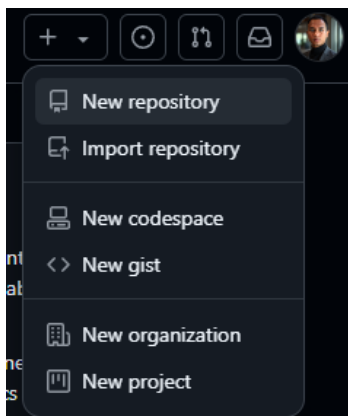


Homework Sesi 18 Bootcamp DevOps Engineer Digital Skola

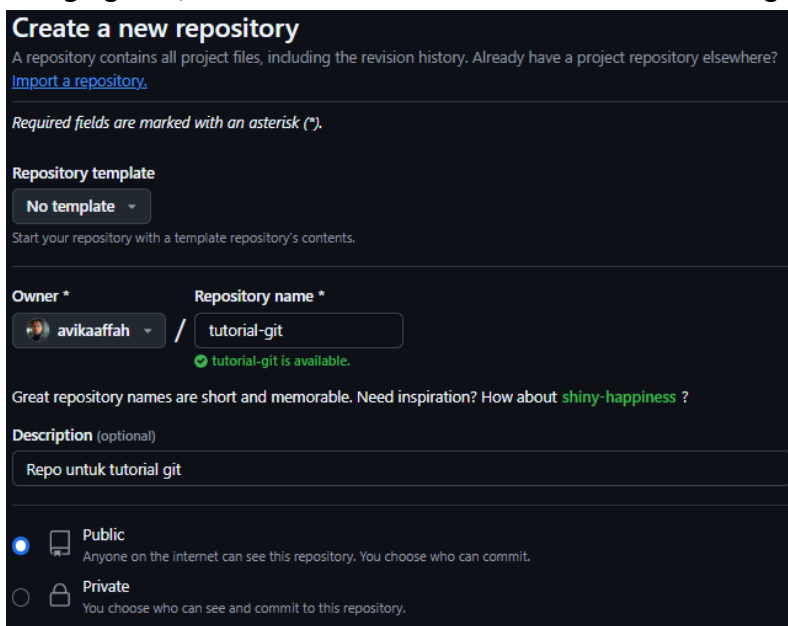
1. Buat 1 repo di github.com dengan public access
2. Clone repo tersebut
3. Create file test1.txt
4. Push file tersebut ke repo

Langkah 1: Membuat Repository di GitHub

1. Buka situs web GitHub di <https://github.com/> dan login ke akun.
2. Setelah login, klik tombol "+" di pojok kanan atas halaman, lalu pilih "New repository".



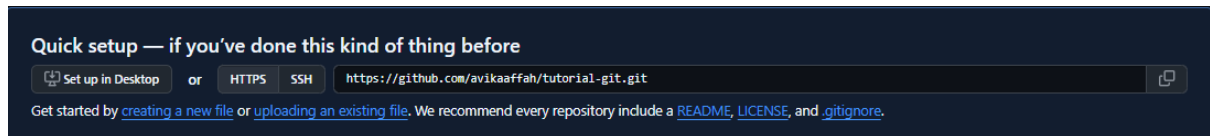
3. Pada halaman "Create a new repository", isi detail berikut:
 - **Repository name:** Berikan nama untuk repository. Contohnya, tutorial-git.
 - **Description (opsional):** Bisa menambahkan deskripsi singkat tentang repository.
 - **Public:** Pastikan opsi "Public" terpilih agar repository dapat diakses oleh siapa saja.
 - **Initialize this repository with:** Bisa memilih untuk menambahkan README file, .gitignore, atau lisensi. Untuk tutorial ini biarkan kosong.

