



Terminal dan IDE

Silver - Chapter 3 - Topic 1

**Selamat datang di Chapter 3 Topik 1 online
course Full Stack Web dari Binar Academy!**





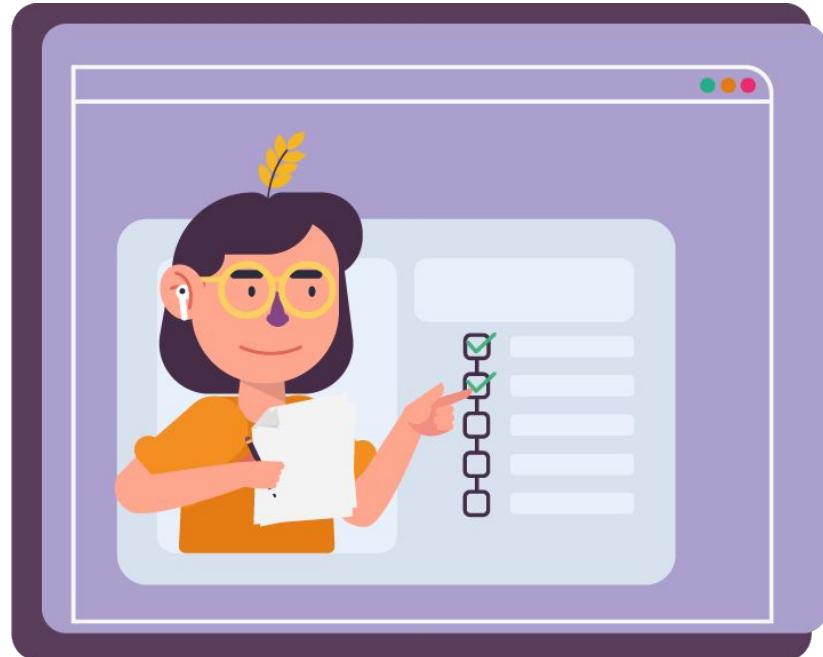
Setelah kamu selesai dijelaskan tentang dasar-dasar membuat halaman web dan juga bahasa pemrograman Javascript, **Chapter 3 ini bakalan ngajak kamu buat belajar cara menjalankan kodingan dan cara berkolaborasi dengan developer lain dalam mengembangkan sebuah sistem.** Mulai dari penjelasan terkait terminal, IDE, GIT, membuat web layout dan responsive design.

Pada topik pertama ini, kita bakal bahas tentang **apa itu Terminal dan IDE.** Cus langsung aja~



Detailnya, kita bakal bahas hal-hal berikut ini:

- Mengenal konsep dan fungsi Terminal
- Mengenal konsep dan fungsi IDE
- Relevansi Terminal dan IDE pada Fullstack Web Developer



Terminal?

Hhmm... bukan terminal yang banyak bus itu ya yang akan kita bahas disini.

Jadi apaan tuh? Aplikasi yang dipake hacker-hacker di film-film hollywood itu kah?

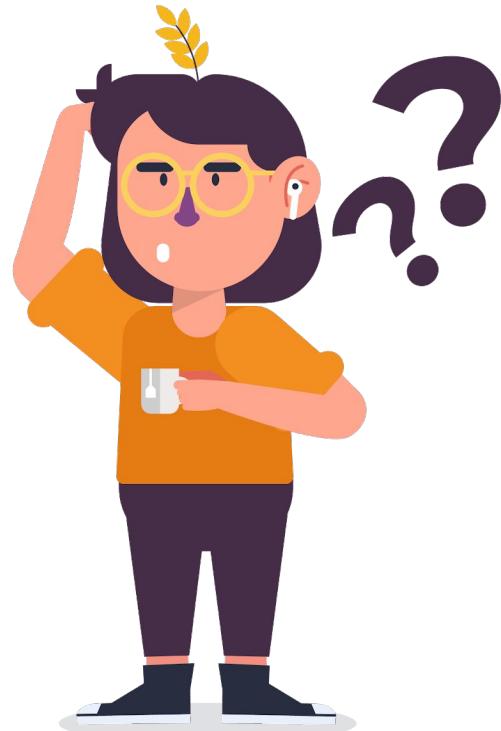
Cekidooot kita langsung aja ya ~





Kita mulai dari Terminal dulu ya~

Yes sobs, jadi terminal itu emang aplikasi yang biasa dipake hacker-hacker di film-film hollywood. Bedanya, kita sebagai full-stack web developer juga butuh aplikasi terminal, bukan buat mangkal bus atau hacking yaa, tapi buat ngembangin website.



