

Mengenal CLI⁺, Git, Github & Remote Repository

Apa itu Version Control System ?

Sistem yang mencatat semua perubahan yang dilakukan pada file sehingga semua riwayatnya akan terekam dan bisa dilihat kembali nanti. Saat developer membuat proyek baru, mereka selalu dan akan terus-menerus melakukan pembaruan terhadap kodenya. Bahkan, setelah proyeknya online, developer tetap harus memperbarui versinya, memperbaiki bug, menambahkan fitur baru, dan lain sebagainya.



Apa itu Git dan Kenapa penting bagi seorang programmer ?

Git adalah salah satu *tool* yang sering digunakan dalam proyek pengembangan software.

Version control system membantu developer melacak perubahan yang mereka lakukan terhadap basis kode. Tak hanya itu, sistem ini juga mencatat siapa saja yang membuat perubahan serta memulihkan kode yang telah dihapus atau dimodifikasi.

Karena Git menyimpan banyak salinan kode di repositori, maka tidak ada kode yang saling tertimpa.

REPOSITORY

Repositori atau repo adalah direktori penyimpanan file proyek.

Di sini, Anda bisa menyimpan apa pun yang berkaitan dengan proyek yang sedang Anda buat, misalnya file kode, gambar, atau audio.

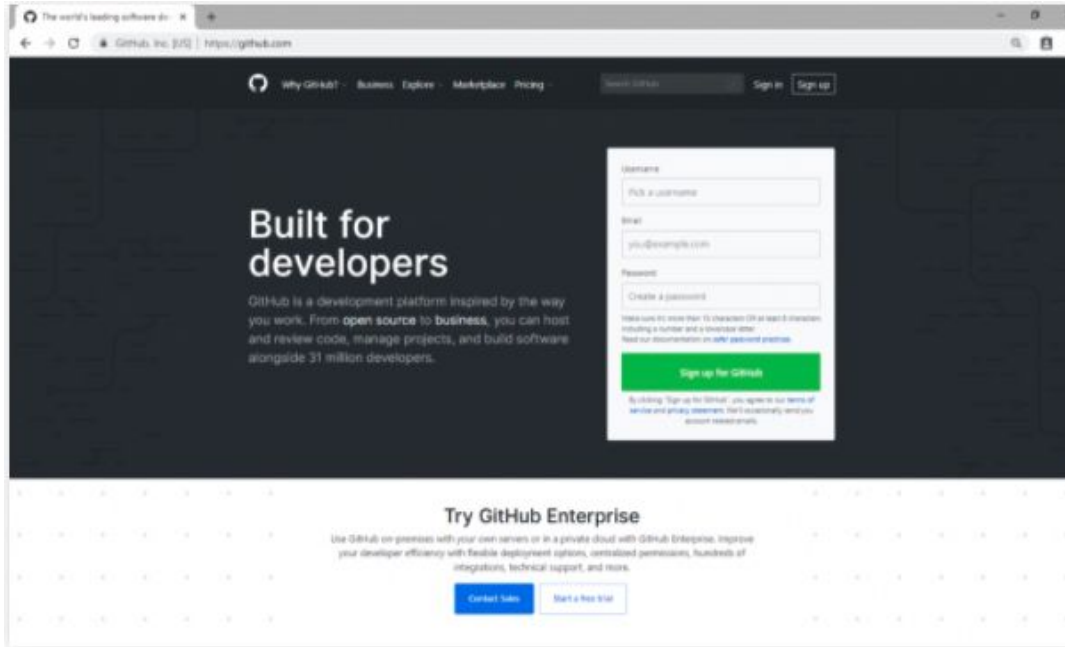
Repo sendiri bertempat di penyimpanan atau *storage* GitHub atau repositori lokal di komputer Anda.

BRANCH



Branch merupakan salinan dari **repository**. Anda bisa menggunakan branch ketika akan melakukan suatu pengembangan atau *development* secara terpisah.

Pekerjaan atau *task* yang Anda kerjakan di branch tidak akan mempengaruhi repository pusat atau branch lainnya. Jika pengembangannya sudah selesai, Anda bisa menggabungkan branch saat ini ke branch lainnya dan juga repository pusat dengan menggunakan *pull request*.



