

Git & Version Control System

Hello, everyone!



Sebelum kita memulai kelas, kita awali dengan:

1. Berdoa
2. siapkan diri
3. pretest





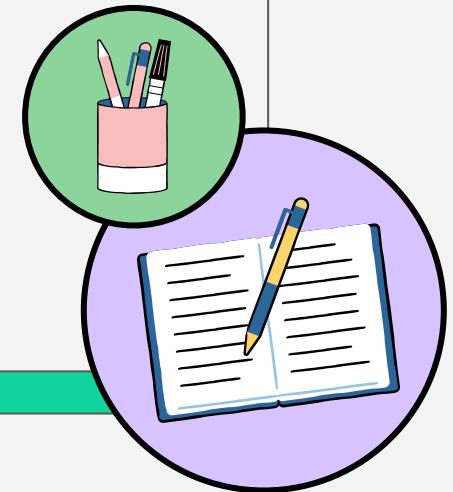
Pertemuan lalu...

Let's Recall

pertemuan sebelumnya kita sudah membahas lanjutan command SQL. Ada yang inget apa aja?

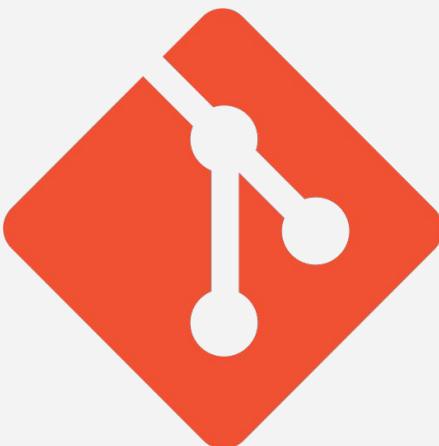
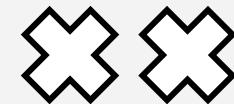
Target

1. Version Control System
2. Git
3. Github





Tools



git



Version Control System (VCS)

Seperti yang dijelaskan di dokumentasi git (<https://git-scm.com/book/id/v2/Memulai-Tentang-Version-Control>).

Version control adalah sebuah sistem yang merekam perubahan-perubahan dari sebuah berkas atau sekumpulan berkas dari waktu ke waktu sehingga kita dapat menilik kembali versi khusus suatu saat nanti. Lalu apa bedanya dengan dropbox dan google drive?

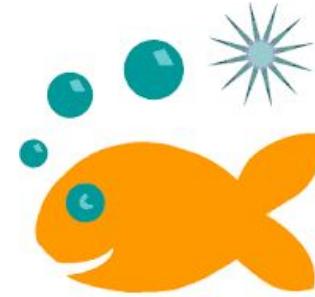
2 software ini juga bisa untuk berkolaborasi. Tapi tidak seflexibel Ketika kita menggunakan version control untuk source code.

1. Sebuah system yang menyimpan rekaman/snapshot perubahan source code
2. Memungkinkan bekerja berkolaborasi dengan lebih baik
3. Mengetahui siapa yang melakukan dan kapan sebuah perubahan terjadi
4. Memungkinkan kita untuk Kembali ke keadaan sebelum perubahan (checkout)

Tanpa VCS

Name
 skripsi_siap
 skripsi_revisi1
 skripsi_revisi2
 skripsi_revisi3
 skripsi_ceksiapfinal
 skripsi_ceksiapfinalrevisi1

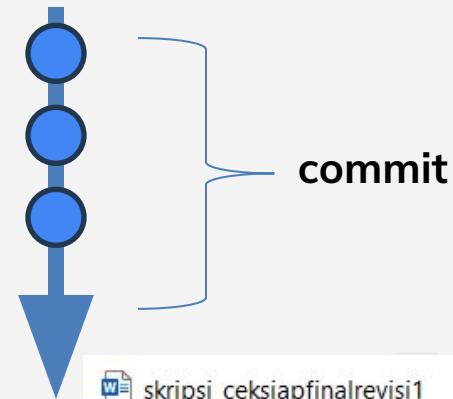
Jenis-jenis Version Control System



GIT

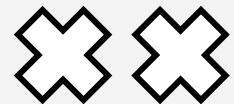
- Git adalah salah satu sistem pengontrol versi (*Version Control System*) yang dapat digunakan untuk mencatat setiap perubahan pada file proyek yang dikerjakan oleh banyak orang maupun sendiri. GIT juga memiliki sifat terdistribusi, sehingga memungkinkan kolaborasi antar banyak pengembang menjadi lebih mudah dengan fitur-fitur bawaan seperti *branch*, *fork*, dan *merging*
- Riwayat perubahan file disimpan menggunakan serangkaian commit
- Penanda untuk setiap commit dinamakan hash atau string.
- Aktor yang melakukan commit.
- Tanggal commit
- Pada saat membuat sebuah repo dan sudah melakukan commit akan tersimpan titik/commit. Contoh pada gambar disamping adalah sudah dilakukan commit sebanyak 3
- Bisa membuat cabang/branch agar tidak mengganggu commit utama
- Menggabungkan branch dinamakan merge.
- Ini bisa dilakukan di lokal

