

Version Control System (VCS)

Adi Wahyu Pribadi



Skripsi v1

1 Januari 2023



Revisi pertama

2 Januari 2023



Skripsi v2

15 Februari 2023



Revisi kedua

16 Februari 2023



Skripsi v3

1 Maret 2023



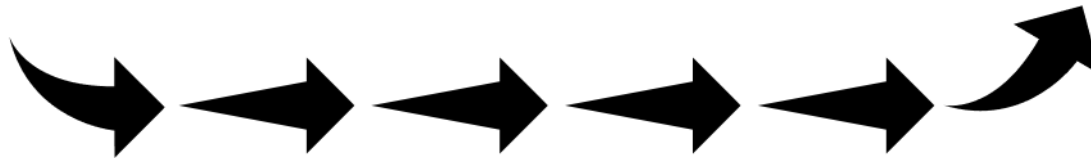
Revisi Ketiga

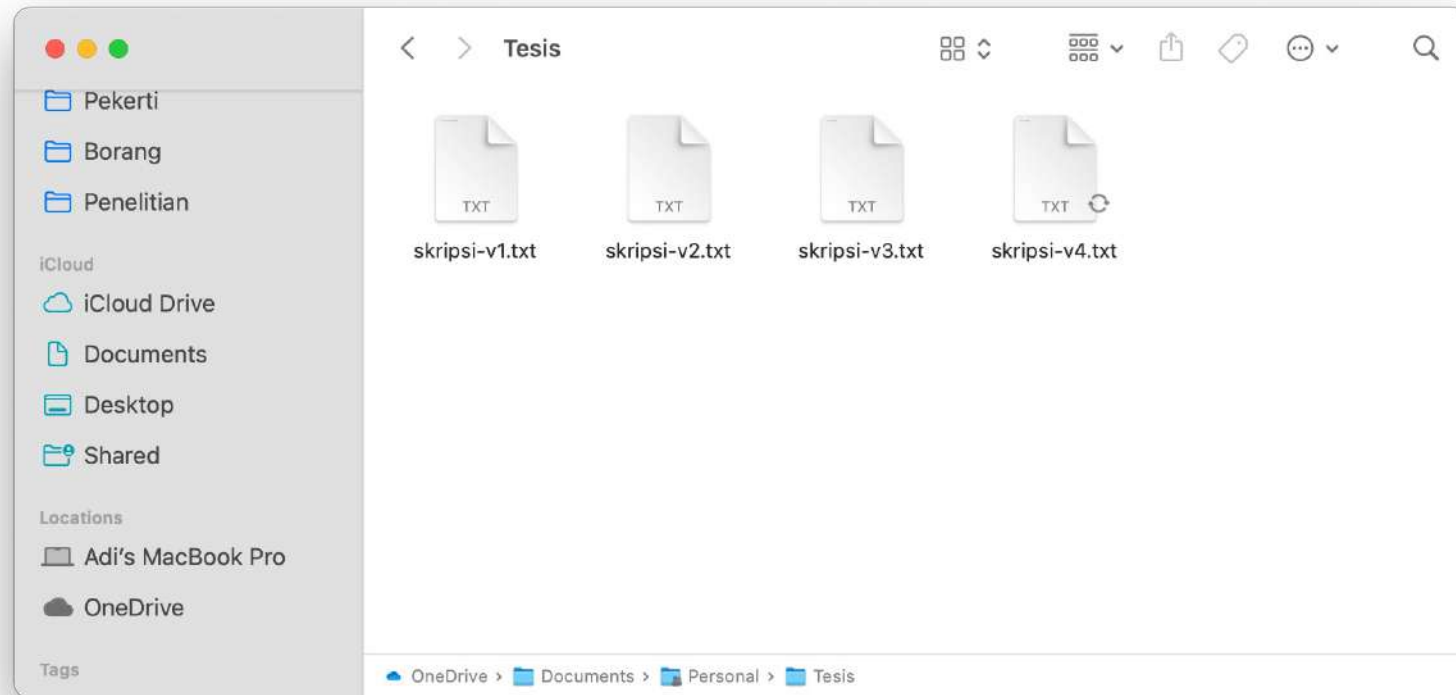
2 Maret 2023



Skripsi v4

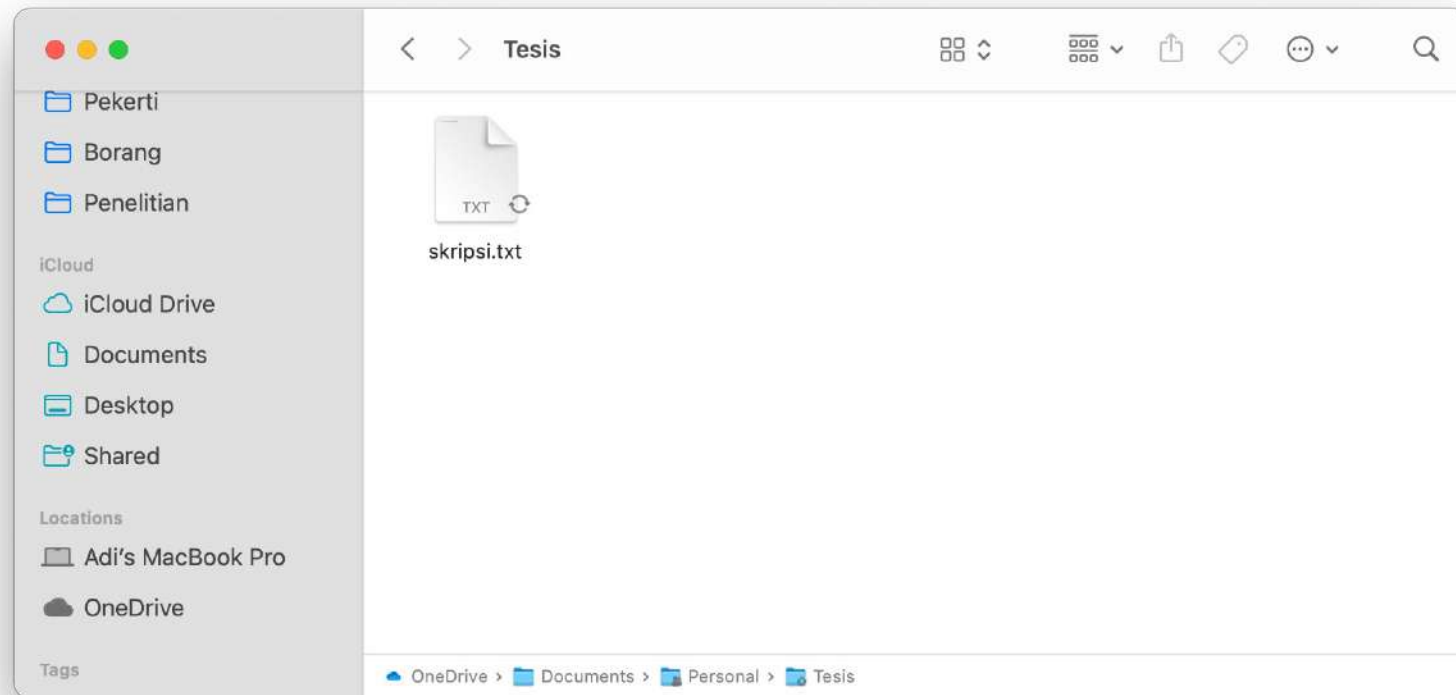
10 Maret 2023





Membuat berkas baru
setiap ada perubahan

Biasanya: save as



Bagaimana kalau langsung save??

skripsinya ditiban lupa pakai versi

Bagaimana cara supaya
bisa kembali ke versi sebelumnya?



Definisi

- Alat bantu untuk melacak perubahan berkas (file) dari waktu ke waktu.
- VCS bekerja dengan menyimpan snapshot file pada waktu yang berbeda.
- Bisa kembali ke snapshot file yang sebelumnya.

Jenis VCS

- Terpusat
 - Menyimpan semua file snapshot di server secara terpusat.
 - Pengguna dapat “checkout” atau mengambil versi tertentu dari file dari server pusat
 - Contoh: [Subversion](#) (SVN) dan [Perforce](#).
- Terdistribusi
 - Setiap pengembang memiliki salinan lengkap repositori di komputer mereka sendiri.
 - Pengembang dapat melakukan perubahan pada file mereka sendiri, dan kemudian menggabungkan perubahan tersebut dengan salinan repository orang lain.
 - Contoh: [Git](#) dan [Mercurial](#).

The mind behind Linux | Linus Torvalds | TED



Git

Git is a free and open source distributed version control system designed to handle everything from small to very large projects with speed and efficiency.

