

LAPORAN PRAKTIKUM 2

DASAR PEMROGRAMAN



Rangga Dwi Saputra

2341720248

Kelas 1B

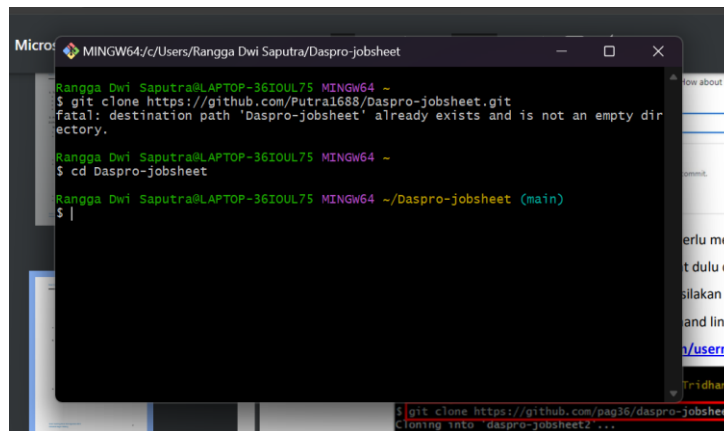
Prodi D-IV Teknik Informatika

JOBSHEET 2

Sistem Version Control dan Kanbar Board

A. Percobaan 1: Menggunakan Github

1. Buka situs web GitHub di <https://github.com>
2. Klik tombol "**Sign up**" (Daftar) untuk membuat akun GitHub.
3. Ikuti petunjuk untuk mengisi informasi yang diperlukan dan verifikasi email Anda.
4. Setelah masuk ke akun GitHub Anda, klik tombol "+", lalu pilih "New repository" (Repositori baru).
5. Isi nama repositori, deskripsi (opsional), dan konfigurasi lainnya.
6. Anda dapat membuat repositori publik atau pribadi sesuai kebutuhan. Jika sudah jangsan lupa untuk klik tombol "**Create repository**".
7. Untuk mengelola repositori secara lokal, Anda perlu mengklonnya ke komputer Anda. Tetapi, perlu melakukan instalasi git client dulu di <https://git-scm.com/downloads> , setelah selesai download silakan melakukan instalasi.
8. Windows run, ketik dan pilih Git Bash
9. Gunakan perintah git clone dari terminal (command line) untuk mengklon repositori. Perintah umumnya (\$ git clone <https://github.com/username/nama-repositori.git>



```
Microsoft Windows [MINGW64/c/Users/Rangga Dwi Saputra/Daspro-jobsheet]
Rangga Dwi Saputra@LAPTOP-36IOUL75 MINGW64 ~
$ git clone https://github.com/putra1688/daspro-jobsheet.git
fatal: destination path 'daspro-jobsheet' already exists and is not an empty directory.

Rangga Dwi Saputra@LAPTOP-36IOUL75 MINGW64 ~
$ cd Daspro-jobsheet

Rangga Dwi Saputra@LAPTOP-36IOUL75 MINGW64 ~/Daspro-jobsheet (main)
$ |
```

10. Buat atau edit berkas-berkas di dalam repositori sesuai kebutuhan. Buka folder repository menggunakan Visual Studio Code -> Open folder lalu drag file Daspro-Jobsheet (sesuai letak tempatnya)
11. Tambahkan file dengan klik kanan – **New File**, berikana nama file tersebut dengan nama "**README.md**"
12. Isikan file "**README.md**" dengan isi seperti berikut

```
welcome  README.md
1  README.md > # Dasar Pemrograman - Jobsheet 2
2  # Dasar Pemrograman - Jobsheet 2
3  Hello Github, ini adalah repository pertama saya
4
```

13. Setelah selesai, simpan perubahan Anda dan komit dengan perintah **git commit**. Anda akan diminta untuk **memberikan pesan komit** yang menjelaskan perubahan yang telah Anda lakukan.

```
Rangga Dwi Saputra@LAPTOP-36IOUL75 MINGW64 /d/Semester 1/Dasar Pemrograman/Daspro-jobsheet (main)
$ git add .

Rangga Dwi Saputra@LAPTOP-36IOUL75 MINGW64 /d/Semester 1/Dasar Pemrograman/Daspro-jobsheet (main)
$ git commit -m "commit pertama saya lho"
Author identity unknown

*** Please tell me who you are.

Run

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

to set your account's default identity.
Omit --global to set the identity only in this repository.

fatal: unable to auto-detect email address (got 'Rangga Dwi Saputra@LAPTOP-36IOUL75.(none)')

Rangga Dwi Saputra@LAPTOP-36IOUL75 MINGW64 /d/Semester 1/Dasar Pemrograman/Daspro-jobsheet (main)
$ git ck=git config --global user.email "ranggadsaputra11@gmail.com"
```