Dengan VCS

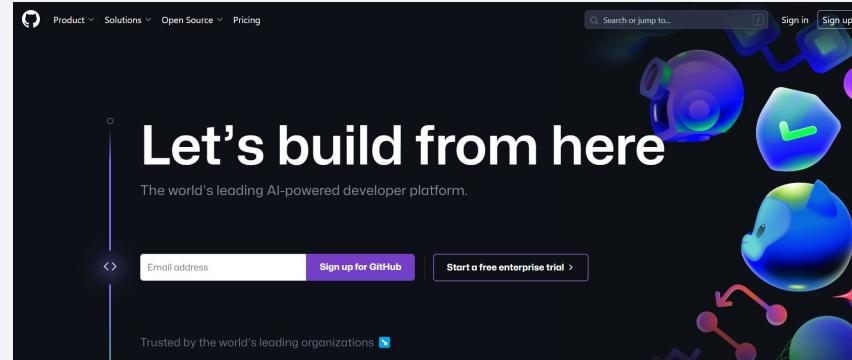




Github

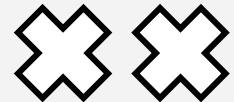


1. Layanan cloud untuk menyimpan dan mengelola project/repo git
2. Bisa membuat repo, membuat commit, branch
3. Dilakukan secara online
4. Website yang didalamnya menggunakan GIT
5. Dapat menggabungkan GIT di local dan GITHUB
6. Mengirimkan source code dari GIT ke github (push)
7. Mengambil source code dari github ke GIT (pull)
8. Push dan pull adalah commit.

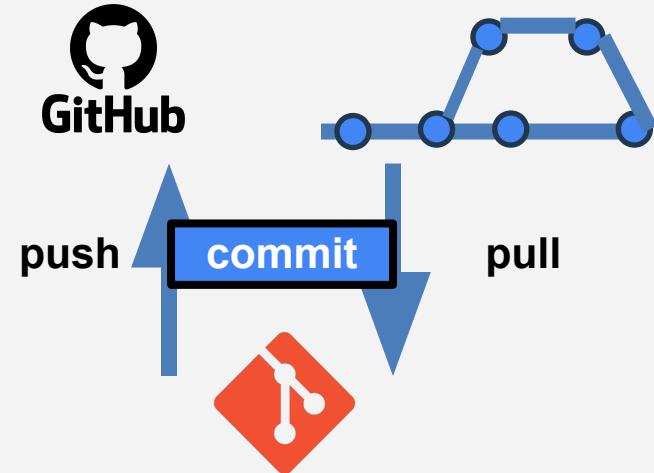
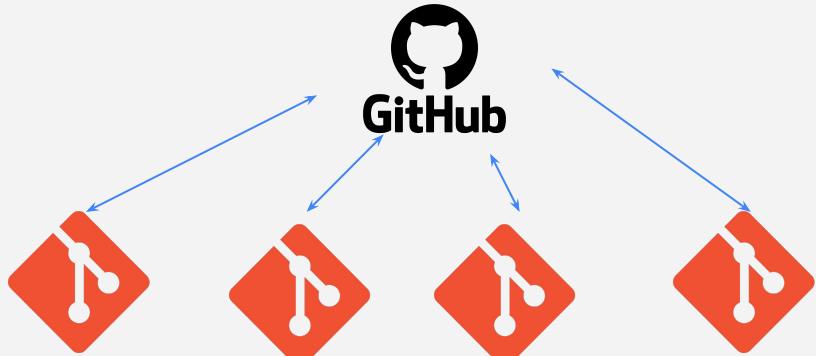


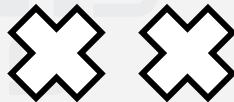


Langkah Git to Github

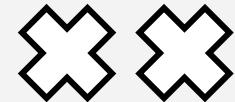


1. Jadikan Github sebagai remote yaitu sumber dari repo. Kita membuat repo di github
2. Repo di clone ke local (git), sehingga di local computer kita juga ada. Sehingga repo di git dan github nyambung.





RECAP



- Version Control System

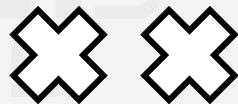
Sistem yang menyimpan dan mengelola rekaman perubahan dari source code

- Git

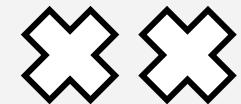
Salah satu software yang bisa melakukan VCS

- Github

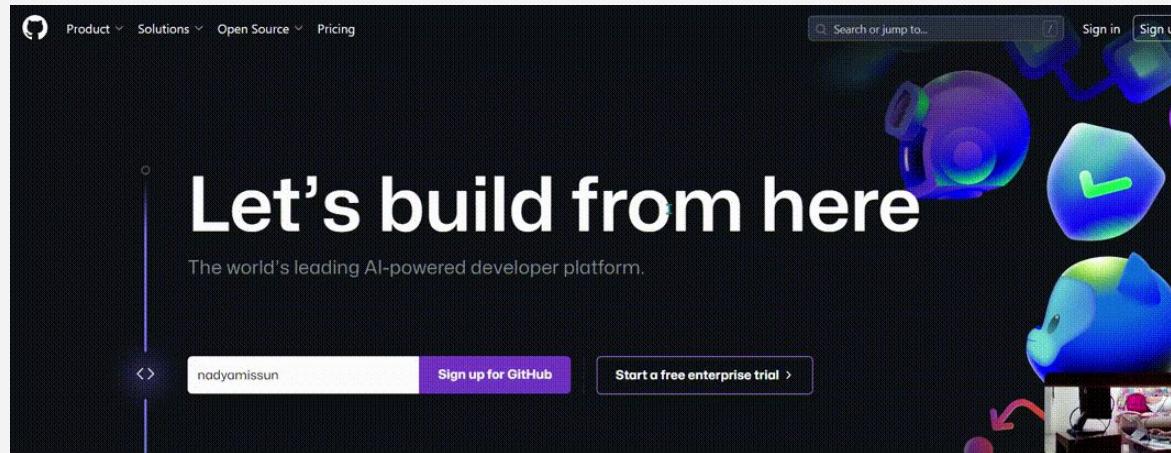
Website/layanan untuk mengelola project GIT