GitHub adalah manajemen proyek dan sistem *versioning code* sekaligus platform jaringan sosial yang dirancang khusus bagi para developer. Lalu, apa saja fungsi GitHub? Dengan platform ini, Anda bisa bekerja bersama-sama dengan rekan dari berbagai belahan dunia, merencanakan proyek, dan bahkan *tracking* (melacak) pekerjaan Anda.

Silahkan Lakukan Sign Up/ Register terlebih dahulu.

Buat akun GitHub

- Masuk ke [Github](https://github.com) dan daftar.
- Tentukan username kamu yang baik, jelas, mudah diingat dan dibaca; dengan huruf kecil (misalnya hacktiv8 atau hacktiv atau Height).
- Konfirmasi email akun GitHub di inbox kamu.
- Kunjungi <https://github.com/settings/profile> lalu lengkapi profil kamu.
- "Update profile" kamu.
- Jika nanti kamu melihat USERNAME, artinya perlu diganti dengan username-kamu. Misalnya username kamu adalah hacktiv8, berarti github.com/USERNAME menjadi github.com/hacktiv8.

NICE TO KNOW

Git merupakan tool utama yang pasti akan kamu gunakan setiap hari, sepanjang karirmu sebagai developer. Dengan ini kamu dapat tahu apa saja yang kamu lakukan setiap saat dan berkolaborasi dengan rekan lainnya juga. Anggaplah Git sebagai mesin waktu untuk berbagai kerjaan dan file/folder yang kamu kelola.

Sedangkan GitHub merupakan tempat sekaligus komunitas untuk para developer berbagai code dan berkolaborasi dalam berbagai project software. Bahkan modern ini, profil GitHub dapat menggantikan resume/CV untuk melamar pekerjaan!

Kami ingin untuk kamu nyaman menggunakan Git dan GitHub sesegera mungkin. Semoga kamu juga bakal ketagihan dan bahkan nggak bisa bayangin hidup tanpa Git dan GitHub!

Kini kamu akan melakukan instalasi dan konfigurasi Git di komputermu, mampu menjelaskan dan membedakan Git dan GitHub, serta memahami manfaat dan kekuatan version control system (VCS) atau nama lainnya source code management (SCM).

Download Git

Silahkan buka website resminya Git (git-scm.com). Kemudian unduh Git sesuai dengan arsitektur komputer kita. Kalau menggunakan 64bit, unduh yang 64bit. Begitu juga kalau menggunakan 32bit.

Git merupakan tool utama yang pasti akan kamu gunakan setiap hari, sepanjang karirmu sebagai developer. Dengan ini kamu dapat tahu apa saja yang kamu lakukan setiap saat dan berkolaborasi dengan rekan lainnya juga. Anggaplah Git sebagai mesin waktu untuk berbagai kerjaan dan file/folder yang kamu kelola.

Sedangkan GitHub merupakan tempat sekaligus komunitas untuk para developer berbagai code dan berkolaborasi dalam berbagai project software. Bahkan modern ini, profil GitHub dapat menggantikan resume/CV untuk melamar pekerjaan!

Kami ingin untuk kamu nyaman menggunakan Git dan GitHub sesegera mungkin. Semoga kamu juga bakal ketagihan dan bahkan nggak bisa bayangin hidup tanpa Git dan GitHub!

Kini kamu akan melakukan instalasi dan konfigurasi Git di komputermu, mampu menjelaskan dan membedakan Git dan GitHub, serta memahami manfaat dan kekuatan version control system (VCS) atau nama lainnya source code management (SCM).


git --fast-version-control

About

Documentation

Downloads

CUI Clients

Logos

Community

The entire **Pro Git** book written by Scott Chacon and Ben Straub is available to read online for free. Dead tree versions are available on [Amazon.com](#).

Download for Windows

Click [here](#) to download the latest (2.39.1) 64-bit version of **Git for Windows**. This is the most recent maintained build. It was released **22 days ago**, on 2023-01-17.

Other Git for Windows downloads

Standalone Installer

[32-bit Git for Windows Setup.](#)

[64-bit Git for Windows Setup.](#)

Portable ("thumbdrive edition")

[32-bit Git for Windows Portable.](#)

[64-bit Git for Windows Portable.](#)

Using winget tool

Install **winget** tool if you don't already have it, then type this command in command prompt or Powershell.

