

MATERI

Hari 3 – Command Line dan Git

Apa itu command line interface?

Command line interface atau yang biasa kita kenal cmd atau command prompt atau terminal adalah software atau antar muka untuk memberikan perintah-perintah dasar pada komputer kita. Biasanya kita mengoperasikan komputer seperti browsing media, mencari file, membuat file atau folder, dan lain-lain dengan GUI (Graphical User Interface) yang sudah disediakan pengembang Sistem Operasi seperti Windows, Linux, atau OSX. Namun pada dasarnya kita juga bisa melakukan operasi-operasi tersebut dalam suatu perintah yang kita kirimkan melalui cmd/terminal. Berikut ini beberapa contohnya:

Catatan: Contoh perintah ditulis dengan awalan `$` (tapi tidak usah ditulis tanda `$` pada terminal/command prompt nya) . Tanda `//` berarti menandakan baris tersebut adalah komentar atau keterangan, tidak perlu ditulis di perintah terminal.

Menuju folder/direktori tertentu

Untuk menuju folder atau direktori tertentu bisa gunakan perintah `cd` atau kepanjangan dari change directory.

```
// Menuju Desktop
$ cd Desktop
// Menuju ke direktori parent (sebelumnya)
$ cd ..
```

Mengetahui alamat dari direktori yang sedang dibuka

Terkadang kita butuh tahu sedang dimana posisi terminal kita berjalan. Perintah untuk fungsi ini yaitu `pwd` atau kependekan dari path of working directory

```
$ pwd
/home/users
// sedang berada di folder
home/users
```

Mengetahui list isi dari sebuah direktori/folder

Kamu bisa gunakan perintah `ls` atau `dir` . contohnya sebagai berikut

```
$ ls
Document Desktop Image index.html
$ dir
Document Desktop Image index.html
```

Membuat folder baru

Perintahnya adalah `mkdir` atau kependekan dari make directory. contohnya sebagai berikut

```
// membuat folder baru bernama new-folder
$ mkdir new-folder
// mengecek apakah sudah berhasil buat folder baru
$ ls
new-folder ... ..
```

Membuat file baru

Gunakan perintah `touch` untuk membuat file baru

```
// membuat file index.html
$ touch index.html
// mengecek apakah sudah berhasil buat file baru
$ ls index.html ... ..
```

Software terminal/command line yang recommended

Untuk OS Windows, terminal bawaannya adalah command prompt atau cmd yang relatif kurang nyaman untuk dipakai dalam tahap development. Oleh karena itu saya rekomendasikan untuk install terminal di bawah ini agar lebih optimal dalam mengerjakan project.