Create Account Github



<https://github.com/>

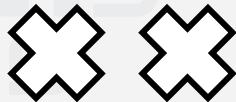


Explor Github

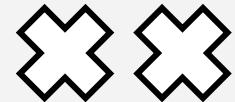
The screenshot shows the GitHub homepage with a blue header bar containing the title "Explor Github". The main content area displays the following sections:

- Top Repositories**: A list of repositories by user "nadyamissun", including "SVM", "ROTATION-FOREST", "Data-Visualisasi", "BayesianFMM-R2WinBUGS", "Regresi-Linear-Sederhana", and "KIMIAINFORMATIKA-KNIME".
- Recent activity**: A placeholder message stating, "When you take actions across GitHub, we'll provide links to that activity here."
- Home**: An informational box titled "Updates to your homepage feed" explaining the merge of the Following and For you feeds. It includes a "Learn more" link.
- Trending repositories**: A list featuring "allenai/OLMo" (Python, 1.4k stars) and "maybe-finance/maybe" (Ruby, 14.3k stars).
- Latest changes**: A list of recent commits:
 - 3 days ago: Gradle starter workflows now automatically submit transitive dependencies
 - 5 days ago: Code faster and better with GitHub Copilot's new features in Visual Studio
 - 5 days ago: GitHub Actions: macOS 14 (Sonoma) is now available
 - 5 days ago: GitHub Actions: Introducing the new M1 macOS runner available to open source!A "View changelog" link is present.
- Explore repositories**: A section showing "cri-o / cri-o" (Go, 4.9k stars) and "akubot / akubot".

Edit profile



Explor Github



The screenshot shows the GitHub homepage with the following elements:

- Top Repositories:** A sidebar on the left lists repositories belonging to the user "nadyamissun".
 - nadyamissun/nama_-_project
 - nadyamissun/SVM
 - nadyamissun/ROTATION-FOREST
 - nadyamissun/Data-Visualisasi
 - nadyamissun/BayesianFMM-R2WinBUGS
 - nadyamissun/Regresi-Linear-Sederhana
 - nadyamissun/KIMIAINFORMATIKA-KNIME
- Recent activity:** A section showing recent actions across GitHub.
- Home:** The main content area.
 - Updates to your homepage feed:** A message explaining the combination of Following and For you feeds, with a "Learn more" link.
 - Trending repositories:** A list of trending repos:
 - allenai/OLMo**: Modeling, training, eval, and inference code for OLMo. (Python, 1.4k stars)
 - maybe-finance/maybe**: The OS for your personal finances. (Ruby, 14.4k stars)
 - Latest changes:** A sidebar listing recent updates from GitHub, such as Gradle starter workflows now automatically submit transitive dependencies and GitHub Actions: macOS 14 (Sonoma) is now available.
 - Explore repositories:** A section for exploring new repositories, featuring "whatwg / html".

Create a new repo

Explor Github

The image shows a screenshot of a GitHub repository page for 'machine_learning' and a code editor window.

Repository Page:

- Header:** nadyamissun / machine_learning
- Code tab:** Code, Issues, Pull requests, Actions, Projects, Wiki, Security, Insights, Settings
- Branches:** main (selected), 1 Branch, 0 Tags
- Commits:** nadyamissun Initial commit, 50691bd - 8 minutes ago, 1 Commits
- Files:** README.md (Initial commit, 8 minutes ago), README
- About:** Repository untuk belajar machine learning, Readme, Activity, 0 stars, 1 watching, 0 forks
- Releases:** No releases published, Create a new release
- Packages:** No packages published, Publish your first package

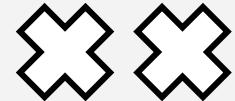
Code Editor:

- Code 55% faster with GitHub Copilot
- Import n
- Use: Control + Shift + m to toggle the tab key moving focus. Alternatively, use: esc then tab to move to the next interactive element on the page.

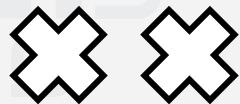
Create a new file



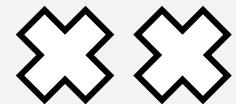
Istilah GIT



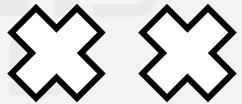
1. Repo: folder project
2. Commit: rekaman/snapshot dari repo
3. Hash: penanda unik pada sebuah commit
4. Checkout: berpindah ke sebuah commit
5. Branch: cabang bebas dari sebuah commit
6. Merge: menggabungkan branch
7. Remote: sumber yang memiliki repo
8. Clone: mengambil repo dari remote
9. Push: mengirimkan commit ke repo
10. Pull: Mengambil commit dari repo



