit.

☐  **Private**

You choose who can see and commit to this repository.

Initialize this repository with:

☐ **Add a README file**

This is where you can write a long description for your project. [Learn more about READMEs.](#)

Add .gitignore

.gitignore template: None ▾

Choose which files not to track from a list of templates. [Learn more about ignoring files.](#)

Choose a license

License: None ▾

Kosongkan bagian **README file**, karena akan dicoba untuk membuat file README menggunakan Git.

**\*Hiraukan tulisan merah**

Lalu pilih **Create Repository**

Initialize this repository with:

☐ **Add a README file**

This is where you can write a long description for your project. [Learn more about READMEs.](#)

Add .gitignore


.gitignore template: None ▾

Choose which files not to track from a list of templates. [Learn more about ignoring files.](#)

Choose a license

License: None ▾

A license tells others what they can and can't do with your code. [Learn more about licenses.](#)

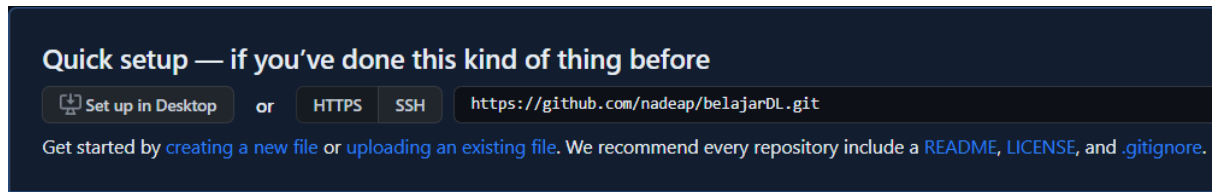
 You are creating a public repository in your personal account.

Create repository

Modul Git 2023

Praktikum Deep Learning

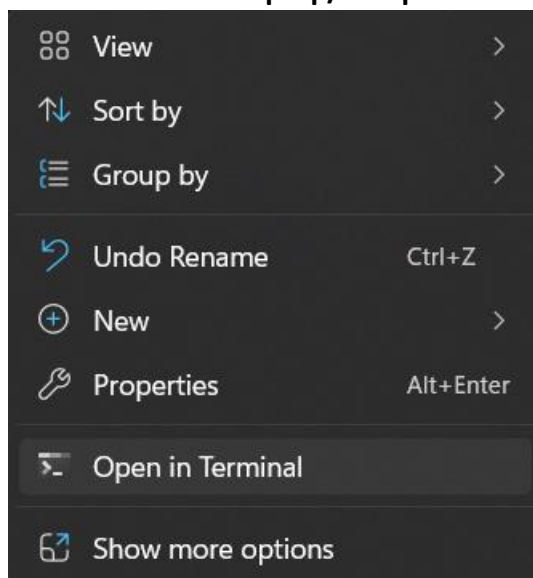
c. Akan Muncul Gambar Berikut:



d. Muncul Seperti Gambar Berikut:



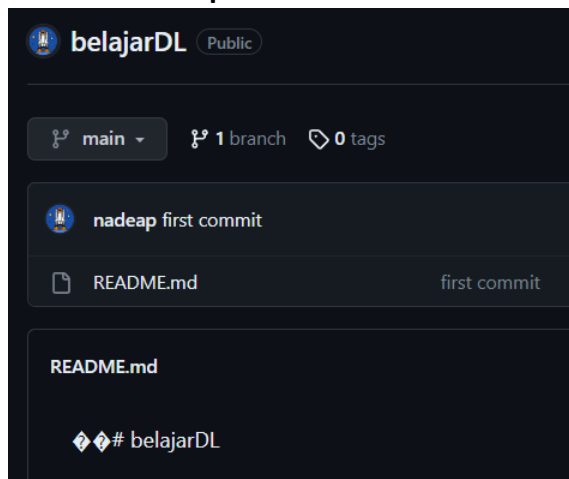
e. Buka Terminal di Laptop/Komputer



f. Copy Code ke Terminal

```
PS C:\Users\HP\OneDrive - ypt.or.id\0 SEMESTER 7\DeepLearning\belajarGIT> echo "# belajarDL" >> README.md
PS C:\Users\HP\OneDrive - ypt.or.id\0 SEMESTER 7\DeepLearning\belajarGIT> git init
Reinitialized existing Git repository in C:/Users/HP/OneDrive - ypt.or.id/0 SEMESTER 7/DeepLearning/belajarGIT/.git/
PS C:\Users\HP\OneDrive - ypt.or.id\0 SEMESTER 7\DeepLearning\belajarGIT> git add README.md
PS C:\Users\HP\OneDrive - ypt.or.id\0 SEMESTER 7\DeepLearning\belajarGIT> git commit -m "first commit"
[master (root-commit) 1a621e8] first commit
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 README.md
PS C:\Users\HP\OneDrive - ypt.or.id\0 SEMESTER 7\DeepLearning\belajarGIT> git branch -M main
PS C:\Users\HP\OneDrive - ypt.or.id\0 SEMESTER 7\DeepLearning\belajarGIT> git remote add origin https://github.com/nadeap/belajarDL.git
PS C:\Users\HP\OneDrive - ypt.or.id\0 SEMESTER 7\DeepLearning\belajarGIT> git push -u origin main
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Writing objects: 100% (3/3), 234 bytes | 117.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/nadeap/belajarDL.git
 * [new branch]      main -> main
branch 'main' set up to track 'origin/main'.
PS C:\Users\HP\OneDrive - ypt.or.id\0 SEMESTER 7\DeepLearning\belajarGIT> |
```

### g. File Sudah Terupload



### h. Tambahkan file baru dari lokal

```
PS C:\Users\HP\OneDrive - ypt.or.id\0 SEMESTER 7\DeepLearning\belajarGIT> git add Modul1-ANN.ipynb
PS C:\Users\HP\OneDrive - ypt.or.id\0 SEMESTER 7\DeepLearning\belajarGIT> git commit -m "Menambahkan file Modul1-ANN.ipynb"
[main b408f00] Menambahkan file Modul1-ANN.ipynb
 1 file changed, 1302 insertions(+)
 create mode 100644 Modul1-ANN.ipynb
PS C:\Users\HP\OneDrive - ypt.or.id\0 SEMESTER 7\DeepLearning\belajarGIT> git push -u origin main
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 4 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 472.67 KiB | 6.47 MiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/nadeap/belajarDL.git
 1a621e8..b408f00 main -> main
branch 'main' set up to track 'origin/main'.
```

### i. File sudah terupload

