LAPORAN PRAKTIKUM ALGORITMA PEMOGRAMAN

COMMIT KE GITHUB DAN PRINTLN PERTAMA



OLEH:

**MUHAMMAD SIDIQ**

(2411533011)

**DOSEN PENGAMPU:**

**DR. WAHYUDI, S.T, M.T**

FAKULTAS TEKNOLOGI INFORMASI

DEPARTEMEN INFORMATIKA

UNIVERSITAS ANDALAS

2024

1. **Pendahuluan**
2. Pengertian GitHub

GitHub adalah [layanan hos web](https://id.wikipedia.org/wiki/Layanan_hos_web" \o "Layanan hos web) bersama untuk membuat suatu proyek pengembangan [perangkat lunak](https://id.wikipedia.org/wiki/Perangkat_lunak" \o "Perangkat lunak) yang menggunakan sistem [kendali versi](https://id.wikipedia.org/wiki/Kendali_versi" \o "Kendali versi) [Git](https://id.wikipedia.org/wiki/Git) dan [layanan hosting internet](https://id.wikipedia.org/wiki/Layanan_hosting_internet). Hal ini banyak digunakan untuk kode komputer. Ini memberikan kontrol akses dan beberapa fitur kolaborasi seperti pelacakan bug, permintaan fitur, manajemen tugas, dan [wiki](https://id.wikipedia.org/wiki/Wiki" \o "Wiki) untuk setiap proyek

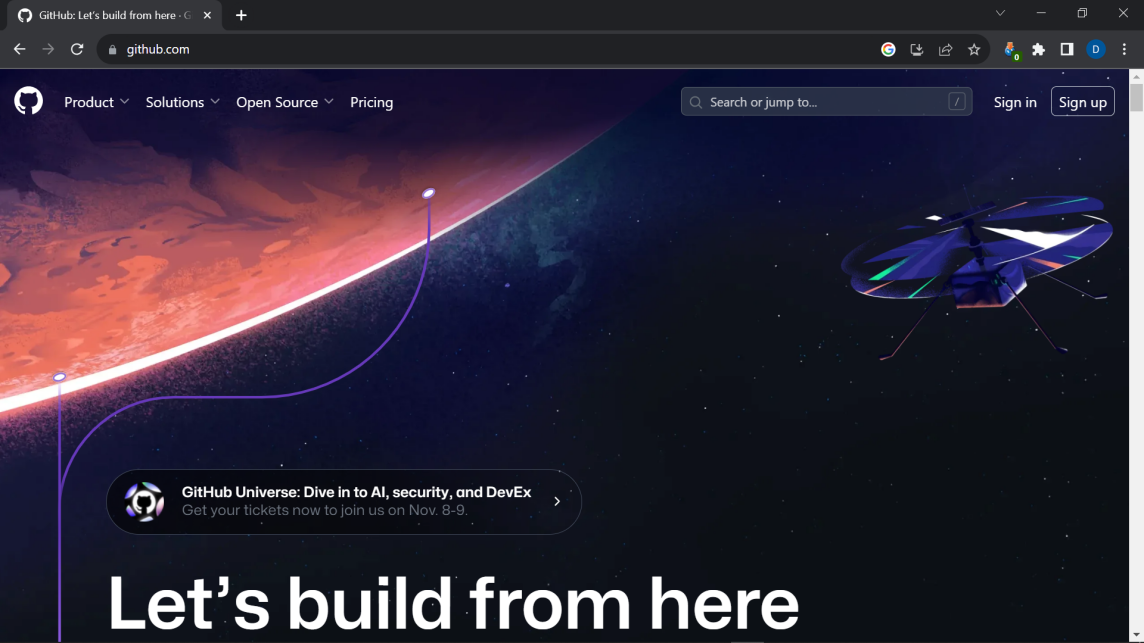
1. Pemograman dalam Java

Program adalah serangkaian instruksi yang harus dilakukan oleh komputer. Dalam program, kita harus mengeksekusi program tersebut. Eksekusi program adalah tindakan melaksanakan instruksi yang terkandung dalam suatu program. Dalam program, diperlukan bahasa pemograman agar kita sebagai manusia dapat membuat instruksi kepada komputer agar program tersebut tereksekusi dengan benar. Bahasa pemograman adalah seperangkat aturan sistematis yang digunakan untuk menggambarkan perhitungan dalam format yand dapat diedit oleh manusia. Pada praktikum kali ini, penulis menggunakan bahasa pemograman Java sebagai bahasa pemograman utama hingga akhir perkuliahan.

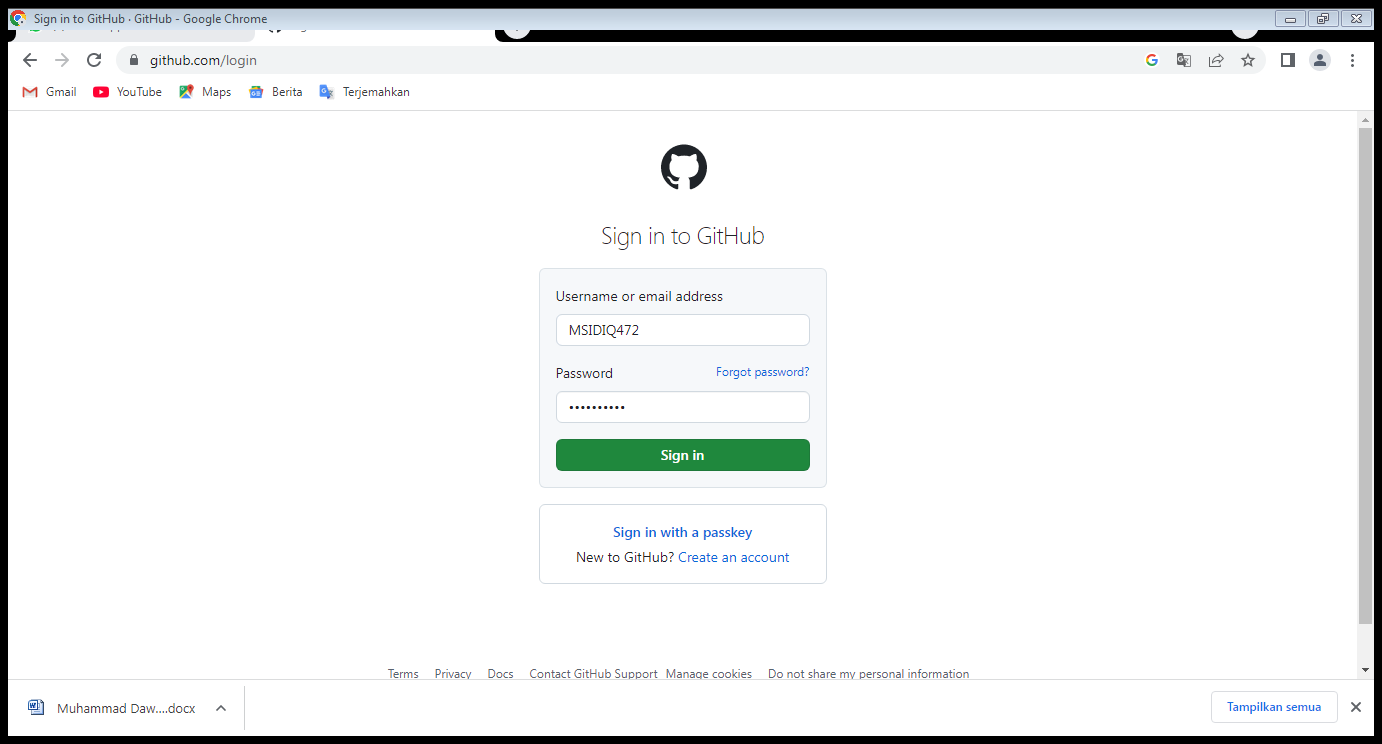
1. **Tujuan**

Tujuan dilakukannya praktikum ini adalah sebagai berikut:

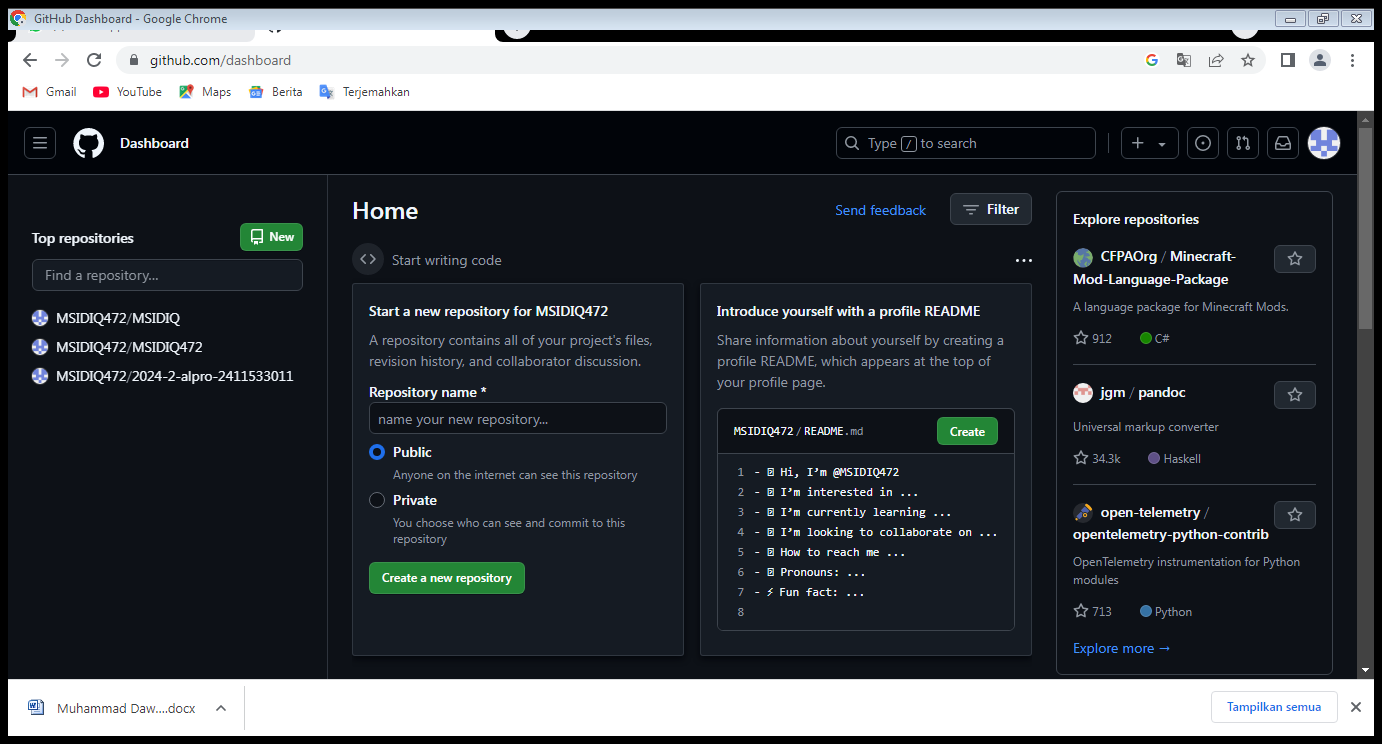
1. Memahami cara membuat akun GitHub dan membuat repository GitHub
2. Memahami cara mengoneksikan Eclipse ke repository GitHub
3. Memahami cara membuat kode di Eclipse dan menjalankannya
4. **Langkah Kerja Praktikum**
5. Membuat akun GitHub dan membuat Repository
6. Buka laman github.com, klik tombol “Sign Up”. Jika sudah memiliki akun, klik tombol “Sign In”.



