

Modul 2 Implementasi Algoritma Link GitHub

2.1. Tujuan

Setelah mengikuti praktikum ini mahasiswa diharapkan dapat:

1. Mengetahui cara pendaftaran akun GitHub
2. Melakukan instalasi Git dan pembuatan repositori lokal
3. Mampu melakukan push file/projek ke dalam akun GitHub masing-masing praktikan

2.2. Alat dan Bahan

Alat & Bahan yang digunakan adalah hardware perangkat laptop/ PC masing-masing praktikan

2.3. Dasar Teori

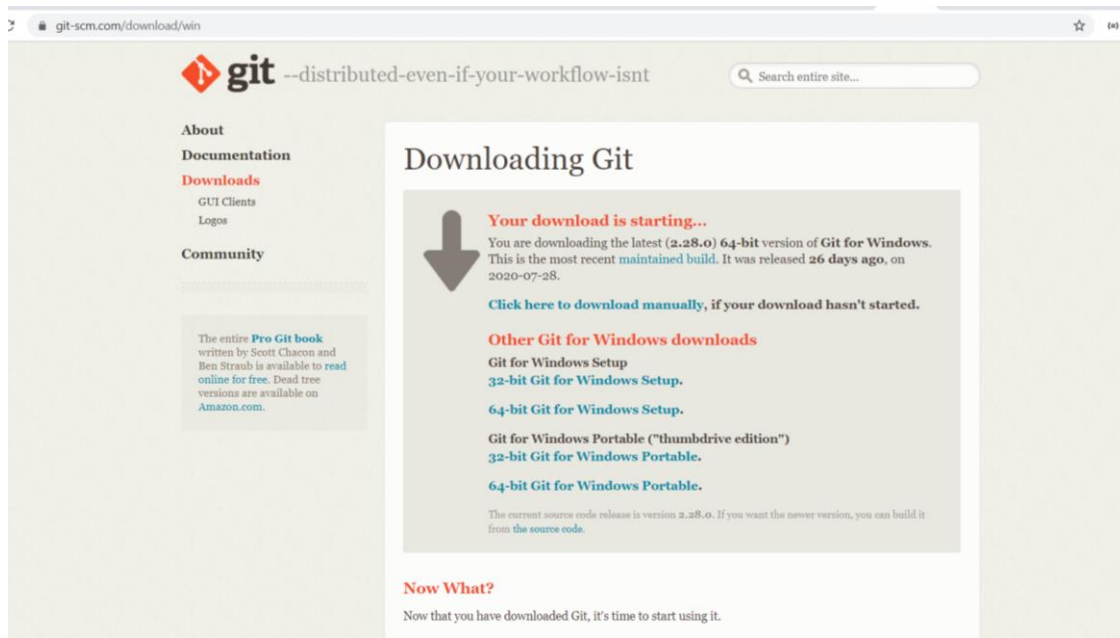
Git merupakan suatu sistem version control (version system control, VCS) terdistribusi yang memiliki sifat open source dan free. Git dirancang untuk dapat digunakan pada projek program dengan segala skala ukuran, mulai dari projek dengan skala kecil sampai sangat besar.

Version control sendiri merupakan suatu sistem yang akan merekam perubahan yang dilakukan pada suatu berkas/ file maupun kumpulan berkas. Perekaman ini dilakukan dari waktu ke waktu, sesuai dengan yang diinputkan oleh programmer. Dengan adanya perekaman ini, programmer dapat melihat kembali pada suatu versi perekaman di kemudian hari. Misal, jika saat ini sudah ada perekaman bagi program Hello World sampai versi kelima, maka programmer masih tetap dapat melihat program pada versi ketiga, atau bahkan versi awalnya.

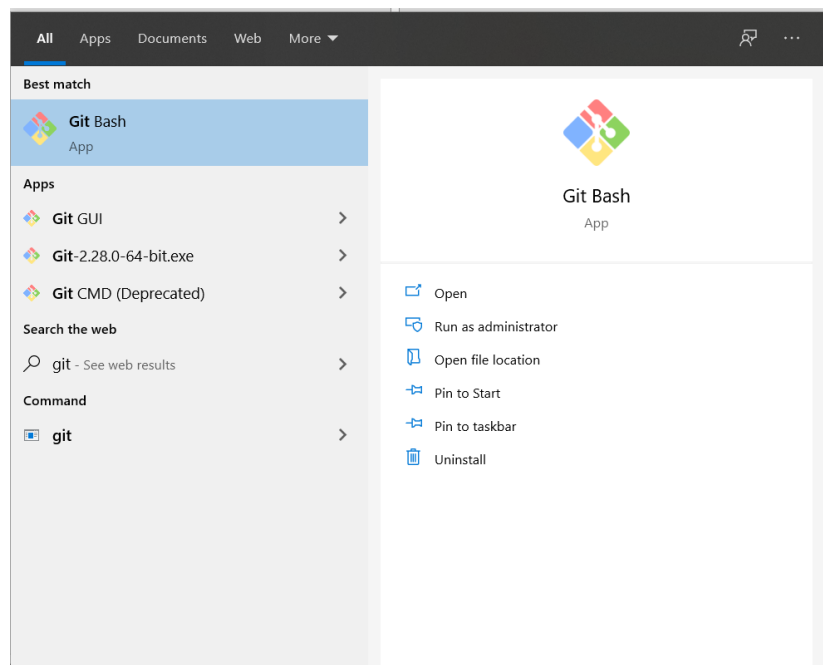
Misal, seorang perancang grafis atau web ingin menyimpan seluruh versi dari gambar atau layout yang telah dibuatnya. Maka, menggunakan Version Control System (VCS) merupakan hal yang bijak untuk dilakukan. Dengan VCS, berkas dan projek yang sudah disimpan dapat dikembalikan ke keadaan sebelumnya, dan perubahan-perubahan yang telah dibuat dapat dibandingkan. Pada VCS juga dapat dilihat siapa yang melakukan perubahan (sehingga bisa diketahui siapa yang mengubah berkas terakhir kali sebelum timbul masalah), demikian juga dapat dilihat siapa dan kapan suatu issu dimulai, dan berbagai keunggulan lainnya.

Salah satu penyedia jasa Git yang banyak digunakan saat ini adalah GitHub. Untuk dapat mengaktifkan VCS dari GitHub, pertama kita harus menginstal Git dengan langkah berikut:

1. Unduhlah Git dari link berikut: <https://git-scm.com/downloads>



2. Lakukan instalasi Git pada komputer Anda. Setelah melakukan instalasi, cek-lah apakah Git sudah dikenali pada komputer Anda dengan menjalankan Git → Git Bash.