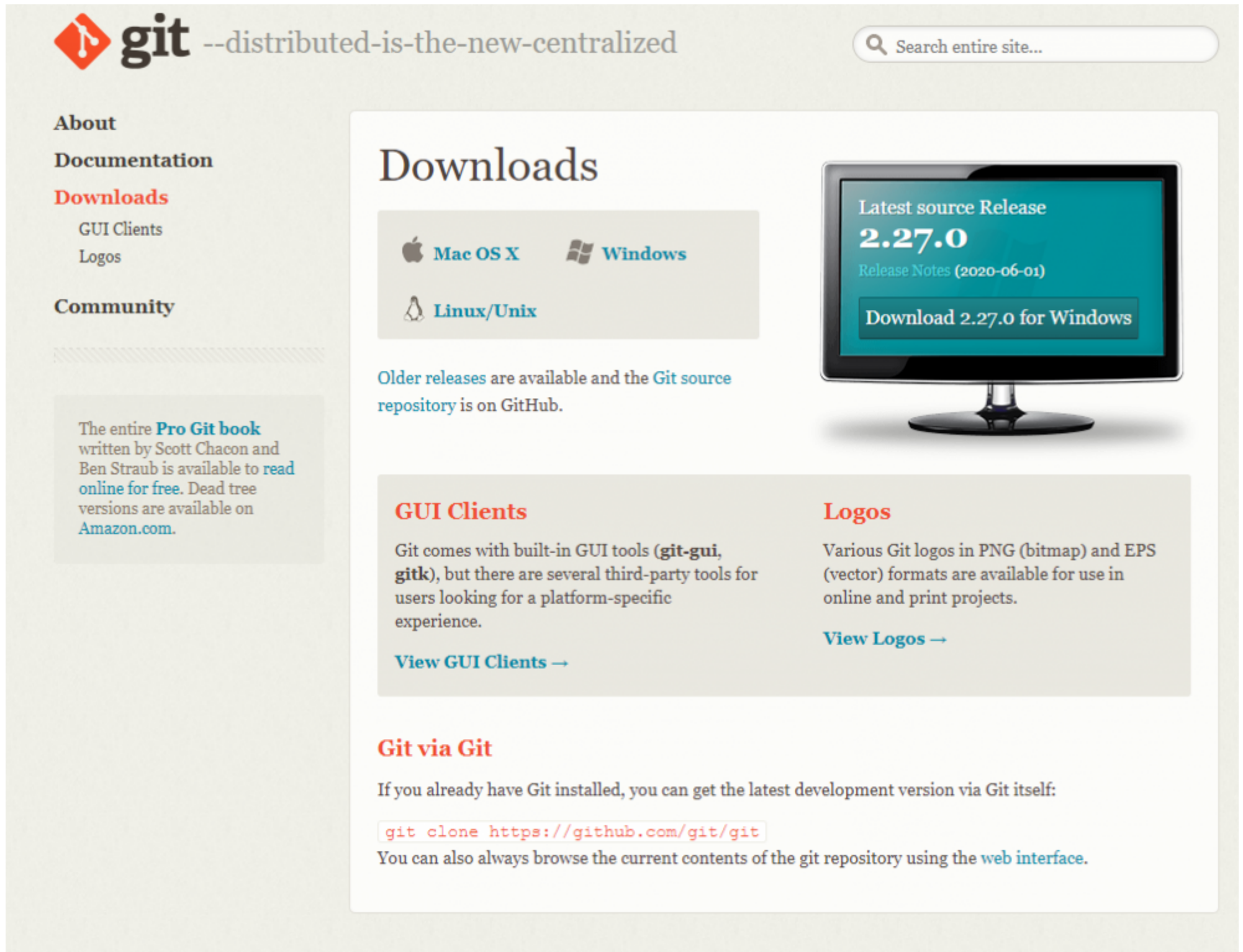
- Cmder, link download: [cmder](#) (Terdapat dua pilihan versi : mini dan full. Mini adalah versi yang minimum dan yang full adalah versi mini ditambah Git dari windows)
- Git bash, Git bash biasanya terinstall jika kita menginstall git di pc/laptop kita.

informasi lain.

Selain itu, Git memungkinkan para pengembang perangkat lunak dari berbagai belahan dunia mengerjakan banyak proyek bersama-sama tanpa mengharuskan bertatap muka.