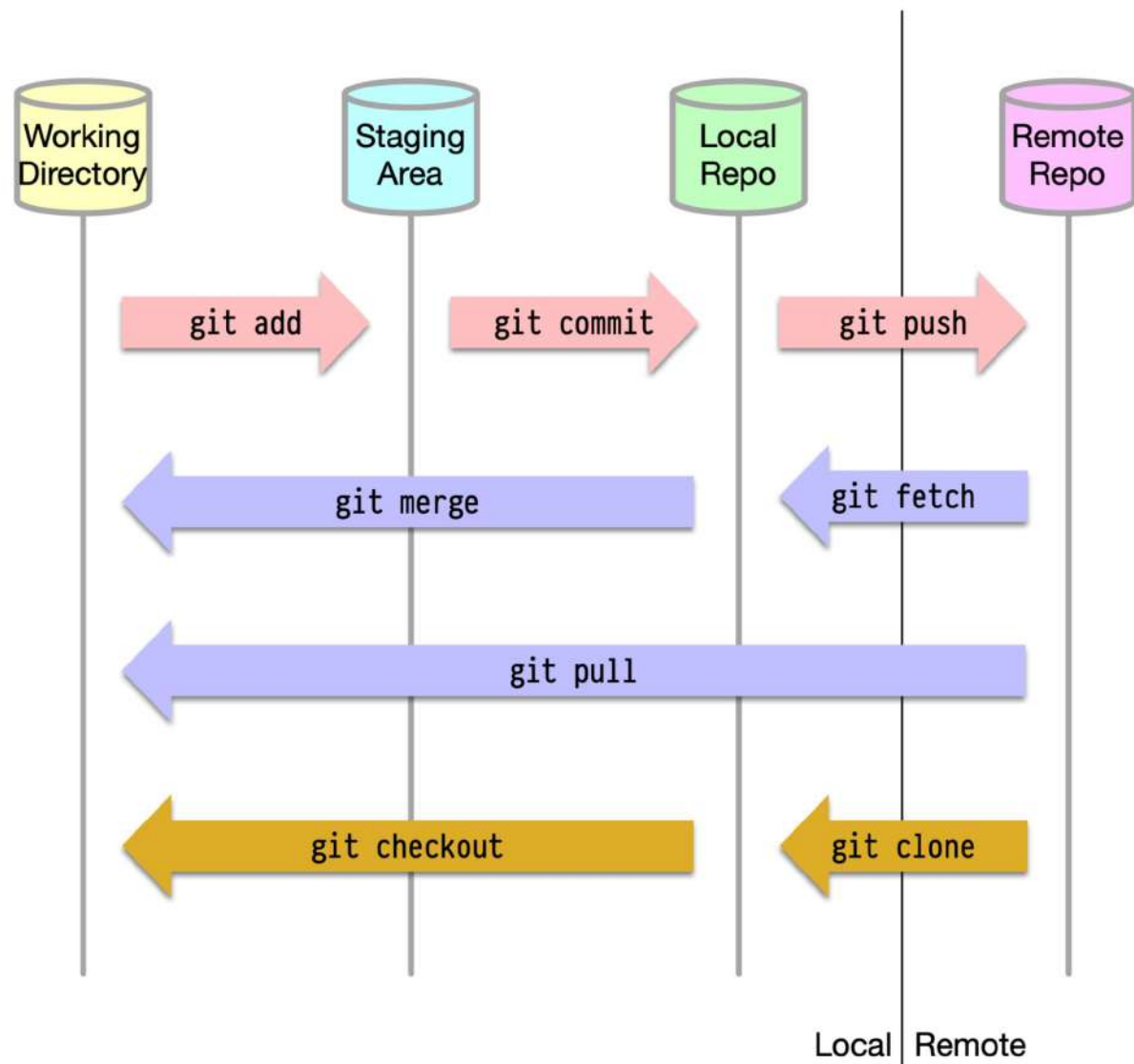
Sejarah Git

- Git adalah system control versi terdistribusi yang pertama kali diciptakan dan dikembangkan oleh Linus Torvalds. Linus Torvalds sendiri adalah pencipta dan pengembang Sistem Operasi Linux. Hingga saat ini, Linus Torvalds memelihara inti system operasi Linux (kernel Linux)
- Sebelum git, proyek kernel linux menggunakan BitKeeper untuk manajemen control versinya. Namun pada 2005, hubungan kedua memburuk sehingga akses gratis ke BitKeeper dihapus.
- Linus mulai merilis git pada April 2005 dan kernel linux mulai menggunakan Git pada 7 Juli 2005.

Fitur Git

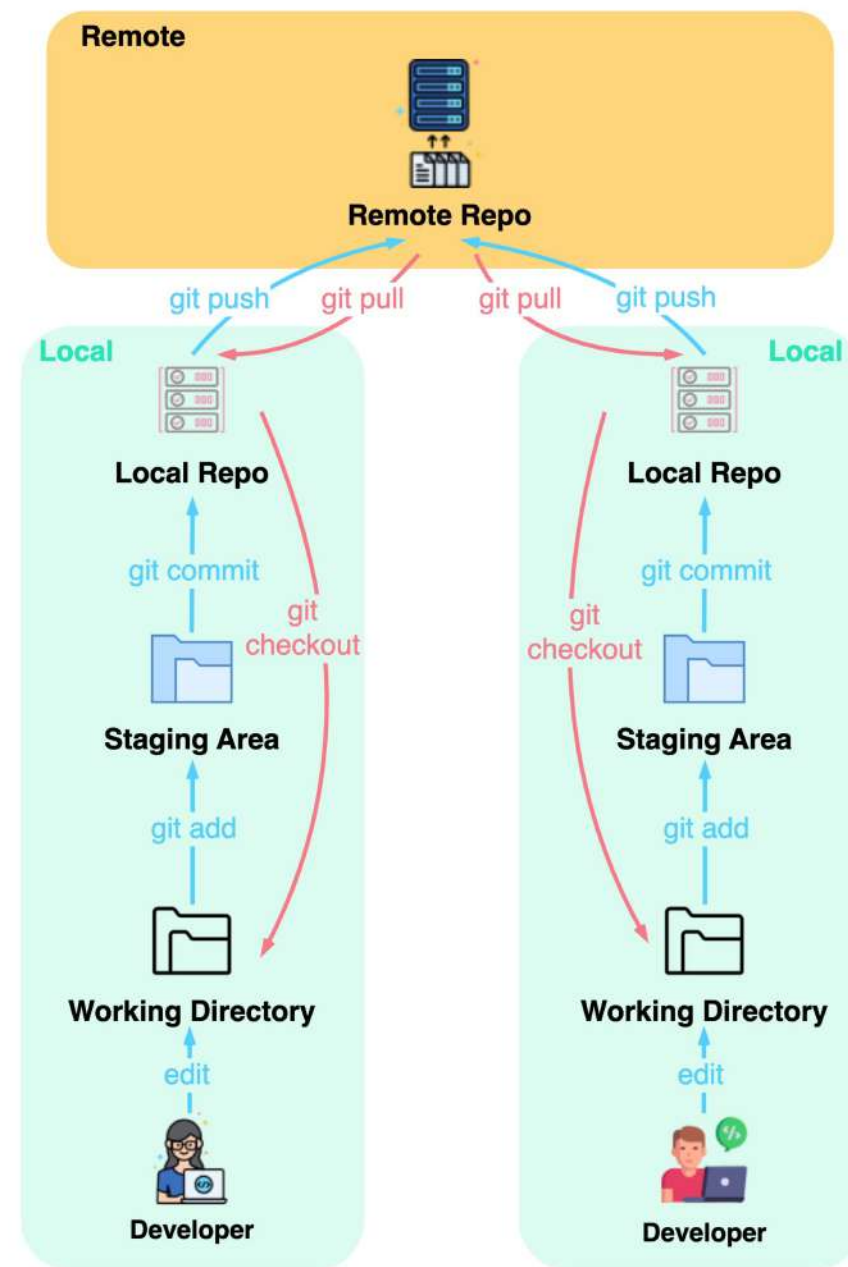
- Distributed architecture
- Integritas Data
- Branching and Merging

How Git Commands work



How does Git Work?

blog.bytebytego.com



Praktikum Git

- Install Git di Windows
- Install Git di Linux/Mac
- Konfigurasi Awal Git

Install Git di Windows

- Unduh Git
 - <https://git-scm.com/download/win>
- Install Git
 - Pilih opsi "*Use Git from the Windows Command Prompt* "
- Verifikasi Git
 - Buka command prompt (cmd)
 - Ketik "git –version" kemudian "enter "
- Integrasi Git dengan Command Prompt (CMD)
 - Buka control panel, pilih Program and Features, pilih Git for Windows
 - Klik Change, pilih "*Configure the Git Bash integration*" klik next
 - Pilih "*Use Windows Terminal instead of Command Prompt*" (opsional)
 - Klik Next, Install, Finish

Install Git di Linux

- Linux (Ubuntu)
 - `$ sudo apt install git`
 - `$ git --version`
- Linux (Fedora)
 - `$ sudo dnf install git`
 - `$ git --version`
- Linux (Suse)
 - `$ sudo zypper install git`
 - `$ git --version`
- Mac
 - Buka terminal
 - (jika belum ada homebrew) Install homebre
 - `$ /usr/bin/ruby -e "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/master/install)"`
 - Install git dengan homebrew
 - `$ brew install git`
 - `$ git --version`