3. Pada *command window* yang muncul, ketikkan git. Jika git sudah ter-instal, maka akan muncul perintah-perintah terkait git pada jendela tersebut

```
MINGW64/c/Users/Cahya
$ git
usage: git [--version] [--help] [-C <path>] [-c <name>=<value>]
      [--exec-path<path>] [--html-path] [--man-path] [--info-path]
      [-p | --paginate] [-P | --no-pager] [--no-replace-objects] [--bare]
      [--git-dir<path>] [--work-tree<path>] [--namespace<name>]
      <command> [<args>]

These are common Git commands used in various situations:

start a working area (see also: git help tutorial)
  clone      Clone a repository into a new directory
  init       Create an empty Git repository or reinitialize an existing one

work on the current change (see also: git help everyday)
  add        Add file contents to the index
  mv         Move or rename a file, a directory, or a symlink
  restore    Restore working tree files
  rm         Remove files from the working tree and from the index
  sparse-checkout  Initialize and modify the sparse-checkout

examine the history and state (see also: git help revisions)
  bisect     Use binary search to find the commit that introduced a bug
  diff       Show changes between commits, commit and working tree, etc
  grep       Print lines matching a pattern
  log        Show commit logs
  show       Show various types of objects
  status     Show the working tree status

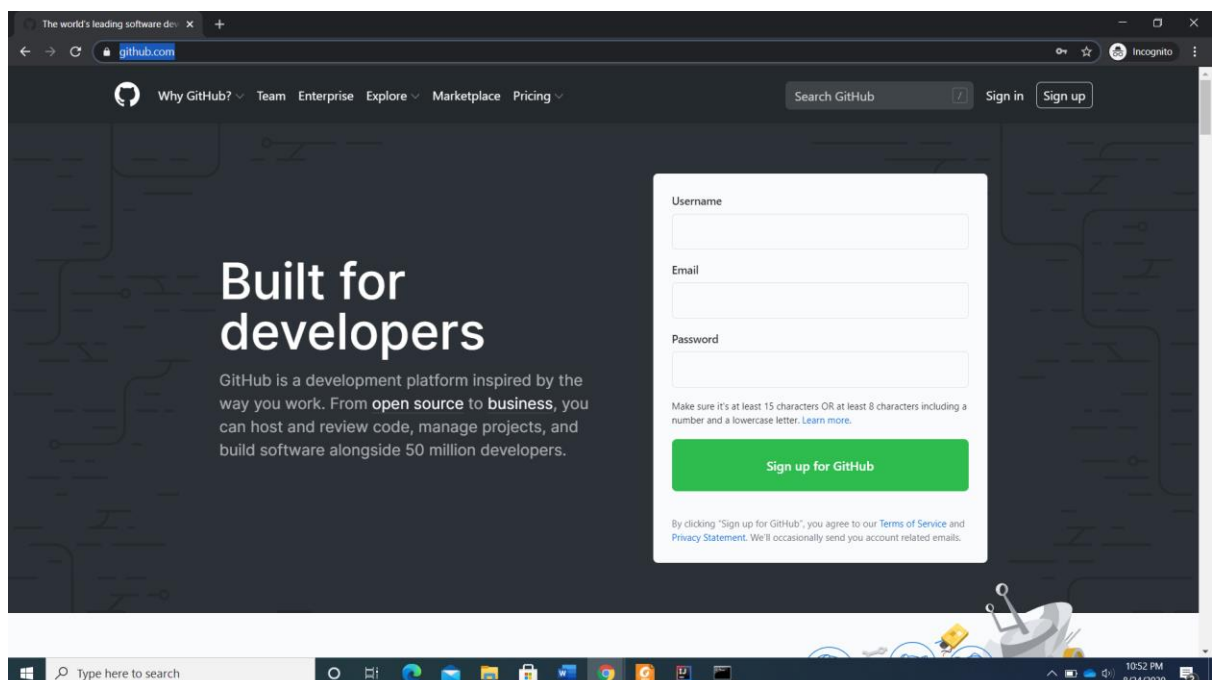
grow, mark and tweak your common history
  branch     List, create, or delete branches
  commit     Record changes to the repository
  merge      Join two or more development histories together
  rebase     Reapply commits on top of another base tip
  reset      Reset current HEAD to the specified state
  switch     Switch branches
  tag        Create, list, delete or verify a tag object signed with GPG

collaborate (see also: git help workflows)
  fetch      Download objects and refs from another repository
  pull       Fetch from and integrate with another repository or a local branch
  push       Update remote refs along with associated objects

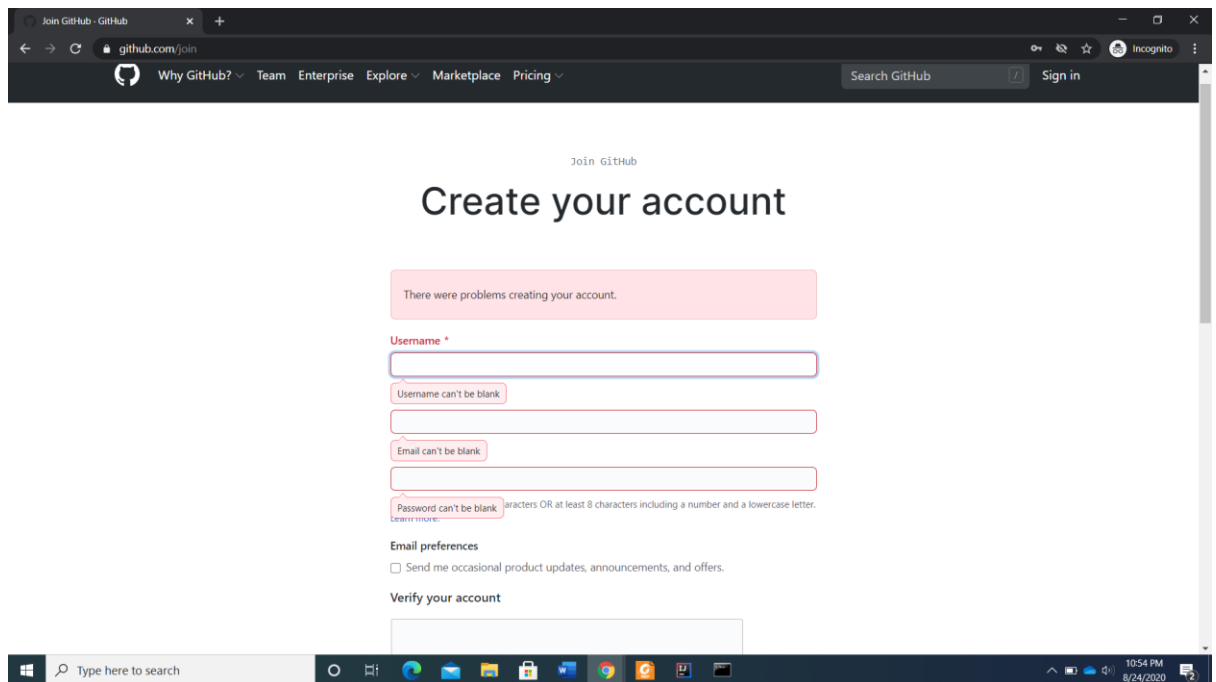
'git help -a' and 'git help -g' list available subcommands and some
concept guides. See 'git help <command>' or 'git help <concept>'
to read about a specific subcommand or concept.
See 'git help git' for an overview of the system.
```

Kemudian, karena kita akan melakukan Version Control System pada GitHub, kita perlu memiliki akun GitHub tersendiri. Karena itu, buatlah akun GitHub Anda dengan mengikuti langkah berikut:

1. Bukalah web GitHub dari link berikut: <https://github.com/>



2. Pilihlah *Sign up* untuk melakukan pendaftaran akun GitHub baru



Join GitHub · GitHub

github.com/join

Why GitHub? Team Enterprise Explore Marketplace Pricing

Search GitHub Sign in

Join GitHub

Create your account

There were problems creating your account.