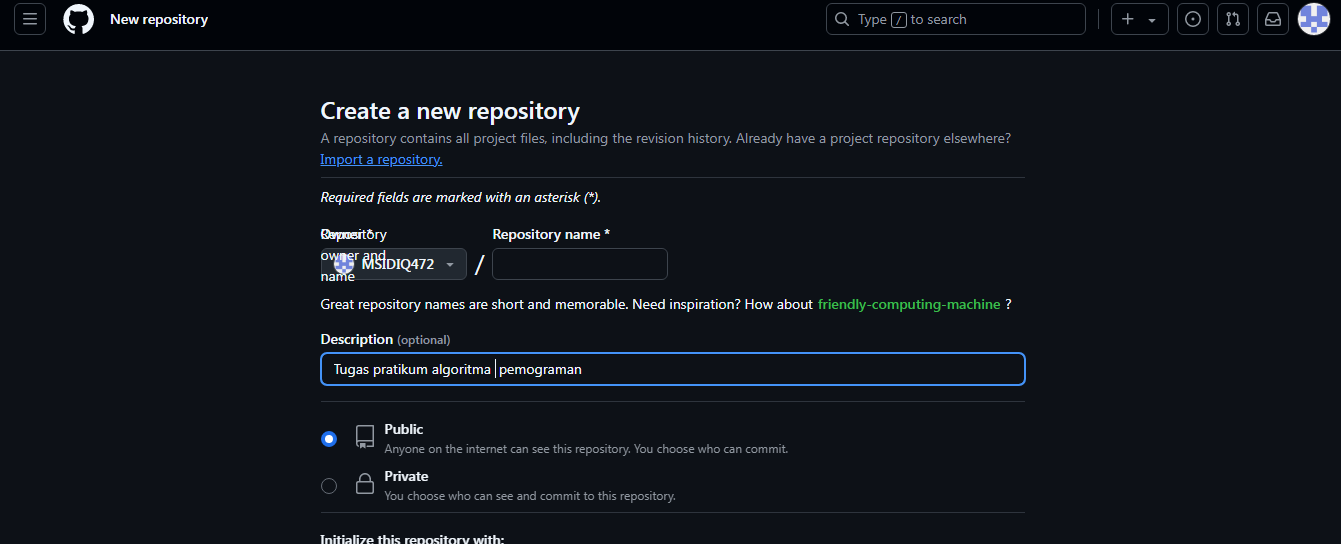
1. Kemudian anda akan diminta memasukkan email, password, dan nama profile.

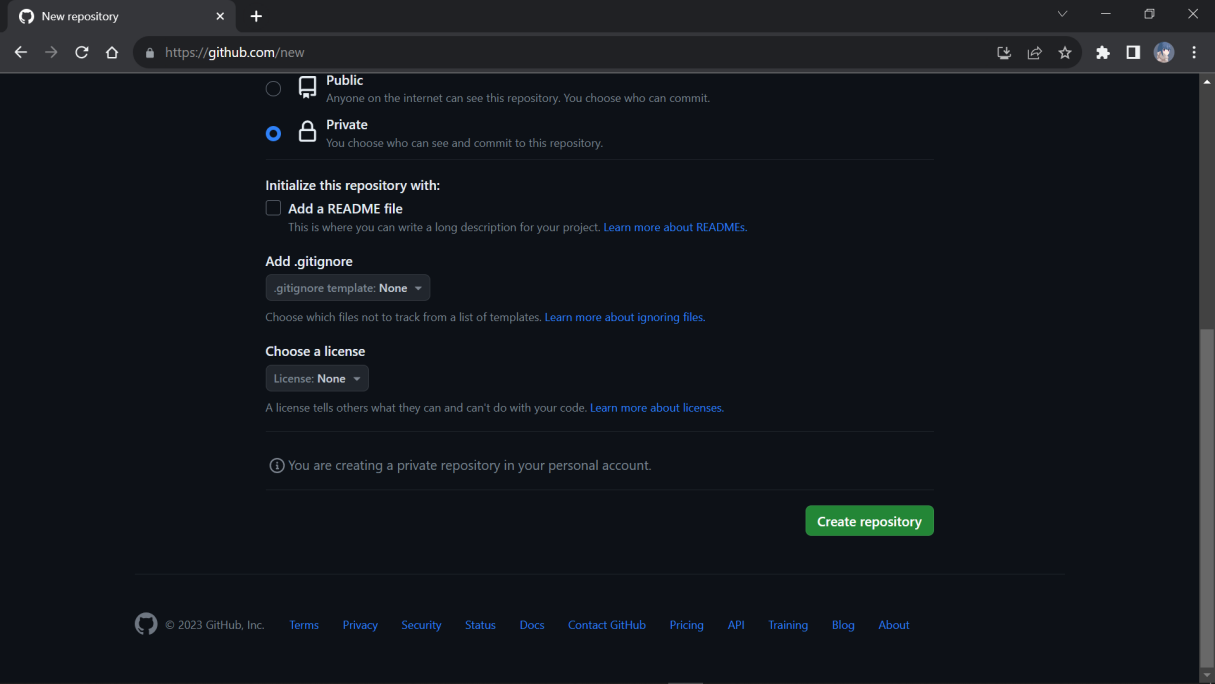


1. Setelah itu, ada beberapa pertanyaan agar akun dipersonalisasikan. Setelah itu, tampilan akun akan menjadi seperti berikut. Lanjutkan dengan membuat repository dengan menekan tombol “Create Repository” atau “new”

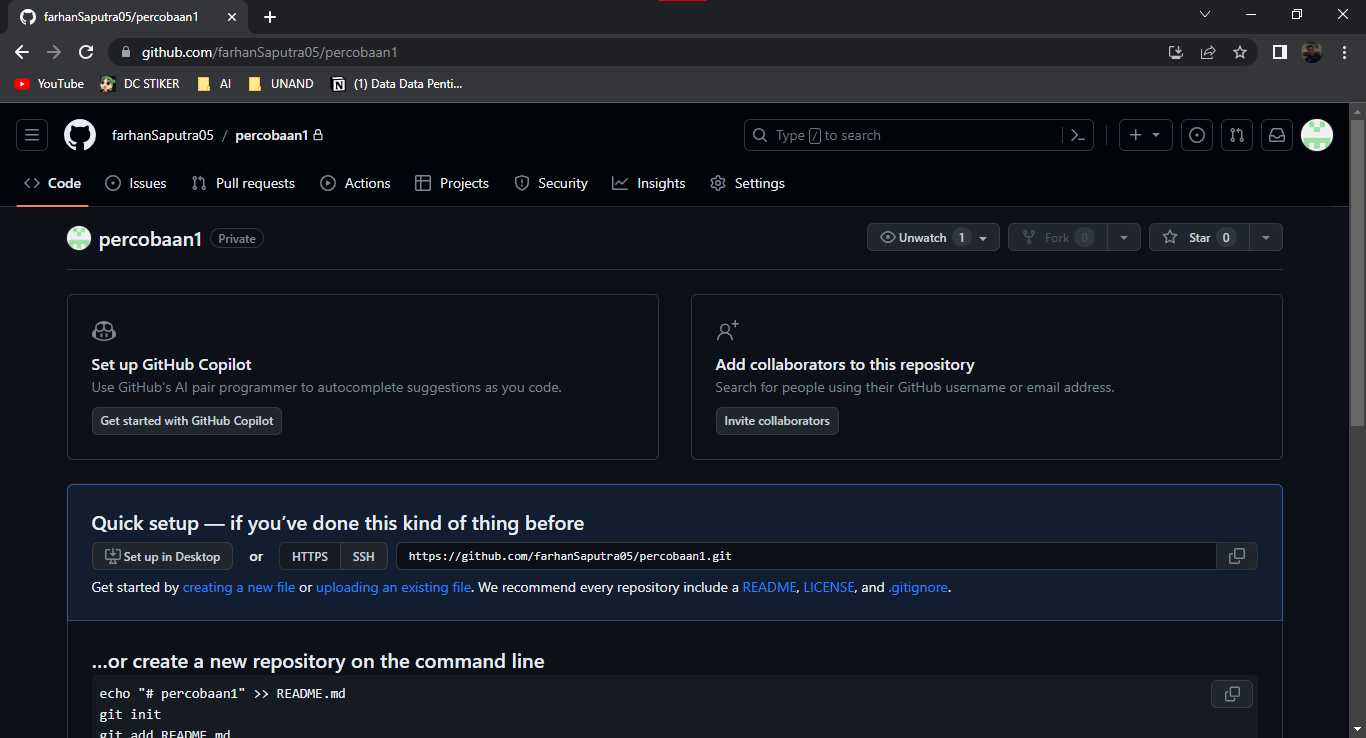


1. Setelah itu pengisian nama repository, deskripsi repository, dan pilih apakah projekmu dapat dilihat secara publik atau hanya undangan. Kemudian klik “Create Repository” di bawah