Meng-install Git

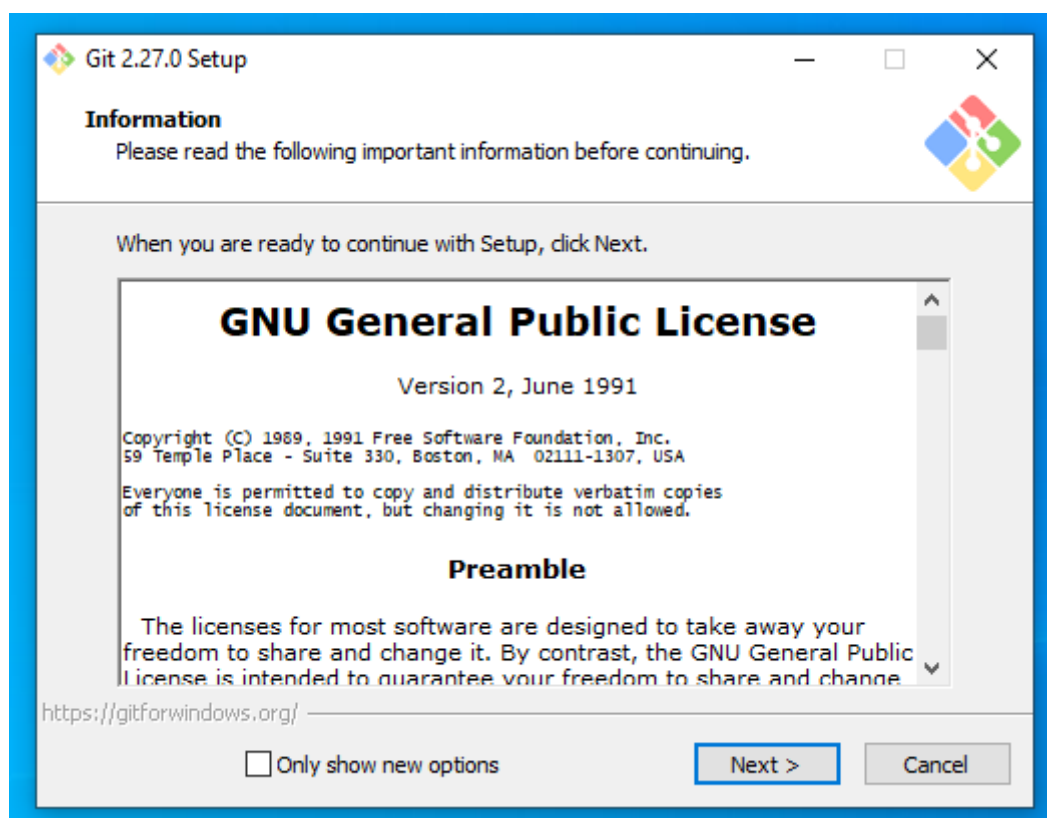
Untuk menginstall git ke dalam windows, download terlebih dahulu file instalasi nya dari halaman [download.git](#). Jika menggunakan OS selain windows bisa menyesuaikan instalasi dengan dokumentasi yang sudah tersedia



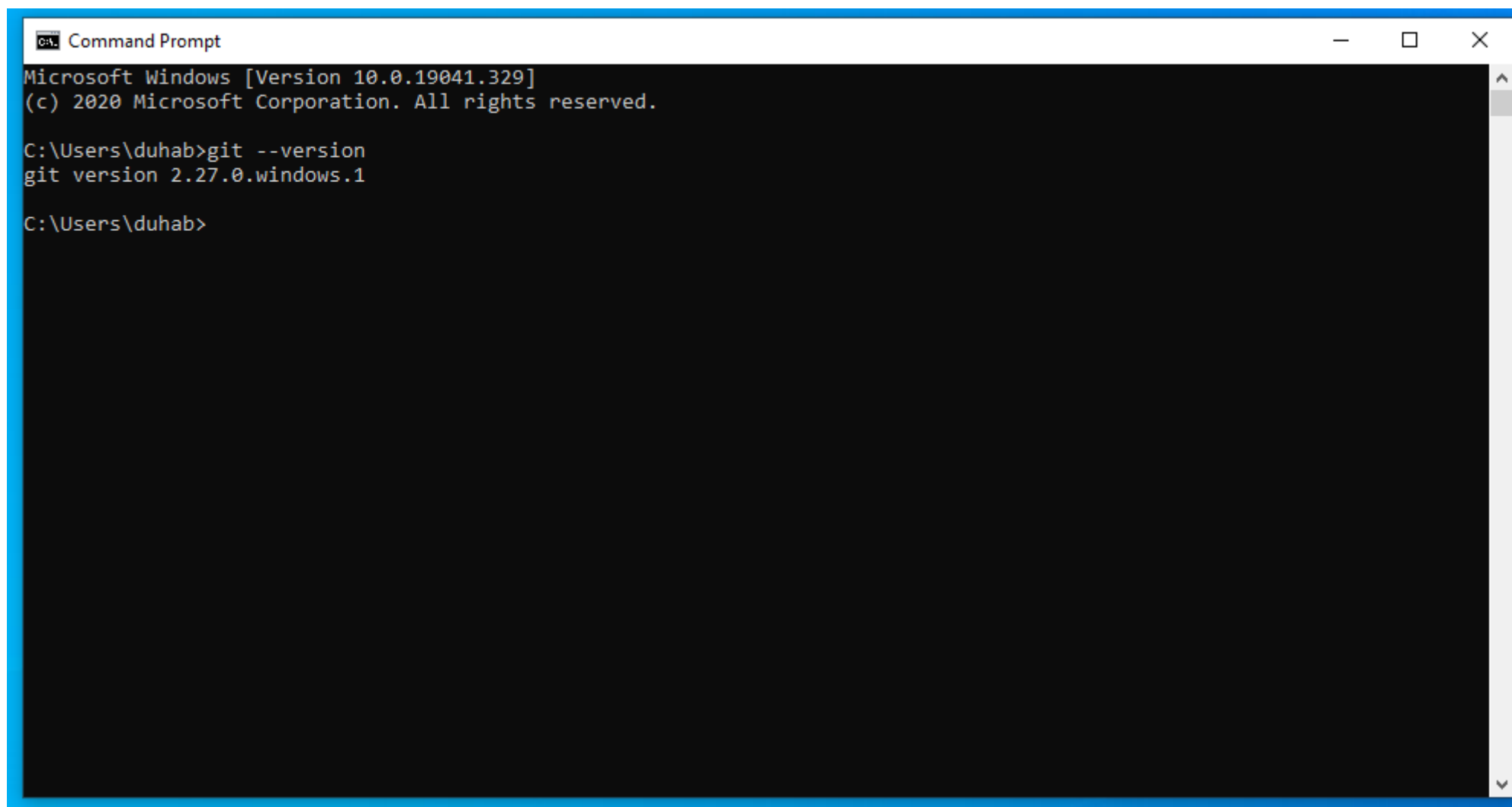
The screenshot shows the Git website homepage. At the top, there's a navigation bar with the Git logo and the tagline "--distributed-is-the-new-centralized". Below this, there's a search bar and a sidebar with links to "About", "Documentation", "Downloads", "GUI Clients", "Logos", and "Community". The main content area is titled "Downloads" and features a large monitor graphic displaying the latest source release "2.27.0" with a "Download 2.27.0 for Windows" button. Below the monitor, there are sections for "GUI Clients" and "Logos". The "GUI Clients" section mentions built-in tools like "git-gui" and "gitk", and provides a link to "View GUI Clients". The "Logos" section mentions various Git logos in PNG and EPS formats and provides a link to "View Logos". At the bottom, there's a section titled "Git via Git" which provides a command to clone the Git repository and a link to the web interface.