Username *

Username can't be blank

Email can't be blank

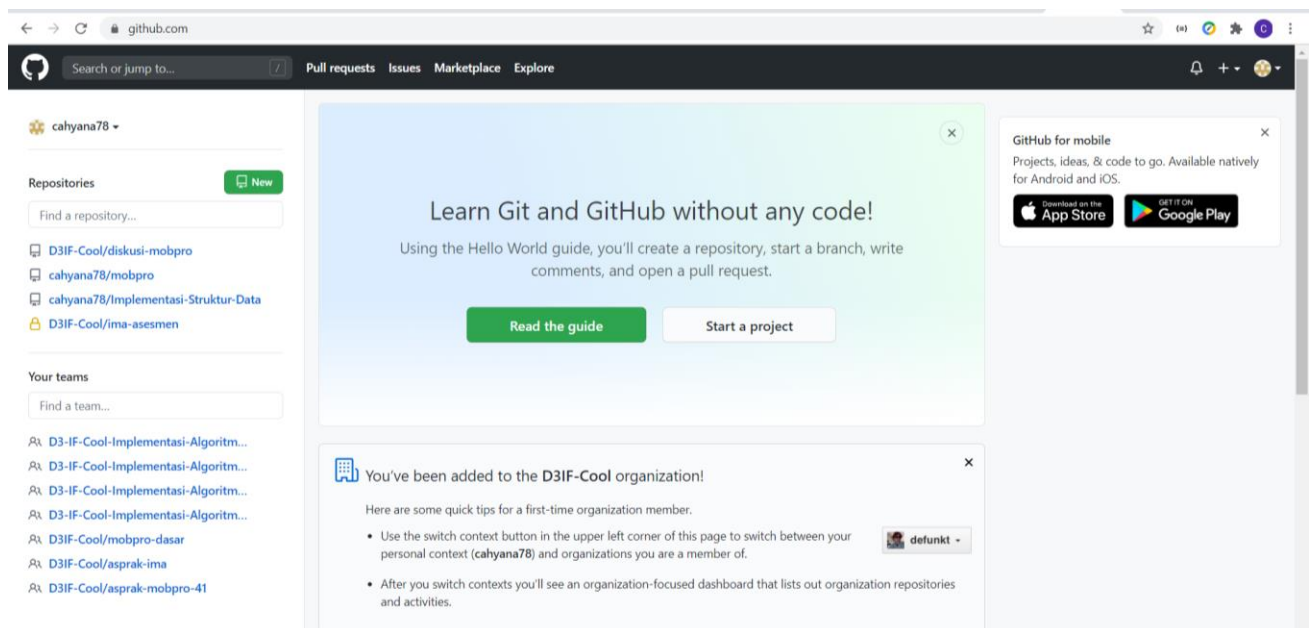
Password can't be blank. 8 characters OR at least 8 characters including a number and a lowercase letter. [Learn more...](#)

Email preferences

☐ Send me occasional product updates, announcements, and offers.

Verify your account

3. Silahkan isikan kolom yang diminta. Untuk *username*, diharapkan praktikan memilih *username* yang mudah dikenali untuk digunakan pada kelas IMA maupun kelas lain pada masa berikutnya (tidak disarankan menggunakan nama ambigu, atau yang sering disebut dengan alay). Jika berhasil, akan muncul tampilan seperti di bawah ini



github.com

Search or jump to...

Pull requests Issues Marketplace Explore

cahyana78

Repositories

Find a repository...

D3IF-Cool/diskusi-mobpro

cahyana78/mobpro

cahyana78/Implementasi-Struktur-Data

D3IF-Cool/ima-asesmen

Your teams

Find a team...

D3-IF-Cool-Implementasi-Algoritma...

D3-IF-Cool-Implementasi-Algoritma...

D3-IF-Cool-Implementasi-Algoritma...

D3-IF-Cool-Implementasi-Algoritma...

D3IF-Cool/mobpro-dasar

D3IF-Cool/asprak-ima

D3IF-Cool/asprak-mobpro-41

Learn Git and GitHub without any code!

Using the Hello World guide, you'll create a repository, start a branch, write comments, and open a pull request.

Read the guide Start a project

You've been added to the D3IF-Cool organization!

Here are some quick tips for a first-time organization member.

- Use the switch context button in the upper left corner of this page to switch between your personal context (cahyana78) and organizations you are a member of.
- After you switch contexts you'll see an organization-focused dashboard that lists out organization repositories and activities.

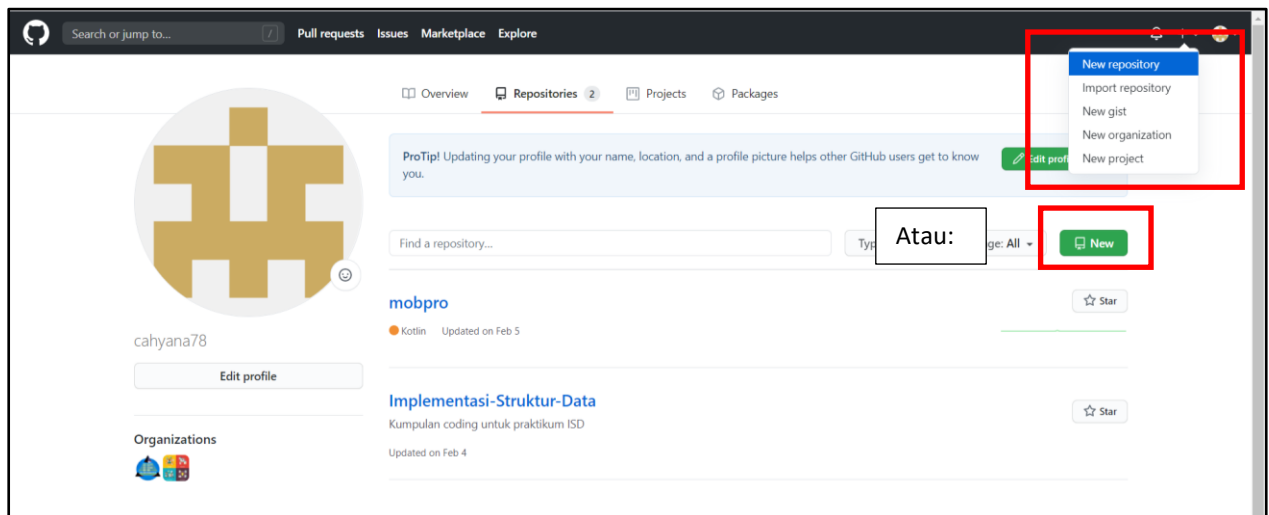
defunkt

GitHub for mobile

Projects, ideas, & code to go. Available natively for Android and iOS.

