LAPORAN PRAKTIKUM ALGORITMA PEMROGRAMAN

“PENGENALAN JAVA DAN GITHUB”



OLEH

MUHAMMAD HAFIZ

NIM 2311532007

MATA KULIAH : ALGORITMA PEMROGRAMAN

DOSEN PENGAMPU : Dr.WAHYUDI,S.T.M.T

FAKULTAS TEKNOLOGI INFORMASI

DEPARTEMEN INFORMATIKA

UNIVERSITAS ANDALAS

PADANG, 27 SEPTEMBER 2023

1. TUJUAN
2. Mengenal eclipse dan github
3. Memahami langkah-langkah untuk menjalankan program pada eclipse
4. Memahami langkah-langkah untuk membuat repository pada github
5. Memahami cara untuk menghubungkan eclipse dengan github
6. KAJIAN TEORI

**Eclips**

Eclipse adalah platform yang dikembangkan untuk membuat berbagai aplikasi yang dapat diintegrasikan. Situs web, perangkat seluler, dll. Eclipse adalah open source dan biasanya digunakan untuk mengembangkan aplikasi berbasis Java yang memungkinkan pengembang perangkat lunak untuk membuat lingkungan pengembangan yang disesuaikan (IDE). Ikuti kursus Android di mana pun Anda berada untuk mengembangkan dan mempelajari platform ini.  
  
Eclipse dimulai oleh IBM pada tahun 2001 ketika IBM menyumbangkan 3 juta baris kode dari alat Java mereka. Tujuan Eclipse adalah untuk membuat dan mengembangkan komunitas IDE open source yang melengkapi komunitas Apache. Eclipse menyediakan model antarmuka pengguna (UI) umum untuk bekerja dengan alat lain. Ini dirancang untuk berjalan di beberapa sistem operasi sambil memberikan integrasi yang kuat dengan sistem operasi yang mendasarinya. Plugin digunakan untuk memprogram Eclipse API portabel dan memodifikasinya di salah satu sistem operasi yang didukung.  
  
Inti dari Eclipse adalah arsitektur untuk menemukan, memuat, dan menjalankan plugin secara dinamis. Platform menangani logistik untuk menemukan dan mengeksekusi kode yang tepat. Platform UI menyediakan model navigasi pengguna standar. Setiap plugin dapat fokus melakukan serangkaian tugas kecil dengan baik. pekerjaan apa? Tentukan, uji, hidupkan, publikasikan, kompilasi, debug, diagram satu-satunya batasan adalah imajinasi Anda.

### ****Fitur eclipse****

### Fitur Eclipse meliputi:

### ****1. Multi-platform**** Target sistem operasi Eclipse adalah Microsoft Windows, Linux, Solaris, AIX, HP-UX, dan Mac OS X. ****2. Multibahasa**** Eclipse dikembangkan menggunakan bahasa pemrograman Java, namun Eclipse mendukung pengembangan aplikasi berbasis bahasa pemrograman lain seperti C/C++, Cobol, Python, Perl, dan PHP. ****3. Serba guna**** Eclipse juga dapat digunakan untuk kegiatan siklus pengembangan perangkat lunak seperti dokumentasi, pengujian perangkat lunak, dan pengembangan web.

### Eclipse awalnya dikembangkan oleh IBM untuk menggantikan perangkat lunak pengembangan IBM Visual Age untuk Java 4.0. Produk Eclipse ini dirilis oleh IBM pada tanggal 5 November 2001. IBM menginvestasikan $40 juta dalam pengembangannya. Sejak 5 November 2001, Eclipse Foundation Consortium telah mengambil alih pengembangan Eclipse lebih lanjut. Eclipse adalah salah satu IDE terpopuler saat ini karena gratis dan open source. Ini berarti siapa pun dapat melihat kode pemrograman untuk perangkat lunak ini. Keuntungan lebih lanjut dari popularitas Eclipse adalah kemampuannya untuk dikembangkan oleh pengguna menggunakan komponen yang disebut plug-in.

### Github

### Melalui homepage pada [GitHub](https://github.com/" \t "_blank), disebutkan bahwa GitHub adalah platform khusus developer yang dibuat karena terinspirasi dari cara bekerja para programmer. Dari open source ke bisnis, kita bisa meng-host dan meninjau sebuah kode, mengelola project, dan membangun perangkat lunak bersama 56 juta lebih pengguna di seluruh dunia.

### Gampangnya, GitHub merupakan manajemen project, sistem versioning code, sekaligus platform jaringan sosial bagi para developer seluruh dunia. Banyak sekali fungsi yang bisa digunakan oleh para developer, sehingga melalui platform ini memudahkan developer dalam mengembangkan sebuah karya.

### Sebelum banyak membahas mengenai GitHub, perlu kamu ketahui bahwa GitHub sendiri memiliki versi induknya, yaitu “git”. Git sendiri adalah sebuah software VCS (Version Controlled System) yang dilakukan secara offline.

### GitHub juga memberikan layanan cloud untuk menyimpan dan mengelola project/repository git. Karena bersifat online, kita meng-edit sebuah repository/project secara bersamaan dengan orang lain di tempat yang berbeda. Oleh karena itu, platform ini sangat membantu tim project dalam menyusun suatu folder yang berisikan files terkait pemrograman.

## Repositori

Repositori atau repo adalah direktori penyimpanan file proyek. Di sini, Anda bisa menyimpan apa pun yang berkaitan dengan proyek yang sedang Anda buat, misalnya file kode, gambar, atau audio. Repo sendiri bertempat di penyimpanan atau *storage*GitHub atau repositori lokal di komputer Anda.

## Branch

Branch merupakan salinan dari repositori. Anda bisa menggunakan branch ketika akan melakukan suatu pengembangan atau *development* secara terpisah.

Pekerjaan atau *task*yang Anda kerjakan di branch tidak akan memengaruhi repositori pusat atau branch lainnya. Jika pengembangannya sudah selesai, Anda bisa menggabungkan branch saat ini ke branch lainnya dah juga repositori pusat dengan menggunakan *pull request*.

## Pull Request

Pull request adalah ketika Anda menginformasikan user bahwa Anda sudah memindahkan perubahan yang Anda lakukan di *branch* ke *master repositori*. *Collaborator*repositori akan menerima atau menolak *pull request.* Segera setelah *pull request* diterima, Anda bisa mendiskusikan dan mengulas proyek bersama dengan *collaborator.*

Berikut beberapa langkah untuk membuat *pull request* di GitHub:

1. Masuk ke repositori dan cari menu *branch*.
2. Di menu *branch*, pilih *branch* yang menyimpan *commit* Anda.
3. Klik opsi **New pull request**yang ada di samping menu *branch*.
4. Masukkan judul dan deskripsi *pull request.*
5. Klik opsi **Create pull request**.

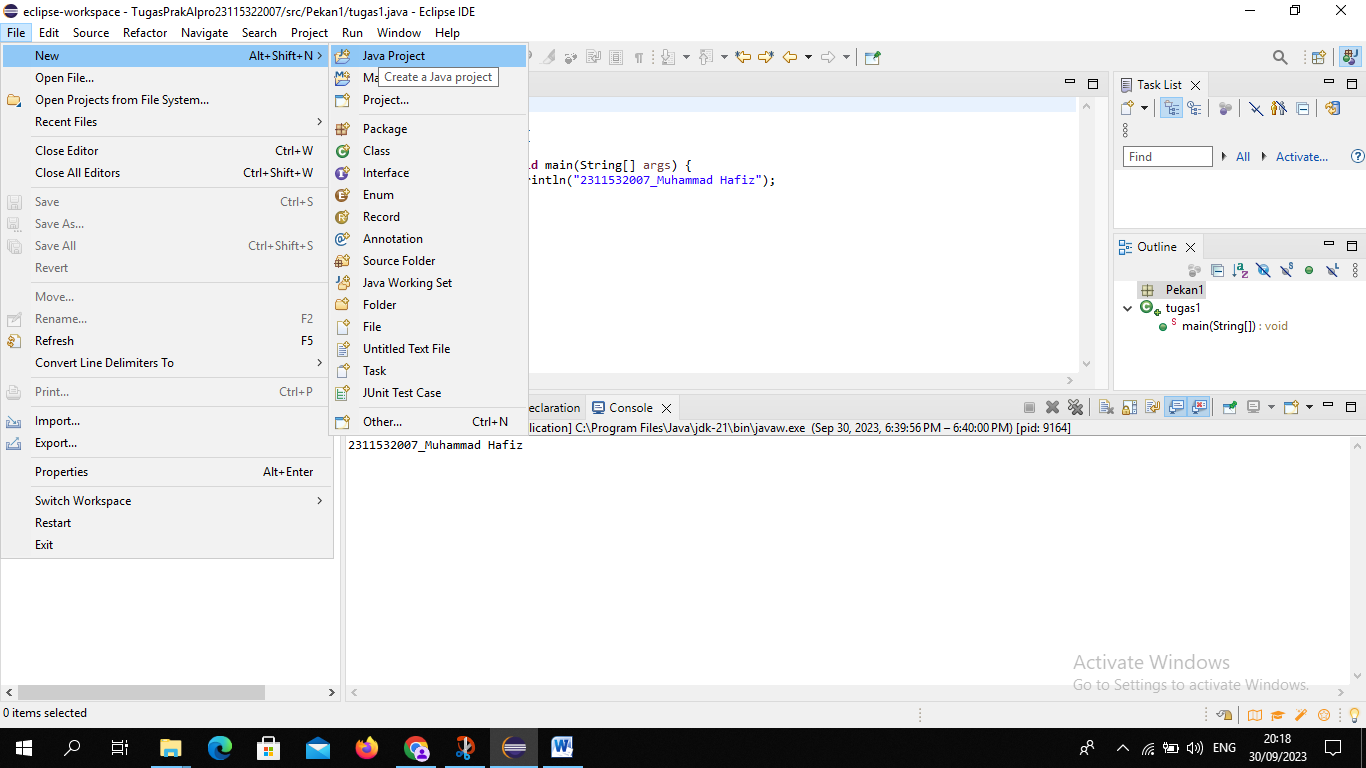
## Forking Repository

Forking repository artinya Anda membuat proyek baru berdasarkan repositori yang sudah ada. Dalam kalimat yang lebih sederhana, forking repo berarti Anda menyalin repositori yang sudah ada, kemudian membuat beberapa perubahan yang Anda perlukan, lalu menyimpan versi terbarunya sebagai repositori baru, dan menjadikannya proyek Anda sendiri.