System tidak mengenali akun mana yang terhubung ke github, maka dari itu perlu menjalankan intruksi git config –global user.email “akun yang terhubung ke github”

```
Rangga Dwi Saputra@LAPTOP-36IOUL75 MINGW64 /d/Semester 1/Dasar Pemrograman/Daspro-jobsheet (main)
$ git ck=git config --global user.email "ranggadsaputra11@gmail.com"
git: 'ck=git' is not a git command. See 'git --help'.

Rangga Dwi Saputra@LAPTOP-36IOUL75 MINGW64 /d/Semester 1/Dasar Pemrograman/Daspro-jobsheet (main)
$ git config --global user.email "ranggadsaputra11@gmail.com"

Rangga Dwi Saputra@LAPTOP-36IOUL75 MINGW64 /d/Semester 1/Dasar Pemrograman/Daspro-jobsheet (main)
$ git commit -m "commit pertama saya lho"
[main (root-commit) 1f84ad6] commit pertama saya lho
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 README.md
```

14. Untuk memperbarui repositori di GitHub dengan perubahan yang telah Anda lakukan secara lokal, gunakan perintah git push.

```
Rangga Dwi Saputra@LAPTOP-36IOUL75 MINGW64 /d/Semester 1/Dasar Pemrograman/Daspro-jobshe
$ git push origin main
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Writing objects: 100% (3/3), 222 bytes | 111.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/Putra1688/Daspro-jobsheet.git
* [new branch]      main -> main
```

Normalnya untuk push ke repository akan meminta user atau password untuk menampilkan system seperti di atas sebagai indikatornya.

Tetapi jika tidak seperti diatas atau tidak diminta untuk memasukkan akun dan password, yang perlu dilakukan adalah membuat token untuk push repository.

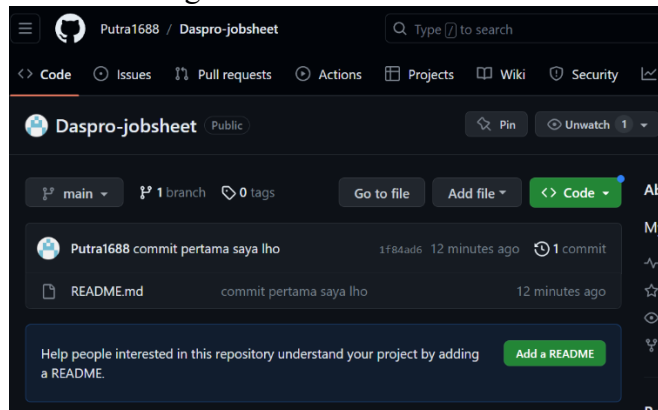
15. Klik akun Anda – Settings – Developer Settings – Tokens (classic) – Generate new token (classic). Isikan bagian Note, Expiration, dan Select scopes. Jika sudah klik tombol Generate token

Simpan token tersebut karena tidak bisa dilihat kembali untuk digunakan push yang selanjutnya.

16. Silakan jalankan perintah git push

[https://\[token\]@github.com/username/namarepository.git](https://[token]@github.com/username/namarepository.git)

17. Cek halaman github anda



18. Jika Langkah 14 sudah berhasil login, maka langsung lompat ke Langkah 17.

Pertanyaan

1. Jelaskan perbedaan perintah git commit dan git push?

Jawab: Git commit menjelaskan perubahan yang kita lakukan. Sedangkan Git push, memperbarui repositori di GitHub dengan perubahan yang telah kita lakukan secara local

2. Apakah bisa alurnya dibalik, membuat folder atau proyek terlebih dahulu kemudian upload (push) ke Github? Jika bisa, buktikan!