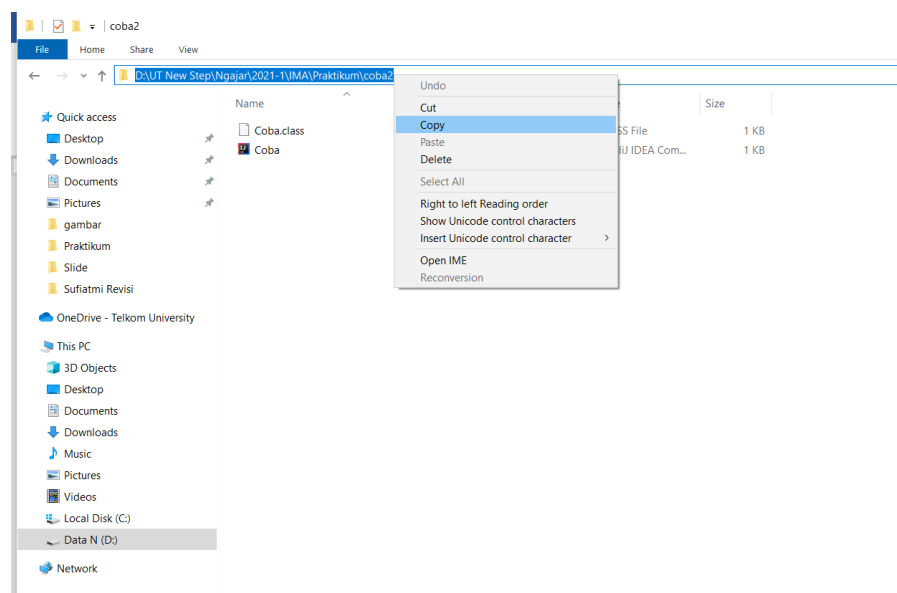
Download on the App Store GET IT ON Google Play

4. Buatlah suatu repository baru pada akun GitHub Anda untuk menyimpan berkas yang akan Anda lakukan VCS-nya.



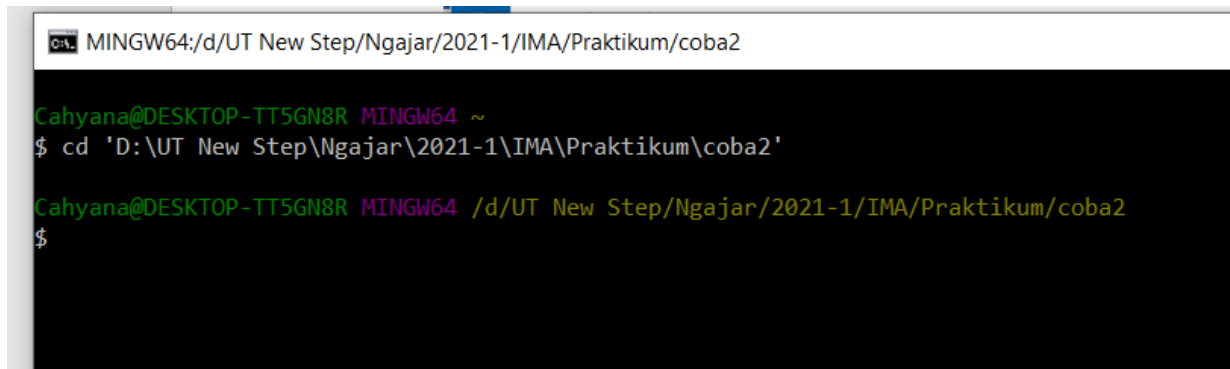
Setelah itu, untuk dapat menyimpan semua berkas yang telah dibuat pada akun GitHub tersebut, kita perlu untuk membuat *clone repository* dari repository pada akun GitHub yang telah Anda buat di dalam komputer Anda. Lakukan langkah berikut untuk melakukan *clone repository* pada komputer Anda:

1. Buka Git Bash
2. Ganti direktori pada Git Bash *command prompt* dengan direktori tempat Anda meletakkan projek/ file yang akan di-klone. Misal, file "Hello World" disimpan di direktori berikut (Untuk mendapatkan direktori, masuklah pada direktori tempat file/ projek disimpan, kemudian klik kanan pada *mouse* dan kopi alamat folder tersebut):



Ketikkan perintah berikut pada command line Git Bash:

```
cd 'D:\UT New Step\Ngajar\2021-1\IMA\Praktikum\coba2'
```