Konfigurasi Awal Git

- Set nama dan email
 - `git config --global user.name "Your name"`
 - `git config --global user.email "your.email@example.com"`
- Set editor default
 - `git config --global core.editor "your_favorite_editor"`

pwd = path working directory
mkdir = make new directory
cd = change directory
git init = inisiasi git aktif di *current folder*
git status = lihat status file di *current folder*

```
adiwp@Adis-MacBook-Pro:~/belajar-git
[Apple] [Home] [~] ..... at 15:13:18
> pwd
/Users/adiwp

[Apple] [Home] [~] ..... at 15:13:32
> mkdir belajar-git

[Apple] [Home] [~] ..... at 15:13:36
> cd belajar-git

[Apple] [Home] [~/belajar-git] ..... at 15:13:42
> git init
Initialized empty Git repository in /Users/adiwp/belajar-git/.git/

[Apple] [Home] [~/belajar-git] on git main ..... at 15:13:47
> git status
On branch main

No commits yet

nothing to commit (create/copy files and use "git add" to track)

[Apple] [Home] [~/belajar-git] on git main ..... at 15:14:29
> 
```

```
vim README.md
```

```
1 Saatnya Belajar Git
2
3 Agar lebih mudah mengingat dan memahaminya, maka kita harus mempraktikkan il
mu yang sudah didapatkan.
```

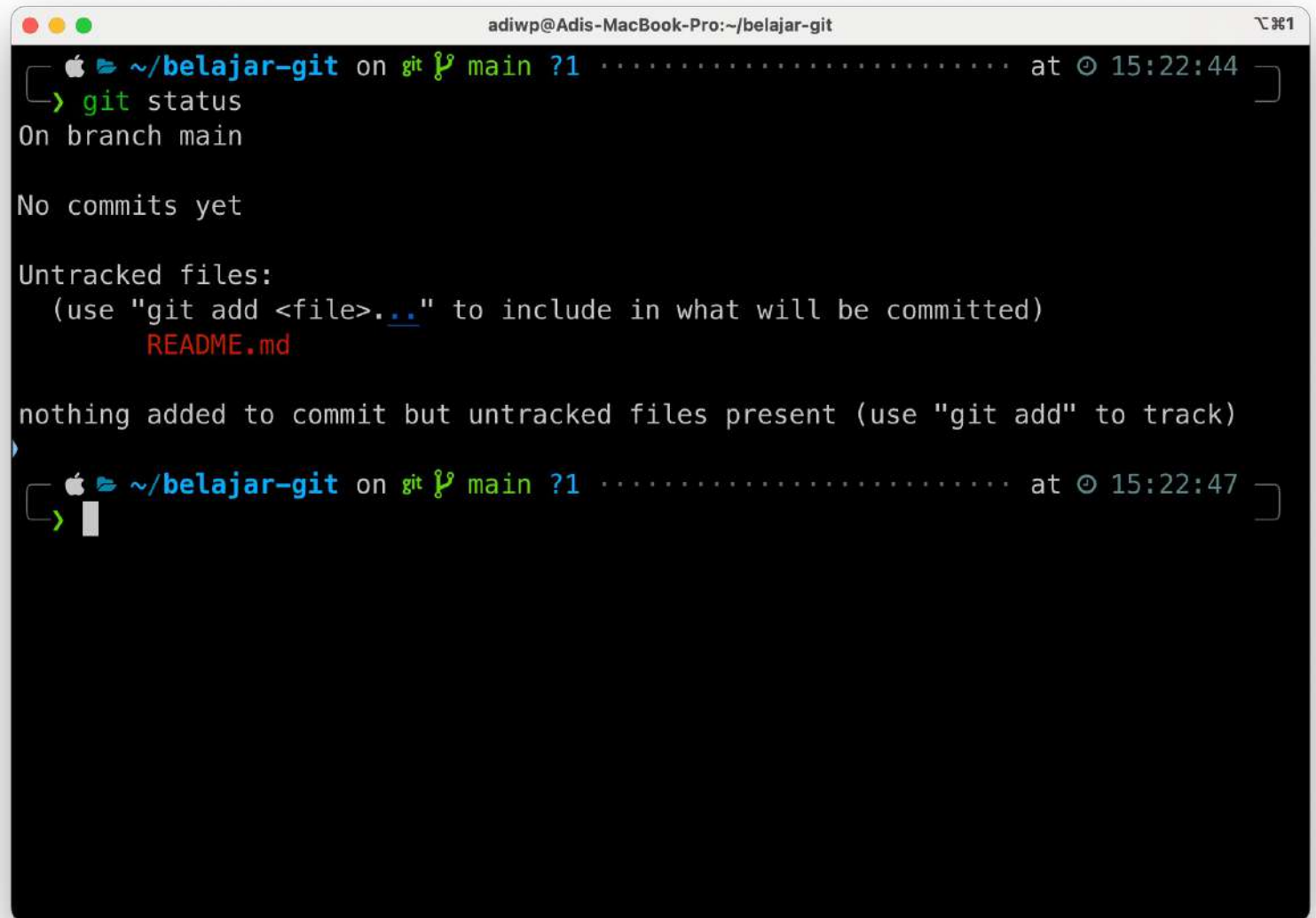
```
"README.md" 3L, 123B
```

Setelah menambah README.md
Kita lihat status belajar-git

\$ git status

Maka terlihat informasi bahwa
Di branch main ada file
Bernama README.md

Belum ditrak dan bewarna merah

A terminal window on a Mac with a dark background. The title bar shows 'adiwp@Adis-MacBook-Pro: ~/belajar-git'. The prompt is 'adiwp@Adis-MacBook-Pro: ~/belajar-git on git main ?1'. The user enters 'git status'. The output shows 'On branch main', 'No commits yet', and 'Untracked files: (use "git add <file>..." to include in what will be committed)'. Below this, 'README.md' is listed in red text. A final line says 'nothing added to commit but untracked files present (use "git add" to track)'. The prompt returns, and the user enters a space character.

```
adiwp@Adis-MacBook-Pro: ~/belajar-git
> git status
On branch main

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        README.md

nothing added to commit but untracked files present (use "git add" to track)
> 
```

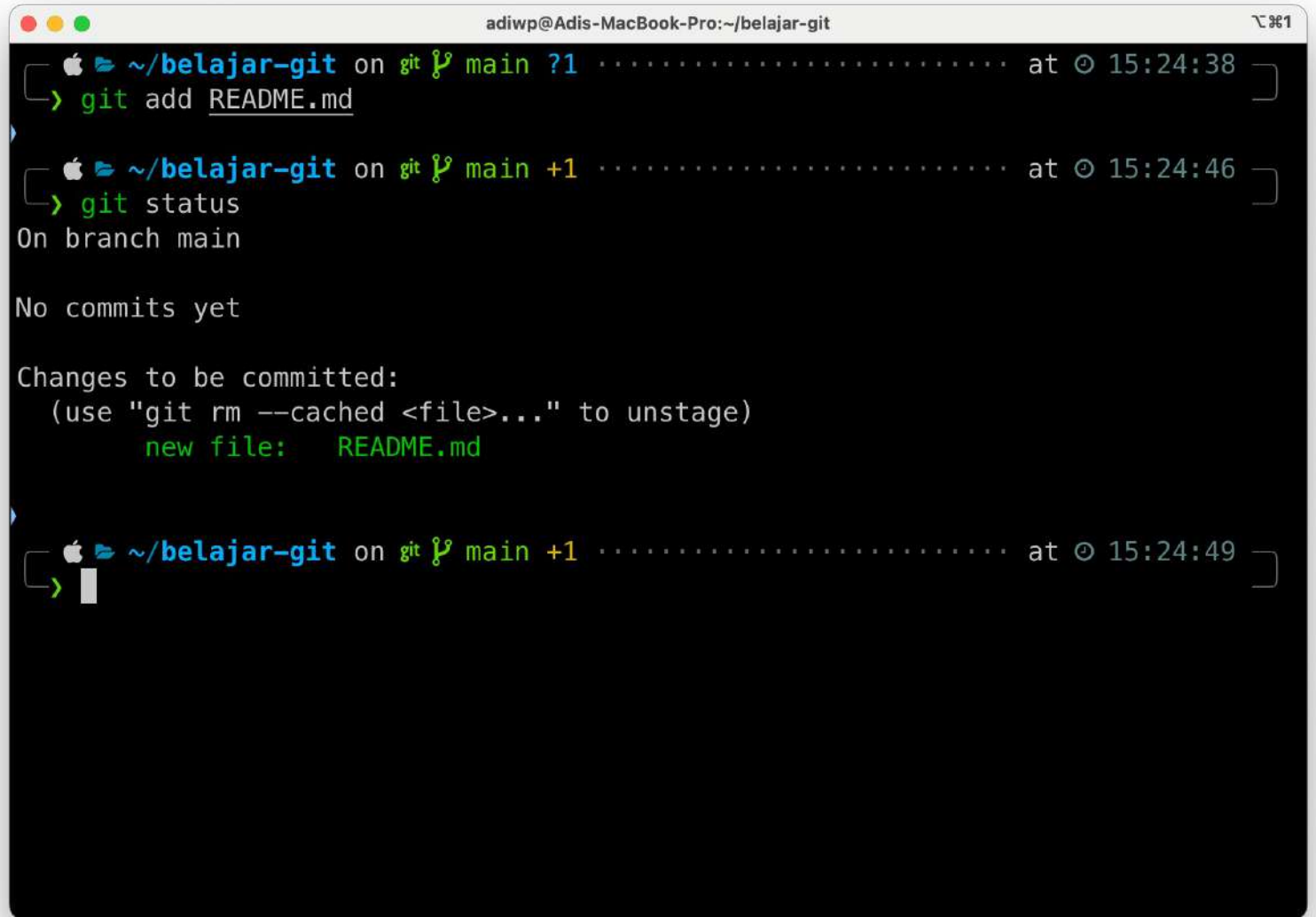
Kita tambahkan README.md ke dalam status staging, menggunakan perintah `git add nama_file`, atau `git add .` untuk semua perubahan

```
$ git add README.md
```