halaman download git dari [git scm](#)

jika sudah terdownload, klik file instalasi lalu ikuti instruksi instalasi (klik next saja sampai selesai).



instalasi git, klik next sampai instalasi selesai



```
Command Prompt
Microsoft Windows [Version 10.0.19041.329]
(c) 2020 Microsoft Corporation. All rights reserved.

C:\Users\duhab>git --version
git version 2.27.0.windows.1

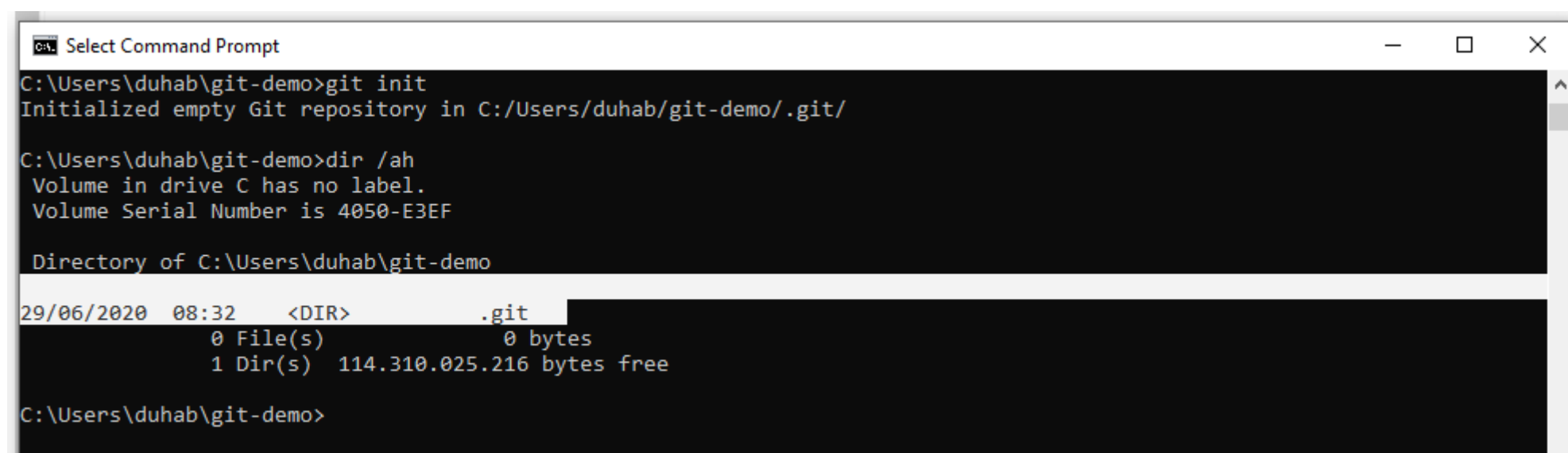
C:\Users\duhab>
```