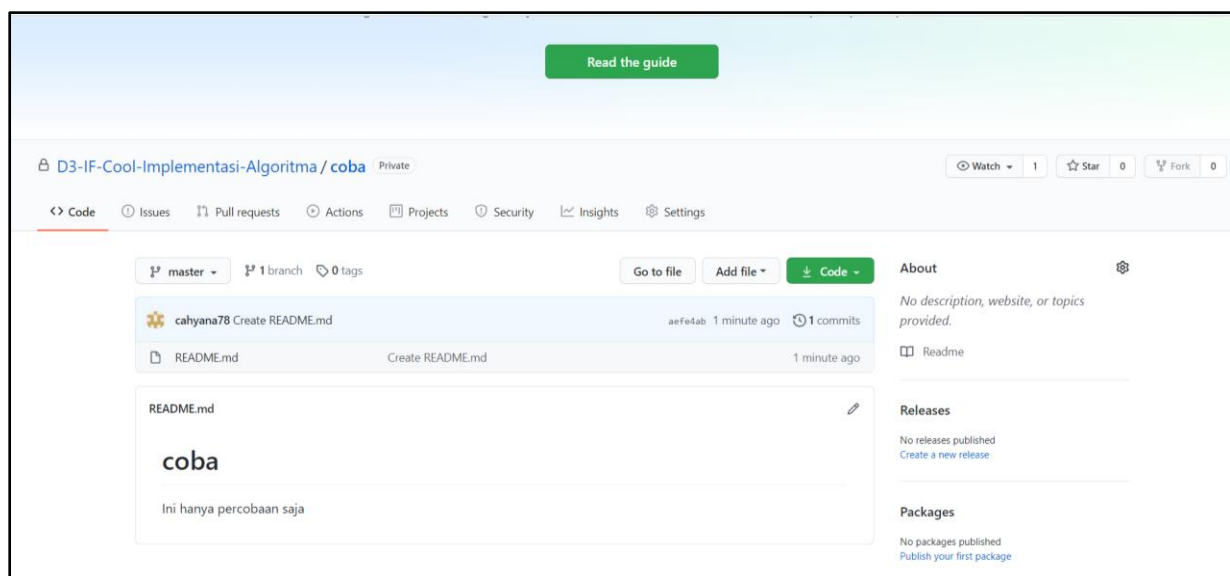


```
C:\% MINGW64:/d/UT New Step/Ngajar/2021-1/IMA/Praktikum/coba2

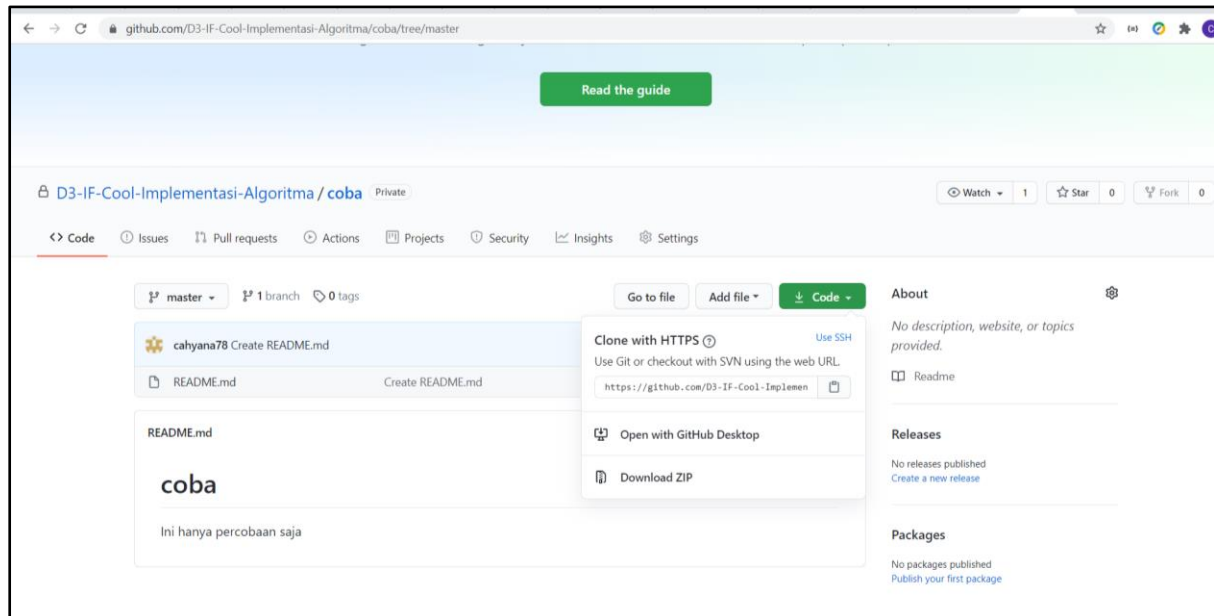
Cahyana@DESKTOP-TT5GN8R MINGW64 ~
$ cd 'D:\UT New Step\Ngajar\2021-1\IMA\Praktikum\coba2'

Cahyana@DESKTOP-TT5GN8R MINGW64 /d/UT New Step/Ngajar/2021-1/IMA/Praktikum/coba2
$
```

3. *Login*-lah ke dalam akun GitHub Anda, kemudian masuk ke repositori yang telah Anda buat dan akan dibuatkan clone-nya.



4. Klik “Clone or download” dan kopi URL nya.



5. Ketikkan perintah git clone diikuti dengan alamat web (URL) yang telah dikopi sebelumnya.
\$git clone <alamat web>

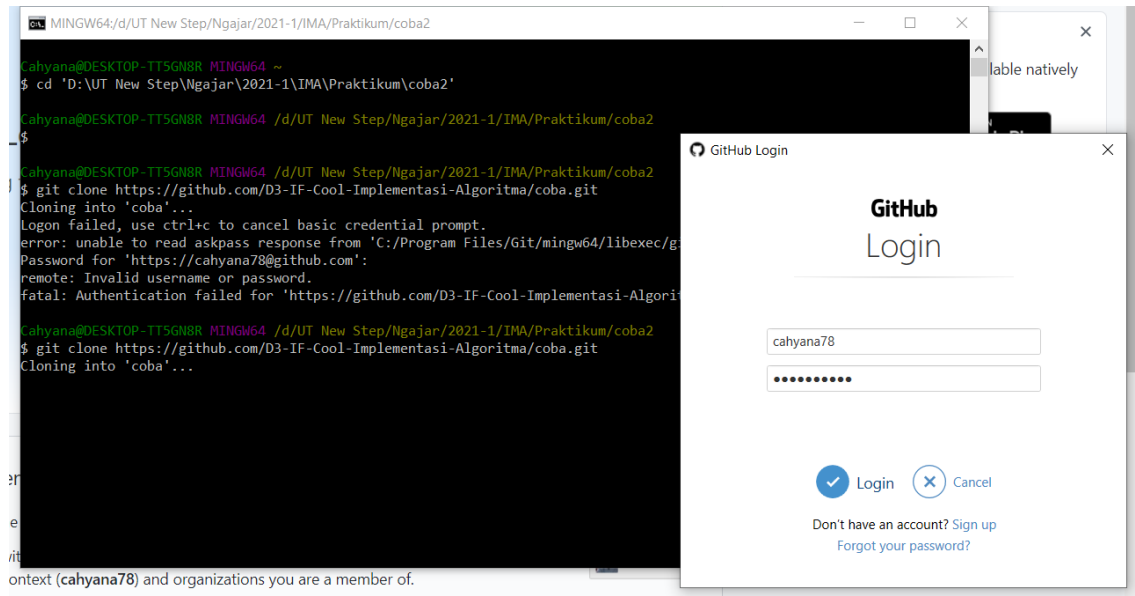
```
MINGW64:/d/UT New Step/Ngajar/2021-1/IMA/Praktikum/coba2

Cahyana@DESKTOP-TT5GN8R MINGW64 ~
$ cd 'D:\UT New Step\Ngajar\2021-1\IMA\Praktikum\coba2'

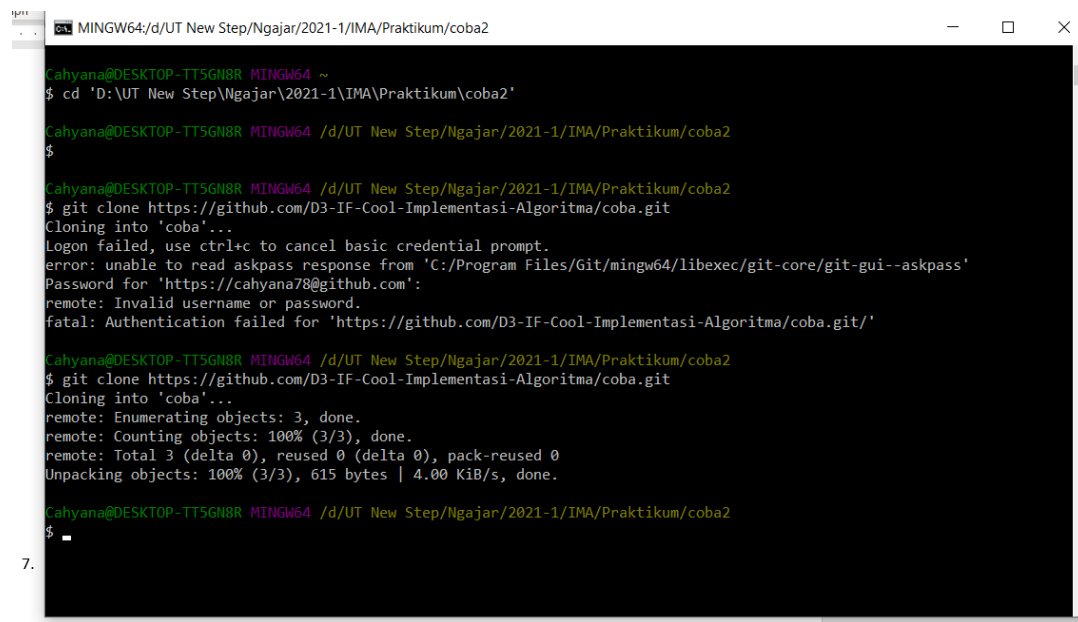
Cahyana@DESKTOP-TT5GN8R MINGW64 /d/UT New Step/Ngajar/2021-1/IMA/Praktikum/coba2
$

Cahyana@DESKTOP-TT5GN8R MINGW64 /d/UT New Step/Ngajar/2021-1/IMA/Praktikum/coba2
$ git clone https://github.com/D3-IF-Cool-Implementasi-Algoritma/coba.git
```