Setelah itu kita lihat statusnya

```
$ git status
```

README.md sudah berada di staging

A terminal window titled 'adiwp@Adis-MacBook-Pro: ~/belajar-git' with a dark background and light green text. It shows three command-line sessions. The first session shows the command 'git add README.md' being entered. The second session shows 'git status' being entered, followed by the output: 'On branch main', 'No commits yet', and 'Changes to be committed: (use "git rm --cached <file>..." to unstage) new file: README.md'. The third session shows the command 'git status' being entered again, with a cursor visible on the line following the prompt. Each session is preceded by a prompt line showing the current directory as '~/belajar-git' and the current branch as 'main'.

```
adiwp@Adis-MacBook-Pro: ~/belajar-git

❯ git add README.md

❯ git status
On branch main

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
        new file:   README.md

❯ git status
```

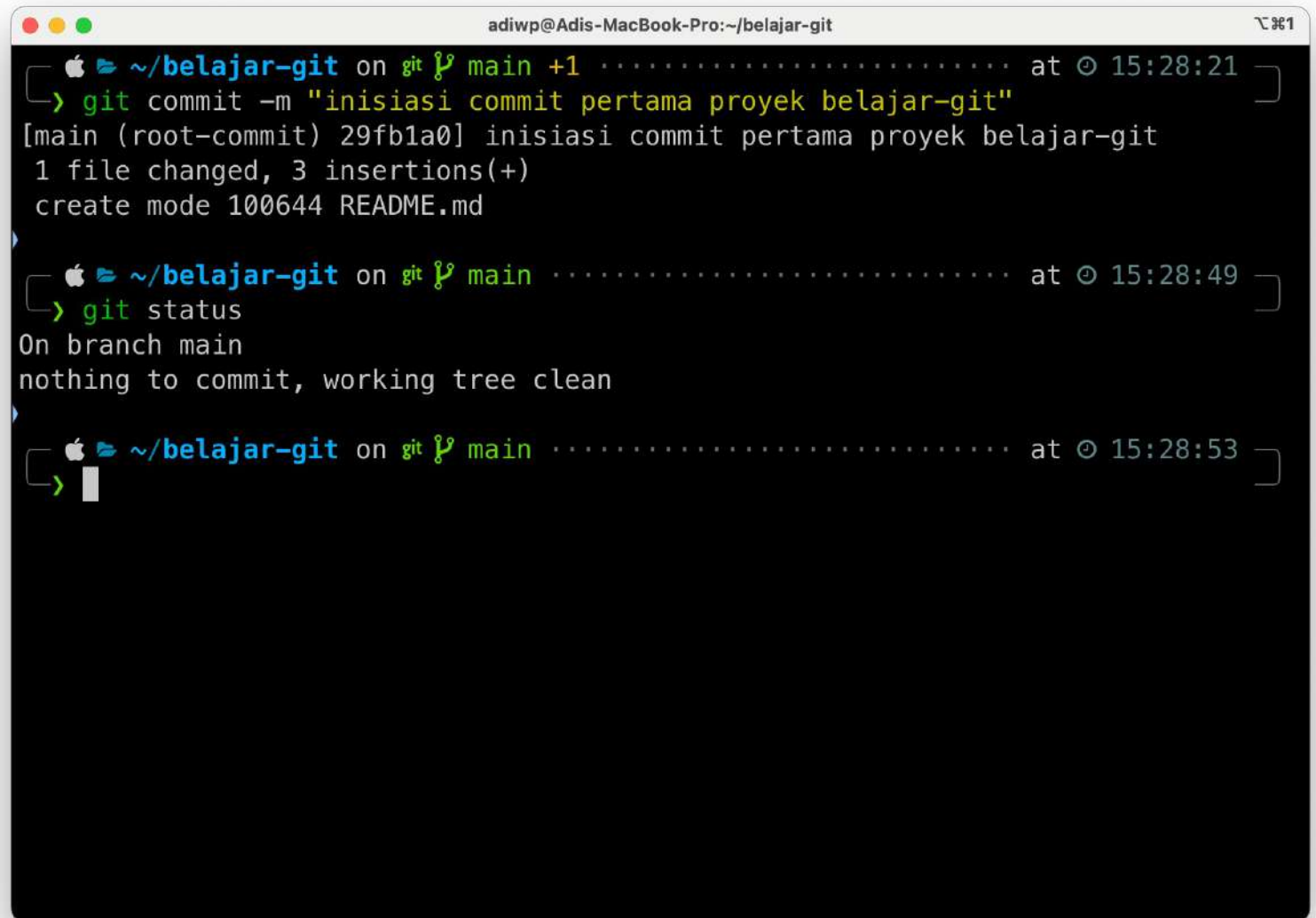
Kita pindah ke local repo dengan perintah `git commit -m "pesan"`

`$ git commit -m "pesan"`

Setelah itu kita cek statusnya

`$ git status`

terlihat bahwa pada branch main tidak ada yang perlu di-commit semuanya bersih/clean.

A terminal window titled 'adiwp@Adis-MacBook-Pro:~/belajar-git' with a macOS-style title bar (red, yellow, green buttons). The terminal shows three command sessions. The first session at 15:28:21 shows 'git commit -m "inisiasi commit pertama proyek belajar-git"' being executed, resulting in a commit hash '29fb1a0' and a message 'inisiasi commit pertama proyek belajar-git'. It shows '1 file changed, 3 insertions(+)' and 'create mode 100644 README.md'. The second session at 15:28:49 shows 'git status' being executed, resulting in 'On branch main' and 'nothing to commit, working tree clean'. The third session at 15:28:53 shows the start of 'git status' with a cursor on the next line.

```
adiwp@Adis-MacBook-Pro:~/belajar-git

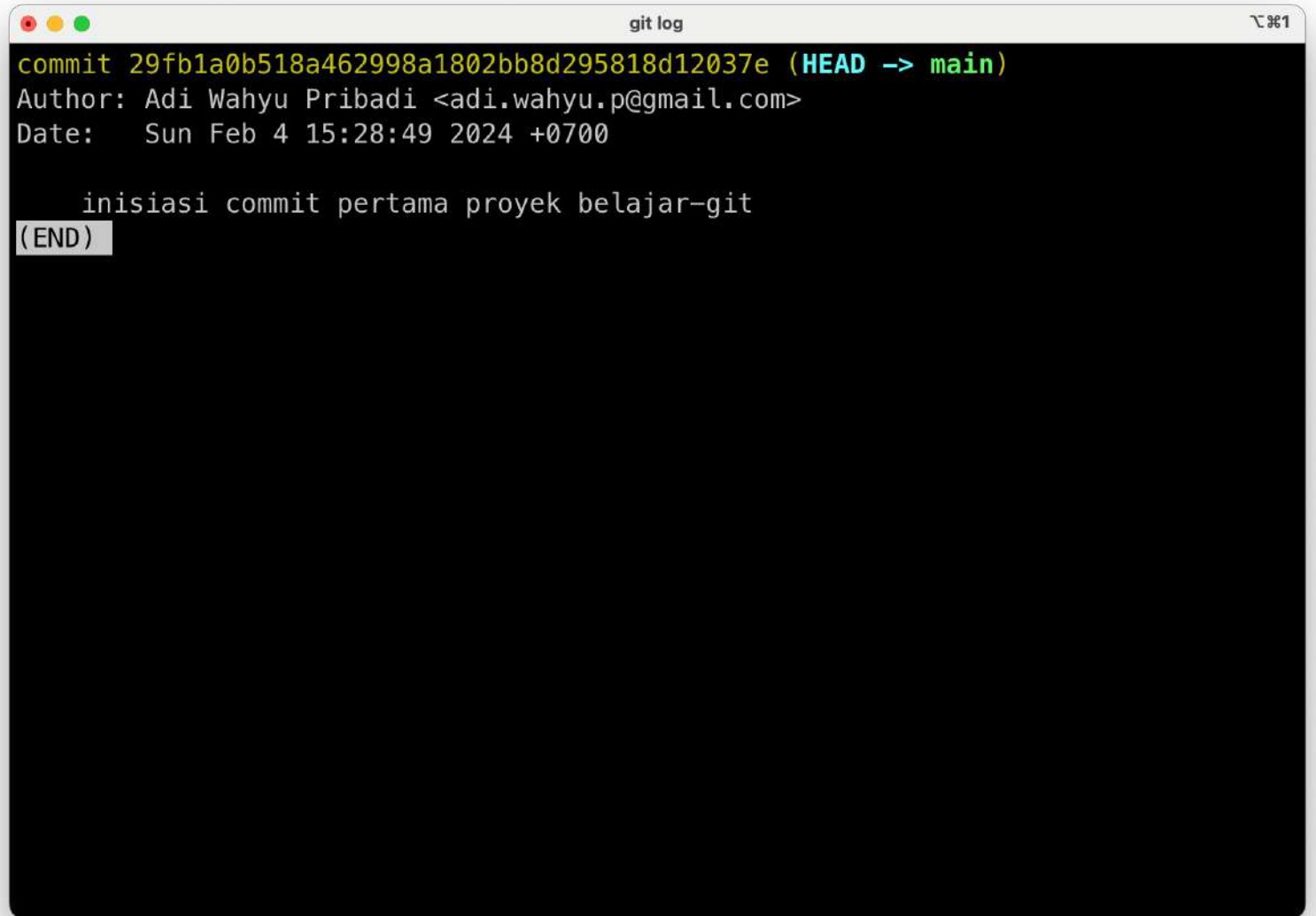
[❯] Apple ~/belajar-git on git main +1 ..... at 15:28:21 [❯]
> git commit -m "inisiasi commit pertama proyek belajar-git"
[main (root-commit) 29fb1a0] inisiasi commit pertama proyek belajar-git
1 file changed, 3 insertions(+)
create mode 100644 README.md

[❯] Apple ~/belajar-git on git main ..... at 15:28:49 [❯]
> git status
On branch main
nothing to commit, working tree clean

[❯] Apple ~/belajar-git on git main ..... at 15:28:53 [❯]
> 
```