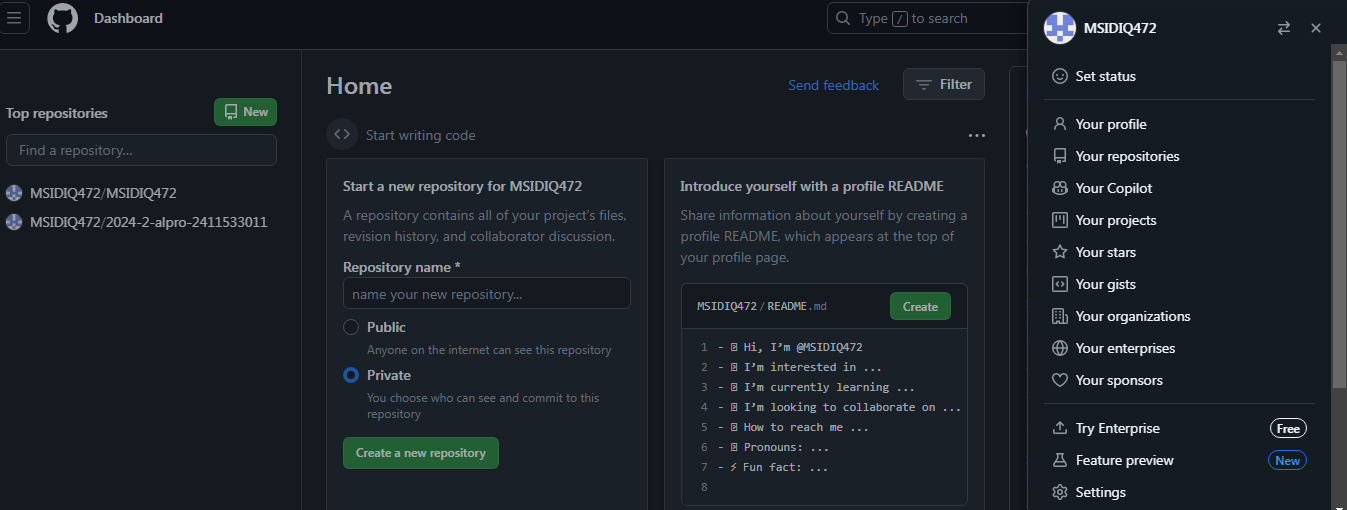




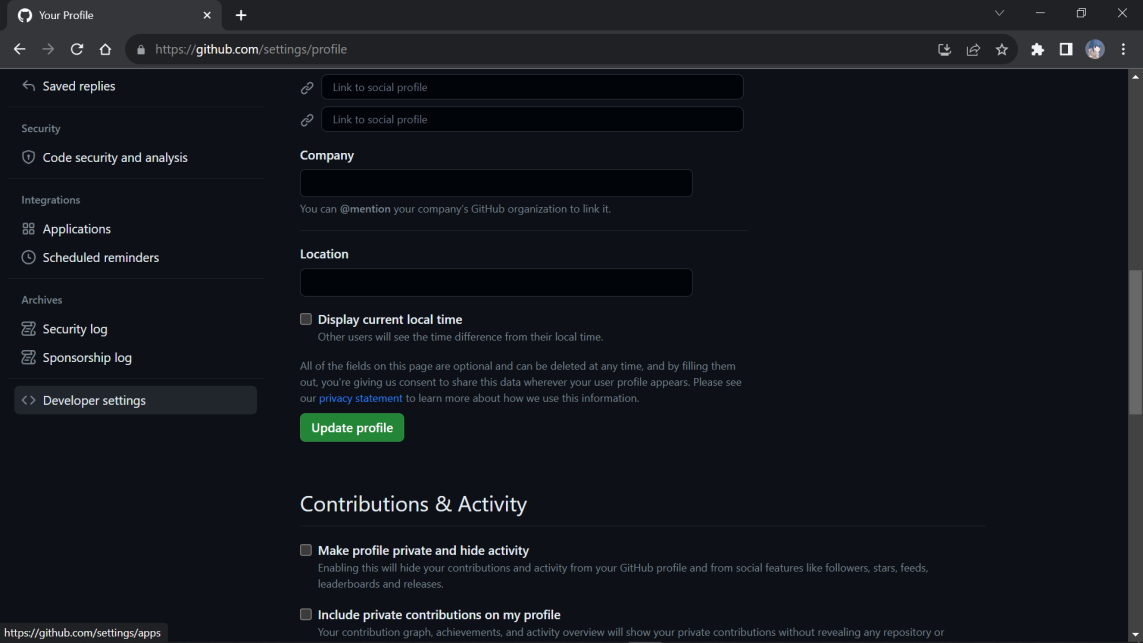
1. Setelah dibuat, tampilan akan seperti berikut. Simpan link pada “Quick setup” karna akan digunakan nanti.



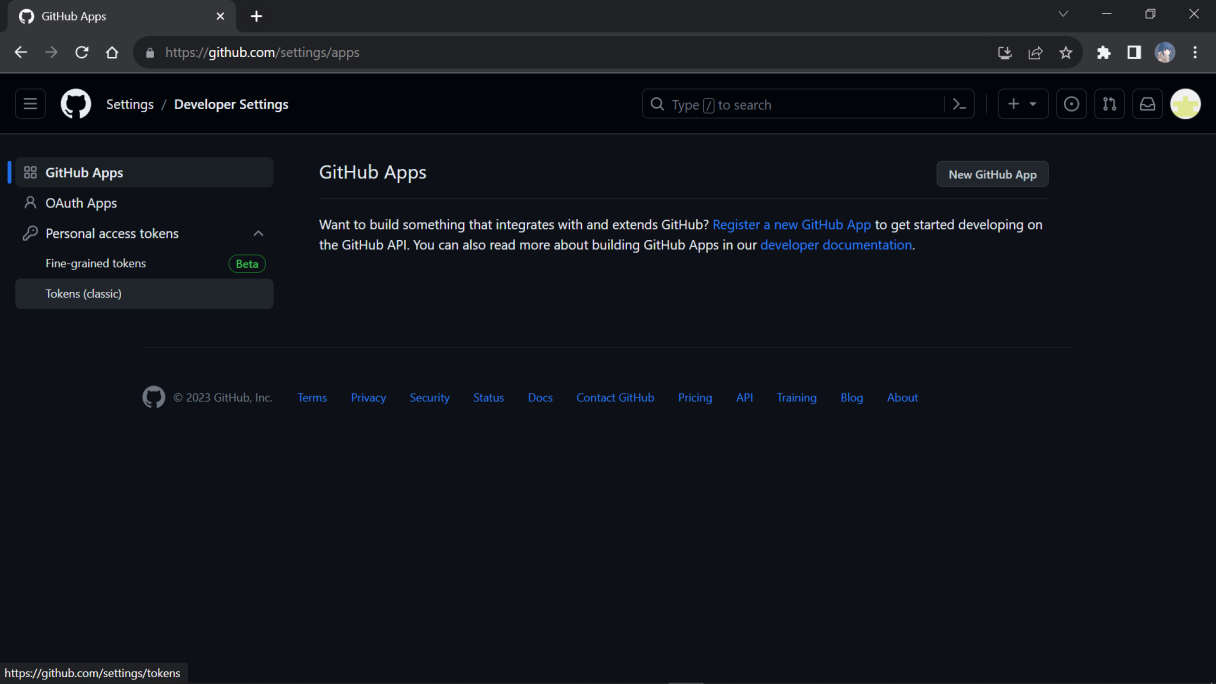
1. Langkah selanjutnya yaitu membuat token untuk menghubungkan Eclipse ke GitHub. klik logo profil, lalu pilih “Setting”



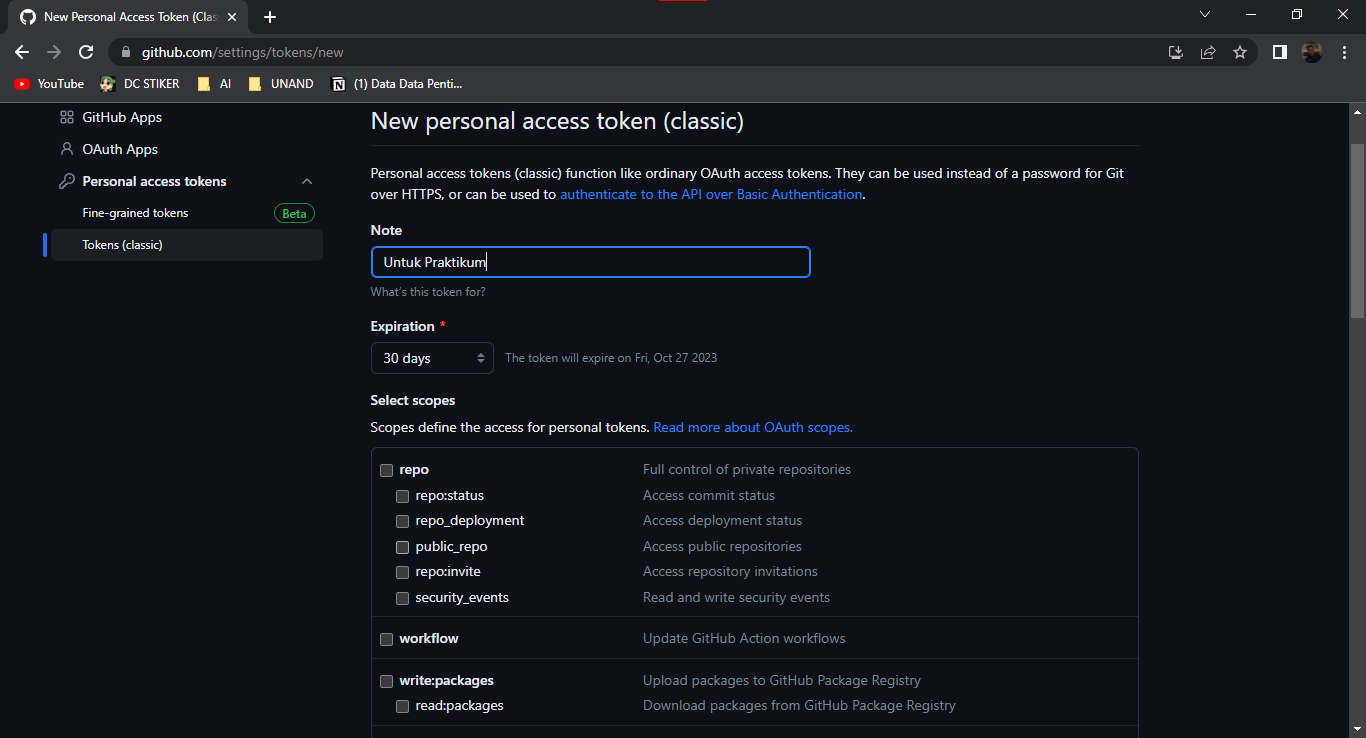
1. Setelah itu, scroll hingga paling bawah sampai menemukan menu “Developer option”, Klik pada menu tersebut.