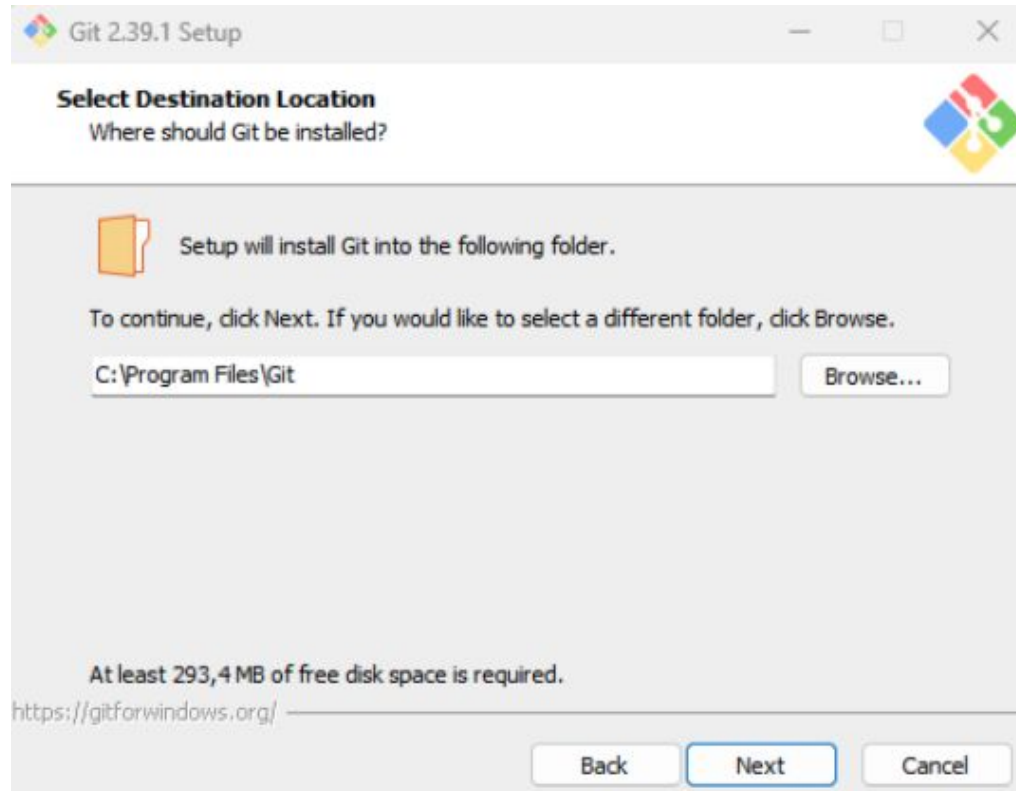
```
winget install --id Git.Git -e --source winget
```

The current source code release is version **2.39.1**. If you want the newer version, you can build it from the [source code](#).

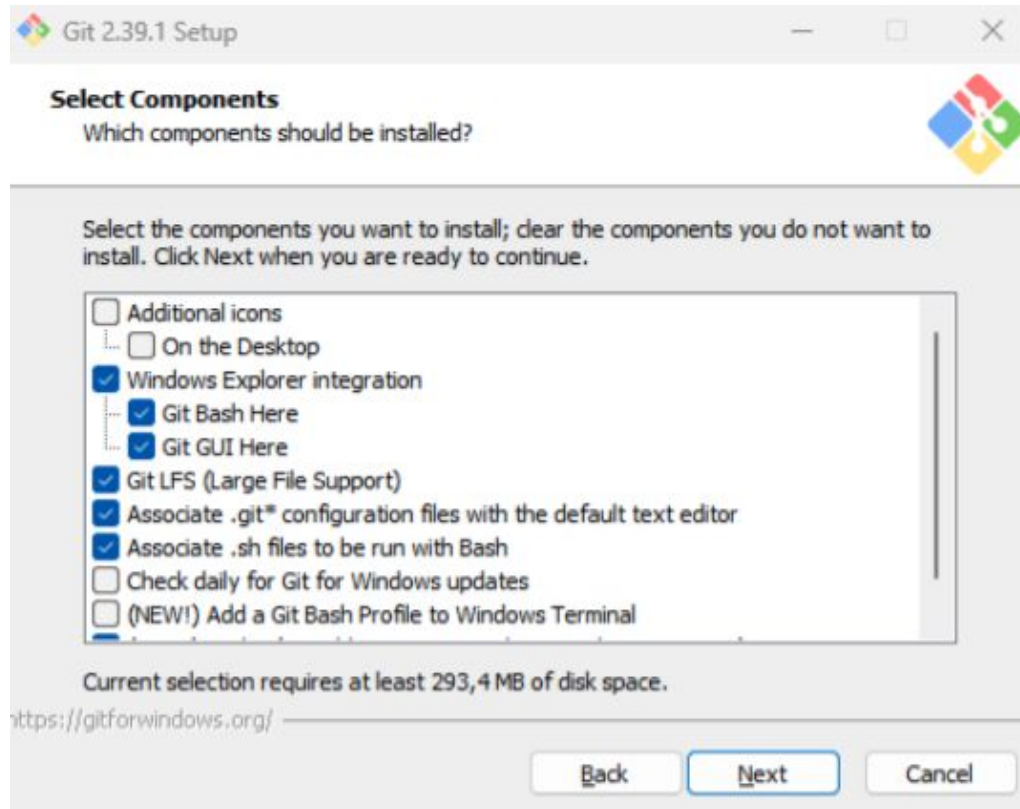
Maka akan muncul informasi lisensi Git, klik *Next* > untuk melanjutkan.



