**Sesi 1**

Halo, Koders.

Perkenalkan, saya Hadi Hidayat Hammurabi biasa dipanggil Robi. Saya adalah seorang freelancer dengan fokus di fullstack developer, cuma lebih sering ambil job di backend. Portofolio lebih lengkap, cek di @hadihammurabi (Linkedin, Instagram, Github, Twitter).

Kali ini, saya akan menyampaikan materi mengenai Git. Level materi kali ini adalah \*\*dasar/mudah\*\*.

Adapun materi yang akan saya sampaikan, berikut ini:

1. \*\*Pengenalan Git\*\*

2. \*\*Perintah dasar Git\*\*

\* Sampai di sini ada yang ingin menyanggah atau menanyakan sesuatu?

**Sesi 2**

Oke, kita lanjutkan.

Masuk ke materi pertama, yaitu pengenalan Git.

Git merupakan salah satu perangkat lunak pengontrol versi.

Git ini membantu kita untuk memanajemen versi dari sebuah proyek yang sedang kita kerjakan.

Misalkan, kita punya sebuah proyek, apabila kita mengubah sesuatu, baik itu menambah maupun menghapus, tentunya kita tidak bisa mengingat semua hal yang sudah kita lakukan. Dengan Git, kita dapat dengan mudah mengetahui bagian mana saja yang pernah kita ubah.

Perangkat lunak pengontrol versi yang lainnya.

1. SVN

2. CVS

3. Mercurial

\* Oke, jelas? Atau ada yang mau nanya? Silakan.

**Sesi 3**

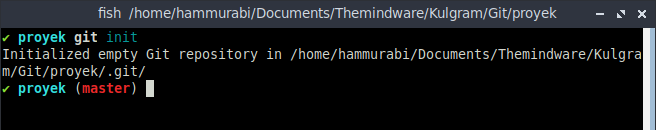
Oke, lanjut lagi ya.

Sekarang kita masuk ke perintah dasar Git. Kita langsung sambil praktik ya.

Silakan temen-temen buat folder kosong dengan nama “proyek” atau apa saja seenak temen-temen, tapi saya pake itu.

**Part 3.1**

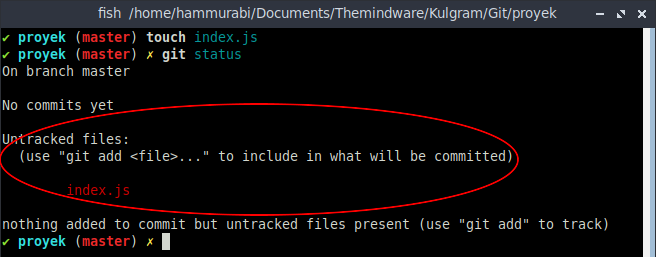
Sekarang, di dalam folder tersebut, temen-temen jalankan perintah ```git init``` dan akan muncul output kayak gini.



Ini artinya kita sudah punya sebuah proyek yang akan kita manajemen menggunakan git.

**Part 3.2**

Setelah itu, silakan temen-temen tambahkan sebuah file ke dalam folder tersebut. Misalkan kita tambahkan file js dengan nama “index.js” lalu temen-temen jalankan perintah ```git status``` maka akan tampil seperti ini.



Pada bagian yang dilingkari, kita bisa lihat file apa saja yang belum direkam oleh git atau disebut unstaged atau untracked. Kita harus membuat git merekam file ini agar kita dapat memanajemennya.

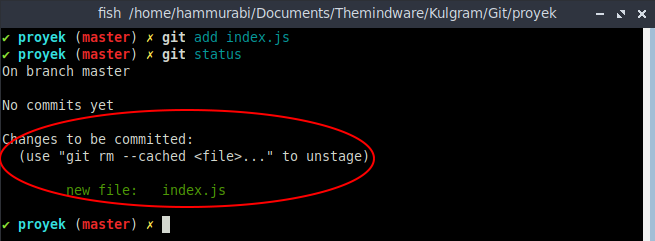
\* Bagaimana? Sudah bingung? Kalau bingung tanya dulu dah yow.

**Part 3.3**

Lanjuuutttt lage.

Selanjutnya kita akan mengubah status file yang unstaged tadi menjadi staged.

Caranya adalah dengan menjalankan perintah ```git add index.js``` maka apabila kita lihat akan seperti pada gambar.



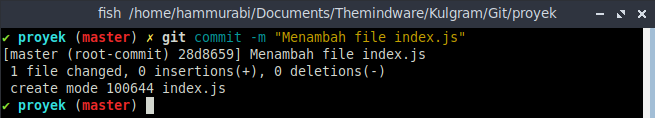
Ini tandanya, file “index.js” sudah staged dan siap untuk dicommit.

Oya, cara ini digunakan apabila kita ingin menambahkan file tertentu, kalau kita punya banyak file yang akan kita stage semua, maka gunakan perintah ```git add .```

**Part 3.4**

Selanjutnya, kita akan commit file kita yang sudah staged tadi.

Caranya dengan menggukan perintah ```git commit -m “Menambah file index.js”``` maka akan tampil seperti gambar berikut.



Ini artinya kita menandai dan menyimpan semua perubahan yang kita lakukan.

Apabila kita lihat status, maka akan kita lihat bahwa proyek kita bersih dan tidak ada perubahan apapun.

Untuk tambahan, silakan temen-temen coba ubah-ubah apapun di dalam proyek, dan lihat statusnya.

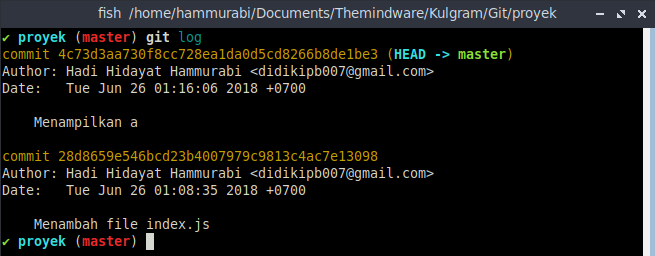
\* Huuffftt… Sampai di sini. Ada yang ditanyakan?

**Part 3.5**

Yaudah, lanjut lagi.

Setelah kita melakukan commit. Tentunya semua commit yang kita lakukan akan dicatat oleh git. Bagaimana cara melihat catatan tersebut?

Gunakan perintah ```git log``` dan akan tampil semua commit yang pernah kita lakukan. Pada kasus ini, saya menambahkan beberapa perubahan dan commit. Seperti pada gambar ini.



**Part 4**

Oke. Ini adalah bagian terakhir dan penutup. Apabila temen-temen masih punya pertanyaan, disilakan untuk bertanya.

Nahh… Sampai di sini, saya harap temen-temen tau apa itu git, manfaatnya, dan dapat menggunakannya dalam keseharian (bekerja).

Saya tutup. Terima kasih atas antusiasme temen-temen semua. Apabila ada kekurangan maupun kesalahan, saya mohon maaf sebesar-besarnya.

Apabila masih ada sesuatu yang terpendam. Dapat didiskusikan setelah kulgram ini.

Terima kasih.

Penuh hormat,

Hadi Hidayat Hammurabi