lihat riwayat commit repository git

\$ git log

A terminal window titled "git log" with a standard macOS-style title bar (red, yellow, green buttons). The terminal output shows a single commit with a yellow hash, a green branch name in parentheses, author information, a date, and a commit message. The text "(END)" is highlighted with a light gray background.

```
commit 29fb1a0b518a462998a1802bb8d295818d12037e (HEAD -> main)
Author: Adi Wahyu Pribadi <adi.wahyu.p@gmail.com>
Date:   Sun Feb 4 15:28:49 2024 +0700

    inisiasi commit pertama proyek belajar-git
(END)
```

Membuat file baru Halo.java
di dalam folder belajar-git

```
$ vim HelloWorld.java
```

```
vim Halo.java
```

```
1 public class HelloWorld {  
2  
3     public static void main(String[] args) {  
4         System.out.println("Halo!");  
5     }  
6 }  
7
```

```
:wq
```

setelah disimpan, lalu lihat statusnya

```
$ git status
```

lalu tambahkan ke staging

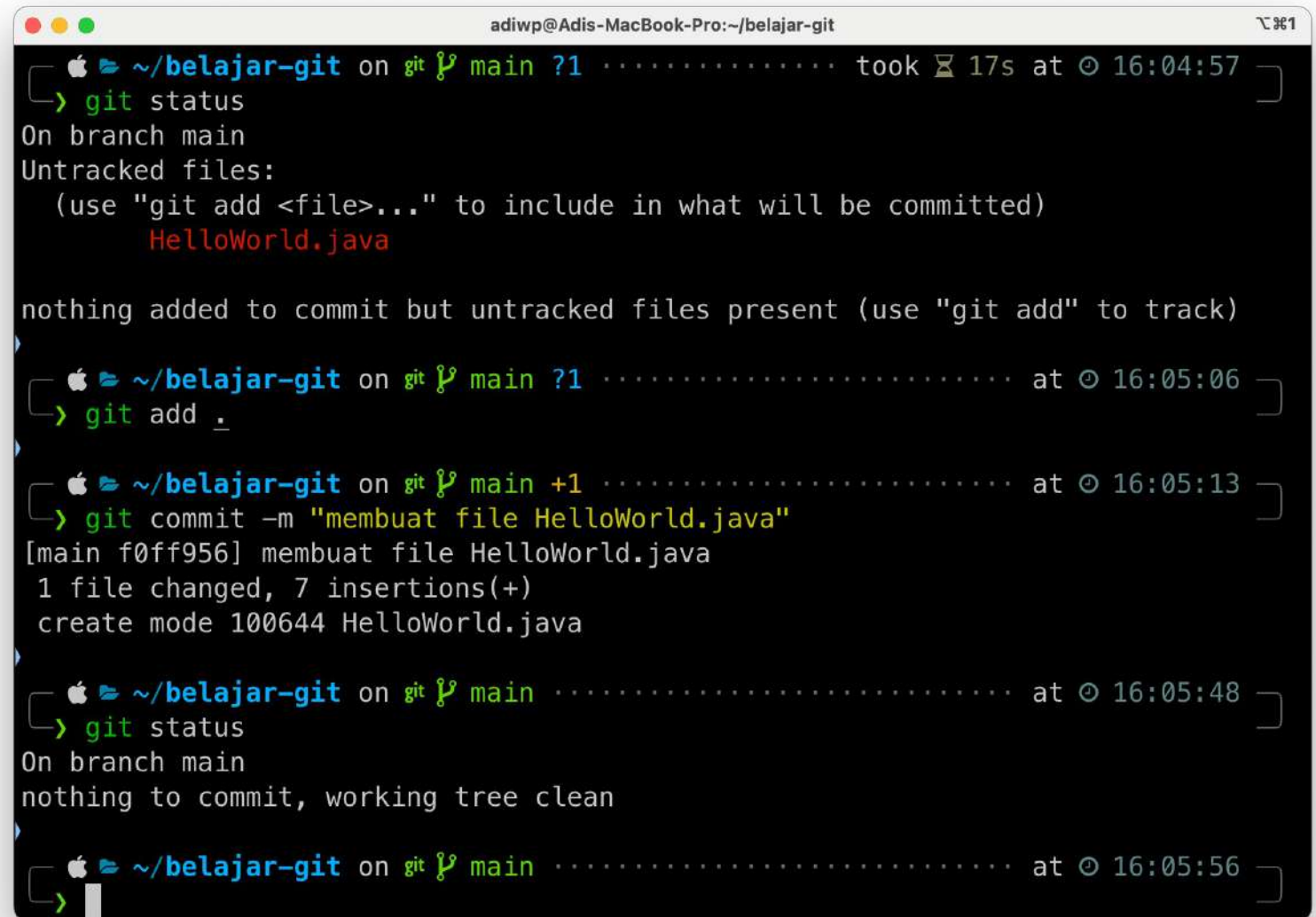
```
$ git add .
```

lalu commit file java tersebut

```
$ git commit -m "pesan baru"
```

lihat statusnya lagi

```
$ git status
```



```
adiwp@Adis-MacBook-Pro:~/belajar-git
[❯] Apple ~/belajar-git on git main ?1 ..... took 17s at 16:04:57
[❯] > git status
On branch main
Untracked files:
  (use "git add <file>..." to include in what will be committed)
        HelloWorld.java

nothing added to commit but untracked files present (use "git add" to track)
[❯] Apple ~/belajar-git on git main ?1 ..... at 16:05:06
[❯] > git add .
[❯] Apple ~/belajar-git on git main +1 ..... at 16:05:13
[❯] > git commit -m "membuat file HelloWorld.java"
[main f0ff956] membuat file HelloWorld.java
1 file changed, 7 insertions(+)
create mode 100644 HelloWorld.java
[❯] Apple ~/belajar-git on git main ..... at 16:05:48
[❯] > git status
On branch main
nothing to commit, working tree clean
[❯] Apple ~/belajar-git on git main ..... at 16:05:56
[❯] >
```