memastikan git sudah terpasang dengan mengecek versi git yang sudah terinstall

Perintah Dasar Git

git init

`git init` adalah perintah untuk memasang git pada sebuah direktori atau folder yang sedang terbuka di command prompt atau terminal.



```
Select Command Prompt
C:\Users\duhab\git-demo>git init
Initialized empty Git repository in C:/Users/duhab/git-demo/.git/

C:\Users\duhab\git-demo>dir /ah
Volume in drive C has no label.
Volume Serial Number is 4050-E3EF

Directory of C:\Users\duhab\git-demo

29/06/2020  08:32    <DIR>          .git
               0 File(s)              0 bytes
               1 Dir(s)  114.310.025.216 bytes free

C:\Users\duhab\git-demo>
```

memasang git pada folder git-demo

pada contoh di atas, setelah memberikan perintah `git init` akan muncul keterangan “Initialized empty Git repository in ...” yang berarti folder git-demo yang sedang dibuka di cmd sudah dapat menggunakan Git di dalamnya.

Jika diberikan perintah di cmd `dir /ah` maka akan ditampilkan bahwa pada direktori sudah terdapat folder yang *hidden* dengan nama folder `.git`. Di dalam folder `.git` itu lah terdapat sistem Git sehingga kita dapat menggunakan perintah-perintah git untuk mengatur project di dalam direktori.

git add

`git add` yaitu perintah git untuk menambahkan perubahan-perubahan pada direktori project sebelum menyimpan perubahan tersebut. Misalkan di folder project kita terdapat file `index.html` yang ingin kita simpan perubahannya.

```
$ git add index.html
```

git commit

setelah memberikan `git add` untuk menambahkan file-file yang ingin disimpan sebagai perubahan(update), selanjutnya kita berikan perintah `git commit`. `git commit` berfungsi untuk memberikan catatan atau pesan yang menandai perubahan yang sudah dibuat. contohnya seperti berikut

```
$ git commit -m "sudah menambahkan HTML dan CSS"
```

Contoh di atas menunjukkan bahwa kita memberikan pesan yaitu “sudah menambahkan HTML dan CSS” pada perubahan yang kita buat di project kita.