A screenshot of the 'Create a new repository' form on GitHub. The form has a dark theme. At the top, it says 'Create a new repository' and provides a brief explanation of a repository. Below this, there is a section for 'Repository template' with a dropdown menu set to 'No template'. The 'Owner' field is set to 'avikaaffah' and the 'Repository name' field is set to 'tutorial-git'. A green checkmark indicates that 'tutorial-git' is available. The 'Description' field is optional and contains the text 'Repo untuk tutorial git'. At the bottom, there are two radio buttons for visibility: 'Public' (selected) and 'Private'. The 'Public' option is described as 'Anyone on the internet can see this repository. You choose who can commit.' The 'Private' option is described as 'You choose who can see and commit to this repository.'

4. Klik tombol "Create repository" di bagian bawah halaman.

Langkah 2: Mengklon Repository

1. Setelah repository berhasil dibuat, akan diarahkan ke halaman repository.
2. Akan terlihat beberapa opsi untuk mendapatkan URL repository. Pilih opsi "SSH" (SSH key sudah dikonfigurasi di GitHub).



3. Buka terminal atau command prompt di komputer. Buat dan pindah ke direktori tempat ingin menyimpan proyek ini.

```
kaisenber@MENOMEN:~$ mkdir project-github
kaisenber@MENOMEN:~$ cd project-github/
kaisenber@MENOMEN:~/project-github$
```

4. Gunakan perintah git clone diikuti dengan URL repository yang sudah disalin.

```
kaisenber@MENOMEN:~/project-github$ git clone git@github.com:avikaaffah/tutorial-git.git
```

5. Setelah perintah git clone selesai dijalankan, sebuah folder dengan nama repository akan dibuat di direktori saat ini.

```
kaisenber@MENOMEN:~/project-github$ ls
tutorial-git
```

Langkah 3: Membuat File test1.txt

1. Pindah ke dalam folder repository yang baru saja di-clone menggunakan perintah cd.

```
kaisenber@MENOMEN:~/project-github$ cd tutorial-git/
```

2. Buat file bernama test1.txt.

```
kaisenber@MENOMEN:~/project-github/tutorial-git$ touch test1.txt
kaisenber@MENOMEN:~/project-github/tutorial-git$ ls
test1.txt
```

Langkah 4: Mendorong (Push) File ke Repository

1. Setelah file test1.txt dibuat dan disimpan di dalam folder repository lokal, perlu menambahkannya ke staging area Git. Jalankan perintah **git add test1.txt** di terminal. Jika ingin menambahkan semua perubahan yang ada di direktori, bisa menggunakan perintah **git add**.

```
kaisenber@MENOMEN:~/project-github/tutorial-git$ git add test1.txt
```

2. Selanjutnya, perlu melakukan commit untuk menyimpan perubahan secara lokal.
git commit -m "Menambahkan file test1.txt"
-m diikuti dengan pesan commit yang menjelaskan perubahan yang dilakukan.

```
kaisenberg@MENOMEN:~/project-github/tutorial-git$ git commit -m "Menambahkan file test1.txt"
[main (root-commit) f96c173] Menambahkan file test1.txt
Committer: kaisenberg <kaisenberg@MENOMEN>
Your name and email address were configured automatically based
on your username and hostname. Please check that they are accurate.
You can suppress this message by setting them explicitly:

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

After doing this, you may fix the identity used for this commit with:

    git commit --amend --reset-author

1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 test1.txt
```

3. Terakhir, untuk mengirim perubahan lokal ke repository di GitHub, perlu melakukan push.

```
kaisenberg@MENOMEN:~/project-github/tutorial-git$ git remote -v
origin  git@github.com:avikaaffah/tutorial-git.git (fetch)
origin  git@github.com:avikaaffah/tutorial-git.git (push)
kaisenberg@MENOMEN:~/project-github/tutorial-git$ git push origin main
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Writing objects: 100% (3/3), 219 bytes | 219.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To github.com:avikaaffah/tutorial-git.git
* [new branch]      main -> main
```

origin adalah nama default untuk remote repository (biasanya mengarah ke repository di GitHub), dan main adalah nama default untuk branch utama. Jika repository menggunakan nama branch master, ganti main dengan master.