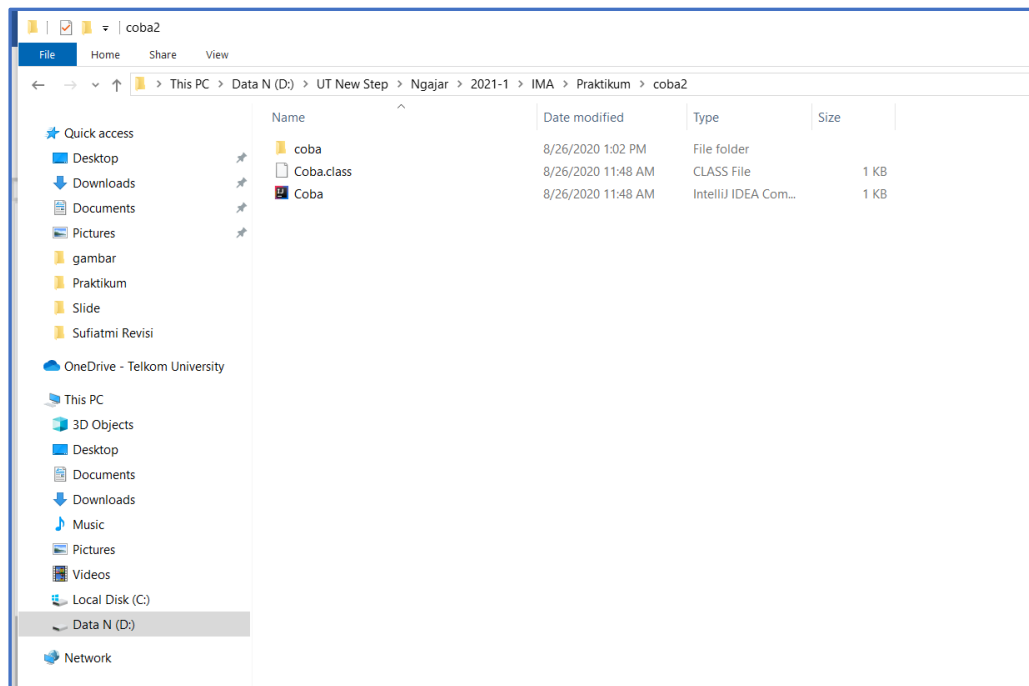
6. Press Enter. Masukkan *username* dan *password* akun GitHub yang telah Anda buat sebelumnya.



7. Jika sudah muncul *line* berikut, *clone* sukses dilakukan



Setelah menambahkan repositori pada GitHub, kita bisa membuat folder baru pada repositori tersebut. Pada contoh sebelumnya, repositori GitHub 'coba' telah di-klon pada folder 'coba2' yang ada Drive D, folder Praktikum (Gambar 1)

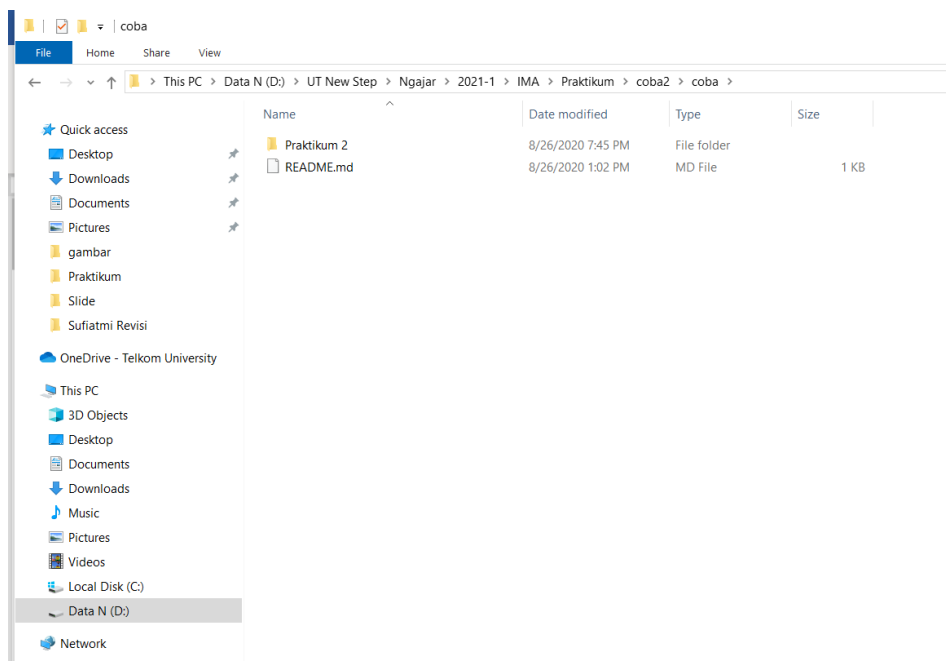


Gambar 1 Hasil clone

Agar terhubung dengan repositori pada akun GitHub yang telah dibuat, maka semua folder, proyek dan file harus diletakkan di dalam folder klon yang sudah terbentuk (pada contoh adalah folder 'coba').