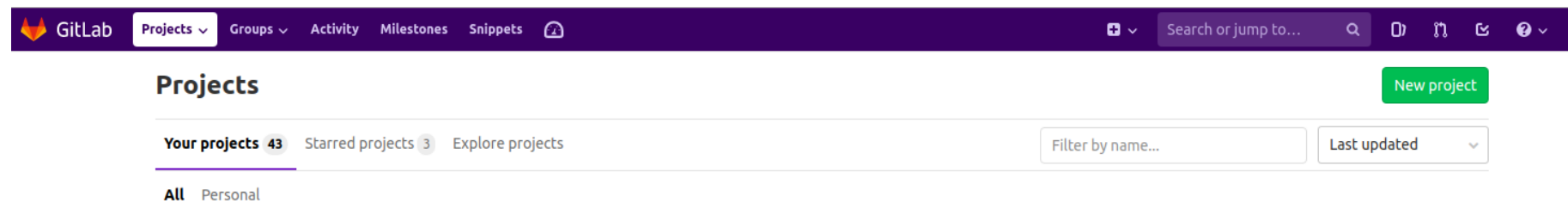
Membuat Repositori Git dan Mengelolanya

Membuat Repository Baru

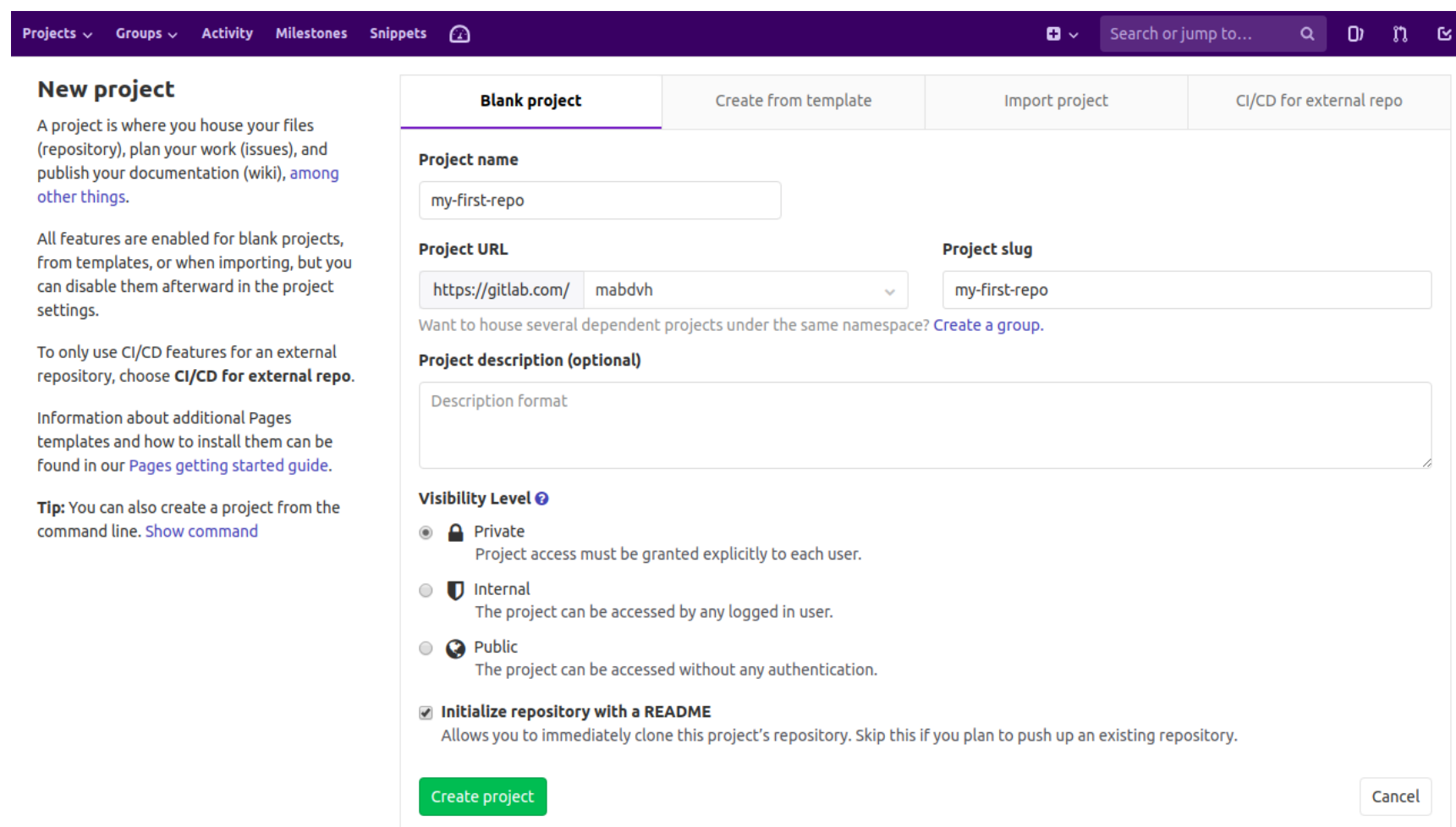
Pada materi ini akan diberikan contoh membuat repositori Git dan mengelola menggunakan layanan Gitlab.

