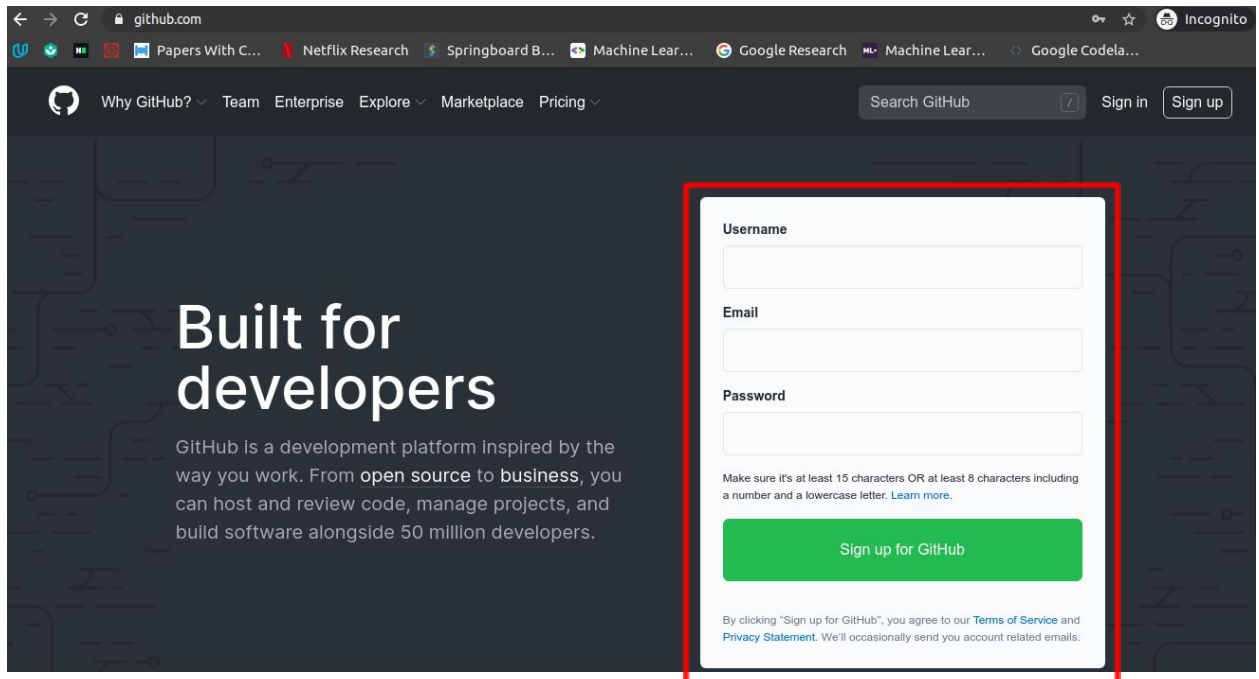


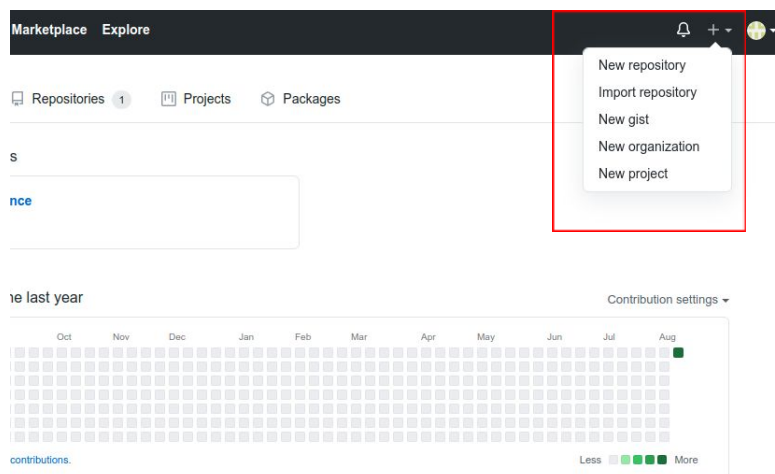
# Membuat akun GitHub

1. Buka website github.com
2. Daftarkan akun baru anda pada section yang diberi kotak berwarna merah dan ikuti step by step nya



# Membuat repository baru pada GitHub

1. Setelah anda berhasil membuat akun GitHub, anda dapat membuat repository baru dengan meng-klik tanda + yang ada di pojok kanan atas seperti gambar dibawah.



2. Isi informasi pada repository
  - a. Nama repository (mandatory)
  - b. Deskripsi repository (optional)
  - c. Pilih apakah repository anda akan available untuk public atau bersifat private / tidak bisa dilihat orang lain
  - d. Centang **Initialize this repository with a README** jika anda ingin memberikan dokumentasi pada repository tersebut

### Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere?