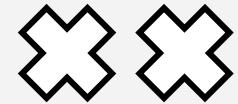
Instal GIT



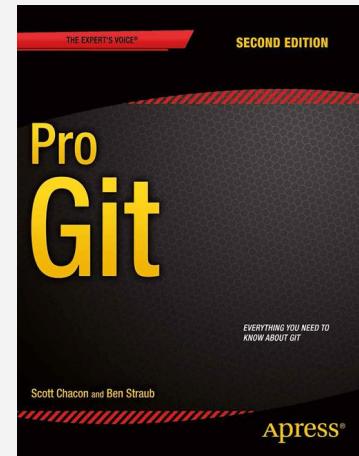
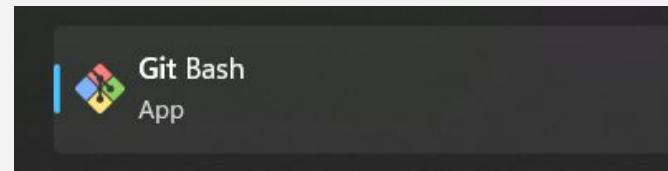
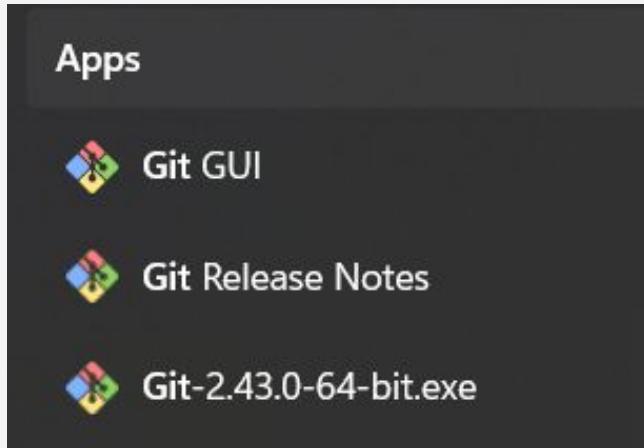
<https://git-scm.com/>



Panduan GIT

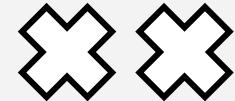


<https://git-scm.com/book/id/v2>





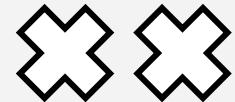
Command GIT



command	fungsi
git init	Untuk memulai repositori
Git add <file(s)>	Untuk menambahkan file ke area staging
Git status	Untuk mengetahui status repo
Git commit	Untuk membuat commit atau mengambil snapshot file secara permanen dalam Riwayat versi
Git config	Untuk memasukan configurasi kedalam git
Git branch	Untuk membuat branch
Git help	Untuk mengetahui sebuah perintah secara cepat

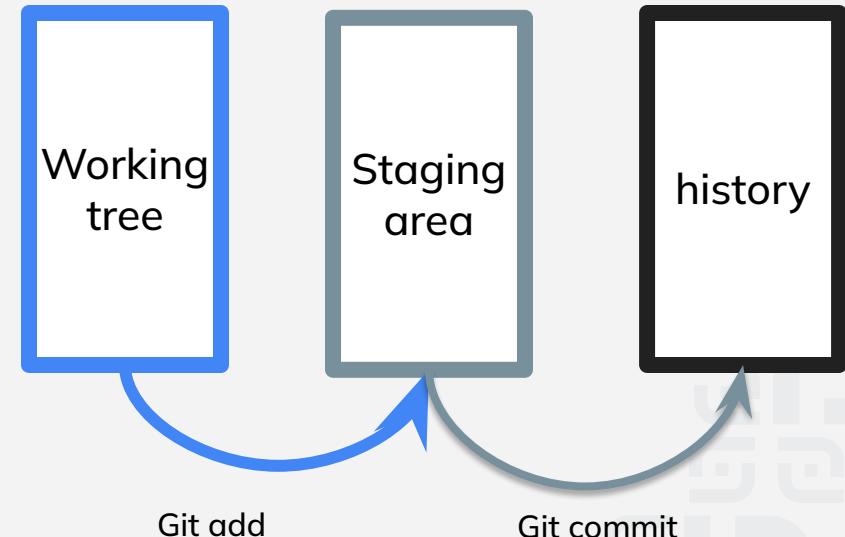


Area Repo GIT



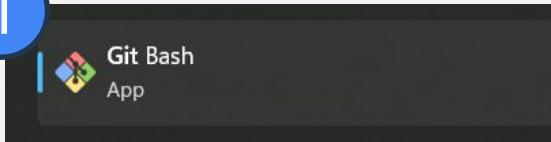
- Working tree
- Staging area
- History

.git



Open GIT Bash

1



Directory aktif

2

```
Nadya Asanul Husna@LAPTOP-PAIDBRTM MINGW64 ~ (master)
$ |
```

A screenshot of a Git Bash terminal window. The title bar shows the path "MINGW64\c\Users\Nadya Asanul Husna". The command prompt shows the user's name, computer name, and current branch "(master)". The cursor is at the end of the line.

3

```
Nadya Asanul Husna@LAPTOP-PAIDBRTM MINGW64 ~ (master)
$ pwd
/c/Users/Nadya Asanul Husna

Nadya Asanul Husna@LAPTOP-PAIDBRTM MINGW64 ~ (master)
$ cd "E:\Great Edu\project_test"

Nadya Asanul Husna@LAPTOP-PAIDBRTM MINGW64 /e/Great Edu/project_test (main)
$ |
```

A screenshot of a Git Bash terminal window. The title bar shows the path "MINGW64\c\Users\Nadya Asanul Husna". The command prompt shows the user's name, computer name, and current branch "(main)". The cursor is at the end of the line. A red box highlights the command "cd "E:\Great Edu\project_test"" and its output.