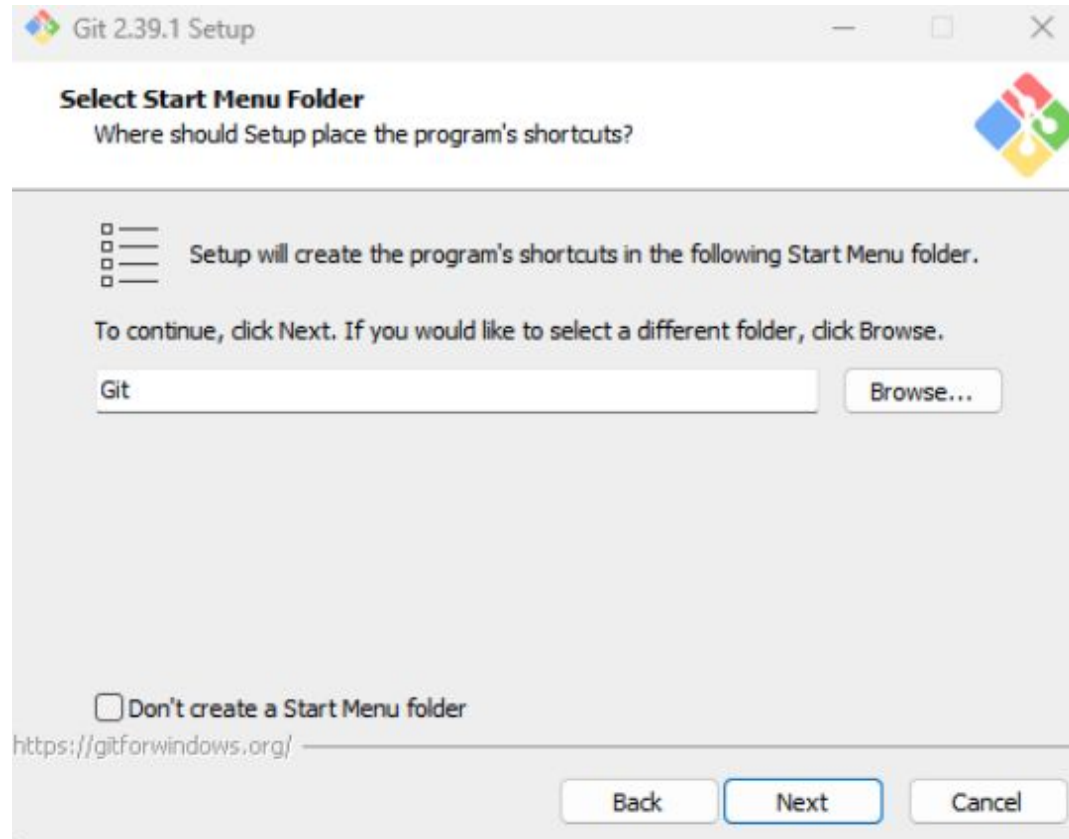
Selanjutnya menentukan lokasi instalasi. Biarkan saja apa adanya, kemudian klik *Next* >.