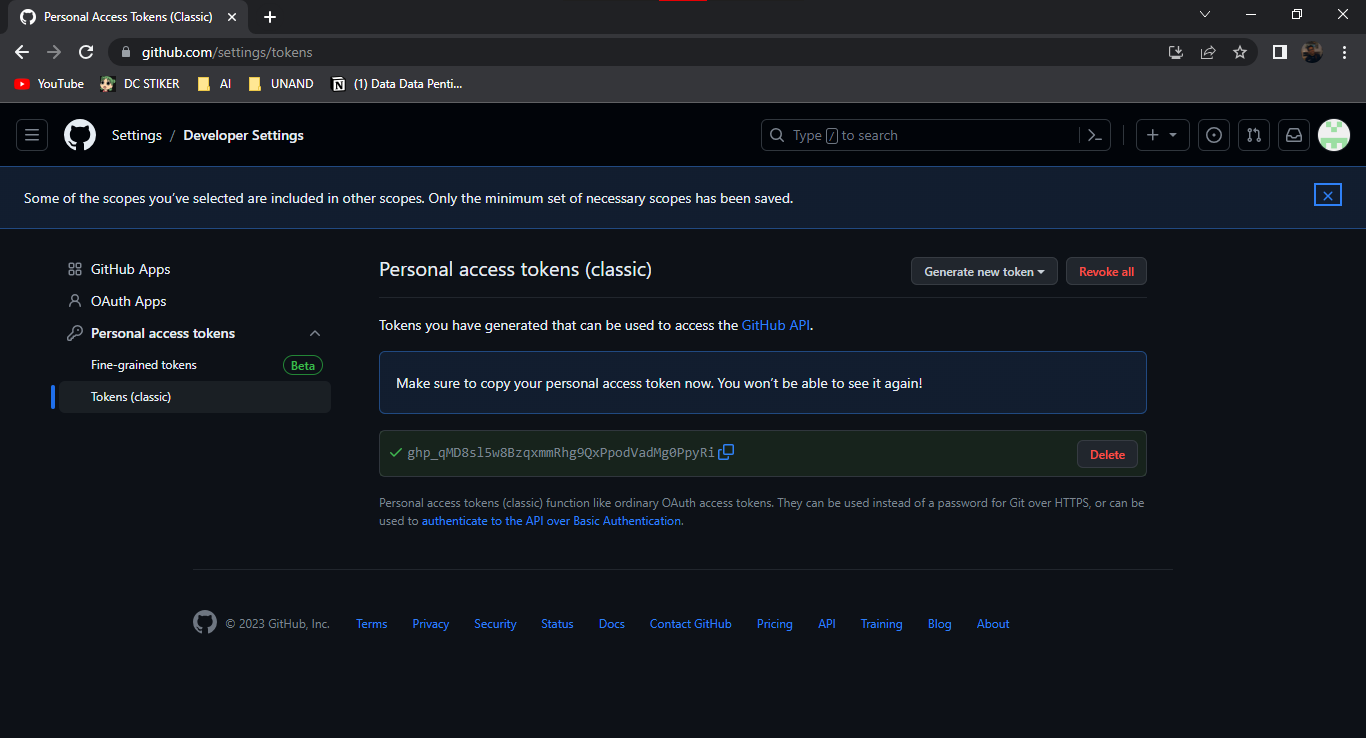
1. Setelah itu, pilih menu “Personal Access Token”, lalu ke “Token (Classic)”, lalu “Generate new token”



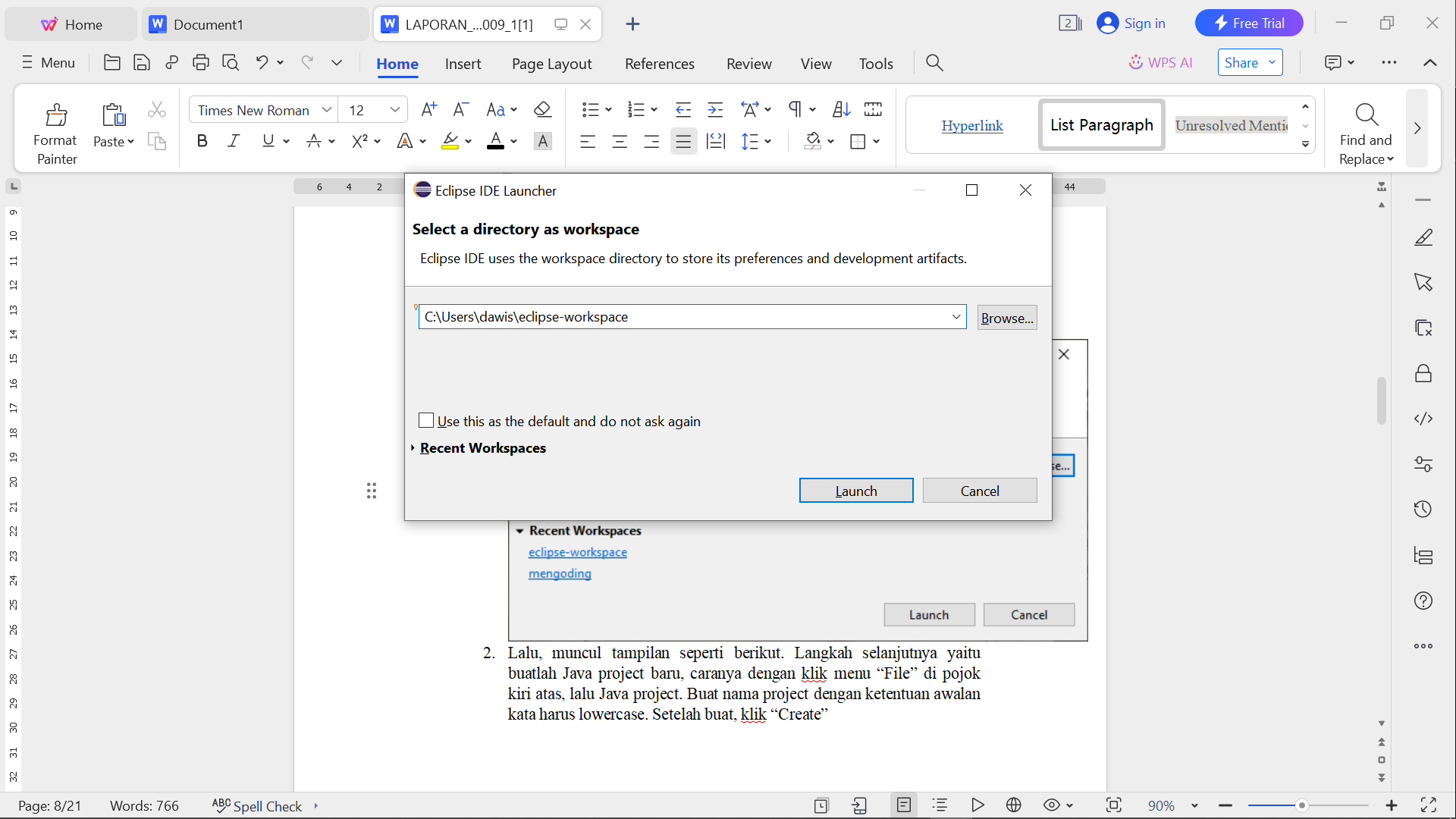
1. Setelah itu, isi note lalu tulis fungsi tokennya dibuat. Expiration date menyesuaikan. Lalu untuk pilihan yang di checklist, pilihan wajib yaitu bagian “Repo”, selebihnya opsional



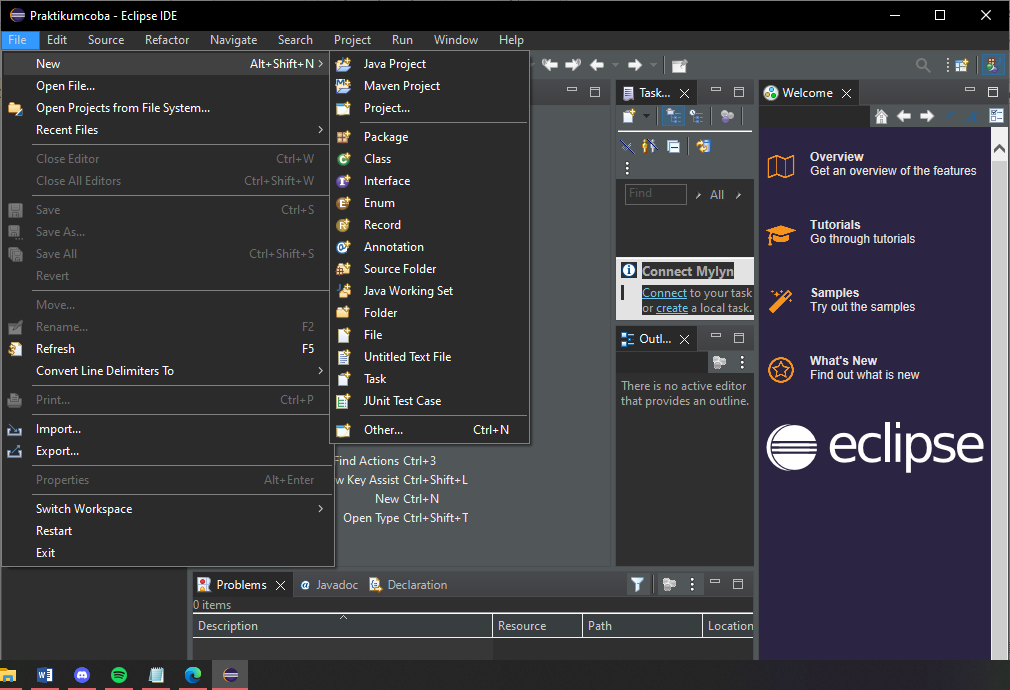
1. Kemudian tampilan akan seperti berikut. Simpan tokennya karena akan digunakan.