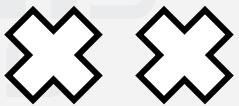
Open GIT Bash

```
MINGW64:/e/Great Edu/project_test
Nadya Asanul Husna@LAPTOP-PAIDBRTM MINGW64 /e/Great Edu
$ ls
'Proposal Data Science.pdf'
'Proposal MSIB6 - Expert.pdf'
'silabus Data Science SIB Cycle 6 - Expert - B.pdf'
'silabus Data Science SIB Cycle 6 - Expert - syllabus v1.pdf'
'bahan ajar'
linky.txt
materi/
oret.pptx
project_test/
skripsi/
supermarket_sales.csv

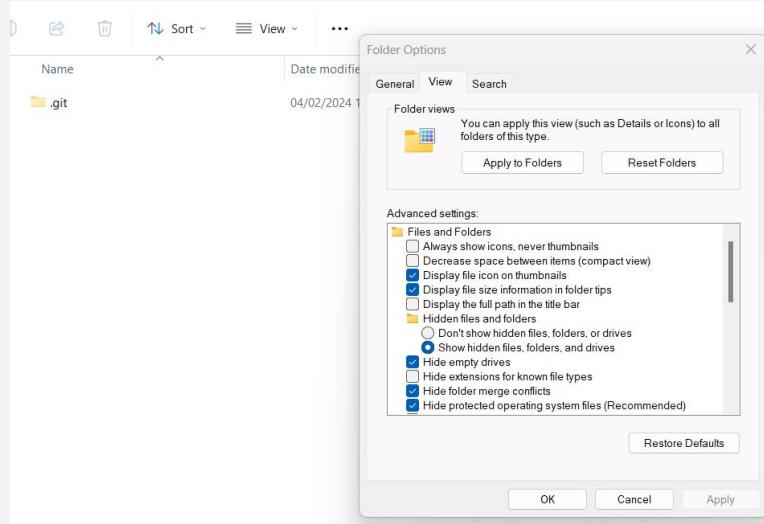
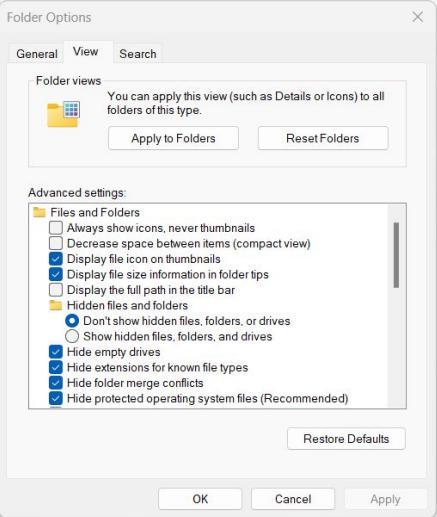
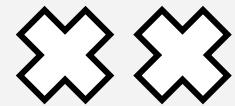
Nadya Asanul Husna@LAPTOP-PAIDBRTM MINGW64 /e/Great Edu
$ cd project_test

Nadya Asanul Husna@LAPTOP-PAIDBRTM MINGW64 /e/Great Edu/project_test (main)
$ git init
Reinitialized existing Git repository in E:/Great Edu/project_test/.git/
Nadya Asanul Husna@LAPTOP-PAIDBRTM MINGW64 /e/Great Edu/project_test (main)
$
```

1. Cek lokasi default
2. Buatlah folder dengan nama great_edu di local
3. Kita akan menggunakan lokasi great_edu
4. Gunakan perintah ls untuk melihat folder atau data yang tersedia. Warna biru adalah folder
5. Buatlah folder project_test didalam great_edu
6. Lalu kita masuk ke project_test dan folder ini baru folder biasa belum repo git
7. Jadikan folder tersebut menjadi repo git.
Caranya git init
8. maka sekarang folder tersebut sudah diawasi oleh git
9. Sekarang kita sudah bisa menyebut repo project_test



Open GIT Bash



Secara sekilas folder project_test hanyalah folder biasa saja. Tapi jika kita ubah pengaturan view folder, didalam folder tersebut terdapat hidden folder .git artinya folder tersebut sudah diawasi oleh git

Open GIT Bash

```
MINGW64:/e/Great Edu/project_test
Nadya Asanul Husna@LAPTOP-PAIDBRTM MINGW64 ~ (master)
$ cd "E:\Great Edu\project_test"

Nadya Asanul Husna@LAPTOP-PAIDBRTM MINGW64 /e/Great Edu/project_test (main)
$ ls
hello.py

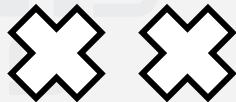
Nadya Asanul Husna@LAPTOP-PAIDBRTM MINGW64 /e/Great Edu/project_test (main)
$ git status
On branch main 1
No commits yet 2
Untracked files:
  (use "git add <file>..." to include in what will be committed)
    hello.py 3
nothing added to commit but untracked files present (use "git add" to track)

Nadya Asanul Husna@LAPTOP-PAIDBRTM MINGW64 /e/Great Edu/project_test (main)
$ |
```

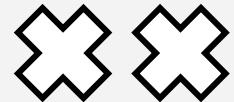
Git status: mengecek perubahan file baik file baru atau ada perubahan pada file.