Jawaban: Bisa, berikut langkah-langkah yang dapat dilakukan:

- 1) Setelah membuat direktori atau folder proyek pada computer, buka terminal atau command prompt dan navigasikan ke direktori proyek tersebut menggunakan perintah cd
- 2) Jalankan perintah git init untuk menginisiasikan Git didalam direktori proyek

- 3) Salin atau buat file dan direktori proyek yang sudah dibuat. Setelah itu, tambahkan file-file ke repositori lokal menggunakan perintah git add
- 4) Commit perubahan yang sudah ditambahkan
- 5) Buka situs github akun anda lalu klik tombol +
- 6) Setelah membuat repositori di GitHub, Anda akan melihat instruksi untuk mengunggah proyek yang sudah ada. Anda dapat mengikuti instruksi ini untuk menghubungkan repositori lokal dengan repositori GitHub Anda. Biasanya, langkah-langkah ini melibatkan perintah git remote add origin untuk menghubungkan repositori lokal dengan repositori GitHub, dan kemudian perintah git push -u origin master untuk mengunggah perubahan ke GitHub.

B. Percobaan 2: Dasar Kolaborasi di Github

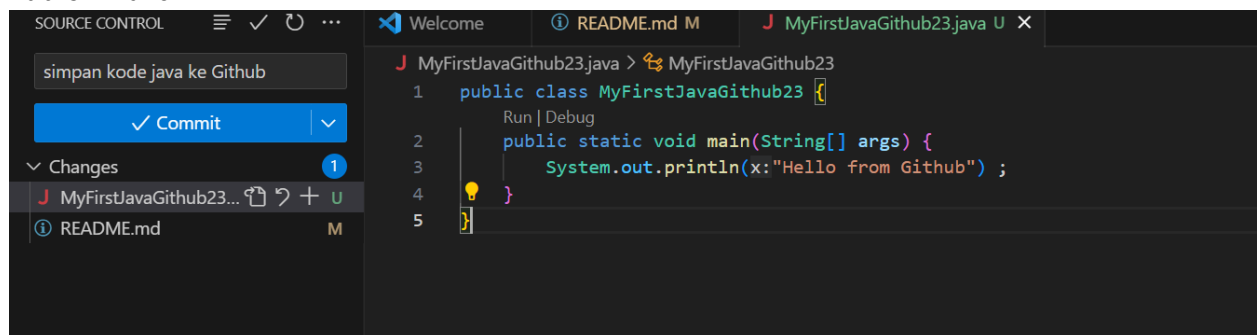
1. Sebelum memulai pekerjaan pada proyek, sebaiknya membuat branch terlebih dahulu untuk mengisolasi perubahan Anda dari cabang utama (biasanya "main" atau "master").
2. Gunakan perintah git branch nama-branch untuk membuat branch baru dan git checkout nama-branch untuk beralih ke branch tersebut.

```
Rangga Dwi Saputra@LAPTOP-36IOUL75 MINGW64 /d/Semester 1/Dasar Pemrograman/Daspr
o-jobsheet (main)
$ git branch devel

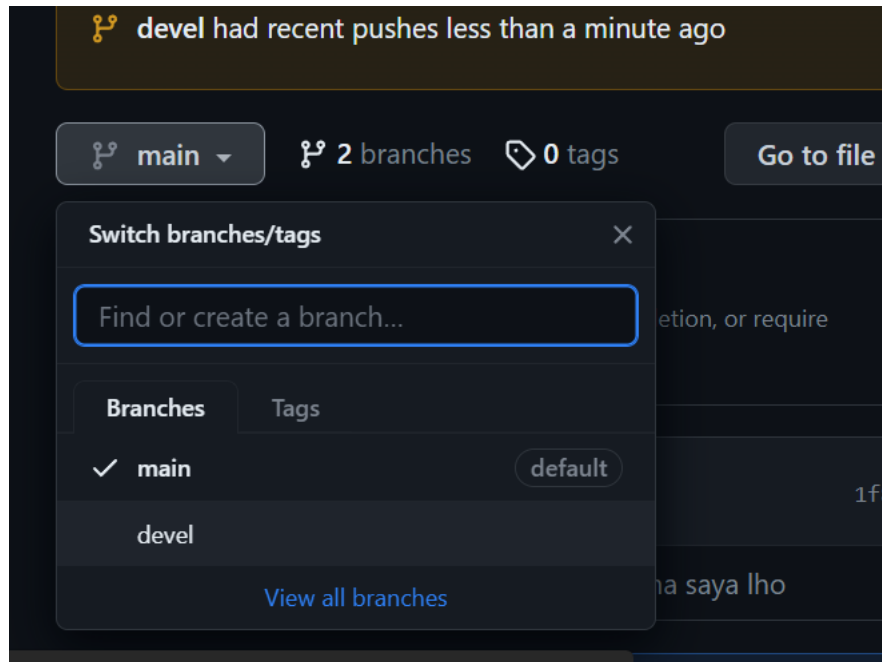
Rangga Dwi Saputra@LAPTOP-36IOUL75 MINGW64 /d/Semester 1/Dasar Pemrograman/Daspr
o-jobsheet (main)
$ git checkout devel
Switched to branch 'devel'

Rangga Dwi Saputra@LAPTOP-36IOUL75 MINGW64 /d/Semester 1/Dasar Pemrograman/Daspr
o-jobsheet (devel)
$ |
```