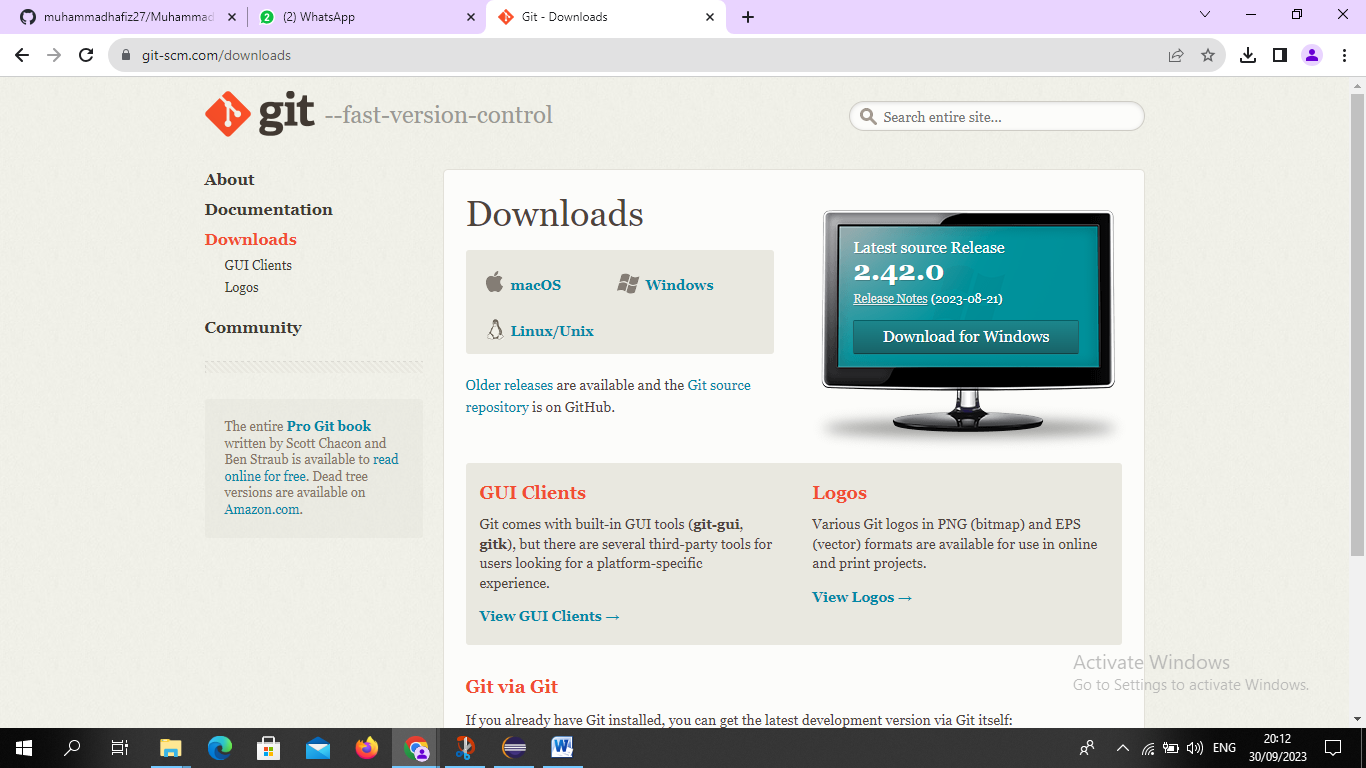
Fitur ini akan memperbaiki serta meningkatkan pengembangan proyek yang Anda lakukan. Karena proyek hasil forking masih baru, maka tidak akan terjadi apa-apa di repositori pusat. Perubahan pada repositori *master* juga dapat Anda terapkan di forking saat ini.

Berikut dua langkah untuk forking repositori GitHub:

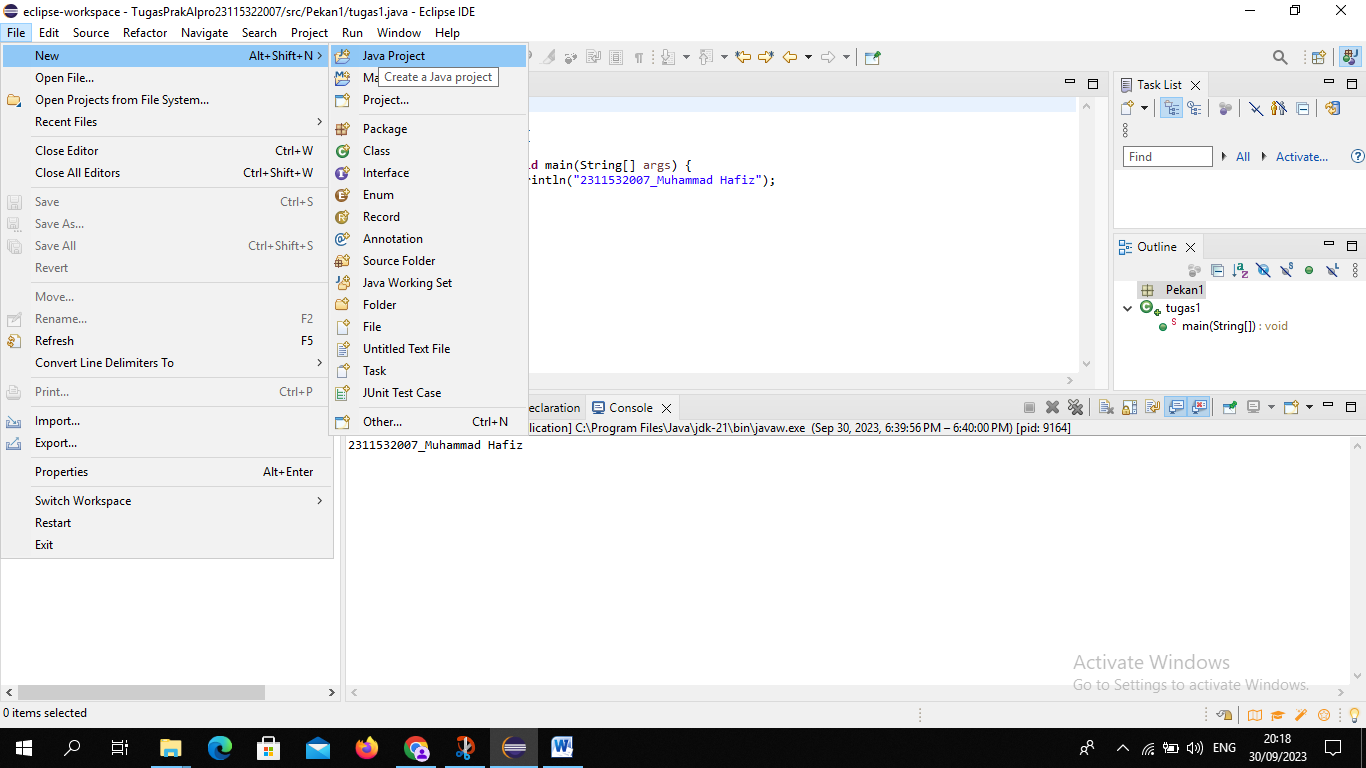
1. Cari repositori yang ingin Anda *forking*.
2. Klik opsi **Fork**.
3. LANGKAH PENGERJAAN
   1. Instal Eclips