Repository atau repo di Gitlab biasa disebut dengan project. Berikut langkah-langkah untuk membuat repository baru:

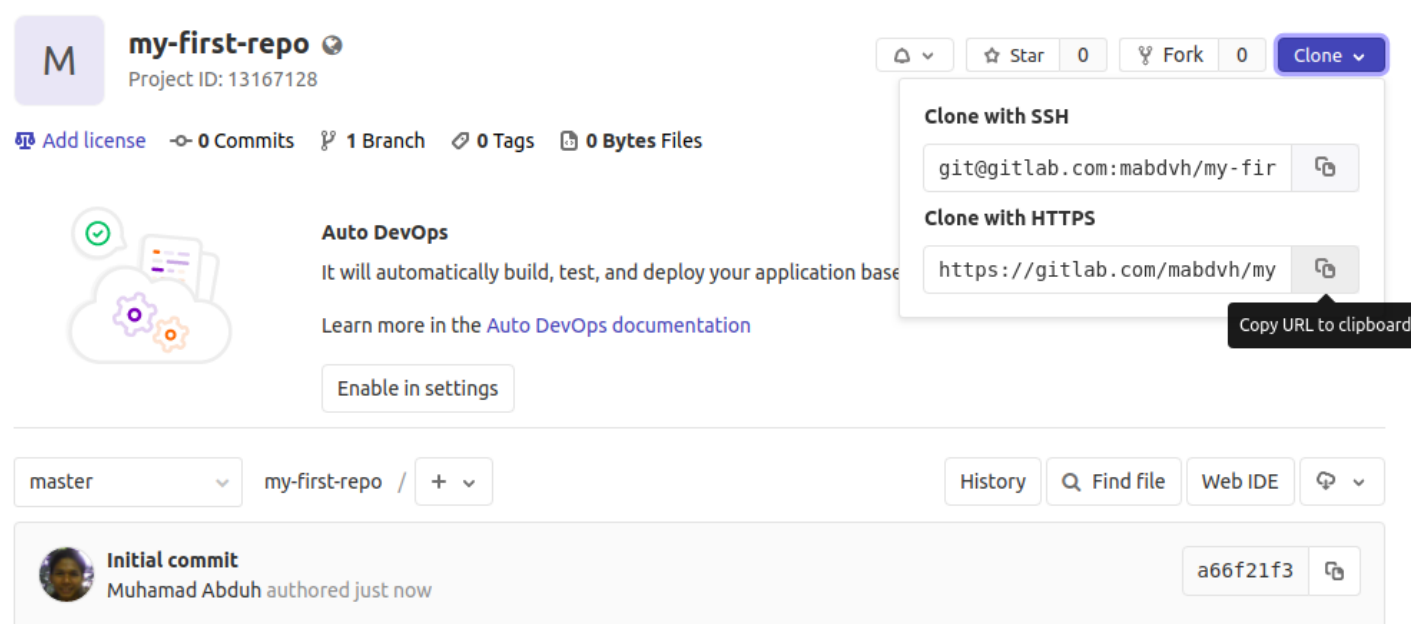
- Setelah login anda akan menuju halaman Projects yang berisi semua repository yang anda miliki. Jika Anda user baru maka belum ada repository yang tersedia. Di halaman tersebut anda akan melihat tombol New Project di sebelah kanan atas. Klik tombol tersebut.



- Pada halaman membuat project baru, isi kolom project-name dengan nama yang Anda inginkan, misalkan: my first repo.
- Pada kolom visibility level Anda akan melihat tiga pilihan yaitu Private, Internal, dan Public. Private artinya repo yang akan kita buat bersifat rahasia, hanya Anda dan orang yang Anda beri akses saja yang bisa membukanya. Sedangkan pilihan Internal berarti repo Anda akan dapat dilihat oleh semua user Gitlab. Pilihan "Public" berarti repo Anda akan terlihat oleh orang dari mana saja walaupun dia tidak memiliki akun gitlab. Pilih yang sesuai kebutuhan misalkan : Public.
- Di bagian bawahnya terdapat pilihan checkbox "Initialize repository with a README". Centang saja jika anda ingin langsung clone repository nya setelah dibuat.