Informasi:

1. On branch master: kita berada di branch master
2. No commits yet: kita belum melakukan commit
3. Untracked files: ada file yang belum di track. Nama file adalah hello.py File hello.py harus disimpan didalam staging area agar dapat di commit. Caranya dengan ketik git add <nama file>



Open GIT Bash



```
Nadya Asanul Husna@LAPTOP-PAIDBRTM MINGW64 /e/Great Edu/project_test (main)
$ git add hello.py 4
```

```
Nadya Asanul 5 Husna@LAPTOP-PAIDBRTM MINGW64 /e/Great Edu/project_test (main)
$ git status
On branch main 6
```

No commits yet

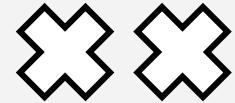
```
Changes to be committed: 7
(use "git rm --cached <file>..." to unstage) 8
  new file:  hello.py
```

```
Nadya Asanul Husna@LAPTOP-PAIDBRTM MINGW64 /e/Great Edu/project_test (main)
```

4. Git add hello.py
5. Lalu kita cek lagi menggunakan git status
6. On branch master: kita berada di branch master
7. Changes to be commited: Sudah ada file didalam staging area dengan nama file hello.py dan siap untuk di commit
8. Jika ingin mengeluarkan lagi dari staging area maka ketik git rm --cached <file>
Sekarang file sudah ada di staging area dan siap untuk di commit. Setiap perubahan di source code tidak selesai dengan save aja. File wajib dilakukan commit.



Open GIT Bash



9. Jika kita langsung ketik git commit maka akan tertampil seperti gambar disamping. Cara keluar dari tampilan tersebut adalah kita klik esc dan :q! lalu enter. Setelah itu akan Kembali ke tampilan awal

```
MINGW64/c/Great Edu/project_test

Nadya Asanul Husna@LAPTOP-PAIDBRTM MINGW64 ~ (master)
$ cd "E:\Great Edu\project_test"

Nadya Asanul Husna@LAPTOP-PAIDBRTM MINGW64 /e/Great Edu/project_test (main)
$ git status
On branch main

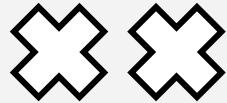
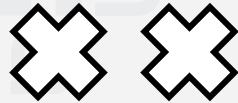
No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
    new file:   hello.py

Nadya Asanul Husna@LAPTOP-PAIDBRTM MINGW64 /e/Great Edu/project_test (main)
$ git commit
Aborting commit due to empty commit message.

Nadya Asanul Husna@LAPTOP-PAIDBRTM MINGW64 /e/Great Edu/project_test (main)
$ |
```

Open GIT Bash



```
Nadya Asanul Husna@LAPTOP-PAIDBRTM MINGW64 /e/Great Edu/project_test (main)
$ git commit -m "menambahkan file hello.py"
[main (root-commit) ea9137b] menambahkan file hello.py
 1 file changed, 1 insertion(+)
 create mode 100644 hello.py
```

10

```
Nadya Asanul Husna@LAPTOP-PAIDBRTM MINGW64 /e/Great Edu/project_test (main)
$ git status
On branch main
nothing to commit, working tree clean
```

11

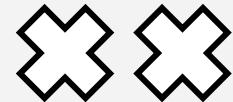
10. Git commit -m “pesan”
Artinya ada commit ke branch master ini commit hash nya yaitu ea9137b dan ini pesannya menambahkan file hello.py
Ada 1 file berubah dan 1 baris yang ditambahkan file tersebut. Jadi sekarang git sudah melacak perubahannya.
11. Jika kita cek git status maka akan ada tulisan nothing to commit artinya kita sudah melakukan commit agar commit tersimpan di story area.

Command GIT

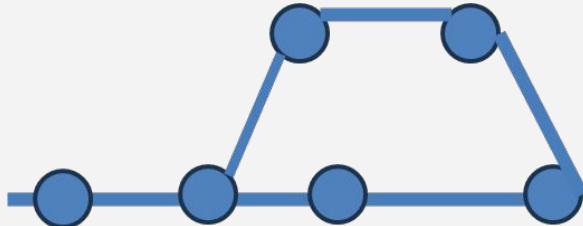
command	fungsi
Pwd (print working directory)	Untuk mengetahui directory active
Cd (change directory)	Untuk mengubah directory
ls	Untuk mengetahui folder di directory
clear	Menghapus script di cmd



GIT Branch

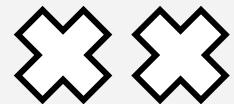


- Branch saat kita membuat cabang bebas dari commit yang kita buat. Ini dilakukan ketika kita menambahkan fitur baru yang kita belum yakin implementasi atau Ketika mengerjakan project secara berkelompok.
- Merge: menggabungkan cabang.





Implementasi GIT Branch

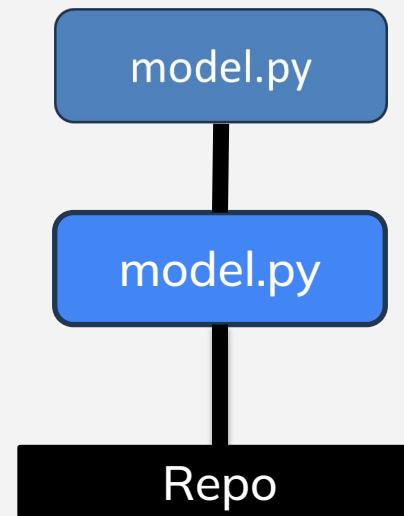


```
Nadya Asanul Husna@LAPTOP-PAIDBRTM MINGW64 /e/Great Edu/project_test (main)
$ git branch
* main 1

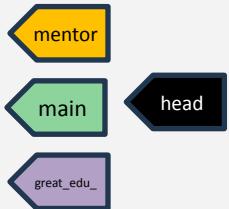
Nadya Asanul Husna@LAPTOP-PAIDBRTM MINGW64 /e/Great Edu/project_test (main)
$ git branch great_edu_
* main 2

Nadya Asanul Husna@LAPTOP-PAIDBRTM MINGW64 /e/Great Edu/project_test (main)
$ git branch mentor 3

Nadya Asanul Husna@LAPTOP-PAIDBRTM MINGW64 /e/Great Edu/project_test (main)
$ git branch
great_edu_
* main 4
mentor
```