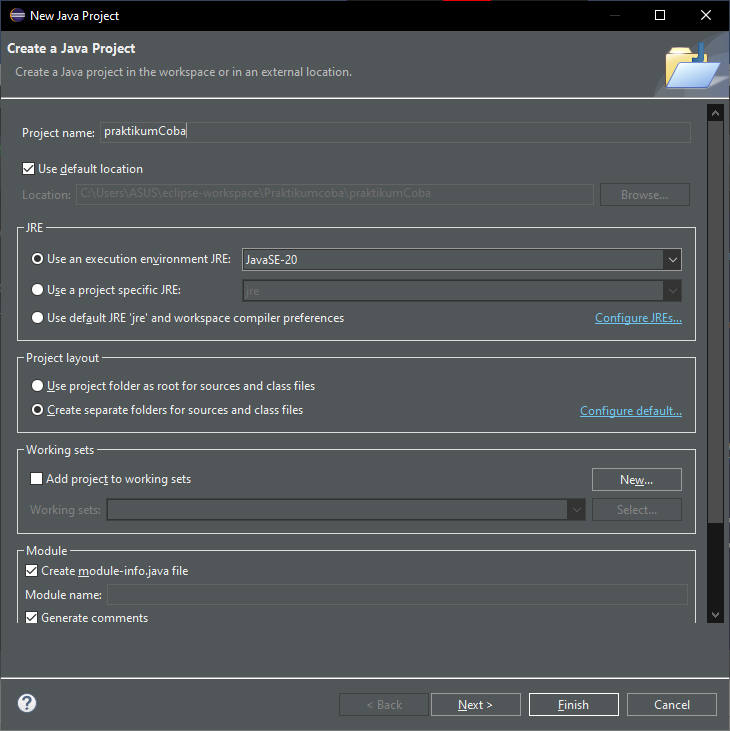


1. Membuat Repository Lokal di Eclipse
2. Langkah pertama, buatlah workspace terlebih dahulu, lalu Launch.

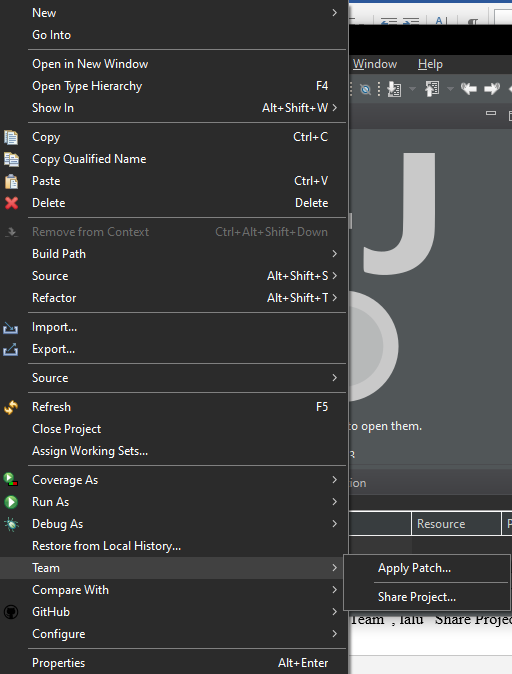


1. Langkah selanjutnya yaitu buatlah Java project baru, caranya dengan klik menu “File” di pojok kiri atas, lalu Java project. Buat nama project dengan ketentuan awalan kata harus lowercase. Setelah buat, klik “Create”

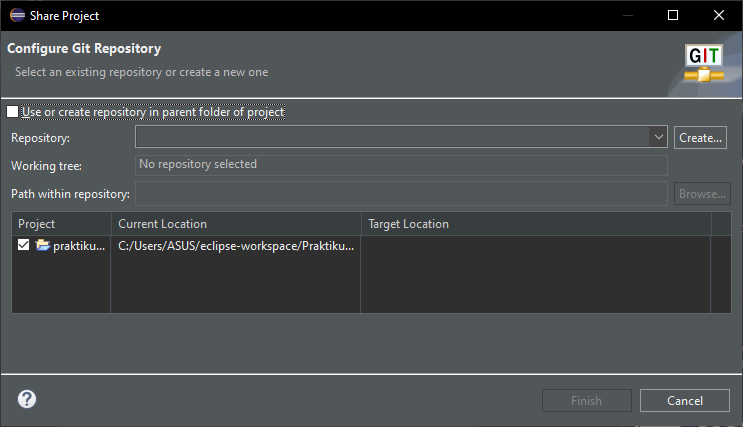




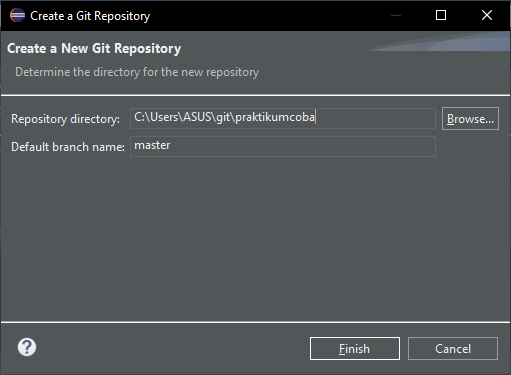
1. Lalu, klik kanan pada nama project yang telah dibuat, lalu ke bagian “Team”, lalu klik “Share Project”



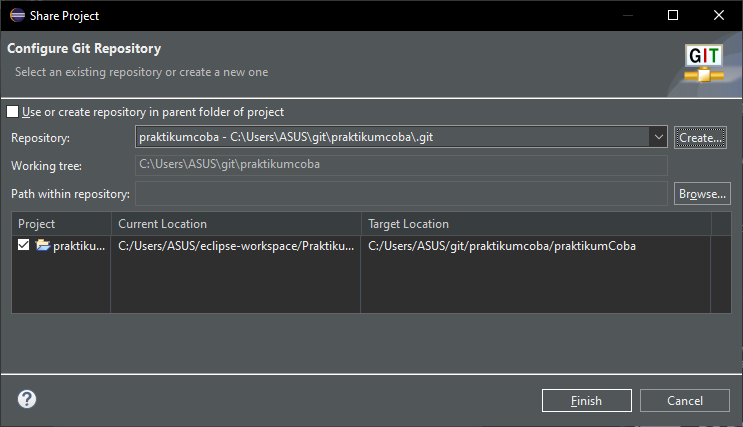
1. Kemudian klik pada menu “Create”



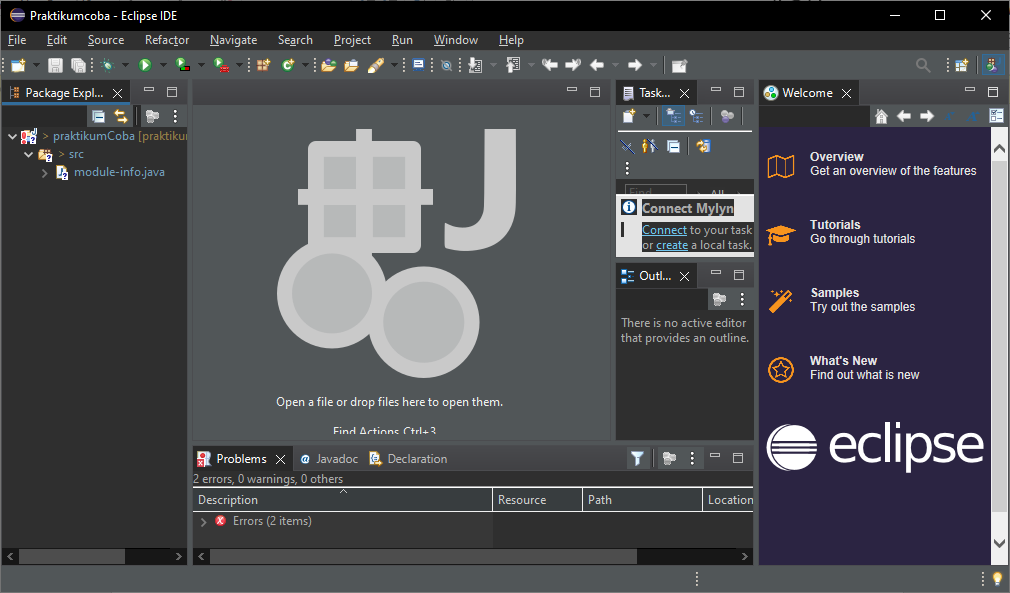
1. Buat nama repository lokal. Untuk default branch name nya dibiarkan. Lalu, klik “Finish”



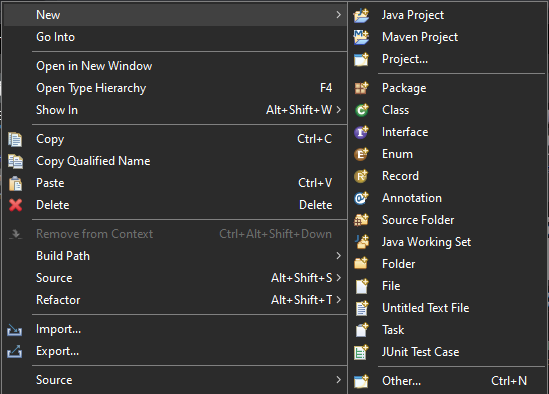
1. Setelah itu, klik “Finish” Sekali lagi