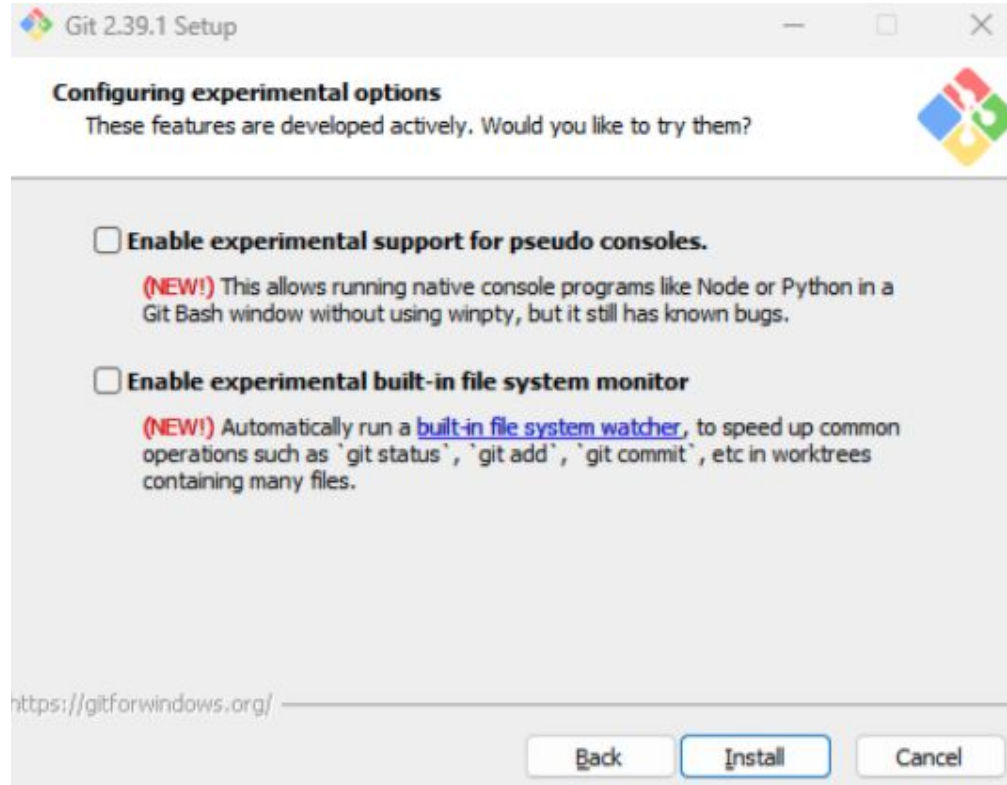
Selanjutnya pemilihan komponen, biarkan saja seperti ini kemudian klik *Next* >.



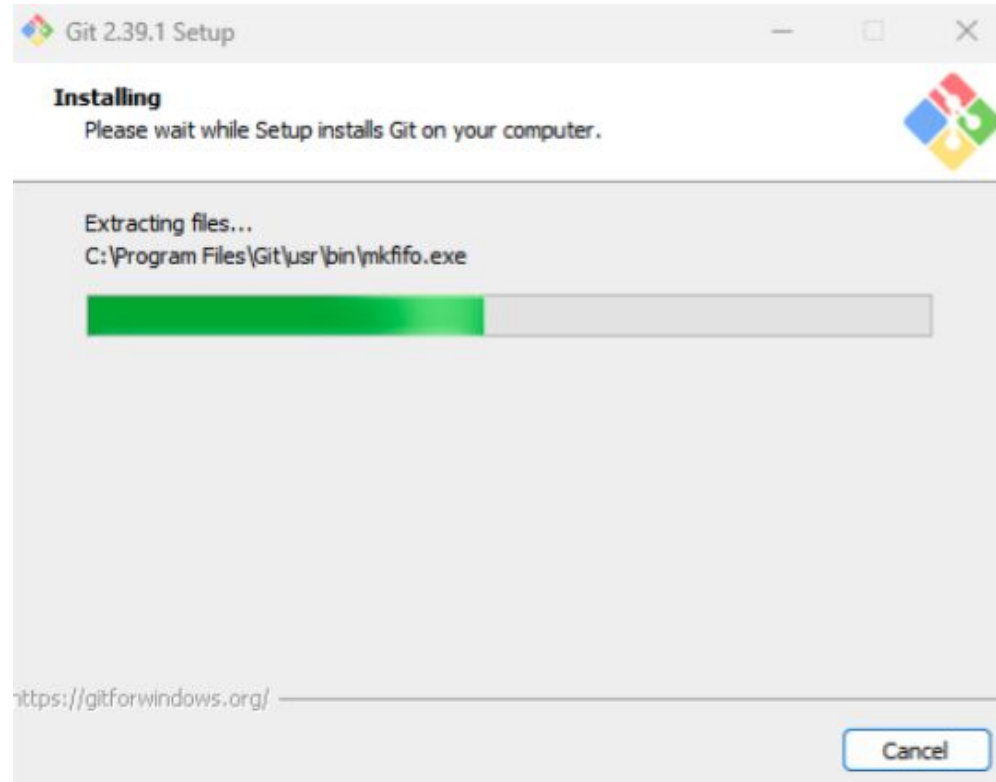
Selanjutnya pemilihan direktori start menu, klik *Next* >.