Commit:wt6571

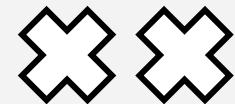


Commit: 10n765

1. Ketika kita menulis git branch maka hanya memunculkan branch yang sudah ada. Jika ada warna hijau dan Bintang artinya branch itu aktif
2. Git branch great_edu_ artinya git akan membuat snapshot atau menduplikat branch di commit yang sama
3. Git branch mentor artinya git akan membuat snapshot atau menduplikat branch di commit yang sama
4. Ketika di cek git branch maka kita sudah memiliki 3 branch. Warna hijau artinya branch aktif



Implementasi GIT Branch



```
Nadya Asanul Husna@LAPTOP-PAIDBRTM MINGW64 /e/Great Edu/project_test (main)
$ git log
commit ea9137bb9f754104a526a50427adeecc3403bfce (HEAD -> main, mentor, great_edu)
Author: id.17.nadyasanulhusna@gmail.com <id.17.nadyasanulhusna@gmail.com>
Date:   Sun Feb 4 14:43:27 2024 +0700
        menambahkan file hello.py
```

1

2

```
Nadya Asanul Husna@LAPTOP-PAIDBRTM MINGW64 /e/Great Edu/project_test (main)
$ git log --all --decorate --oneline --graph
* ea9137b (HEAD -> main, mentor, great_edu) menambahkan file hello.py
```

3

```
Nadya Asanul Husna@LAPTOP-PAIDBRTM MINGW64 /e/Great Edu/project_test (main)
$ alias graph="git log --all --decorate --oneline --graph"
```

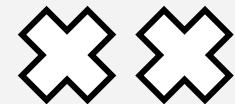
4

```
Nadya Asanul Husna@LAPTOP-PAIDBRTM MINGW64 /e/Great Edu/project_test (main)
$ graph
* ea9137b (HEAD -> main, mentor, great_edu) menambahkan file hello.py
```

1. Kita bisa melihat perjalanan branch kita dengan graph menggunakan git log
2. Output git log adalah history yang sudah dilakukan (HEAD -> main, mentor, Great_edu)
3. Git log --all --decorate --oneline --graph : Menampilkan visualisasi branch dalam bentuk graph
4. Kita bisa melakukan alias pada kolom 3 agar dikemudian mau mengecek branch kita cukup memanggil branch



Connect Git Bash dan Github



Connect to GitHub

GitHub
Sign in

Browser/Device Token

[Sign in with your browser](#)

[Sign in with a code](#)

Don't have an account? [Sign up](#)

Authorize Git Credential Manager



Git Credential Manager by [Git Ecosystem](#)

[Cancel](#)

[Authorize git-ecosystem](#)

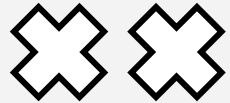
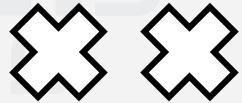


Authentication Succeeded

You may now close this tab and return to the application.

Ketika akan push file dari lokal git bash ke github maka kita harus sign in akun github terlebih dahulu

Implementasi GIT Clone



1. Perbedaan clone dengan download adalah kalau clone kita bisa terhubung dengan github dan kalau download saja hanya dapat folder nya dan tidak terhubung
2. Tampilan folder tutorial_greate sudah ada di folder great_edu

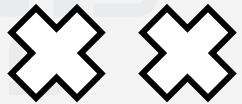
```
MINGW64/e/Great Edu/tutorial_greate
Nadya Asanul Husna@LAPTOP-PAIDBRM MINGW64 ~ (master)
$ cd "E:\Great Edu"

Nadya Asanul Husna@LAPTOP-PAIDBRM MINGW64 /e/Great Edu
$ git clone https://github.com/nadyamissun/tutorial_greate.git
Cloning into 'tutorial_greate'...
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (3/3), done.
```

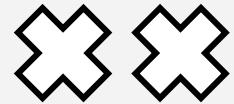
```
Nadya Asanul Husna@LAPTOP-PAIDBRM MINGW64 /e/Great Edu
$ ls
'Proposal Data Science.pdf'
'Proposal MSIB6 - Expert.pdf'
'Silabus Data Science SIB Cycle 6 - Expert - B.pdf'
'Silabus Data Science SIB Cycle 6 - Expert - Syllabus v1.pdf'
'bahan ajar'
'linky.txt'
materi/
oret.pptx
pertemuan/
pippy.png
project_test/
skripsi/
supermarket_sales.csv
tutorial_greate/

Nadya Asanul Husna@LAPTOP-PAIDBRM MINGW64 /e/Great Edu
$ cd tutorial_greate

Nadya Asanul Husna@LAPTOP-PAIDBRM MINGW64 /e/Great Edu/tutorial_greate (main)
$ |
```



Implementasi GIT Remote



```
Nadya Asanul Husna@LAPTOP-PAIDBRTM MINGW64 /e/Great Edu/tutorial_create (main)
$ git remote
origin
```

1

```
Nadya Asanul Husna@LAPTOP-PAIDBRTM MINGW64 /e/Great Edu/tutorial_create (main)
$ git remote -v
origin  https://github.com/nadyamissun/tutorial_create.git (fetch)
origin  https://github.com/nadyamissun/tutorial_create.git (push)
```