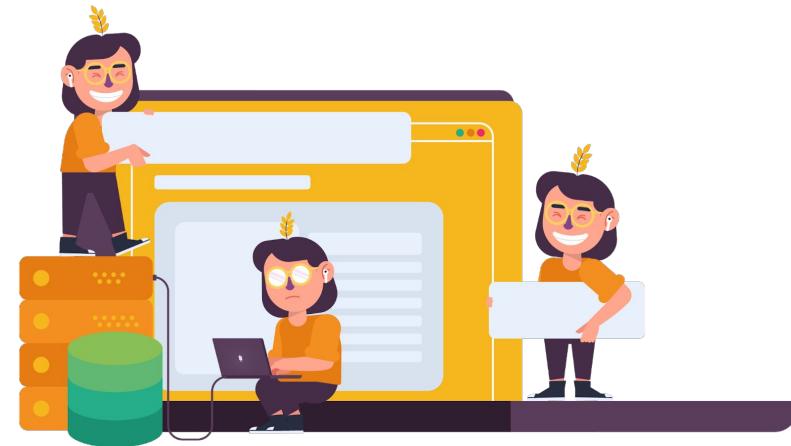


Ngembangin website itu bisa pakai terminal lho ternyata.

Bener sobs, ketika kita sedang ngembangin sebuah website, kita pastinya butuh ngejalani website tersebut di laptop atau komputer kita masing-masing untuk :

- Memastikan apakah kode kita berjalan dengan baik
- Memastikan tampilan website kita sesuai dengan desainnya

Nah, salah satu cara paling efektif untuk ngejalani website kita adalah dengan menggunakan terminal. Dengan menggunakan terminal, kita **dapat mengetikkan perintah-perintah canggih buat nyuruh-nyuruh laptop atau komputer kita tanpa perlu GUI**.