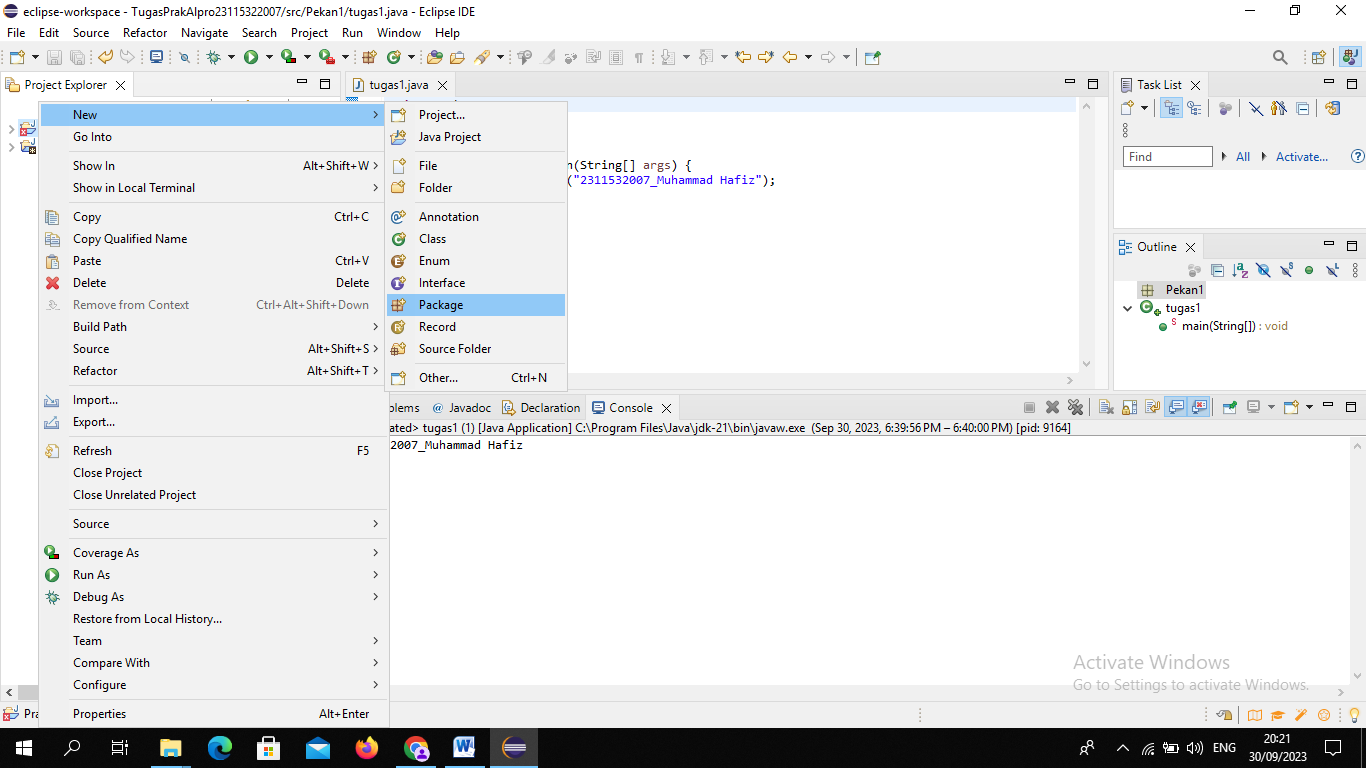
* 1. Instal Gitbash



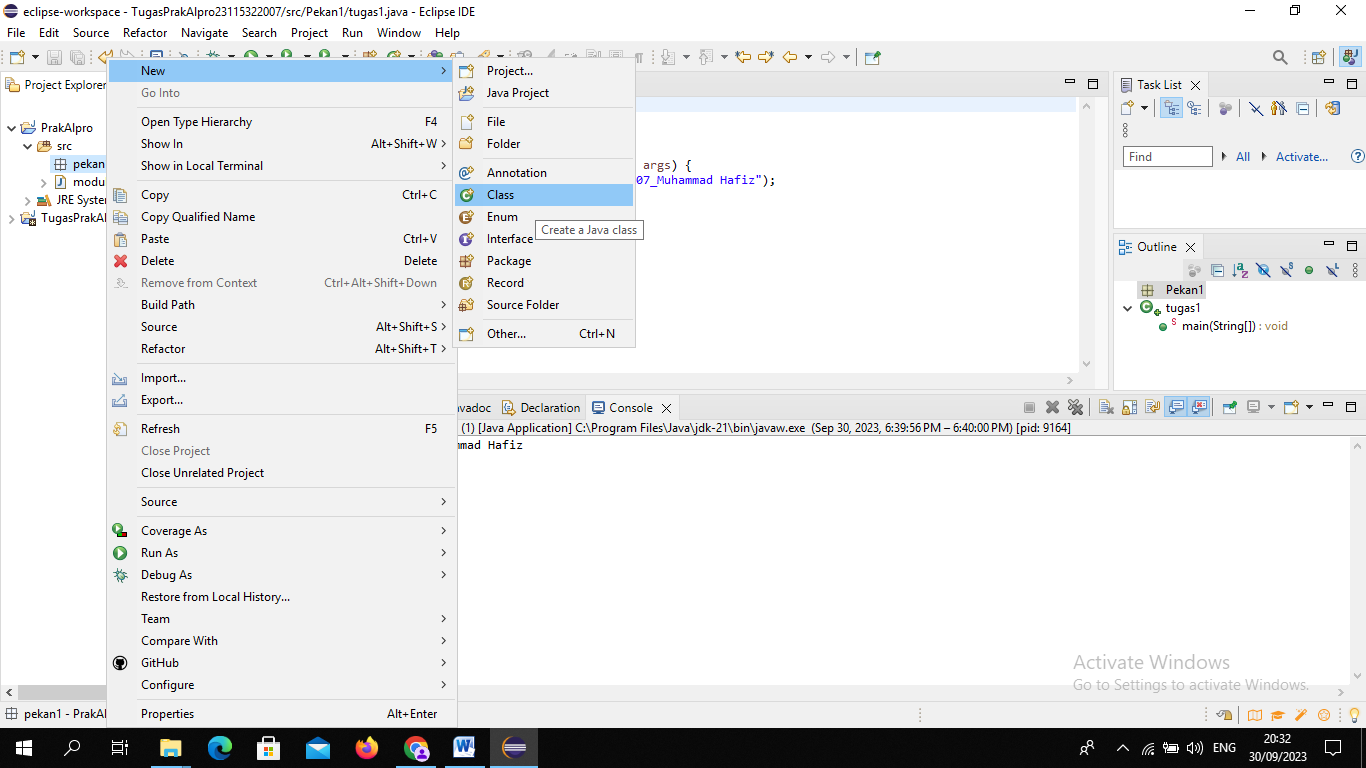
* 1. Setelah keduanya terpasang masuk kedalam eclips dan pilih *directory* file yang akan dijadikan tempat utama



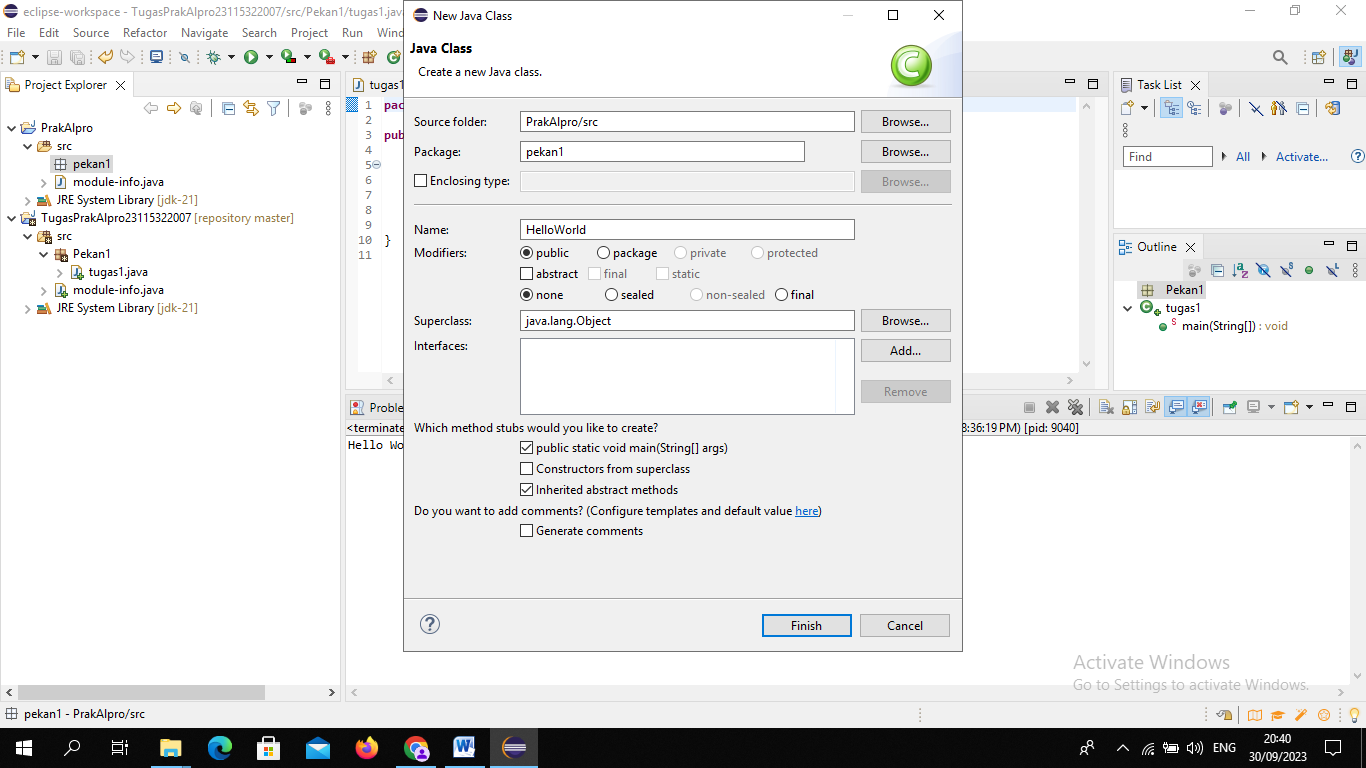
* 1. Klik kiri pada java project yang sudah dibuat lalu tekan tombol “New” dan klik “Package”



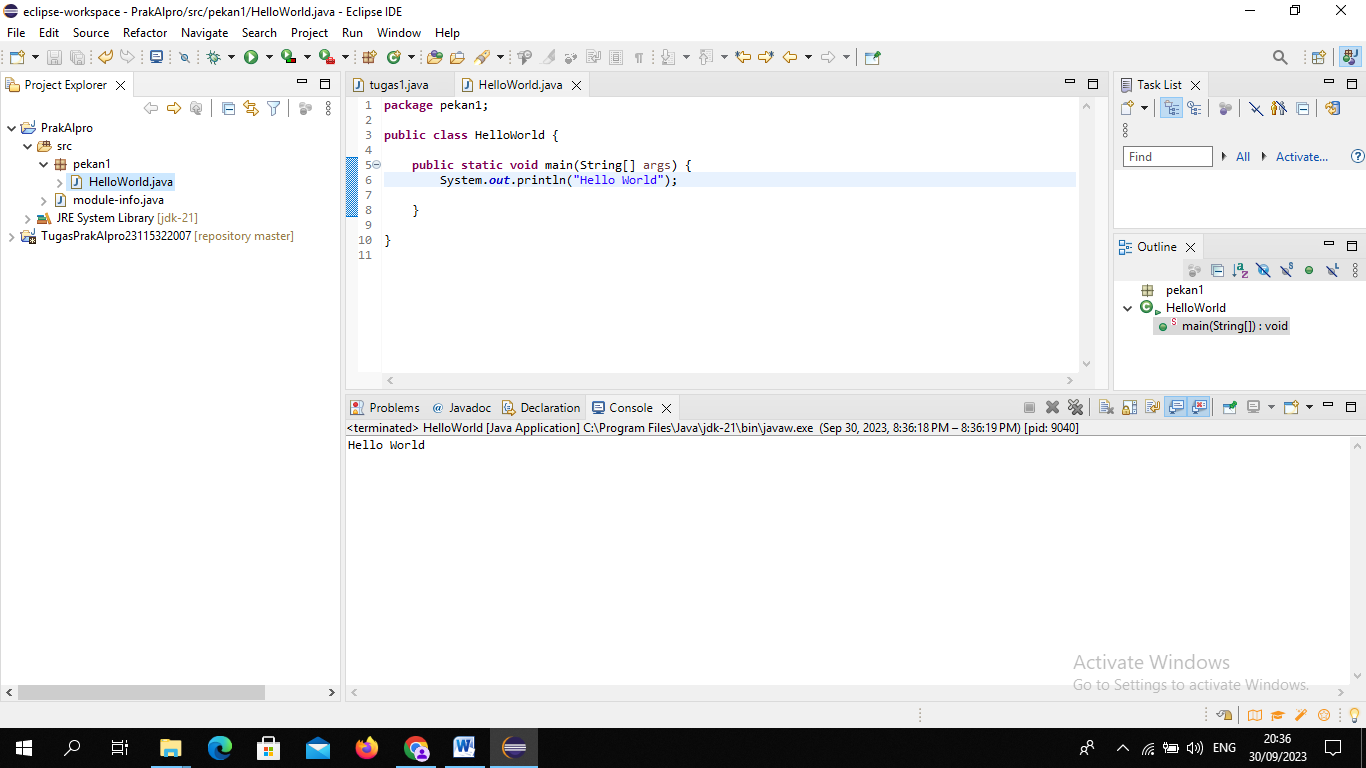
* 1. Klik kiri pada package yang dibuat tadi lalu tekan tombol “New” dan klik “Class”