lihat riwayat commit repository git
terbaru

\$ git log

```
git log
commit f0ff956af004fa880cf3dffc4ca203b0b855a736 (HEAD -> main)
Author: Adi Wahyu Pribadi <adi.wahyu.p@gmail.com>
Date:   Sun Feb 4 16:05:48 2024 +0700

    membuat file HelloWorld.java

commit 29fb1a0b518a462998a1802bb8d295818d12037e
Author: Adi Wahyu Pribadi <adi.wahyu.p@gmail.com>
Date:   Sun Feb 4 15:28:49 2024 +0700

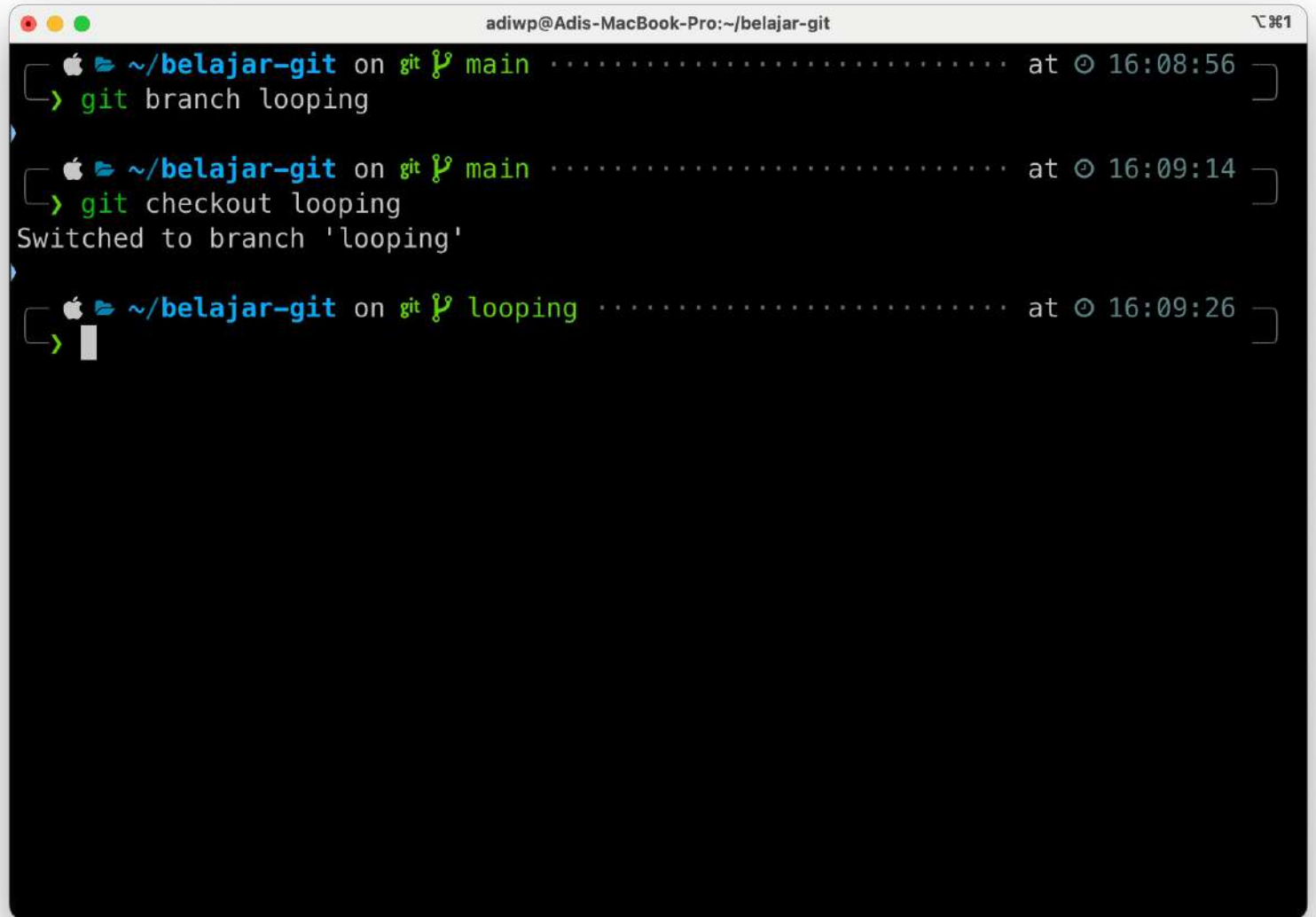
    inisiasi commit pertama proyek belajar-git
(END)
```

membuat cabang baru/new branch
dengan perintah

\$ git branch looping

di sini kita membuat cabang untuk
fitur looping di java

perhatikan setelah pindah cabang
ke looping, ada keterangan
Switched to branch 'looping'

A screenshot of a macOS terminal window titled 'adiwp@Adis-MacBook-Pro:~/belajar-git'. The terminal shows three sequential git commands and their outputs. The first command is 'git branch looping' at 16:08:56. The second command is 'git checkout looping' at 16:09:14, which outputs 'Switched to branch 'looping''. The third command is 'git' at 16:09:26, which is partially visible. The terminal uses a dark theme with green and blue text for the prompts and status indicators.

```
adiwp@Adis-MacBook-Pro:~/belajar-git
❯ git branch looping
❯ git checkout looping
Switched to branch 'looping'
❯ git
```

ketika kita lihat isi dari direktori belajar-git ada file:

HelloWorld.java

README.java

Kemudian mari kita buat file

HelloWorldLoop.java

\$ vim HelloWorldLoop.java

```
adiwp@Adis-MacBook-Pro:~/belajar-git
[❯] Apple ~/belajar-git on git 🍌 main ..... at 16:08:56
> git branch looping

[❯] Apple ~/belajar-git on git 🍌 main ..... at 16:09:14
> git checkout looping
Switched to branch 'looping'

[❯] Apple ~/belajar-git on git 🍌 looping ..... at 16:09:26
> ls
HelloWorld.java README.md

[❯] Apple ~/belajar-git on git 🍌 looping ..... at 16:12:04
> ls -al
total 16
drwxr-xr-x  5 adiwp  staff   160 Feb  4 16:04 .
drwxr-x---+ 65 adiwp  staff 2080 Feb  4 16:12 ..
drwxr-xr-x 12 adiwp  staff  384 Feb  4 16:09 .git
-rw-r--r--  1 adiwp  staff  118 Feb  4 16:04 HelloWorld.java
-rw-r--r--  1 adiwp  staff  123 Feb  4 15:22 README.md

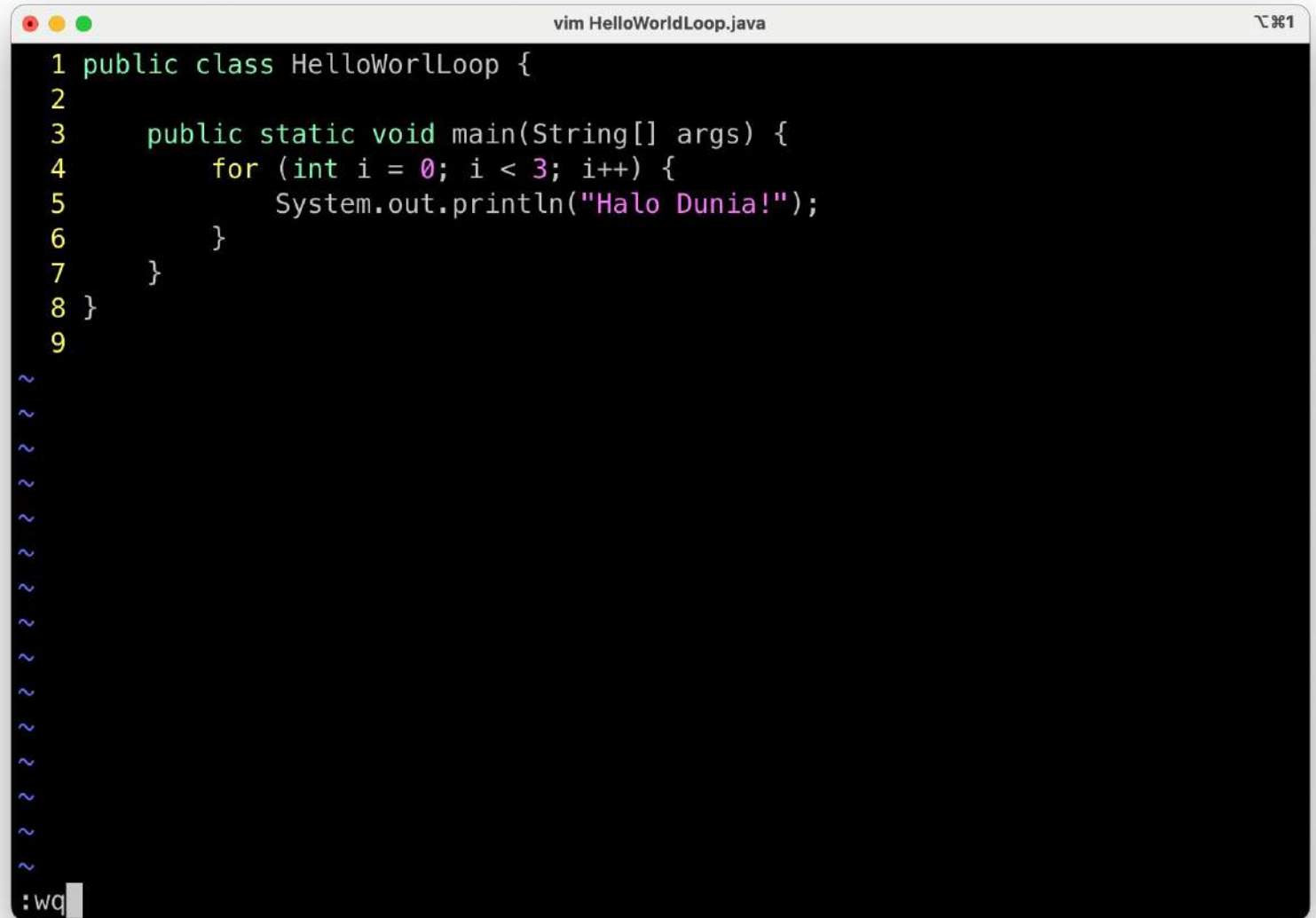
[❯] Apple ~/belajar-git on git 🍌 looping ..... at 16:12:12
> 
```

Isi dari HelloWorldLoop.java

jangan lupa ketik huruf i
untuk memulai edit di VIM

ketikkan source code

lalu ESC dan ketikkan
:wq
untuk simpan



```
1 public class HelloWorldLoop {
2
3     public static void main(String[] args) {
4         for (int i = 0; i < 3; i++) {
5             System.out.println("Halo Dunia!");
6         }
7     }
8 }
9
```

:wq

setelah simpan, jangan lupa

\$ git status

\$ git add .