- Jika berhasil maka Anda akan diarahkan menuju halaman project atau repo yang telah dibuat. Jika sebelumnya kita centang "Initialize repository with a README" maka repository kita akan berisi file README.md.
- Pada Bagian kanan atas terdapat tombol clone. Ketika diklik maka akan keluar dua pilihan "Clone with SSH" dan "Clone with HTTPS". Jika ingin clone dengan SSH maka anda harus setting dulu akun anda agar terhubung dengan komputer Anda. Atau pilih saja clone dengan HTTPS. Klik tombol salin/copy supaya langsung otomatis tercopy pada clipboard.



- Setelah itu menuju ke terminal atau cmd. Arahkan menuju folder dimana anda ingin menyimpan repository yang sudah Anda buat. Berikan perintah `git clone <link_anda_yang_barusan_dicopy>` (tanda <> hanyalah sebagai penanda template tidak usah ditulis di perintah cmd)

```
git clone <your_repository_link_url>
```

- Jika sudah berhasil diclone maka akan ditemukan folder dengan nama repository di Gitlab. Cara mengeceknya dengan perintah "ls" atau "dir".

```
$ ls my-first-repo
```

Mengupload perubahan project dengan git push

Ketika kita sudah berhasil clone repository kita di Gitlab maka tentu kita ingin melakukan update atau penambahan code pada project kita. Agar perbaruan yang kita simpan di local (komputer) kita dapat kita perbaharui juga di repository Gitlab maka kita bisa lakukan dengan git push. Berikut langkah-langkahnya:

- Ketika kita sudah memberikan perubahan terhadap suatu file atau membuat folder-folder baru pada folder project kita maka cara mengecek status perbaruan adalah dengan git status

```
$ git status
On branch master
Your branch is up to date with 'origin/master'.
Changes not staged for commit: (use "git add <file>..." to update what will be committed) (use "git checkout -- <file>..." to discard changes in working directory)
    modified:   README.md
no changes added to commit (use "git add" and/or "git commit -a")
```

- Setelah git status maka akan ditampilkan list file yang "modified" atau "untracked". "modified" artinya adalah kita mengubah file yang sebelumnya ada sedangkan "untracked" artinya file tersebut sama sekali baru di repository. Dalam contoh di atas terdapat perubahan pada file "README.md".
- Untuk menaikkan perubahan (stage) yang kita buat maka berikan perintah git add

```
$ git add README.md
// Jika ada terdapat banyak file dan kita ingin naikkan semuanya maka perintahnya:
$ git add .
```

- Sekarang perubahan tadi sudah pada tahap "staged". jika kita cek status nya lagi maka akan didapati berbeda:

```
$ git status
On branch master
Your branch is up to date with 'origin/master'.
Changes to be committed: (use "git reset HEAD <file>..." to unstage)
    modified:   README.md
```

- Selanjutnya adalah memberikan laporan atau commit terhadap perubahan yang kita buat dengan git commit.

```
$ git commit -m "Pesan saya: Mengubah README.md"
```

- tanda "-m" pada perintah commit artinya kita ingin memberikan pesan atau laporan terhadap perubahan yang sudah dibuat. Dalam hal ini laporannya adalah "Pesan saya: mengubah README.md"


```
$ git push origin master
```

Jika branch yang sedang kita miliki di local adalah branch lain selain master maka cara push nya adalah exit: Ctrl+

```
$ git push origin <nama_branch>
```

Jika langkah tersebut sudah maka kita sudah berhasil mengupload perubahan yang kita buat di repository kita. 🍷🍷

NB : jika terjadi error permission denied

- cara 1 : <https://saannjaay.medium.com/gitlab-remote-http-basic-access-denied-and-fatal-authentication-309ec178e010>
- cara 2 : <https://www.petanikode.com/gitlab-ssh/>

Tutorial Video

Untuk selanjutnya materi dapat dicoba melalui video kelas di link berikut:

- **Review Sanbercode** : <https://www.youtube.com/watch?v=23T5wV7KQBM&list=PL8YCzrSDCR7hm9zNLMVadGuKBgTPnvX5i&index=1>
- **Referensi lain** : [Intro Terminal & Git](#)

Rating - Feedback

Berikan Rating pada posting ini:



Berikan kritik dan saran..

Submit