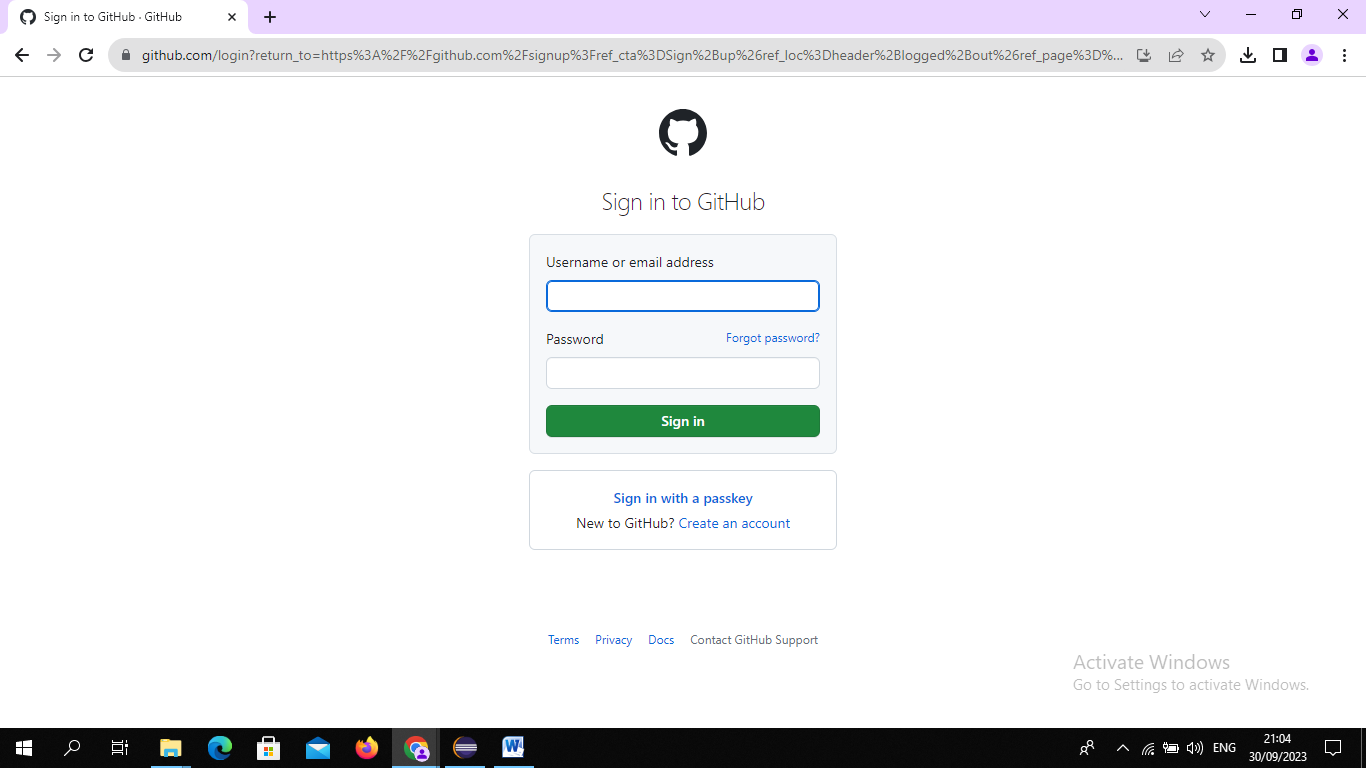
* 1. Setelah itu beri nama pada class yang akan dibuat dan beri tanda centang pada bagian public static void main



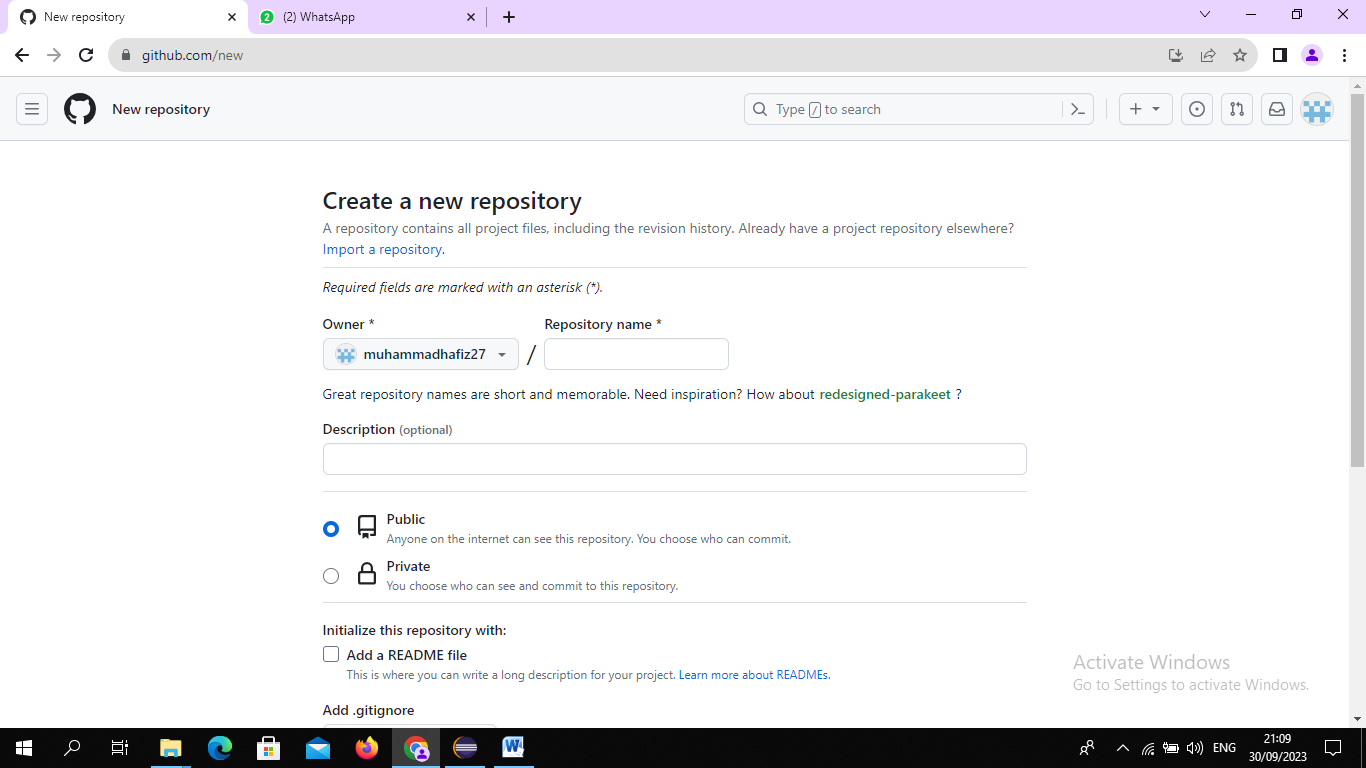
* 1. Untuk dapat menjalankan program harus diketik terlebih dahulu "System.***out***.println("");” barulah kalimat dimasukan di dalam tanda petik dua dan untuk menjalankan program dapat diklik “Run”



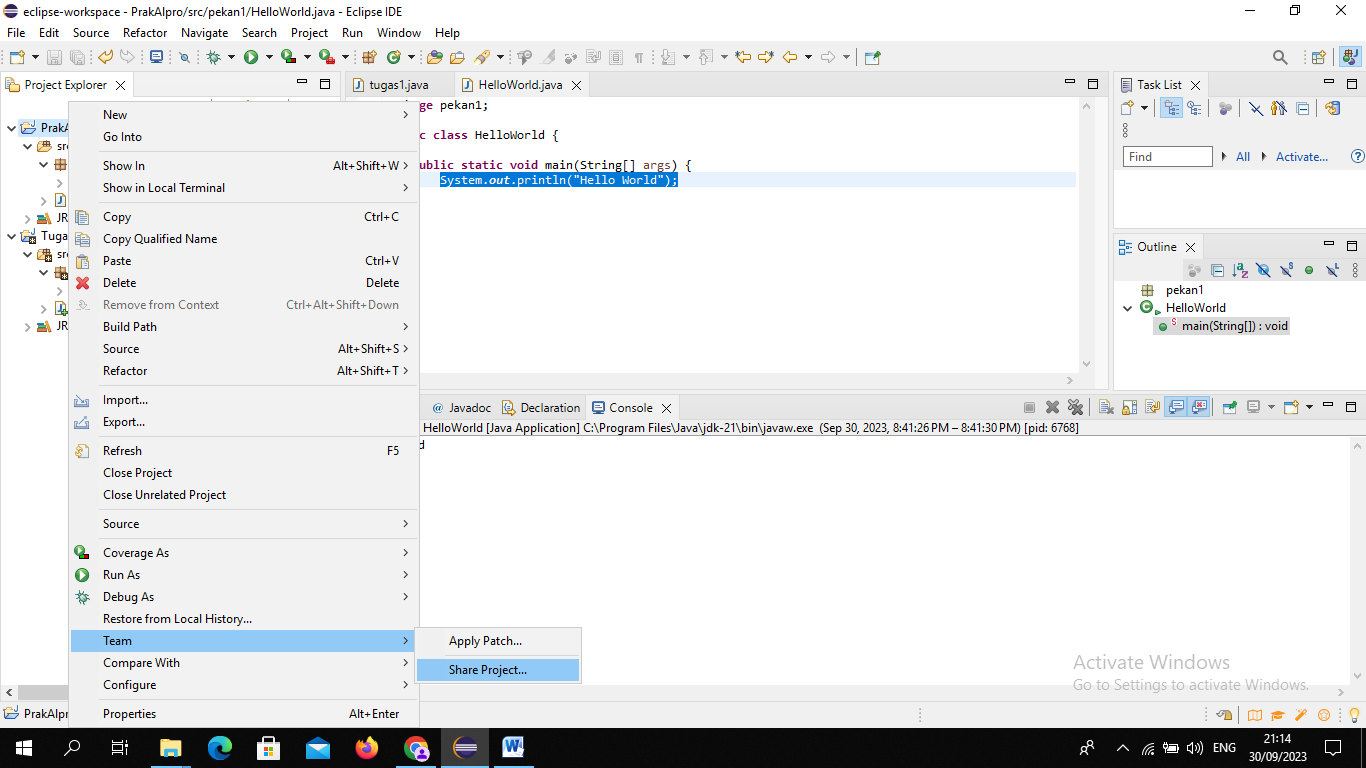
* 1. Log in ke dalam akun git hub



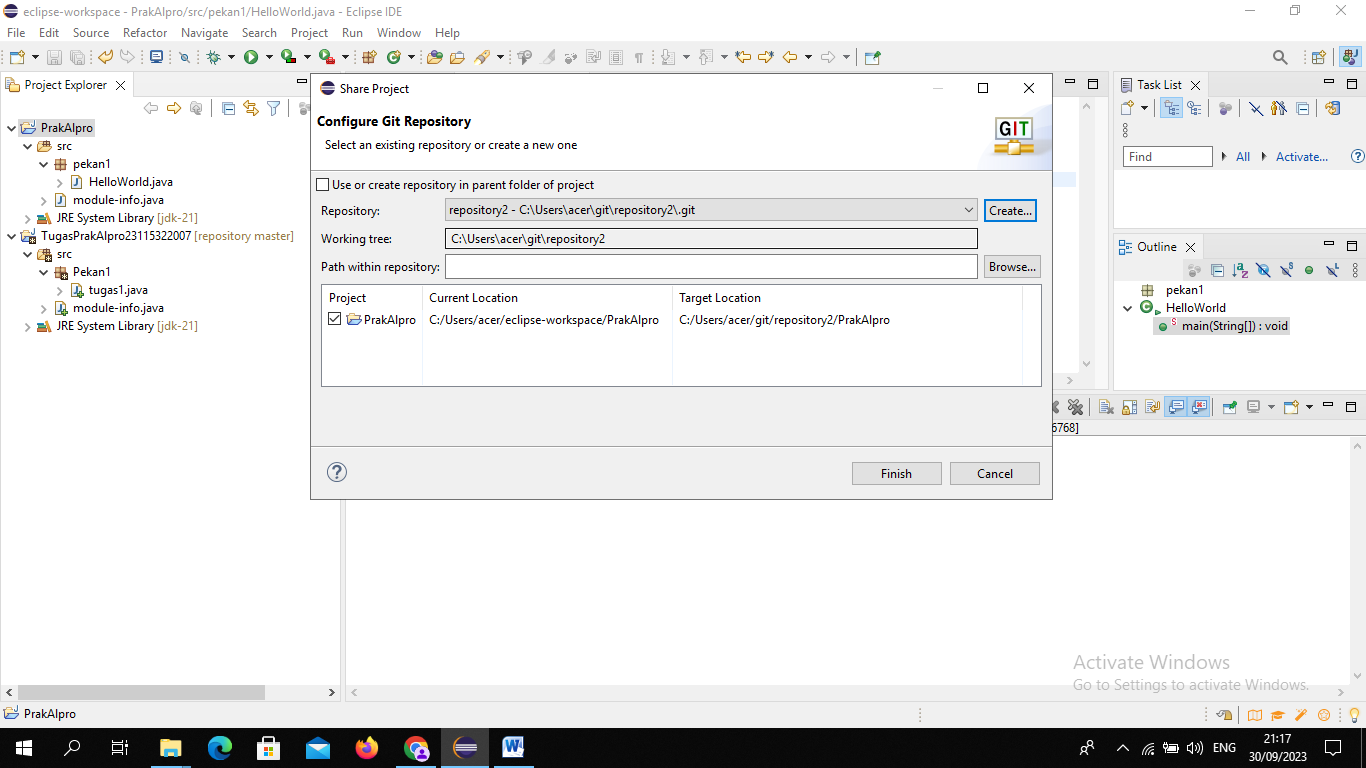
* 1. Setelah berhasil log in ke dalam git hub buatlah repository baru