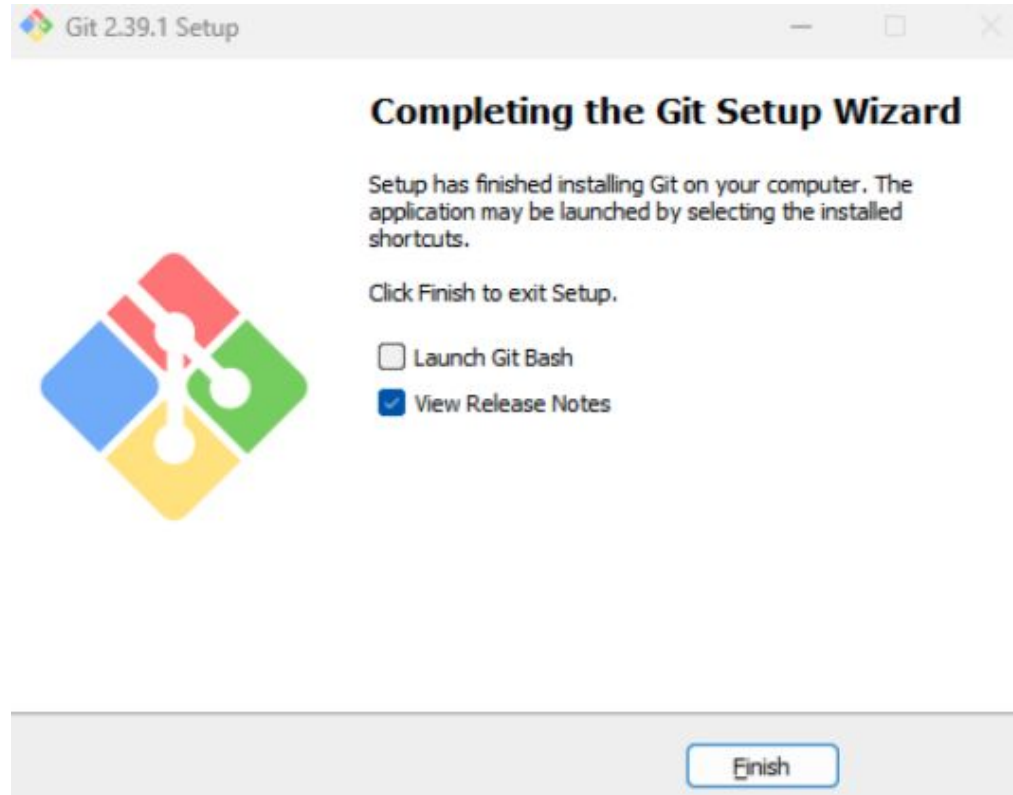
Klik tombol ***Install*** dibawah ini :