\$ git commit -m "pesan"

```
adiwp@Adis-MacBook-Pro:~/belajar-git
❯ git status
On branch looping
Untracked files:
  (use "git add <file>..." to include in what will be committed)
        HelloWorldLoop.java

nothing added to commit but untracked files present (use "git add" to track)
❯ git add .
❯ git commit -m "looping HelloWorld"
[looping 6805005] looping HelloWorld
1 file changed, 9 insertions(+)
create mode 100644 HelloWorldLoop.java
❯ git status
On branch looping
nothing to commit, working tree clean
```

Lihat perbandingan isi kedua direktori
dari branch main dan looping

di 'looping'

```
$ ls -al
```

pindah branch ke main

```
$ git branch main
```

```
$ ls -al
```

```
adiwp@Adis-MacBook-Pro:~/belajar-git
❯ ls -al
total 24
drwxr-xr-x  6 adiwp  staff   192 Feb  4 16:16 .
drwxr-x---+ 65 adiwp  staff 2080 Feb  4 16:18 ..
drwxr-xr-x 12 adiwp  staff  384 Feb  4 16:17 .git
-rw-r--r--  1 adiwp  staff  118 Feb  4 16:04 HelloWorld.java
-rw-r--r--  1 adiwp  staff  179 Feb  4 16:16 HelloWorldLoop.java
-rw-r--r--  1 adiwp  staff  123 Feb  4 15:22 README.md

❯ git checkout main
Switched to branch 'main'

❯ ls -al
total 16
drwxr-xr-x  5 adiwp  staff   160 Feb  4 16:18 .
drwxr-x---+ 65 adiwp  staff 2080 Feb  4 16:18 ..
drwxr-xr-x 12 adiwp  staff  384 Feb  4 16:18 .git
-rw-r--r--  1 adiwp  staff  118 Feb  4 16:04 HelloWorld.java
-rw-r--r--  1 adiwp  staff  123 Feb  4 15:22 README.md
```

Sekarang pindah lagi ke branch looping

\$ git checkout looping

Lalu cek riwayat commit

\$ git log

terlihat bahwa kedua cabang memiliki log yang berbeda.

```
git log
commit 6805005b22343ecf7fe98a6913eb9ed44f07589f (HEAD -> looping)
Author: Adi Wahyu Pribadi <adi.wahyu.p@gmail.com>
Date: Sun Feb 4 16:17:00 2024 +0700

    looping HelloWorld

commit f0ff956af004fa880cf3dffc4ca203b0b855a736 (main)
Author: Adi Wahyu Pribadi <adi.wahyu.p@gmail.com>
Date: Sun Feb 4 16:05:48 2024 +0700

    membuat file HelloWorld.java

commit 29fb1a0b518a462998a1802bb8d295818d12037e
Author: Adi Wahyu Pribadi <adi.wahyu.p@gmail.com>
Date: Sun Feb 4 15:28:49 2024 +0700

    inisiasi commit pertama proyek belajar-git
(END)
```


sekarang kita akan merge atau gabung
branch looping ke branch main

pindah dulu ke main branch

\$ git checkout main

lalu merge looping ke main

\$ git merge looping

```
adiwp@Adis-MacBook-Pro:~/belajar-git
❯ git checkout main
Switched to branch 'main'

❯ git merge looping
Updating f0ff956..6805005
Fast-forward
 HelloWorldLoop.java | 9 ++++++++
 1 file changed, 9 insertions(+)
 create mode 100644 HelloWorldLoop.java

❯
```


Sekarang kita cek log/riwayat commit di branch main

\$ git log

Terlihat bahwa commit terakhir (HEAD -> main, looping)

sudah merging branch main dan looping

```
git log
commit 6805005b22343ecf7fe98a6913eb9ed44f07589f (HEAD -> main, looping)
Author: Adi Wahyu Pribadi <adi.wahyu.p@gmail.com>
Date: Sun Feb 4 16:17:00 2024 +0700

    looping HelloWorld

commit f0ff956af004fa880cf3dffc4ca203b0b855a736
Author: Adi Wahyu Pribadi <adi.wahyu.p@gmail.com>
Date: Sun Feb 4 16:05:48 2024 +0700

    membuat file HelloWorld.java

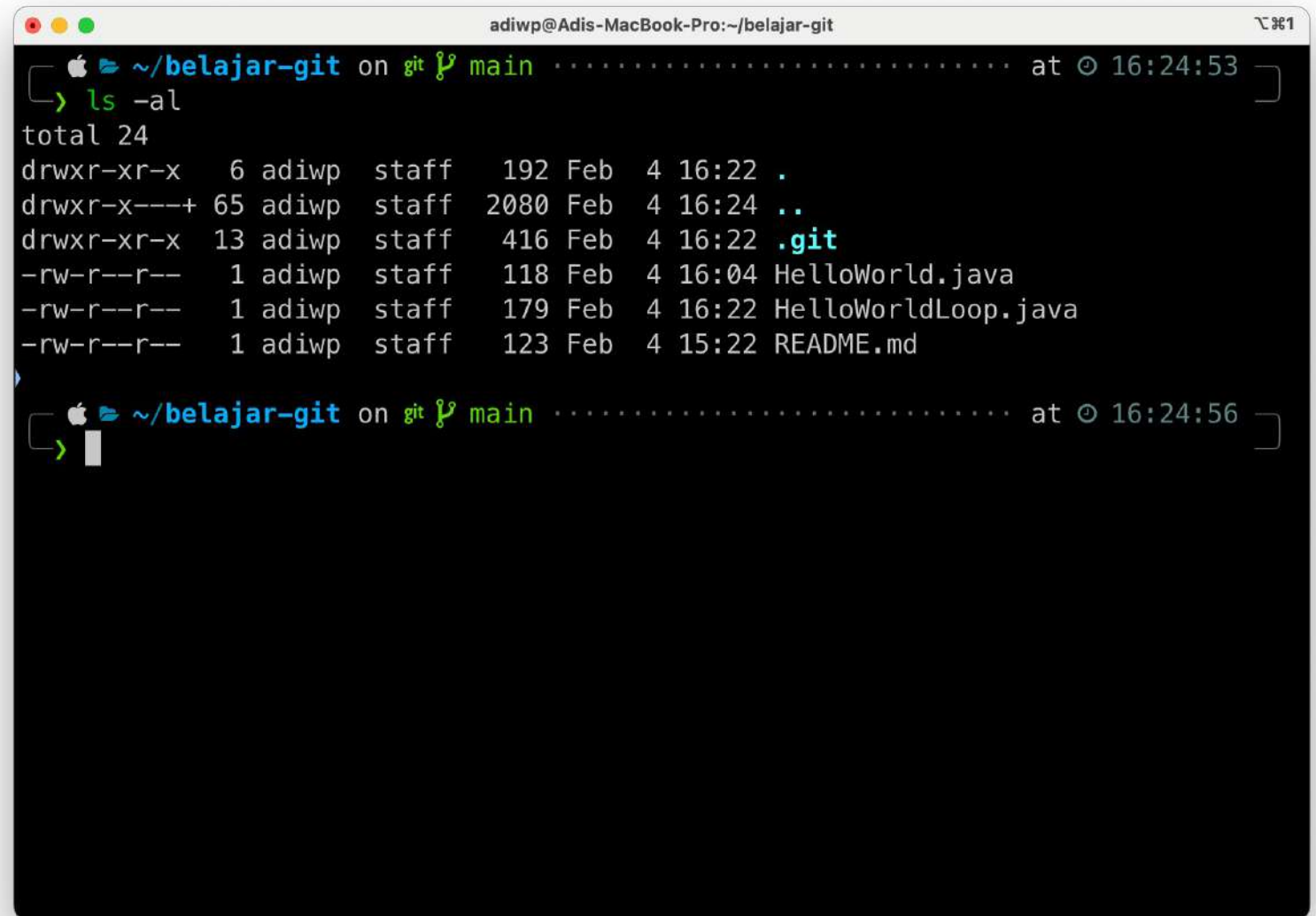
commit 29fb1a0b518a462998a1802bb8d295818d12037e
Author: Adi Wahyu Pribadi <adi.wahyu.p@gmail.com>
Date: Sun Feb 4 15:28:49 2024 +0700

    inisiasi commit pertama proyek belajar-git
(END)
```