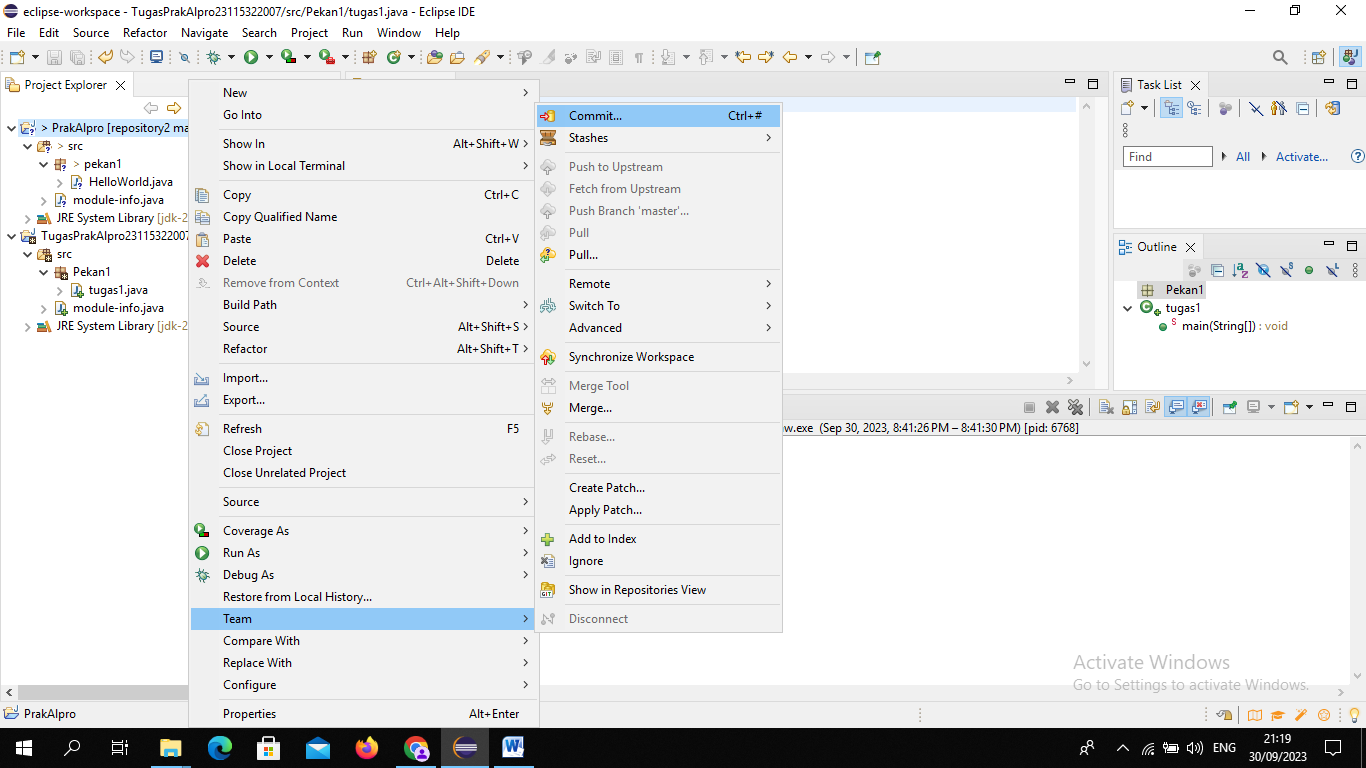
* 1. Untuk menghubungkan eclipse dan github klik kanan pada package yang ingin dihubungkan ke github lalu pilih “team” dan klik “share project”



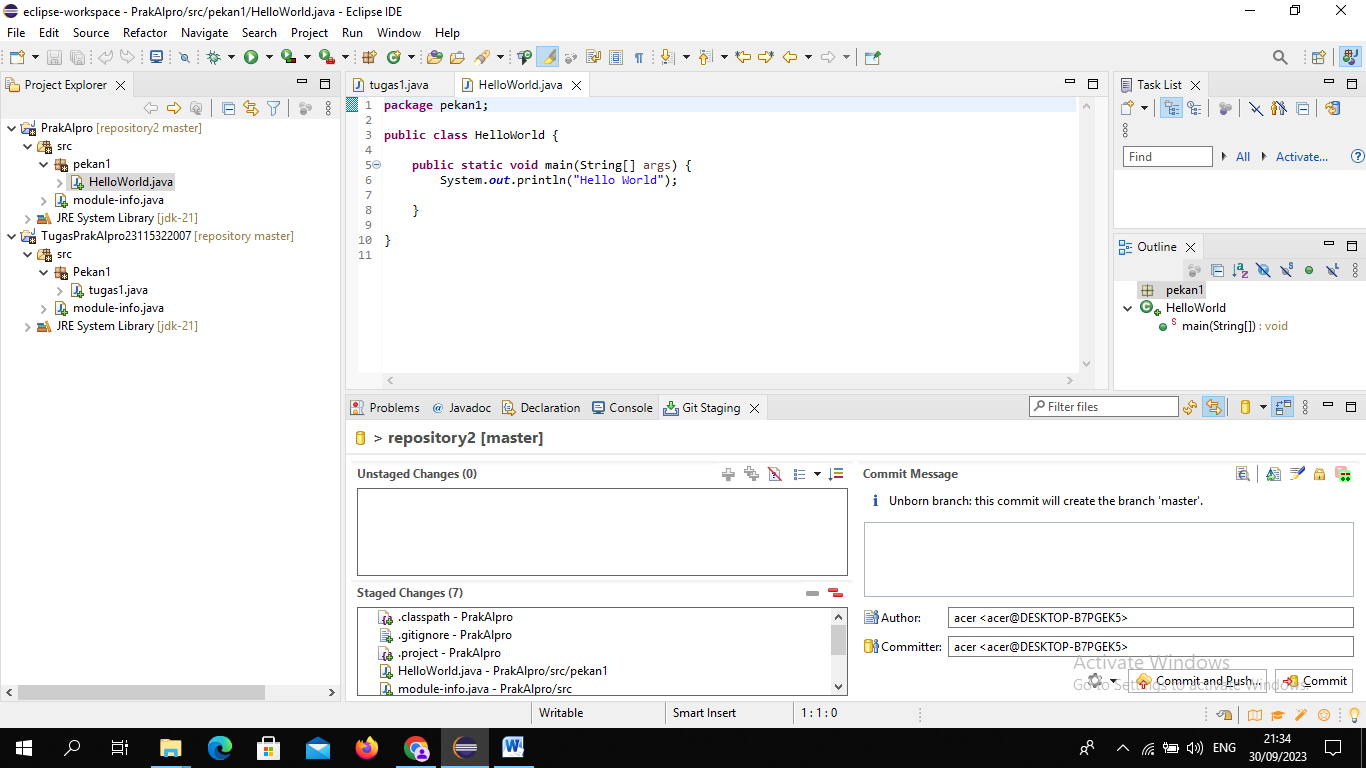
* 1. Lalu tekan “*create a new repository*” dan tekan “*finish*”



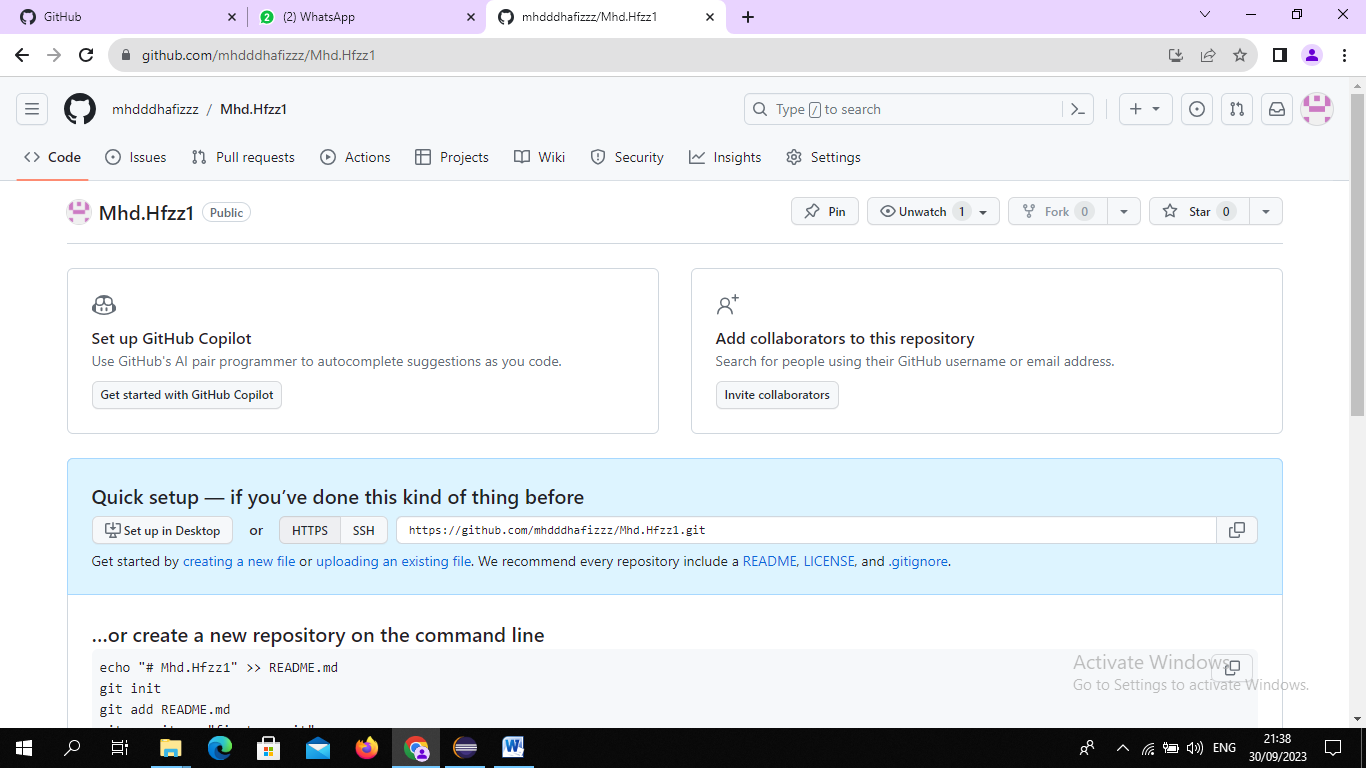
* 1. Setelah itu klik kanan kembali package yang ingn dihubungkan tadi dan pilih “team” kembali dan klik “commit”



