# Ringkasan mengenai Git dan Github

# Git:

# Git adalah sistem kontrol versi terdistribusi yang memungkinkan pengembang melacak perubahan kode, berkolaborasi, dan mengelola berbagai versi proyek secara efisien. Beberapa fitur utama Git:

# Branching dan Merging: Memungkinkan pembuatan cabang (branch) untuk mengerjakan fitur baru tanpa mengganggu kode utama.

# Pelacakan Riwayat: Setiap perubahan disimpan dengan riwayat lengkap.

# Sistem Terdistribusi: Setiap pengguna memiliki salinan penuh dari repositori, memungkinkan pekerjaan dilakukan secara offline.

# GitHub:

# GitHub adalah platform berbasis cloud untuk menyimpan dan mengelola repositori Git secara online. Selain itu, GitHub memfasilitasi kolaborasi dengan menyediakan fitur-fitur seperti:

# Pull Requests: Memungkinkan pengajuan perubahan untuk ditinjau dan digabungkan ke dalam kode utama.

# Issues: Digunakan untuk melaporkan bug atau mengusulkan fitur baru.

# Actions: Memfasilitasi otomatisasi seperti CI/CD (Continuous Integration/Continuous Deployment).

# GitHub juga sering digunakan sebagai portofolio publik untuk menampilkan keterampilan pemrograman.

# Ide Proyek Menggunakan Git dan GitHub

# Aplikasi Daftar Tugas Kolaboratif

# Deskripsi: Kembangkan aplikasi daftar tugas (to-do list) yang memungkinkan beberapa pengguna untuk menambahkan, mengedit, dan menandai tugas.

# Langkah Implementasi:

# Gunakan Git untuk membuat cabang per fitur (misalnya: login, UI, notifikasi).

# Manfaatkan GitHub untuk meng-host kode dan berkolaborasi dengan tim Anda.

# Gunakan GitHub Issues untuk melacak tugas dan bug.

# Terapkan GitHub Actions untuk mengotomatisasi pengujian setiap kali ada perubahan kode.

# Proyek ini bisa menjadi latihan menarik untuk memahami GitHub sekaligus mengembangkan aplikasi praktis dengan fitur kolaborasi.

# SINTAKS GIT DAN GITHUB 1. Konfigurasi Awal

git config --global user.name "Nama Anda"  
git config --global user.email "email@example.com"  
Fungsi: Mengatur nama dan email pengguna yang akan tercatat pada setiap commit.

# 2. Membuat atau Mengkloning Repositori

git init  
Fungsi: Membuat repositori Git baru di folder lokal.  
  
git clone <url-repo>  
Fungsi: Mengunduh (clone) repositori dari GitHub ke lokal.

# 3. Menambah dan Melacak Perubahan

git status  
Fungsi: Menampilkan status file, apakah ada perubahan atau file baru yang belum dilacak.  
  
git add <nama-file> # Menambahkan satu file  
git add . # Menambahkan semua perubahan  
Fungsi: Menambahkan file ke staging area agar siap di-commit.

# 4. Menyimpan Perubahan ke Repositori

git commit -m "Pesan commit"  
Fungsi: Menyimpan perubahan dari staging area ke repositori lokal dengan pesan deskriptif.  
  
git commit -am "Pesan commit"  
Fungsi: Menambahkan dan commit perubahan sekaligus (hanya untuk file yang sudah pernah ditambahkan).

# 5. Mengelola Branch (Cabang)

git branch <nama-branch>  
Fungsi: Membuat branch baru.  
  
git checkout <nama-branch>  
Fungsi: Berpindah ke branch tertentu.  
  
git merge <nama-branch>  
Fungsi: Menggabungkan branch ke branch saat ini.

# 6. Mengirim dan Mengambil Perubahan dari GitHub

git push origin <nama-branch>  
Fungsi: Mengirim perubahan dari branch lokal ke GitHub.  
  
git pull origin <nama-branch>  
Fungsi: Mengambil dan menggabungkan perubahan terbaru dari GitHub.

# 7. Melihat Log atau Riwayat Perubahan

git log  
Fungsi: Menampilkan riwayat commit.

# 8. Mengatasi Konflik

Saat dua orang mengubah file yang sama, konflik mungkin terjadi. Ini bisa diselesaikan dengan:  
# Edit manual file yang konflik, lalu  
git add <file-yang-sudah-diperbaiki>  
git commit -m "Resolve conflict"

# 9. Menghapus atau Mengatur Ulang Perubahan

git reset <nama-file>  
Fungsi: Menghapus file dari staging area tapi tidak mengubah kontennya.  
  
git revert <commit-id>  
Fungsi: Membuat commit baru yang membatalkan perubahan dari commit tertentu.

# 10. Kerja Kolaboratif di GitHub

- Fork: Membuat salinan repositori milik orang lain ke akun sendiri.  
- Pull Request: Meminta pemilik repositori asli untuk menerima perubahan yang kita buat.  
- Issue: Melaporkan bug atau mendiskusikan ide pengembangan.