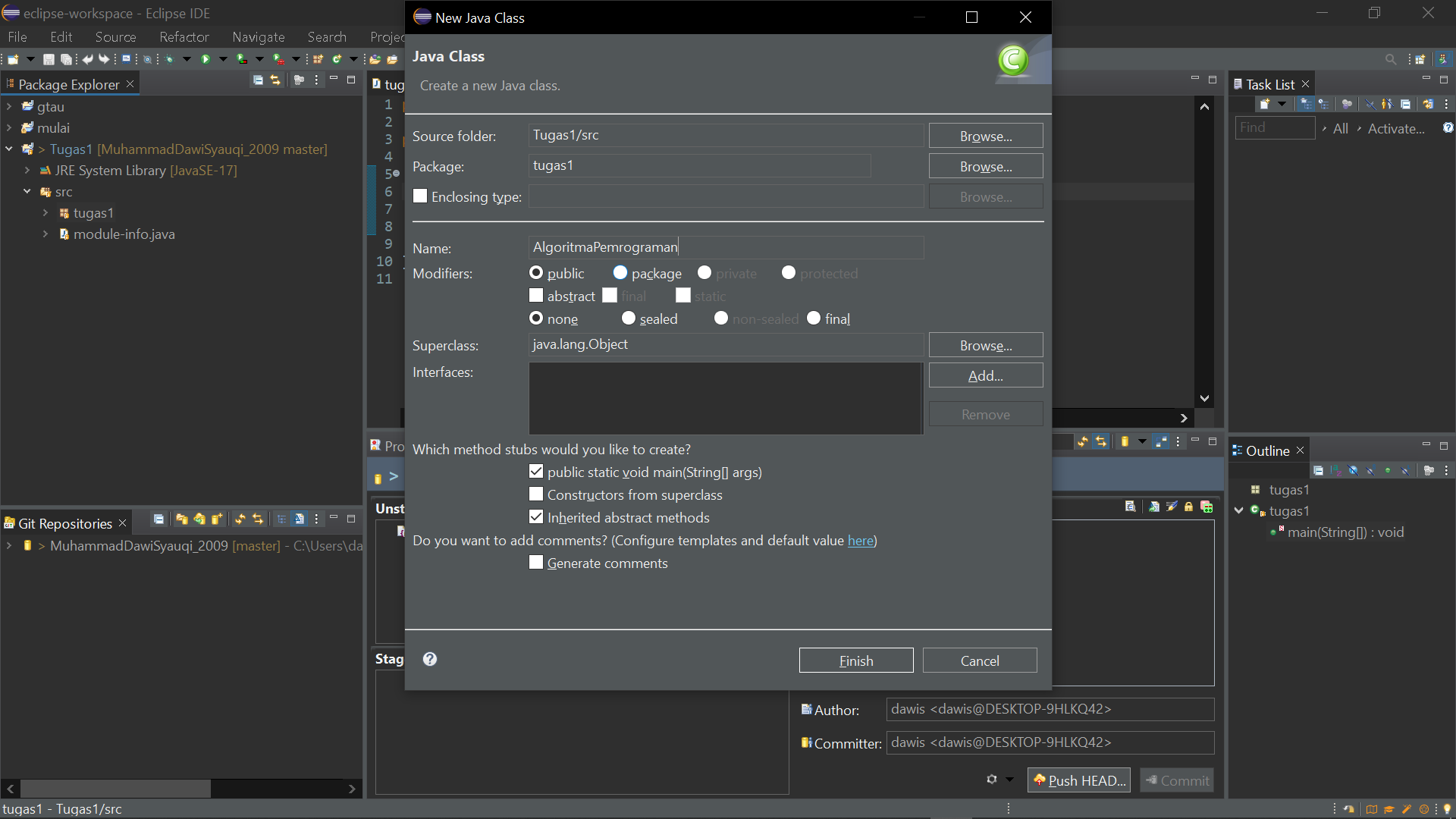


1. Maka muncul repositary yang telah dibuat pada samping kanan project

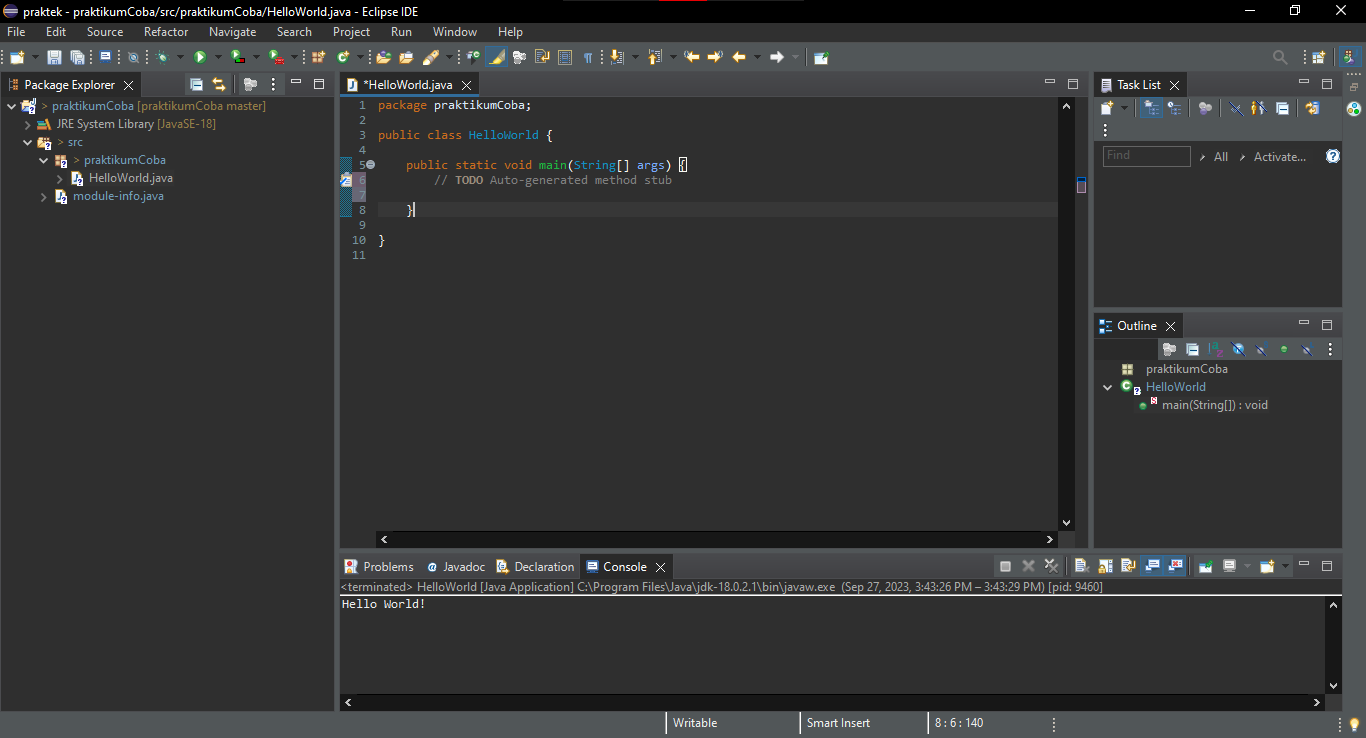


1. Program pertama
2. Buatlah package terlebih dahulu dengan mengklik tengah folder src, lalu pilih “New”, lalu pilih class. Buatlah nama dengan ketentuan nama harus Uppercase pada awal kalimat dan tidak boleh “space”, lalu centang tanda “public static void main (string[] args)”