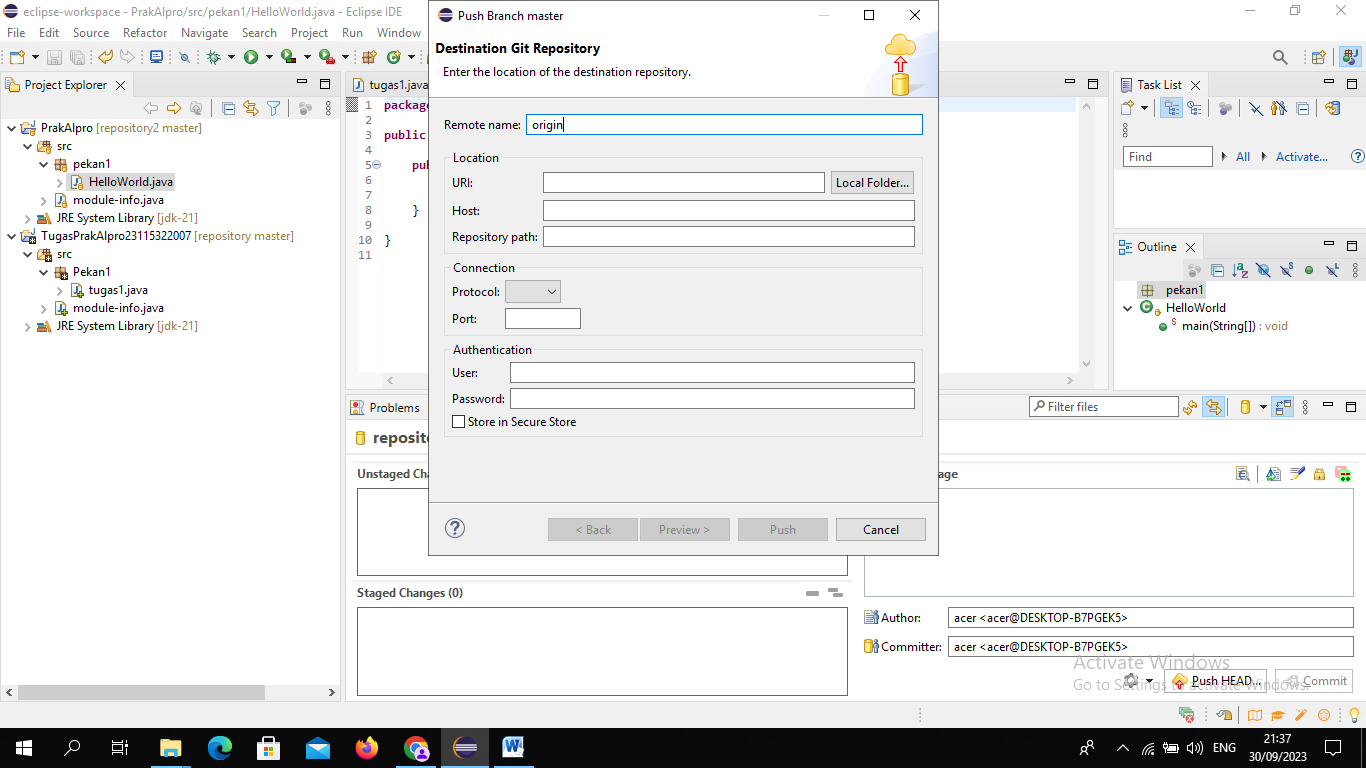
* 1. Lalu tekan tanda tambah dua dan klik “commit and push”



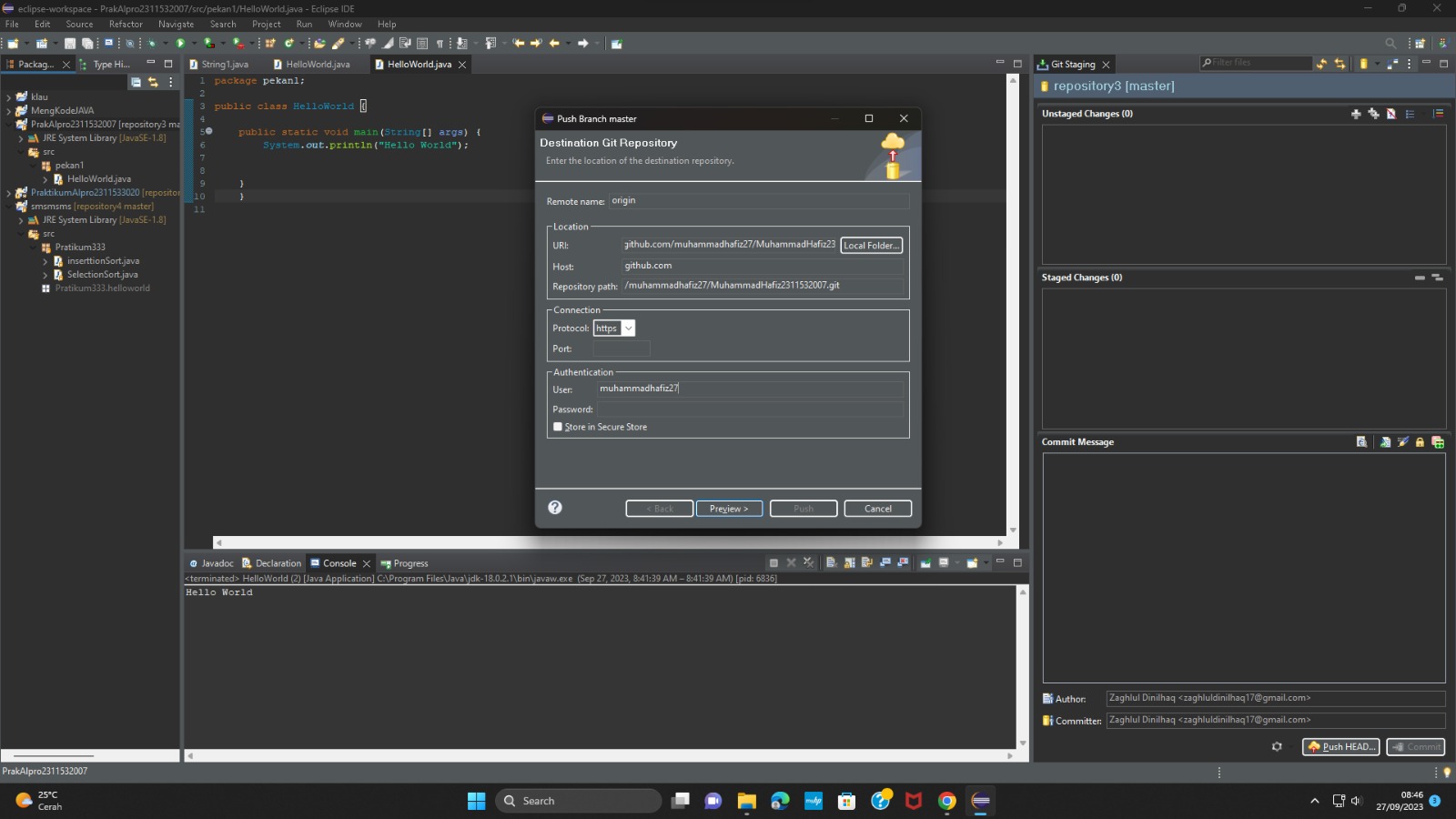
* 1. Salin URL yang ada pada github



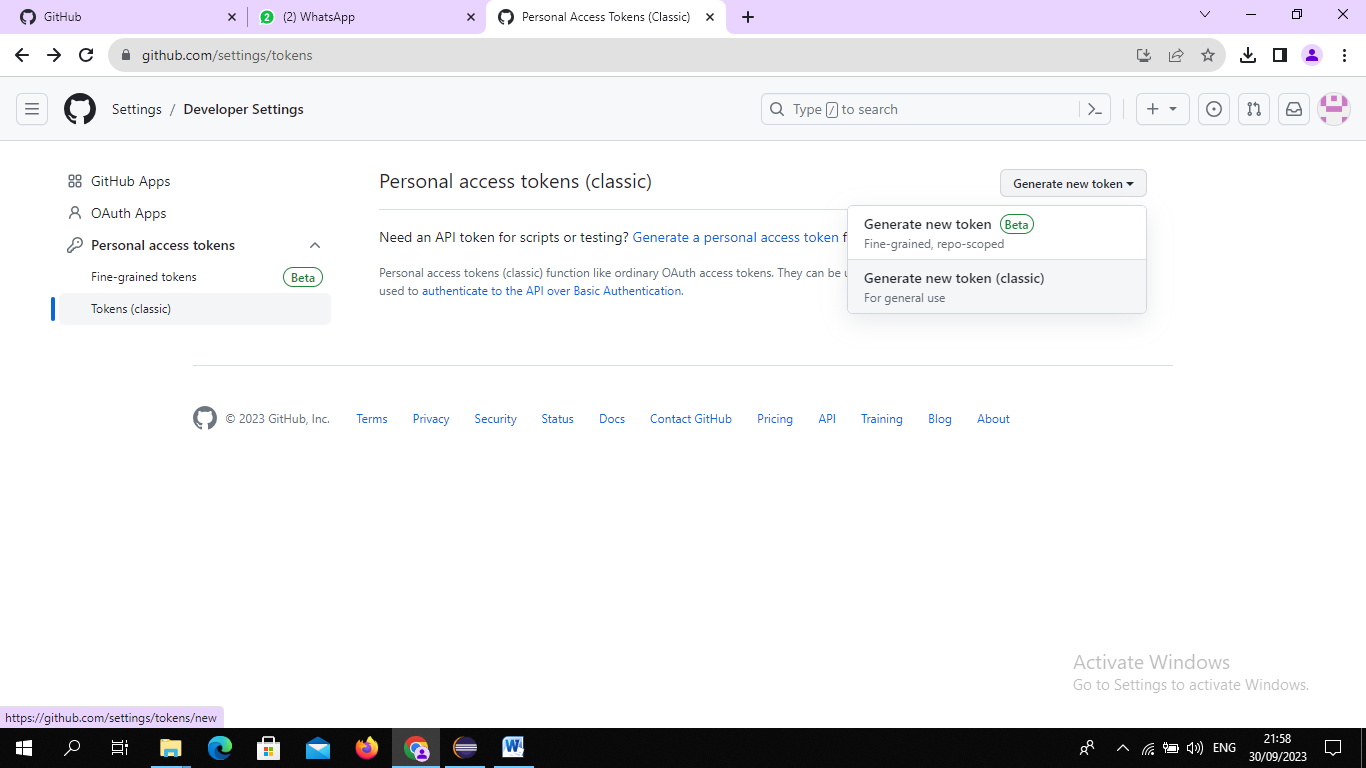
* 1. Setelah itu klik kanan kembali package yang ingn dihubungkan tadi dan pilih “team” kembali dan klik “push HEAD….”