Sekarang kita lihat isi dari folder belajar-git di branch main

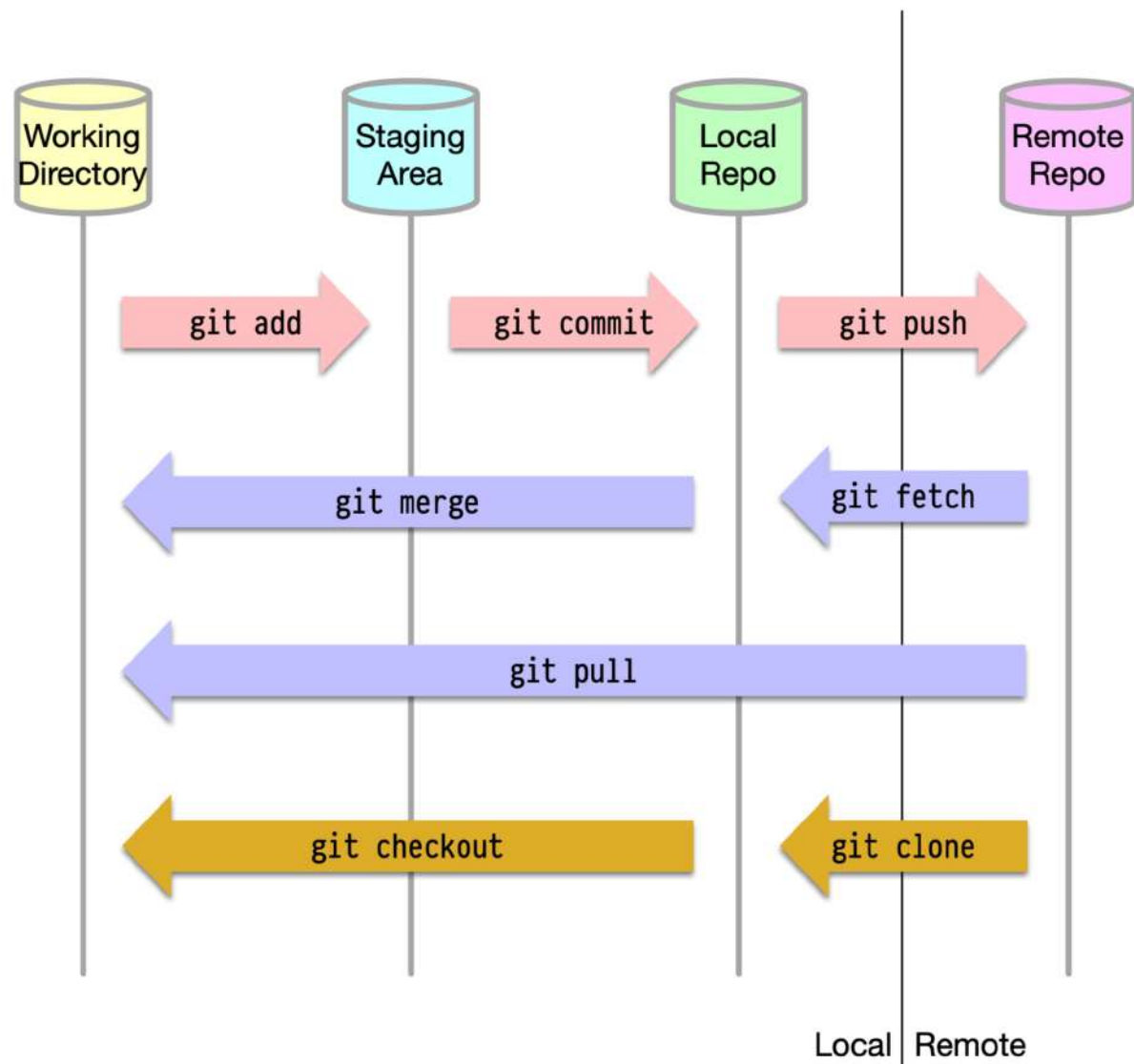
```
$ ls -al
```

terlihat ada file
HelloWorld.java
HelloWorldLooping.java
README.md



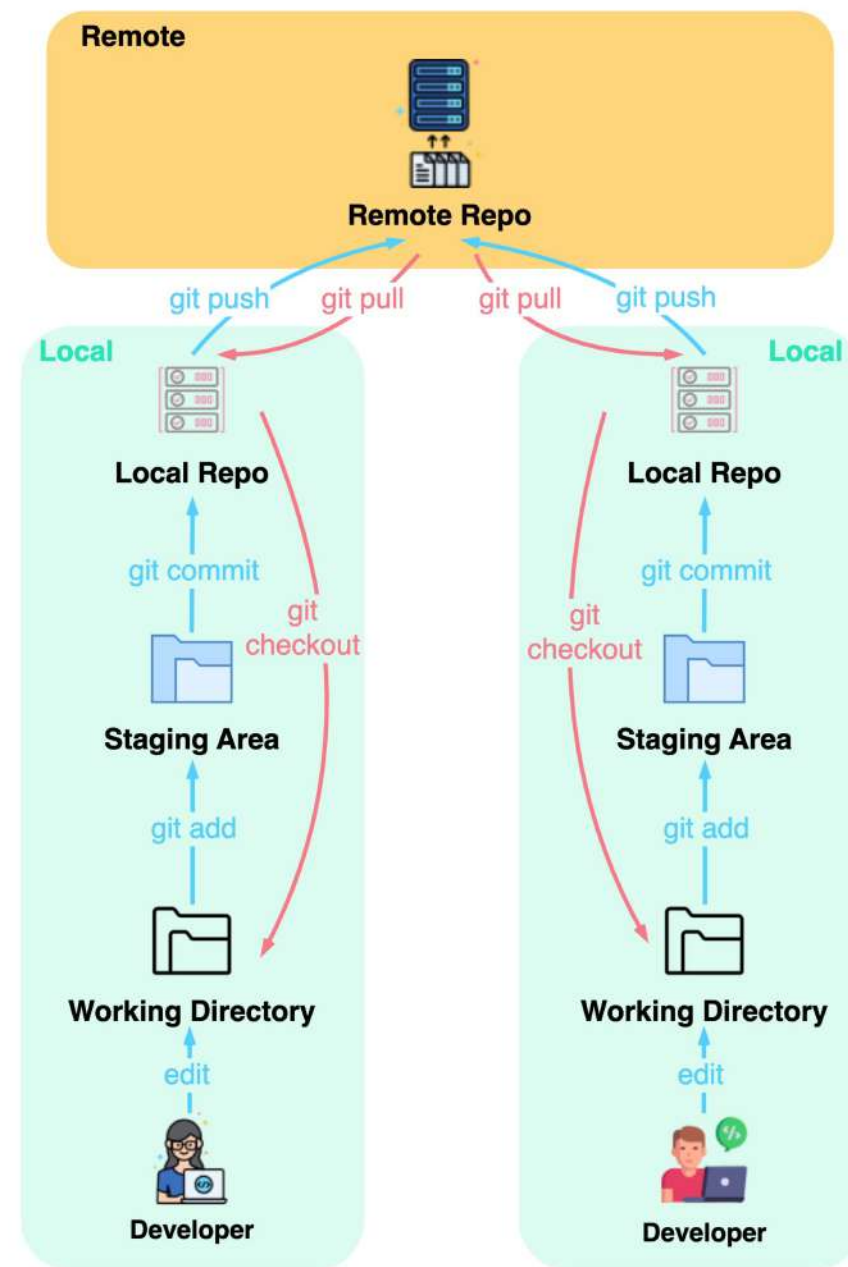
```
adiwp@Adis-MacBook-Pro:~/belajar-git  
[Apple] ~/belajar-git on git main ..... at 16:24:53  
> ls -al  
total 24  
drwxr-xr-x  6 adiwp  staff   192 Feb  4 16:22 .  
drwxr-x---+ 65 adiwp  staff 2080 Feb  4 16:24 ..  
drwxr-xr-x 13 adiwp  staff  416 Feb  4 16:22 .git  
-rw-r--r--  1 adiwp  staff  118 Feb  4 16:04 HelloWorld.java  
-rw-r--r--  1 adiwp  staff  179 Feb  4 16:22 HelloWorldLoop.java  
-rw-r--r--  1 adiwp  staff  123 Feb  4 15:22 README.md  
[Apple] ~/belajar-git on git main ..... at 16:24:56  
> █
```

How Git Commands work



How does Git Work?

blog.bytebytego.com



Ada Pertanyaan?

Tugas