2

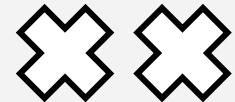
```
Nadya Asanul Husna@LAPTOP-PAIDBRTM MINGW64 /e/Great Edu/tutorial_create (main)
$ git status
On branch main
Your branch is up to date with 'origin/main'.
nothing to commit, working tree clean
```

3

1. GIT REMOTE akan mengeluarkan remote, yaitu secara default remote harus mempunyai nama dan ketika kita mengcloning repo dari github otomatis nama default adalah origin. Jika kita bikin sendiri bisa bebas dalam memberi nama -v adalah verbose, yaitu kita melihat lebih detail dan tidak hanya nama remote.
2. Arti dari your branch is up adalah branch yang kita punya di lokal sudah sama di origin/main. Jadi setiap ada perubahan branch dikedua tempat uptodate



Implementasi GIT Push



```
Nadya Asanul Husna@LAPTOP-PAIDBRTM MINGW64 /e/Great Edu/tutorial_greate (main)
$ git push
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 12 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 312 bytes | 312.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/nadyamissun/tutorial_greate.git
  a502820..c0ff6a54  main -> main
```

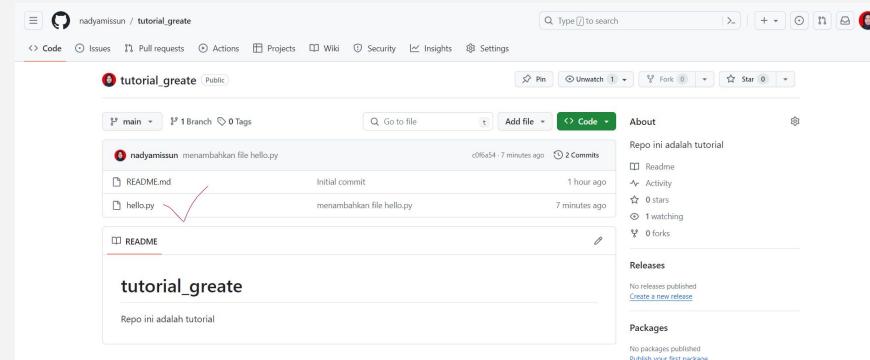
1. GIT Push : mengirimkan commit ke repo. Misal di lokal sudah terdapat file baru yang disimpan didalam folder yang telah kita clone. Lalu kita lakukan push agar file baru tersebut tersimpan di github.

```
MINGW64/e/Great Edu/tutorial_greate
Nadya Asanul Husna@LAPTOP-PAIDBRTM MINGW64 ~ (master)
$ cd "E:/Great Edu/tutorial_greate"
Nadya Asanul Husna@LAPTOP-PAIDBRTM MINGW64 /e/Great Edu/tutorial_greate (main)
$ ls
README.md hello.py
Nadya Asanul Husna@LAPTOP-PAIDBRTM MINGW64 /e/Great Edu/tutorial_greate (main)
$ git status
On branch main
Your branch is up to date with 'origin/main'.
Untracked files:
  (use "git add <file>..." to include in what will be committed)
    hello.py
nothing added to commit but untracked files present (use "git add" to track)
```

```
Nadya Asanul Husna@LAPTOP-PAIDBRTM MINGW64 /e/Great Edu/tutorial_greate (main)
$ git add hello.py
Nadya Asanul Husna@LAPTOP-PAIDBRTM MINGW64 /e/Great Edu/tutorial_greate (main)
$ git status
On branch main
Your branch is up to date with 'origin/main'.
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:   hello.py
```

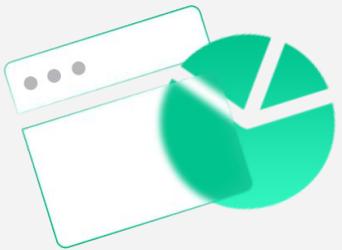
```
MINGW64/e/Great Edu/tutorial_greate
Nadya Asanul Husna@LAPTOP-PAIDBRTM MINGW64 /e/Great Edu/tutorial_greate (main)
$ git add hello.py
Nadya Asanul Husna@LAPTOP-PAIDBRTM MINGW64 /e/Great Edu/tutorial_greate (main)
$ git status
On branch main
Your branch is up to date with 'origin/main'.
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:   hello.py
Nadya Asanul Husna@LAPTOP-PAIDBRTM MINGW64 /e/Great Edu/tutorial_greate (main)
$ git commit
Aborting commit due to empty commit message.
Nadya Asanul Husna@LAPTOP-PAIDBRTM MINGW64 /e/Great Edu/tutorial_greate (main)
$ git commit -m "menambahkan file hello.py"
[main c0f6a54] menambahkan file hello.py
 1 file changed, 1 insertion(+)
create mode 100644 hello.py
```

```
MINGW64/e/Great Edu/tutorial_greate
create mode 100644 hello.py
Nadya Asanul Husna@LAPTOP-PAIDBRTM MINGW64 /e/Great Edu/tutorial_greate (main)
$ git status
On branch main
Your branch is ahead of 'origin/main' by 1 commit.
  (use "git push" to publish your local commits)
nothing to commit, working tree clean
Nadya Asanul Husna@LAPTOP-PAIDBRTM MINGW64 /e/Great Edu/tutorial_greate (main)
$ git push
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 12 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 312 bytes | 312.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/nadyamissun/tutorial_greate.git
  a502820..c0f6a54 main -> main
```



NEXT!!

Programming I:
Introduction to Python with Google Colab Environment



Terima Kasih

SIB Cycle 6 | 2024



www.greatedu.co.id