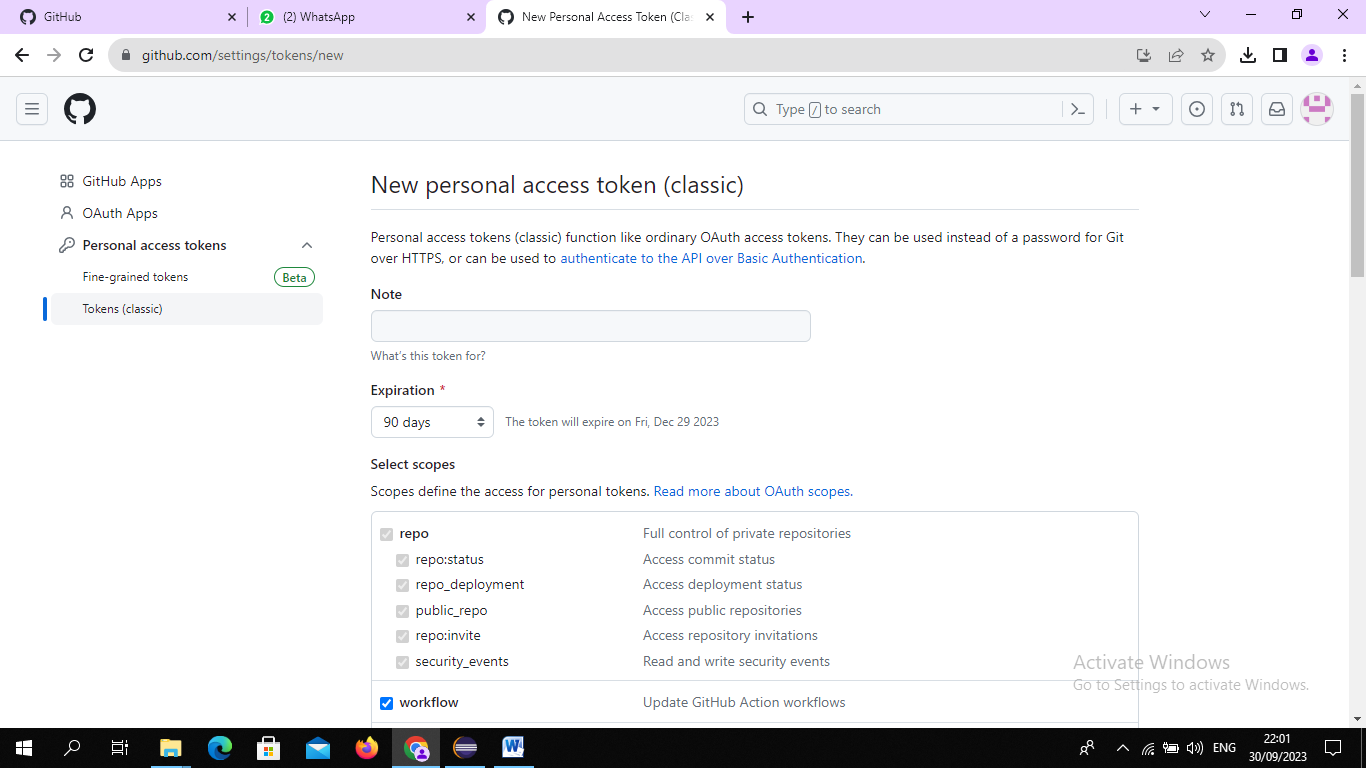
* 1. Lalu tempel URL yang sudah disalin dari git hub tadi pada menu URL



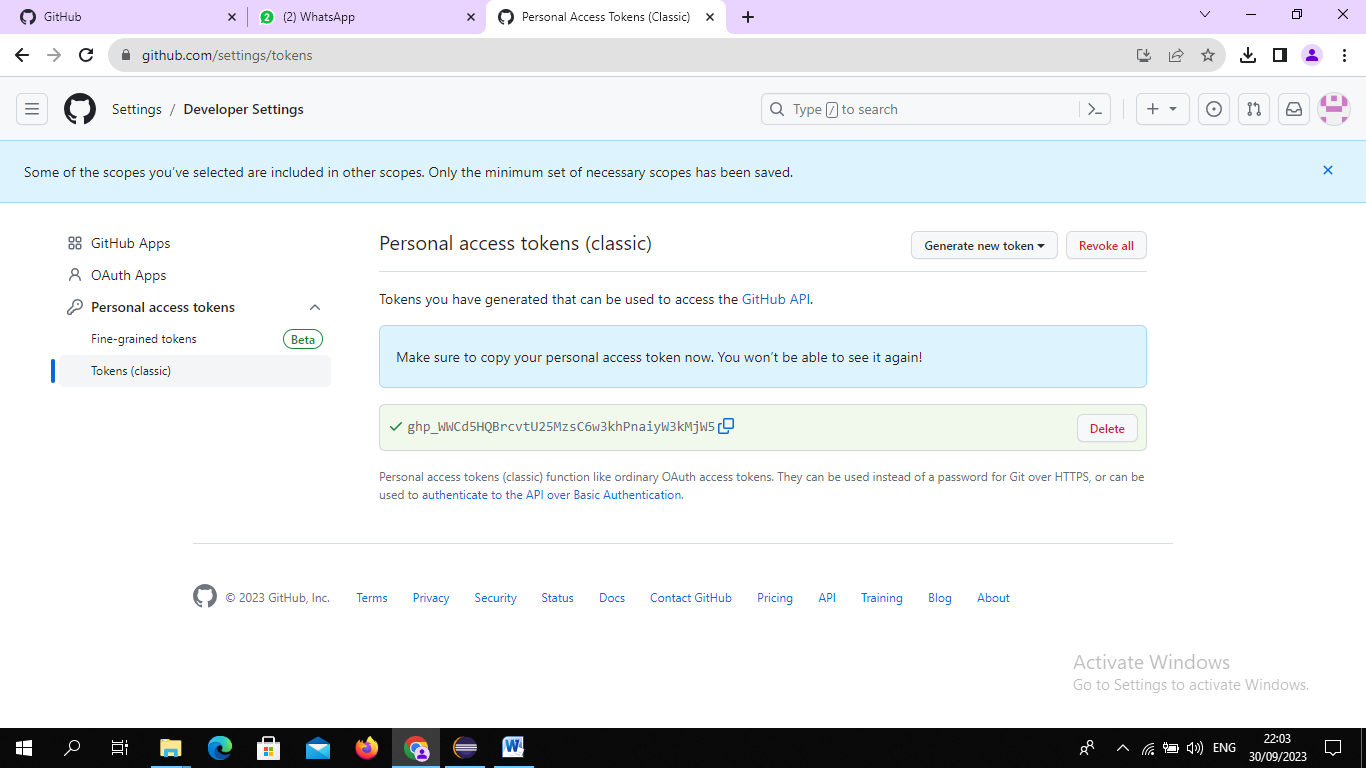
* 1. Lalu klik pada bagian sudut kanan atas dan pilih “setting” lalu klik bagian “Developer Settings” dan tekan “Personal access tokens” lalu pilih “tokens (classic)” lalu pilih “generate new token (classic)”



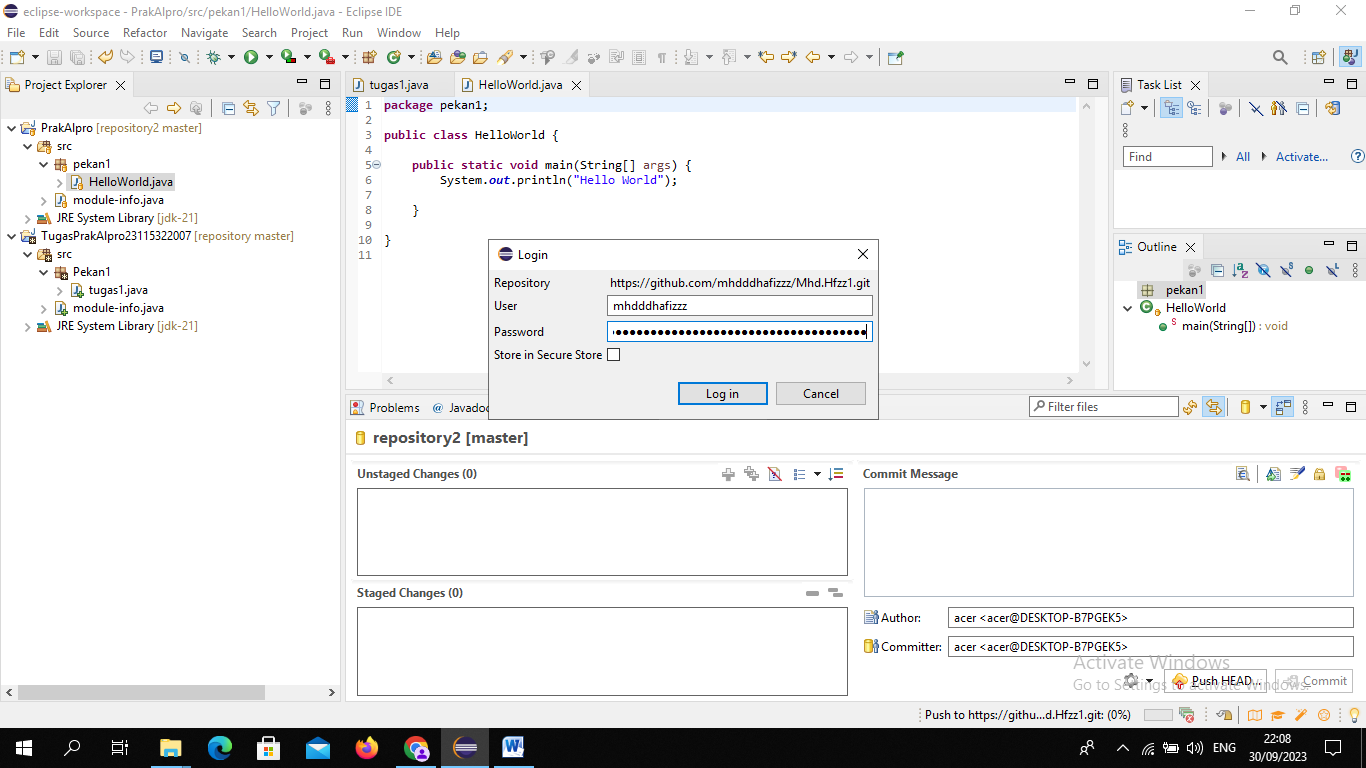
* 1. Setelah itu centang bagian “repo” lalu tekan “generate token” pada bagian paling bawah