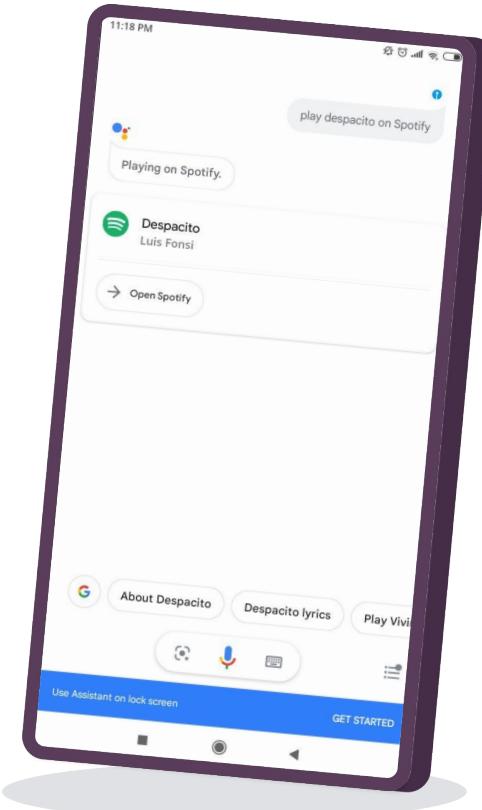




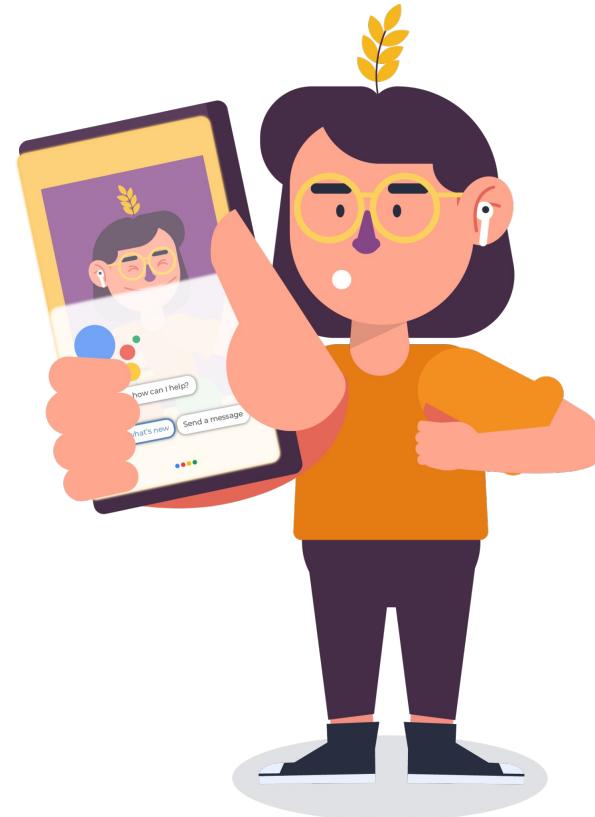
Kamu pernah pake Google Assistant, Siri, atau Alexa kan?

Google Assistant, Siri, atau Alexa ini cara kerjanya mirip terminal. Ketika kita mengetikkan atau menyebutkan sesuatu ke mereka, mereka akan melakukan sesuai perintah kita. Contohnya kaya gambar di samping, disitu kita minta Google Assistant untuk nyetel lagu Despacito di Spotify.

Nah, kalo kalian perhatikan, disitu kita memasukkan input berupa teks, yaitu **play** despacito on Spotify, yang mana Google Assistant langsung **menginterpretasikan perintah tersebut dan menjalankannya dengan membuka aplikasi Spotify dan memainkan lagu Despacito**.



Nah cara kerja Terminal mirip dengan Google Assistant yang tadi sudah kita bahas. Bedanya, kalau Google Assistant bisa terima perintah dalam bentuk suara, kalau **terminal hanya menerima perintah dalam bentuk teks saja.**





Terus, contoh aplikasi terminal itu ada apa aja sih? 🤔

Buanyak hehehe, bahkan ada yang spesifik untuk OS yang kalian pakai.

- Kalo kalian pake Windows, maka ada aplikasi terminal emulator yang bernama **Powershell**.
- Kalo kalian pake MacOS, maka ada aplikasi terminal emulator yang bernama **iTerm**
- Kalo kalian pake GNU/Linux, maka ada aplikasi terminal emulator yang bawaan biasanya **xterm**.





Kan tadi disebut terminal ada banyak, cara pakenya beda ga tuh, terus kita harus belajar yang mana?

Nah, terminal itu ada emang banyak jenisnya, tapi cara pakenya tuh sebenarnya sama aja. Karena sebenarnya **yang bakalan kita ketik di dalam sebuah terminal itu adalah sebuah teks yang kita sebut sebagai shell**. Shell ini sebenarnya ada banyak, cuma kita bisa pisahin jadi 2 jenis aja, yaitu:

- Unix-like shell (zsh, dan bash) (Linux, MacOS, WSL)
- Non Unix-like shell (Powershell) (Windows)





Kedua jenis shell tadi dibuat dengan prinsip yang sama dan cara pakainya juga mirip. Di topik ini kita bakalan pake dan bahas jenis Unix-like shell aja, karena mayoritas server yang banyak digunakan berbasis Linux.