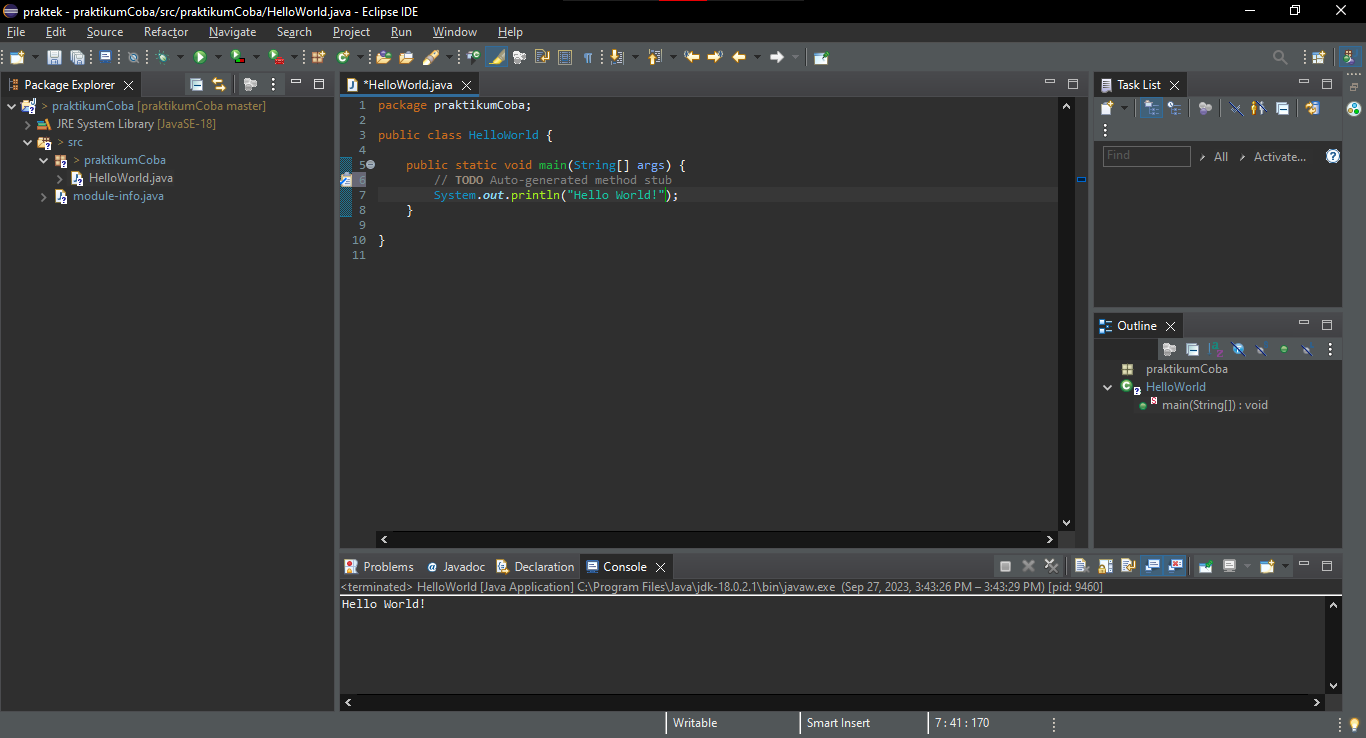


1. Maka Tampilan akan seperti berikut

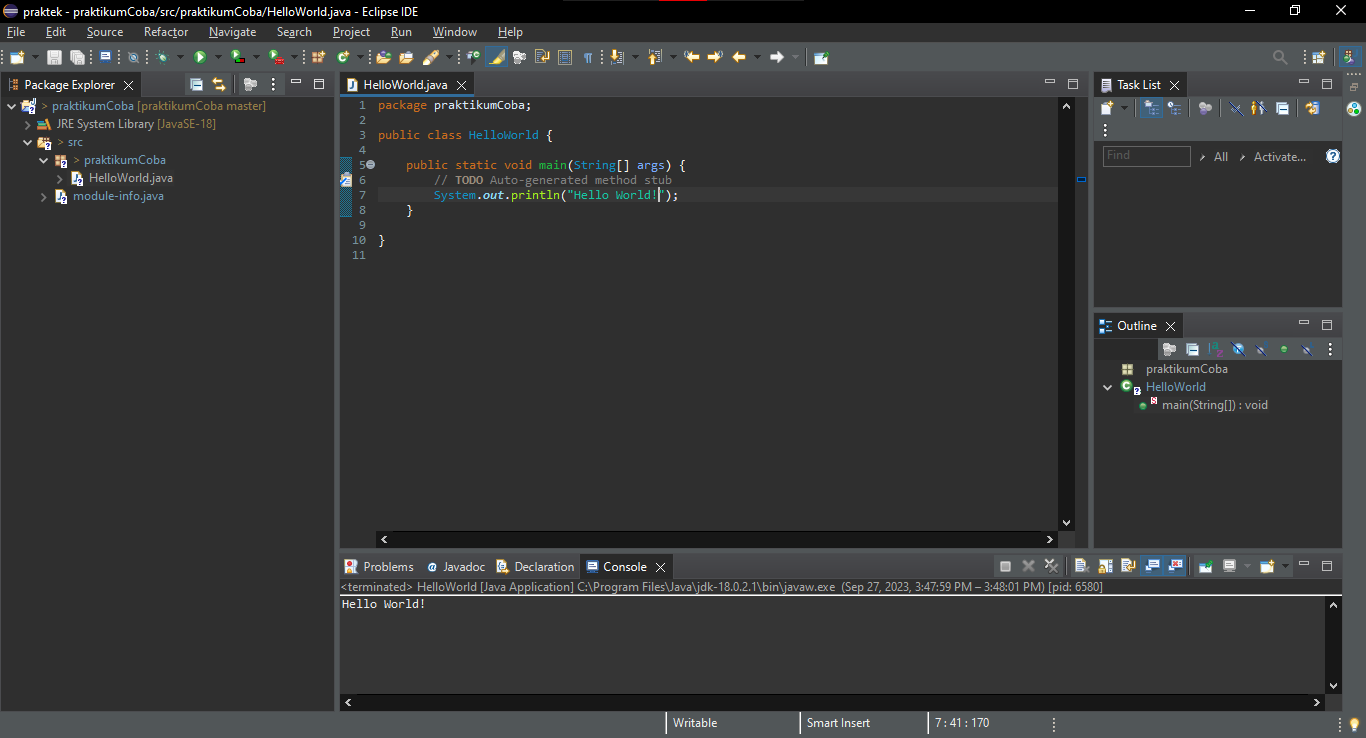


1. Lalu, masukkan syntax sebagai berikut:

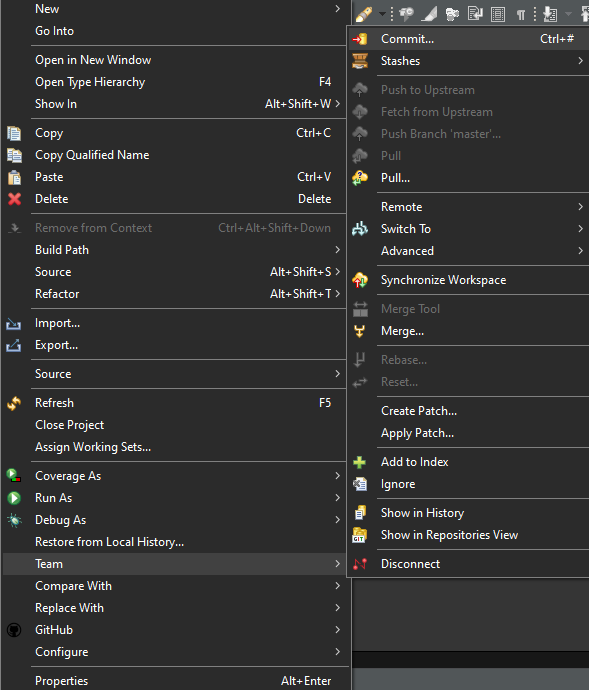
System.out.println(“Hello, World!”);



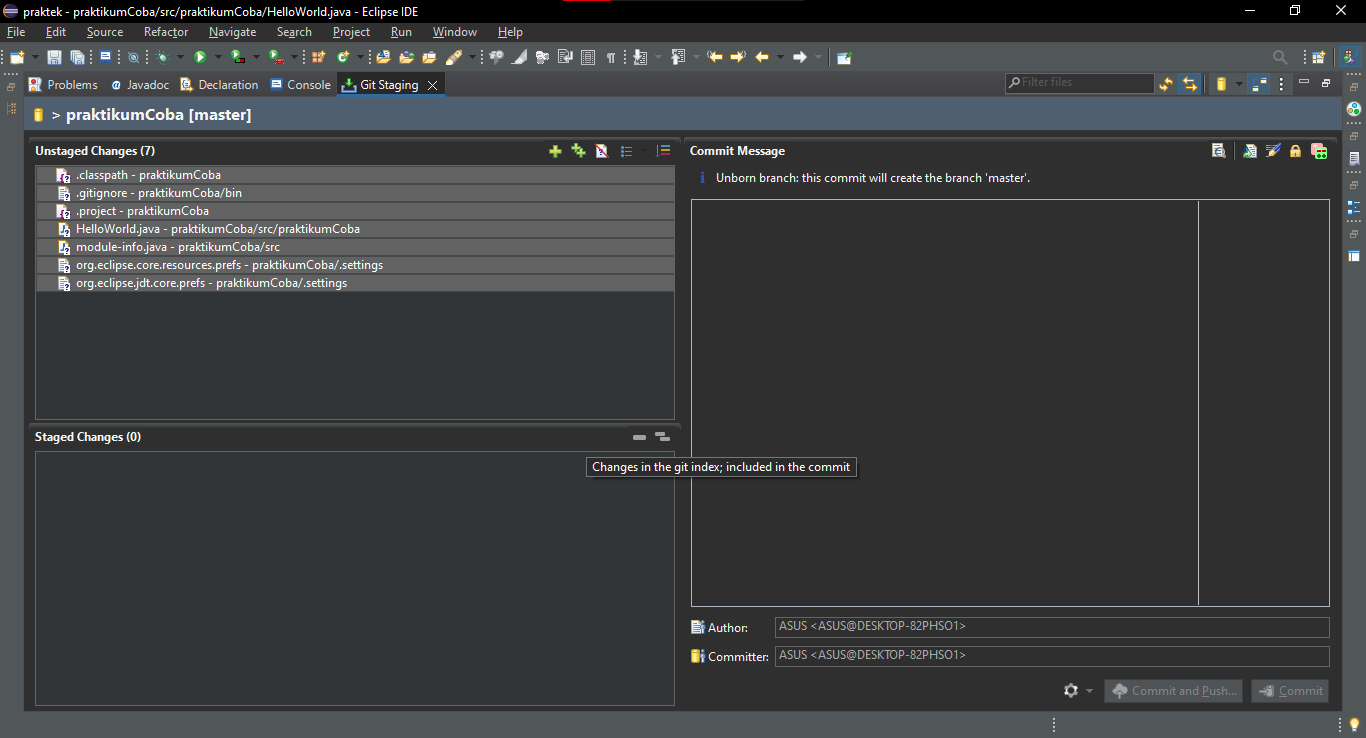
1. Jalankan dengan mengklik bulatan hijau di bar menu (Run)



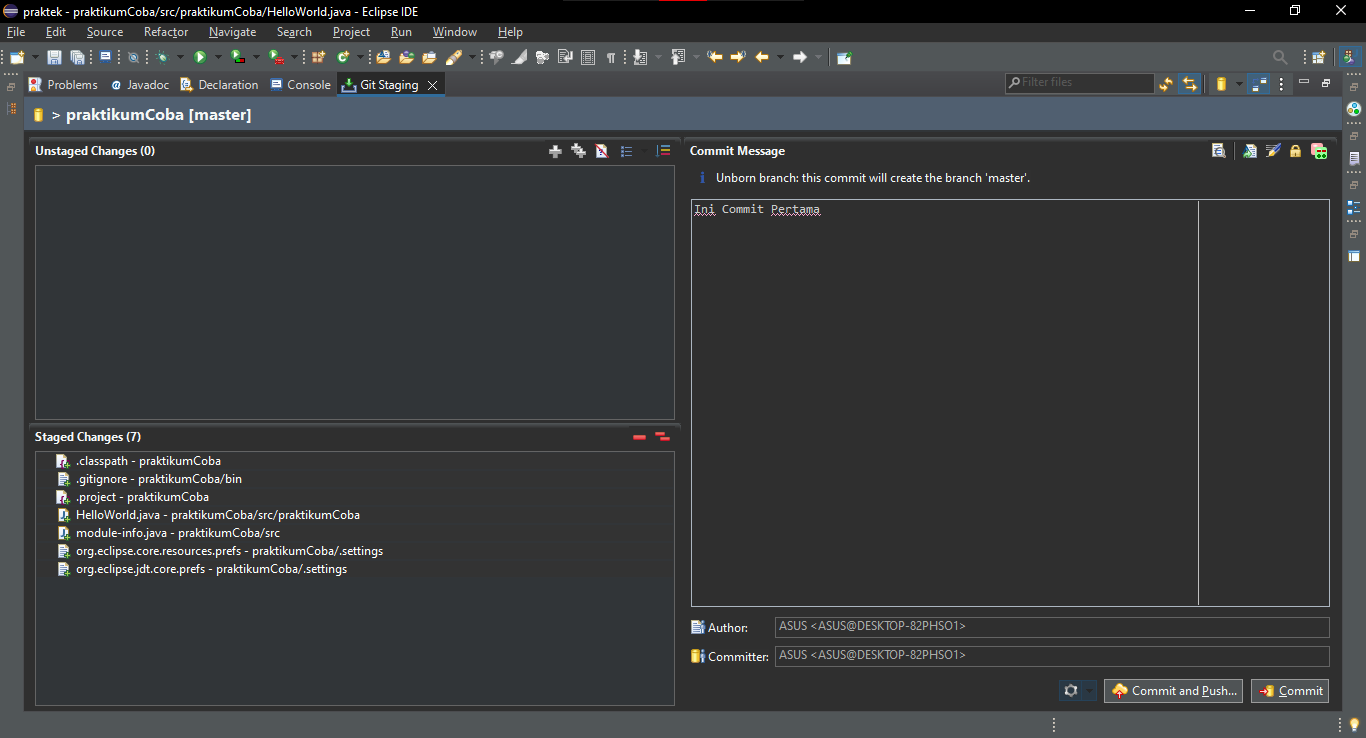
1. Menghubungkan Eclipse ke Github
2. Klik kanan project yang telah dibuat, lalu pilih “Team”, lalu pilih “Commit”