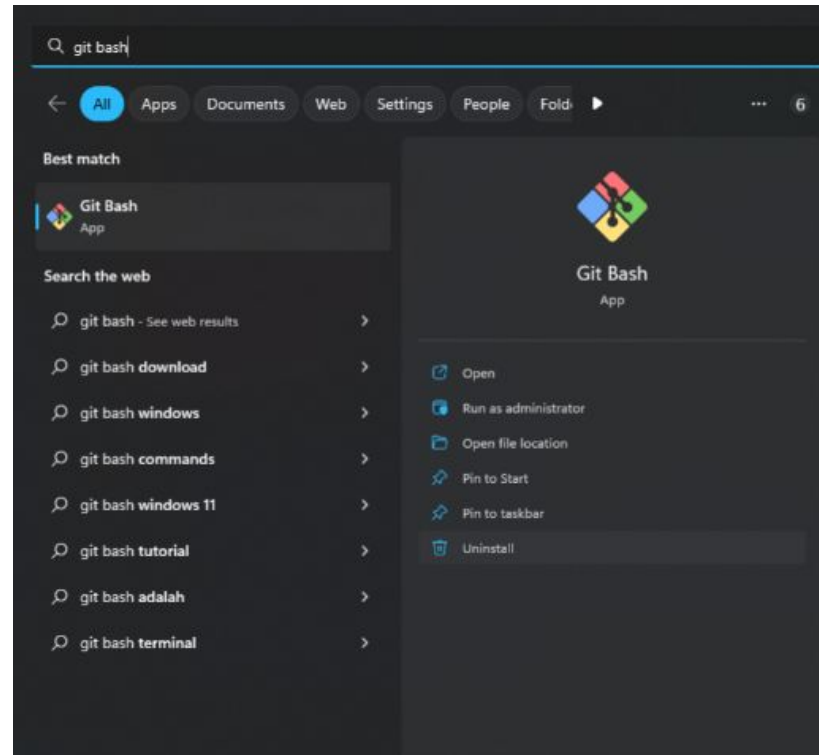
Selanjutnya silahkan tunggu sampai proses install selesai.



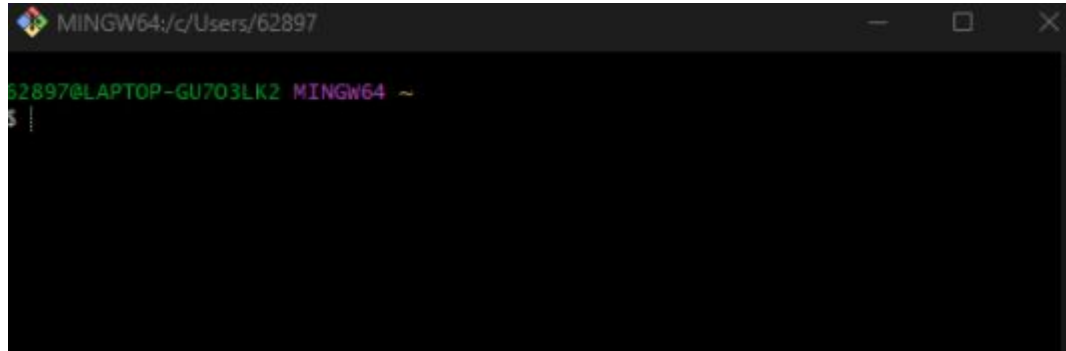
Selanjutnya pemilihan emulator terminal. Pilih saja yang bawah, kemudian klik *Next* >.



Selanjutnya setelah selesai instalasi jalankan git bash, seperti gambar dibawah ini



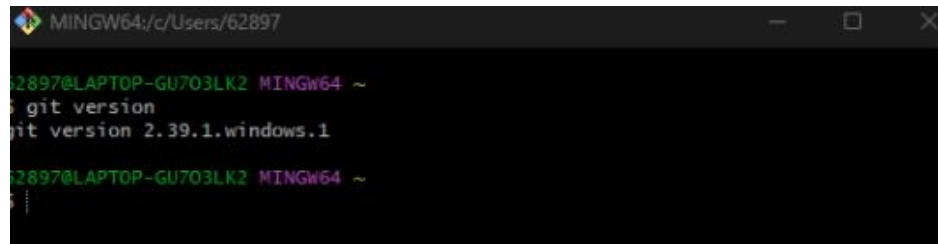
Selanjutnya akan muncul kotak hitam yang dinamakan git bash seperti dibawah ini :



```
MINGW64: c:/Users/62897

62897@LAPTOP-GU703LK2 MINGW64 ~
$
```

Coba cek versi git dengan command : git version



```
MINGW64: c:/Users/62897

62897@LAPTOP-GU703LK2 MINGW64 ~
$ git version
git version 2.39.1.windows.1

62897@LAPTOP-GU703LK2 MINGW64 ~
$
```



Selamat, Git sudah terinstal di Windows. Untuk mencobanya, silahkan buka CMD atau PowerShell, kemudian ketik perintah `git --version`.

Konfigurasi Awal yang Harus Dilakukan

Ada beberapa konfigurasi yang harus dipersiapkan sebelum mulai menggunakan Git, seperti *name* dan *email*.