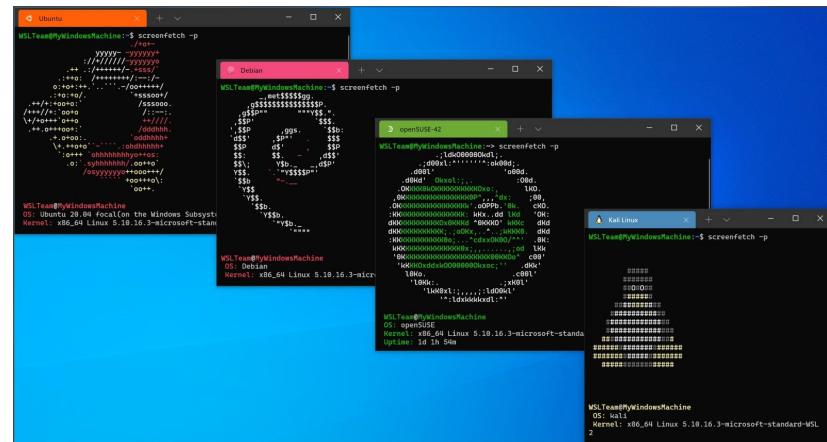
Kalo aku pake windows, terus gimana dong?

Bill Gates sekarang lagi deket-deketnya nih sama Linux, jadi sekarang di Windows sudah ada fitur namanya WSL (Windows Sub System Linux). Yang mana dengan menggunakan WSL, kita bisa pake Terminal Linux di Windows. Sehingga kamu ga perlu ganti OS untuk pake Terminal yang ada di Linux dan MacOS.

Untuk instalasi WSL, bisa kamu liat di link berikut:

[Video: WSL Installation](#)

Sangat disarankan untuk instal distribusi Ubuntu





Setelah instalasi, kita coba aplikasi terminal emulator nya yuk!

- Untuk Windows User, silahkan buka aplikasi bernama **Ubuntu** setelah kalian selesai melakukan instalasi WSL di laptop atau komputer kalian.
- Untuk MacOS user, silahkan buka aplikasi bernama iTerm.
- Dan untuk GNU/Linux user, kamu pasti tau lah harus buka apa hehe, biasanya bisa dibuka pake shortcut **CTRL + SHIFT + ENTER**.



Kita coba langkah pertama untuk tulis perintah di terminal yuk~

Kita coba menulis perintah pertama yang berguna untuk memunculkan suatu teks di dalam terminal. Kamu bisa coba tulis perintah ini di dalam terminal dan tekan **enter**.

```
echo "Hello World"
```

Output yang akan muncul di dalam terminal nanti akan terlihat seperti gambar disamping.



```
λ ~/ echo "Hello World"
Hello World
λ ~/ |
```

Nah, dalam menulis di dalam terminal, kita akan sering bersinggungan dengan yang namanya **perintah dan argumen**.

Benda apa ya itu? Letsgow kita langsung pelajari ~



Perintah dan Argumen

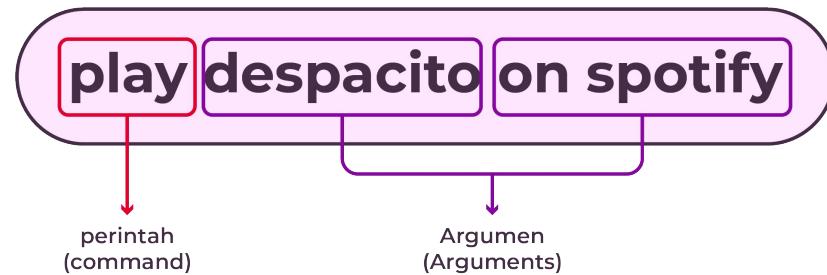
Kita balik lagi ke contoh perintah play despacito on Spotify tadi. Perintah yang kita tulis dalam contoh tadi bisa kita jabarkan menjadi dua, yaitu perintah dan argumen.

- **Perintah** ini biasanya adalah sebuah aktivitas yang ada di dalam komputer kita.
- **Argumen** adalah data yang kita berikan kepada perintah atau aplikasi tersebut.