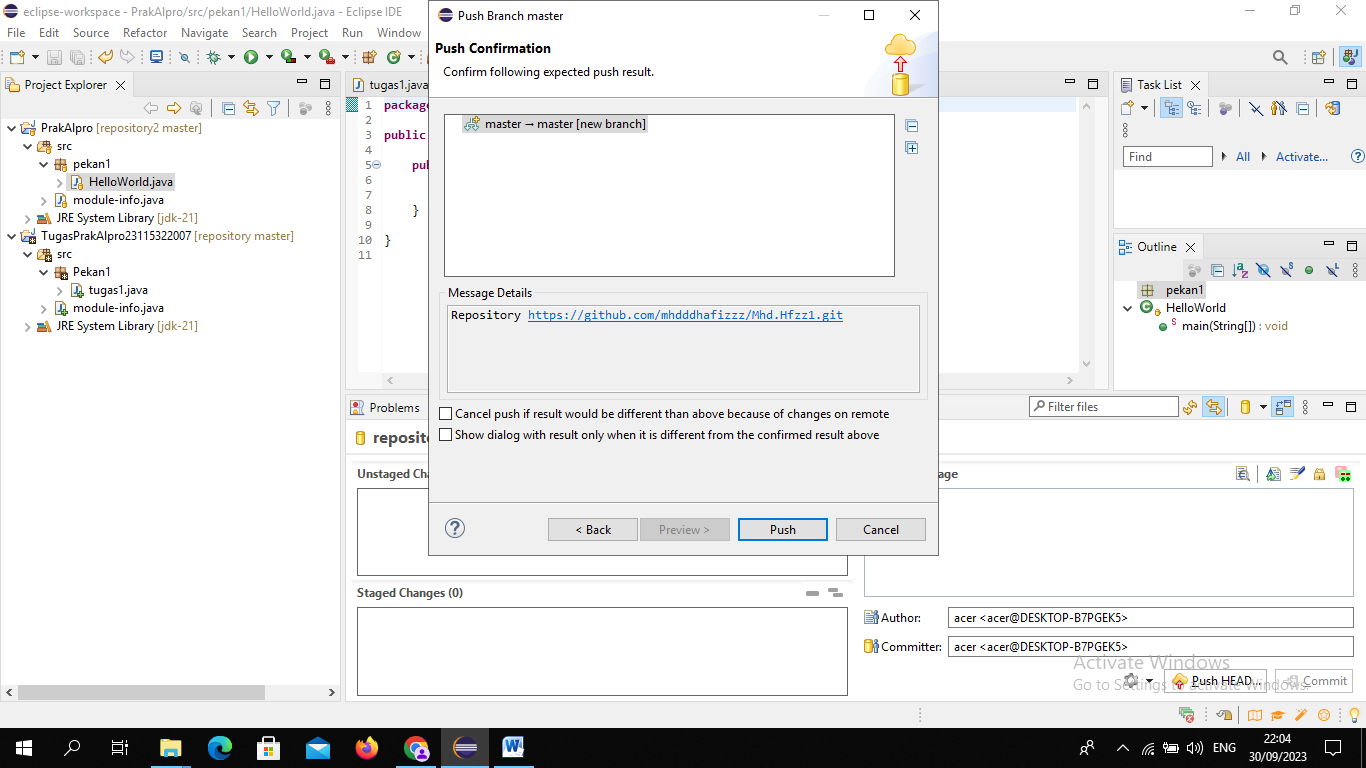
* 1. Lalu salin token tersebut



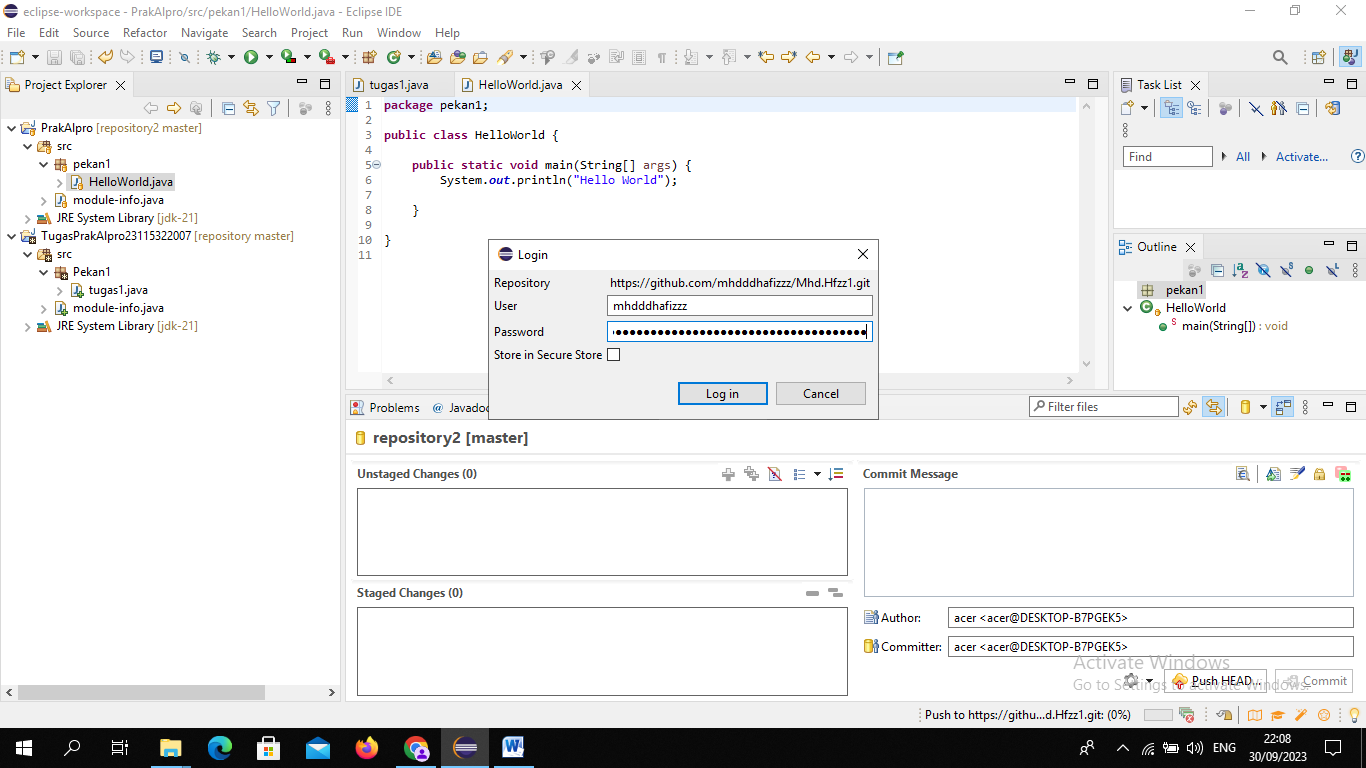
* 1. Pada menu log in masukan nama user github akun kita ke dalam bagian “user” dan untuk ”password” tempel token yang sudah disalin tadi



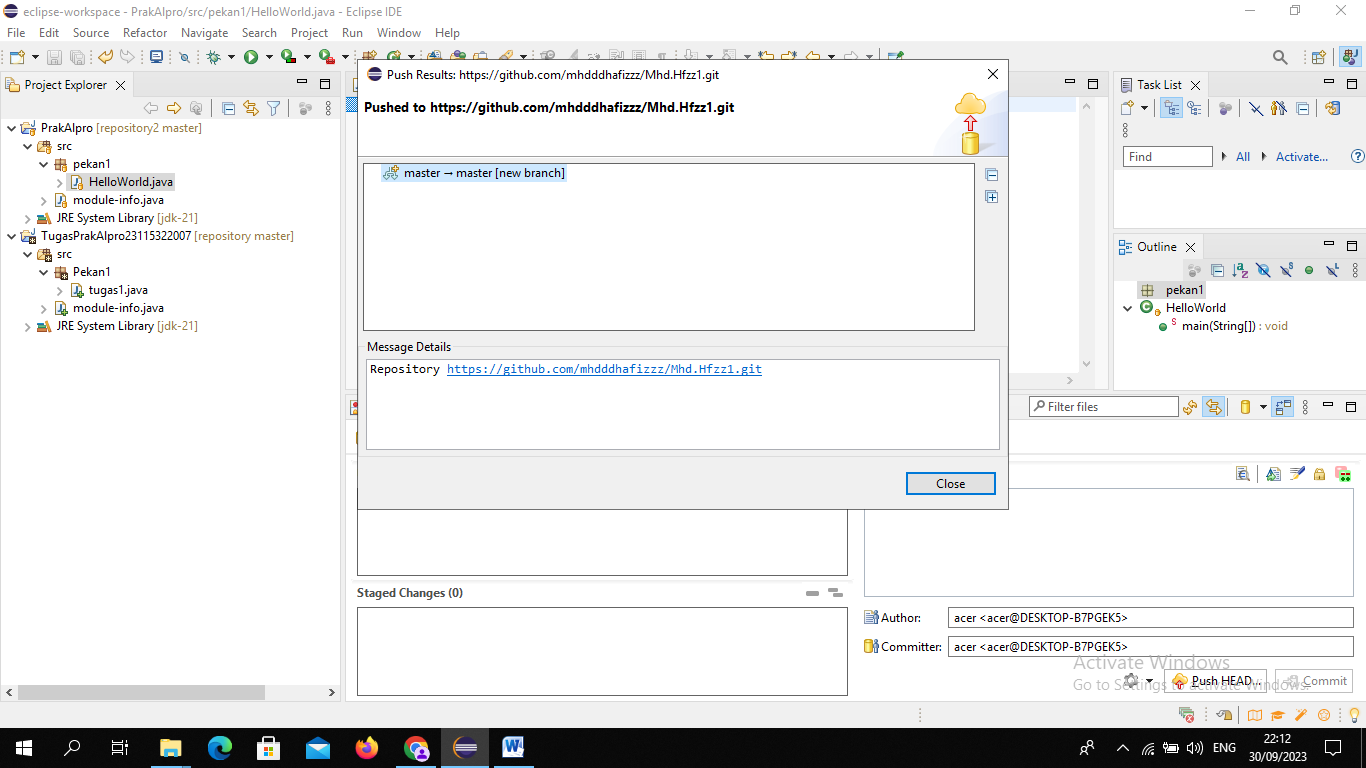
* 1. Setelah itu tekan “push”



* 1. Lalu ulangi log in dengan masukan nama user github akun kita ke dalam bagian “user” dan untuk ”password” tempel token yang sudah disalin tadi



* 1. Setelah itu tekan “close”



D. Hasil Praktikum