Buatlah folder baru yang berisi file "Hello World" dari Praktikum 1 di dalam folder klon. Lakukan langkah berikut:

1. Buat folder baru di dalam repositori klon.



2. Pindahkan *file* yang dibuat pada Praktikum 1 ke dalam folder baru tersebut (folder Praktikum 2).
3. Bukalah folder tersebut, kopi alamat folder. Kemudian ketikkan `cd` dan alamat folder pada *command prompt* Git Bash. Bisa juga dilakukan dengan cara mengetikkan `cd` + men-drag dan drop folder yang akan dimasukkan.

```
$ cd "alamat folder"
```

```
Cahyana@DESKTOP-TT5GN8R MINGW64 /d/UT New Step/Ngajar/2021-1/IMA/Praktikum/coba2
$ cd "D:\UT New Step\Ngajar\2021-1\IMA\Praktikum\coba2\coba\Praktikum 2"
Cahyana@DESKTOP-TT5GN8R MINGW64 /d/UT New Step/Ngajar/2021-1/IMA/Praktikum/coba2/coba/Praktikum 2 (master)
$
```

4. Tambahkan file

```
$ git add 'yourFileName.java'
```

```
Cahyana@DESKTOP-TT5GN8R MINGW64 /d/UT New Step/Ngajar/2021-1/IMA/Praktikum/coba2
$ cd "D:\UT New Step\Ngajar\2021-1\IMA\Praktikum\coba2\coba\Praktikum 2"
Cahyana@DESKTOP-TT5GN8R MINGW64 /d/UT New Step/Ngajar/2021-1/IMA/Praktikum/coba2/coba/Praktikum 2 (master)
$ git add 'Coba.java'
Cahyana@DESKTOP-TT5GN8R MINGW64 /d/UT New Step/Ngajar/2021-1/IMA/Praktikum/coba2/coba/Praktikum 2 (master)
$
```

5. Lakukan commit pada file tersebut

```
$ git commit -m 'yourFileName.java'
```

```
Cahyana@DESKTOP-TT5GN8R MINGW64 /d/UT New Step/Ngajar/2021-1/IMA/Praktikum/coba2
$ cd "D:\UT New Step\Ngajar\2021-1\IMA\Praktikum\coba2\coba\Praktikum 2"
Cahyana@DESKTOP-TT5GN8R MINGW64 /d/UT New Step/Ngajar/2021-1/IMA/Praktikum/coba2/coba/Praktikum 2 (master)
$ git add 'Coba.java'
Cahyana@DESKTOP-TT5GN8R MINGW64 /d/UT New Step/Ngajar/2021-1/IMA/Praktikum/coba2/coba/Praktikum 2 (master)
$ git commit -m 'Coba.java'
[master 7a0da29] Coba.java
1 file changed, 7 insertions(+)
create mode 100644 Praktikum 2/Coba.java
Cahyana@DESKTOP-TT5GN8R MINGW64 /d/UT New Step/Ngajar/2021-1/IMA/Praktikum/coba2/coba/Praktikum 2 (master)
$
```

6. Lakukan push untuk membuat perubahan pada GitHub

```
$ git push origin master
```

```
Cahyana@DESKTOP-TT5GN8R MINGW64 /d/UT New Step/Ngajar/2021-1/IMA/Praktikum/coba2
$ cd "D:\UT New Step\Ngajar\2021-1\IMA\Praktikum\coba2\coba\Praktikum 2"

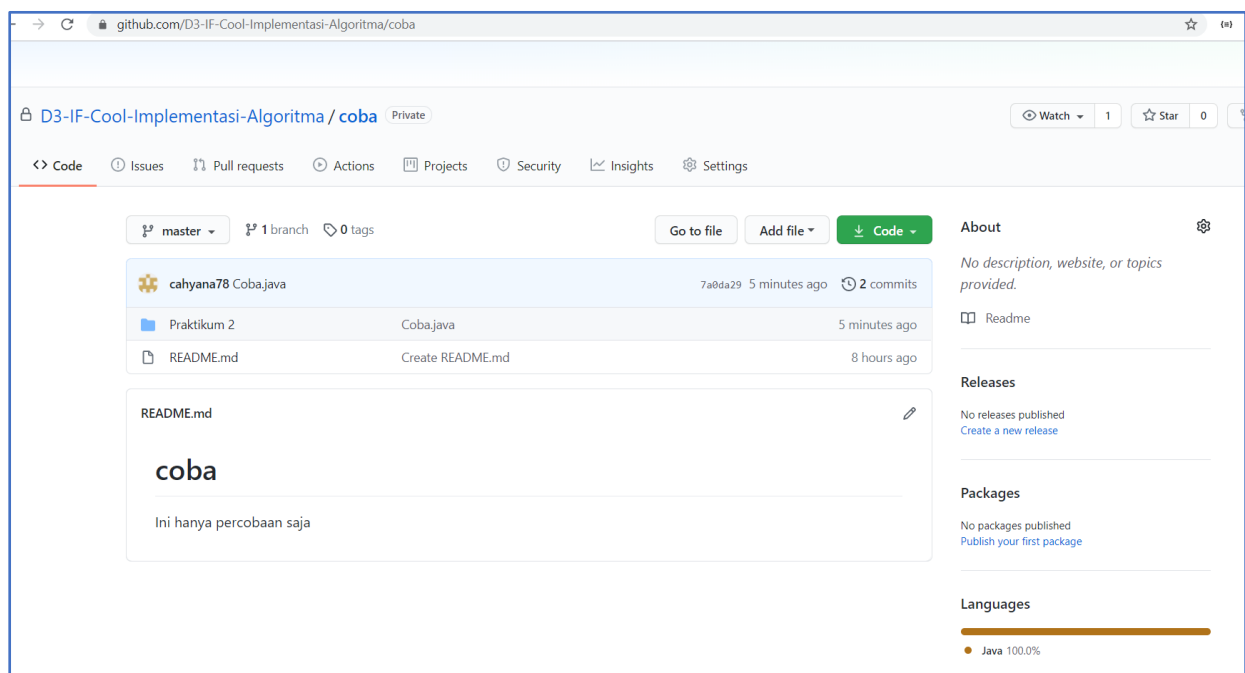
Cahyana@DESKTOP-TT5GN8R MINGW64 /d/UT New Step/Ngajar/2021-1/IMA/Praktikum/coba2/coba/Praktikum 2 (master)
$ git add 'Coba.java'

Cahyana@DESKTOP-TT5GN8R MINGW64 /d/UT New Step/Ngajar/2021-1/IMA/Praktikum/coba2/coba/Praktikum 2 (master)
$ git commit -m 'Coba.java'
[master 7a0da29] Coba.java
1 file changed, 7 insertions(+)
create mode 100644 Praktikum 2/Coba.java

Cahyana@DESKTOP-TT5GN8R MINGW64 /d/UT New Step/Ngajar/2021-1/IMA/Praktikum/coba2/coba/Praktikum 2 (master)
$ git push origin master
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 4 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (4/4), 432 bytes | 108.00 KiB/s, done.
Total 4 (delta 0), reused 0 (delta 0)
To https://github.com/D3-IF-Cool-Implementasi-Algoritma/coba.git
aefe4ab..7a0da29 master -> master

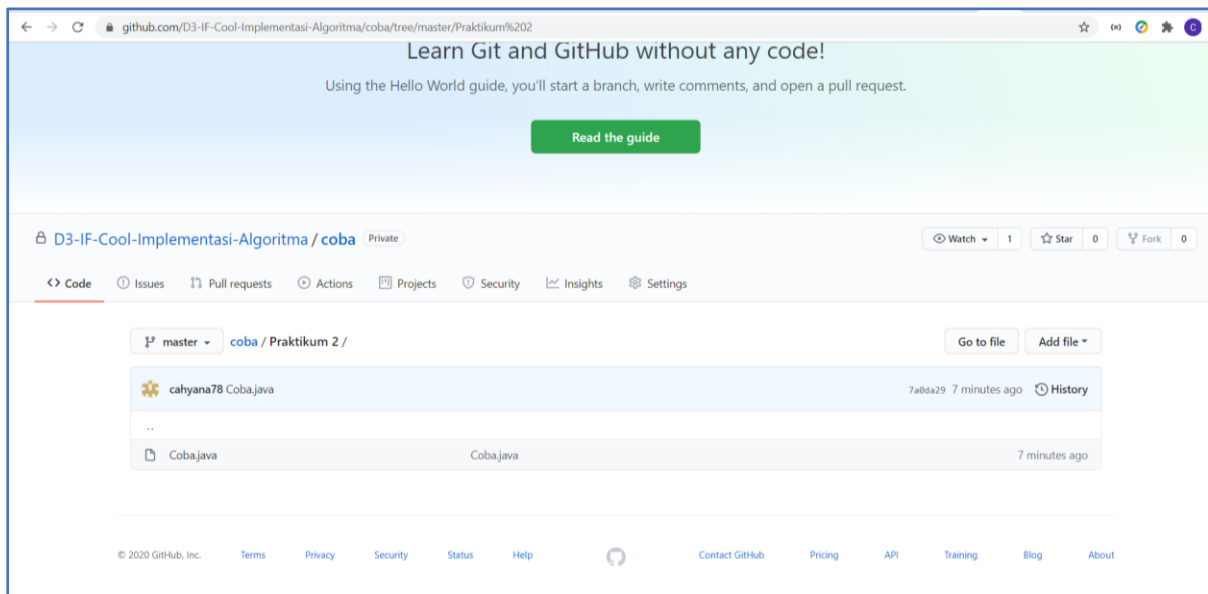
Cahyana@DESKTOP-TT5GN8R MINGW64 /d/UT New Step/Ngajar/2021-1/IMA/Praktikum/coba2/coba/Praktikum 2 (master)
$
```