Silahkan lakukan konfigurasi dengan perintah berikut ini.

```
git config --global user.name "Nama_Register_Github"  
git config --global user.email user@email.com
```

Kemudian periksa konfigurasinya dengan perintah:

```
git config --list
```

Apabila berhasil tampil seperti gambar berikut ini, berarti konfigurasi berhasil.

```
Maliks-Air:~ swijaya$ git config --list
credential.helper=osxkeychain
user.name=wijaysali
user.email=wijaysali@gmail.com
core.autocrlf=input
filter.lfs.clean=git-lfs clean -- %f
filter.lfs.smudge=git-lfs smudge -- %f
filter.lfs.process=git-lfs filter-process
filter.lfs.required=true
Maliks-Air:~ swijaya$
```

Konfigurasi `core.editor` bersifat opsional. Sedangkan *name* dan *email* wajib.

Jika kamu memiliki akun Github, Gitlab, Bitbucket atau yang lainnya...maka *username* dan *email* harus mengikuti akun tersebut agar mudah diintegrasikan.

Directions

Pastikan kamu siap akan pakai Git dengan command-line

Kamu perlu pengetahuan terminal atau command prompt serta CLI. Kamu harus sudah bisa:

- Navigasi ke berbagai directory/folder
- Membuat file dan folder baru
- Menghapus file dan folder
- Melihat berbagai file yang ada di folder
- Mengetahui lokasi folder di mana kita berada
- Memindahkan file ke berbagai folder
- Menyalin file atau folder ke tempat berbeda
- Membuka atau mengubah file ke dalam editor teks/code

Kamu bisa menggunakan referensi dibawah ini untuk mempelajari command line:

<https://lifehacker.com/5633909/who-needs-a-mouse-learn-to-use-the-command-line-for-almost-anything>

Best Practice

buka terminal/cmd arahkan ke folder projects)

terus lakukan :

Git init

Git add .

Git status

git commit -m 'your message'

git remote add origin 'your_url_name'

git push -u origin master

kembali browser lagi terus refresh browser untuk cek sudah masuk ke github belum..

kalo sudah masuk berarti sudah berhasil dan selesai.



#Git configuration

```
git config --global user.name "Your Name"  
git config --global user.email "you@gmail.com"
```

#Starting A Project

```
git init [project name]  
git clone [project url]
```

#Remove file from directory

```
git rm [file]
```

#Status of working directory

```
git status
```

#Add a file to the staging area

```
git add [file]
```

#Discard changes in working directory

```
git checkout -- [file]
```

GIT CHEATSHEET

#Commit to local

```
git commit
```

#Revert your repository

```
git reset [file]
```

#List all local branches

```
git branch [-a]
```

#Fetch changes from the remote and merge current branch with its upstream

```
git pull [remote]
```

#Join specified [from name] branch

```
git merge [from name]
```

#Create new branch

```
git branch [branch_name]
```

#Push local changes to the remote

```
git push [--tags] [remote]
```

#Remove selected branch

```
git branch -d [name]
```

#Fetch changes from the remote

```
git fetch [remote]
```

#Switch current branch to specified branch

```
git checkout [-b][branch_name]
```

Objectives

Melatih penggunaan dasar command line serta Git dan GitHub lebih jauh akan memudahkan kamu seiring pengembangan web kamu. Eksperimen ini dapat secara bebas dilakukan, tidak harus sama persis dengan petunjuk, yang penting esensinya didapat.

Directions

- Melatih setup repositori Git dari awal

Buatlah folder baru sebagai YourFirstSite.

Inisialisasi Git dalam folder tersebut dengan git init.

Kini kamu bisa cek status repositori dengan git status.

Tambahkan file teks misalnya try.txt atau README.md, lalu isi dengan beberapa tulisan.

Gunakan perintah touch jika mau cepat.

Cek kembali status repositori. Latihlah untuk melakukan ini setiap saat.

Add dan commit perubahan yang ada.



- Melatih push dan pull

Buatlah repositori GitHub baru bernama sama dengan repositori di local. Kemudian langsung Create repository.

Bacalah petunjuk yang ada pada halaman utama repositori kosong tersebut.

Add remote dengan git remote add origin [GIT HTTPS URL](#)

Kirim semua perubahan dengan git push -u origin master

Ubah isi file yang ada di GitHub langsung di web-nya, kemudian commit.

Dapatkan perubahan yang ada ke local dengan git pull origin master.

Gunakan File Portfolio pada challenge di pertemuan sebelumnya.