[Import a repository.](#)

Owner \*



naufalabdila4 ▾

Repository name \*

Great repository names are short and memorable. Need inspiration? How about **special-parakeet?**

Description (optional)



**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.

Skip this step if you're importing an existing repository.



**Initialize this repository with a README**

This will let you immediately clone the repository to your computer.

Add .gitignore: None ▾

Add a license: None ▾



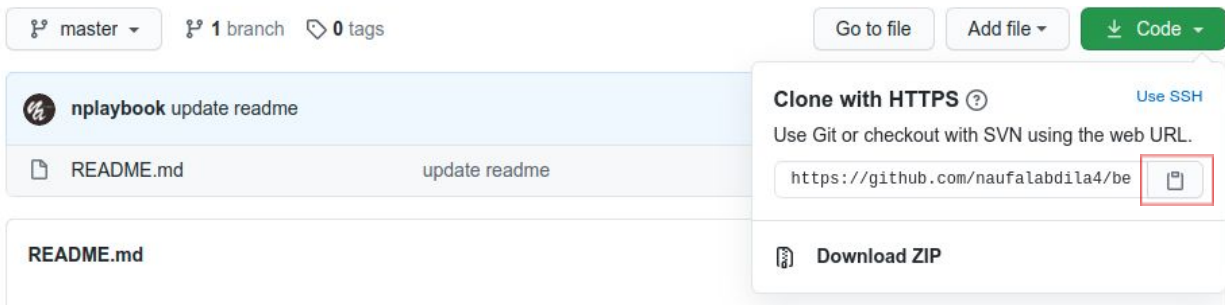
Create repository

3. Setelah mengisi semua informasi, klik Create Repository

## Kloning repository pada GitHub ke laptop / local

1. Setelah anda berhasil membuat repository baru, maka anda akan masuk ke halaman repository tersebut
2. Untuk melakukan cloning repository, klik **Code** pada tombol yang berwarna hijau

3. Kemudian copy alamat URL yang muncul dengan klik tombol clipboard yang ada di sebelah kanan (ditandai dengan kotak merah)



4. Kemudian, buka terminal (jika menggunakan Mac atau Linux) atau Git Bash (jika menggunakan windows)
5. Arahkan direktori (menggunakan fungsi **cd** pada terminal) ke folder machine\_learning\_workshop dan berikutnya arahkan kembali ke Part 4 - Data Analysis Process
6. Kemudian ketik **git clone** dan paste **URL** yang sebelumnya sudah anda copy, lihat contoh di bawah (URL di bawah adalah contoh, gunakan URL dari repository yang anda buat).

```
git clone https://github.com/naufalabdila4/asca.git
```

7. Jika proses sudah selesai, anda bisa cek folder Part 4 - Data Analysis Process maka disana anda akan menemukan folder baru dengan nama repository yang anda buat.

## Menyimpan project di GitHub

1. Untuk bisa menyimpan project di GitHub, pastikan file anda sudah di save atau dimasukkan ke dalam folder hasil kloning dari GitHub
2. Berikutnya, buka terminal / Git Bash anda dan arahkan ke folder tersebut (arahkan ke data **D**, kemudian gunakan **cd** untuk pindah directory)
3. Kemudian, anda ketik **git status** seperti pada contoh di bawah dan anda akan melihat file berwarna merah yang belum di track oleh git

```
belajar_data_science$ git status
```

```
No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
  .gitkeep
  Part 3 - conda-environment.ipynb

nothing added to commit but untracked files present (use "git add" to track)
```

4. Anda bisa memasukkan data berwarna merah atau yang belum di track tersebut ke dalam list data yang ingin anda track dengan menggunakan **git add** .

```
belajar_data_science$ git add .
```

5. Jika kita cek lagi menggunakan **git status** maka file yang sebelumnya berwarna merah akan berubah menjadi berwarna hijau, artinya setiap perubahan atau modifikasi pada file tersebut akan di track oleh git

```
belajar_data_science$ git status
```

```
No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
    new file:   .gitkeep
    new file:   Part 3 - conda environment.ipynb
```

6. Langkah berikutnya adalah melakukan **git commit**, yang terjadi pada git commit adalah kita menyimpan file tersebut ke dalam git sehingga yang awalnya kita punya beberapa file dengan status **new file** atau **modified** akan di save.

```
belajar_data_science$ git commit -m "ini commit pertama saya"
```

Kalimat setelah **-m** adalah commit message dimana kita memberikan penjelasan singkat apakah ada file baru atau ada perubahan yang kita lakukan pada file tersebut.

7. Langkah terakhir adalah dengan melakukan **git push**. Dengan melakukan git push, maka kita akan menyimpan file yang sudah di commit ke repository GitHub.

```
belajar_data_science$ git push origin master
```

8. Jika git push sudah berhasil, kita bisa cek di GitHub dan pastikan directory kita sudah terupdate.