Pada praktikum kali ini, kita melakukan penambahan file pada *branch* master, karena kita belum menambahkan *fork branch* pada repositori GitHub yang kita buat. Hasil commit dan push yang telah dilakukan dapat dilihat pada repositori akun GitHub seperti terlihat pada Gambar 2.



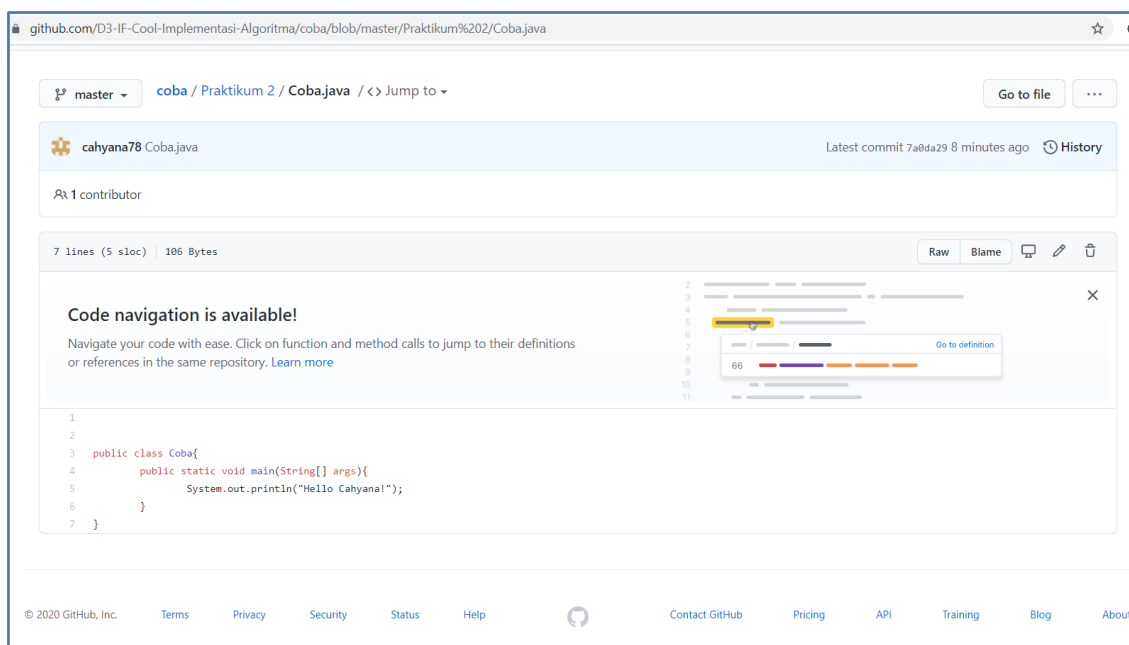
Gambar 2 Hasil add folder

Ketika folder Praktikum 2 dibuka, maka akan terlihat file yang baru saja kita push ke dalam GitHub (Gambar 3).



Gambar 3 Hasil push

Isi dari file yang sudah kita push adalah sama dengan yang kita simpan pada folder di komputer kita (Gambar 4).



Gambar 4 Isi dari file

Reference

Git, *--distributed-even-if-your-workflow-isnt*, Git. <https://git-scm.com/> , diakses tanggal 24-08-2020

Git, *Memulai - Tentang Version Control*, Git. <https://git-scm.com/> , diakses tanggal 24-08-2020(2012)

Jean-Christophe-Chouinard, *Clone a Github Repository on Your Computer*,
<https://www.jcchouinard.com/clone-github-repository-on-windows/>, diakses tanggal 26-08-2020

Jean-Christophe-Chouinard, *Add a File to Github Using Git Bash*, <https://www.jcchouinard.com/add-a-file-to-github-with-git-bash/>, diakses tanggal 26-08-2020