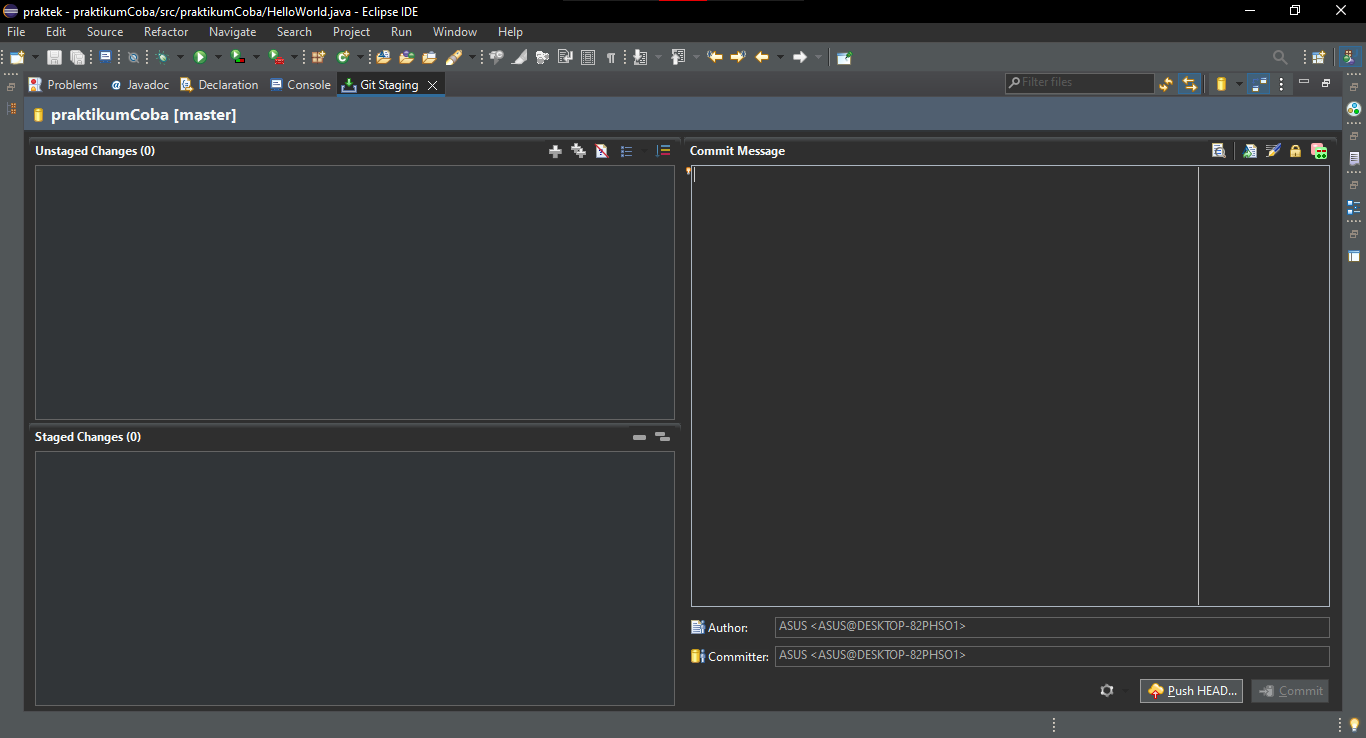
1. Block semua yang berada di atas dengan , lalu tekan logo tambah



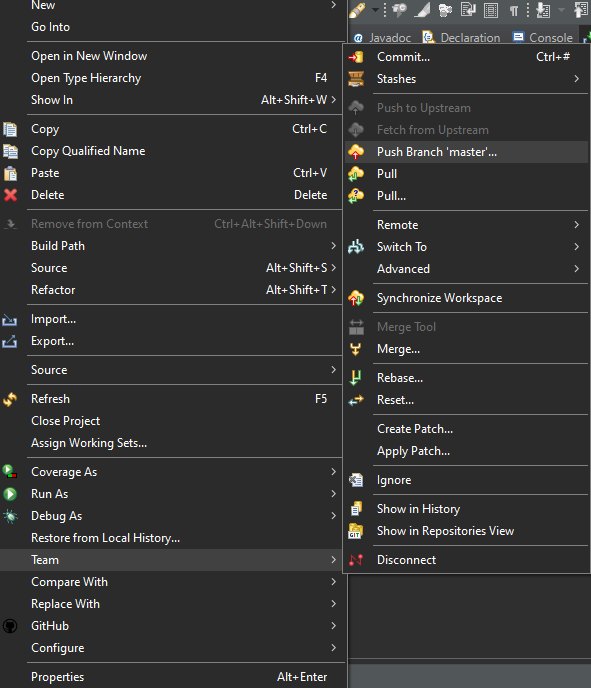
1. Setelah ditambah, buat pesan commitnya, lalu klik “Commit”



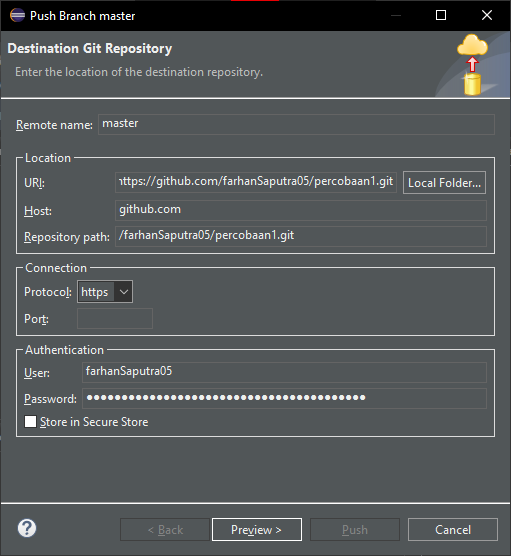
1. Jika tampilan sudah hilang, maka item berhasil di commit.



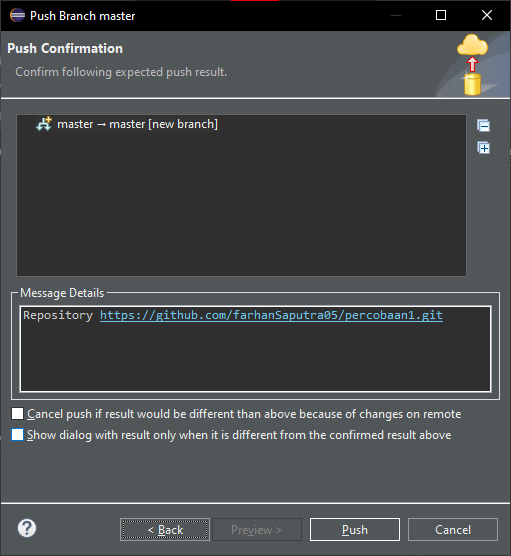
1. Kembali ke menu awal dengan cara mengklik pojok kiri atas, logo kotak dua, lalu klik kanan kembali project, lalu pilih team, lalu pilih push branch master.



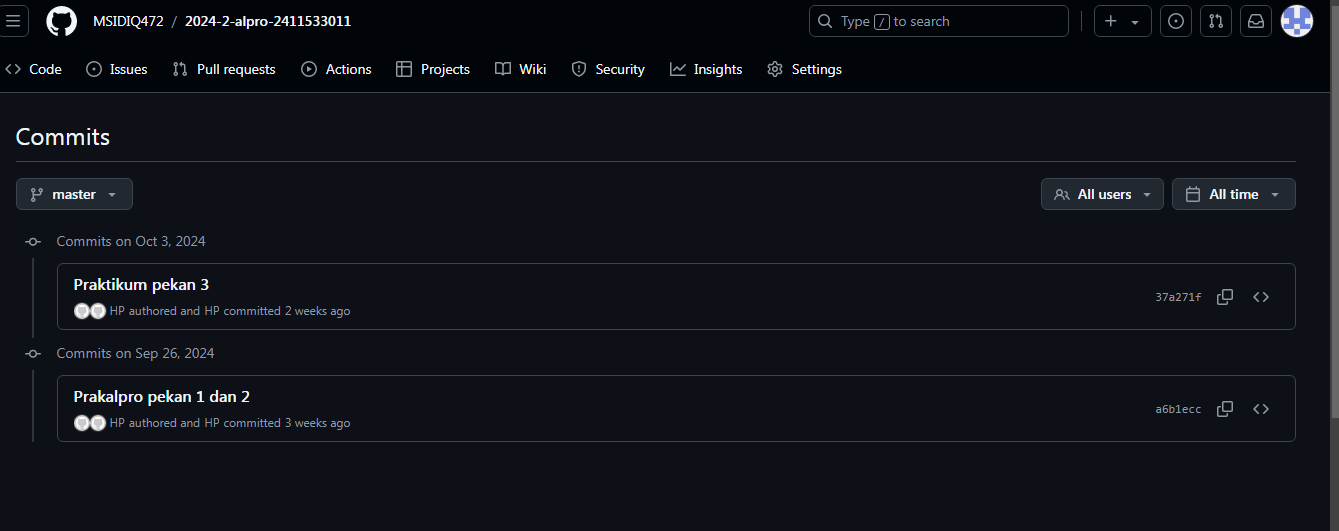
1. Masukkan link URL yang anda simpan tadi berdasarkan GitHub Repository yang telah dibuat di awal, untuk bagian user diisikan dengan nama usernam profile, dan untuk password diisikan dengan token yang telah dibuat di awal, lalu klik “Preview”



1. Setelah itu, tekan “Push”, maka akan tampil seperti berikut.



1. Maka file yang berada di Eclipse berhasil disalin di Repository GitHub. Anda bisa mengecek apakah sudah masuk di repository GitHub di akun anda



1. Kesimpulan

Berdasarkan praktikum, kita dapat menyimpulkan bahwa dalam bahasa pemograman Java, kita memerlukan IDE untuk membuat dan mengedit kode yang akan kita buat. IDE juga perlu untuk tempat mengcompile hasil codingan kita dan memudahkan kita untuk mengeksekusi code kita tanpa harus membuka lewat command prompt. Repository juga sangat penting untuk dokumentasi code agar saat berkerja dengan tim kita tidak menjadi kebingungan. GitHub menyediakan repository berbasis online agar memudahkan para programmer untuk berkerja dari jarak jauh dan berkerjasama dengan tim.