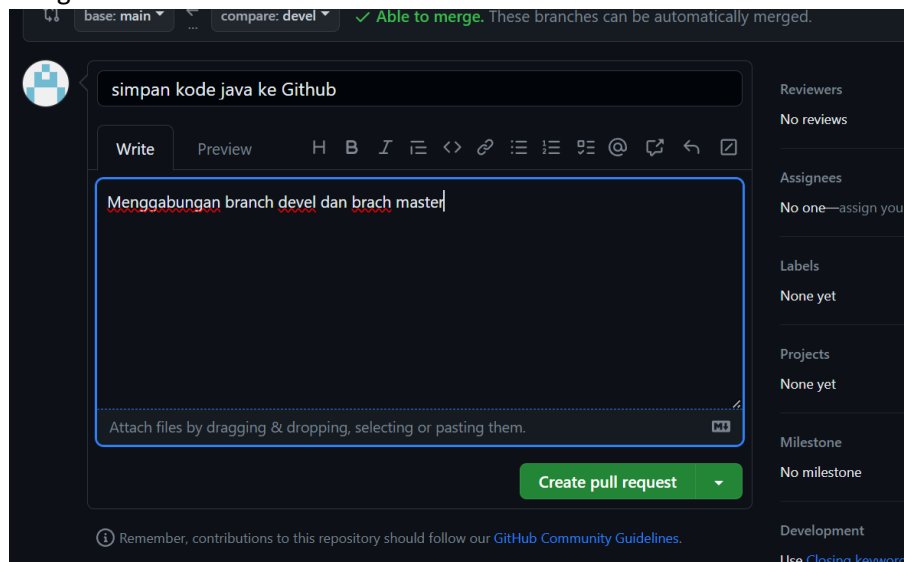
3. Buatlah file MyFirstJavaGithubNoAbsen.java (Gunakan No Absen Masing-masing).
4. Simpan perubahan tersebut di local dengan cara commit kemudian push ke Github menggunakan Visual Studio Code. Jangan lupa memberikan pesan ketika akan melakukan commit. Caranya klik icon ranting – isikan pesan commit – klik tombol Commit – klik tombol Publish Branch



5. Silakan menuju ke halaman Github, seharusnya akan muncul branch devel yang beberapa waktu telah di-push.



6. Klik tombol Compare & pull request, Anda dapat memilih branch mana yang akan digabungkan (devel ke master). Isikan pesan dan klik tombol Create pull request, tunggu beberapa saat kemudian klik tombol Merge pull request. Terakhir, klik tombol Confirm merge.



7. Pindah ke tab Code, kemudian amati hasil antara branch main dan branch devel.

Pertanyaan

- 1) Jelaskan fungsi dari Pull requests!

Jawaban: Pull request adalah fitur yang digunakan dalam sistem control versi seperti Git dan Github untuk memungkinkan kontributor mengusulkan perubahan (biasanya dalam bentuk kode) ke dalam proyek open source atau repositori yang dikelola oleh orang lain

2) Mengapa kita perlu membuat sebuah branch, manfaatnya apa?

Jawaban: untuk memungkinkan kita dapat melacak perubahan, membuat kolaborasi tim menjadi lebih baik, efisiensi dan proses code review yang terstruktur.