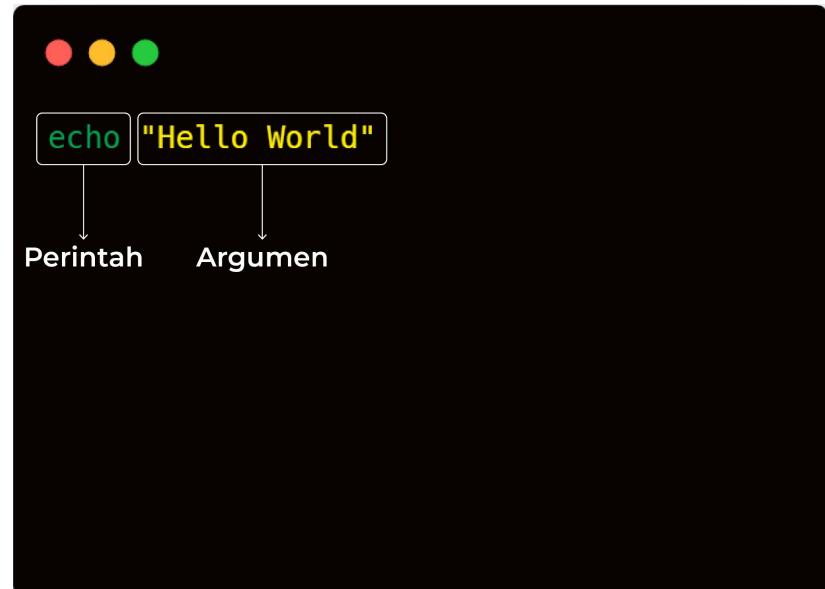


Coba kita balik lagi ke contoh perintah play despacito on Spotify tadi~

Kalo kita korelasikan lagi dengan analogi despacito yang menjadi **perintah adalah play**, yang mana ini pastinya di dalam komputer kita ada sebuah program bernama play. **Dan argumen dari perintah tersebut adalah despacito on Spotify**, yang mana data ini akan diolah oleh perintah tersebut.



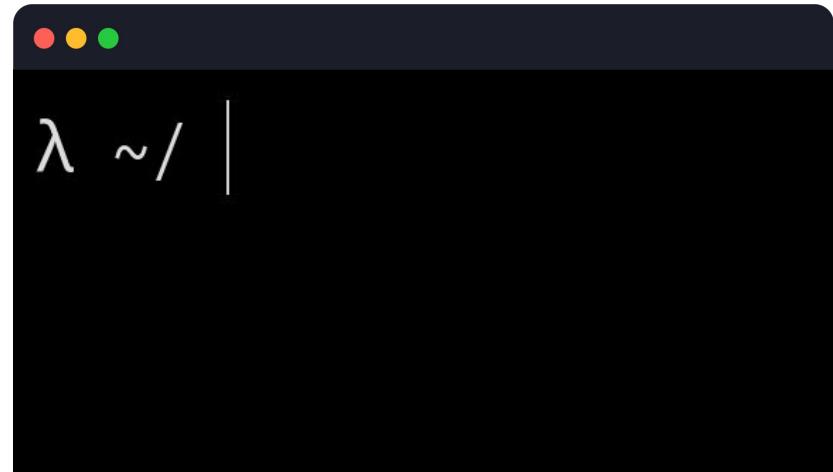
Kalau kita ambil contoh echo "Hello World", sebenarnya kita sedang memanggil perintah atau aplikasi yang bernama **echo**. Dengan argumen, **Hello World**. Yang mana argumennya akan menjadi output di dalam terminal.



Navigasi

Ketika kita membuka aplikasi terminal, kita akan masuk ke dalam sebuah direktori didalam komputer kita. Direktori tersebut adalah direktori user, yang biasa dilambangkan dengan menggunakan **tilde (~)**.

Di terminal kita juga bisa melakukan navigasi untuk pindah ke direktori lain lho. Caranya gimana ya?



Sebelum kita mempelajari gimana caranya navigasi di dalam terminal, kita perlu tau dulu nih, apa itu path.

Path

Adalah sebuah **alamat dari suatu file atau direktori di dalam komputer** kita.

- Kalo di Windows, path itu biasanya kayak gini:
C:\Program Files\Tetris\Tetris.exe
- Tapi kalo di sistem operasi yang mirip dengan Unix, biasanya kayak gini:
/home/fain/Pictures/Tetris.png

Nah, nantinya kita akan lebih sering menggunakan versi Unix~



Nah, path sendiri ada dua jenis, yaitu absolute path, dan relative path~

Absolute Path

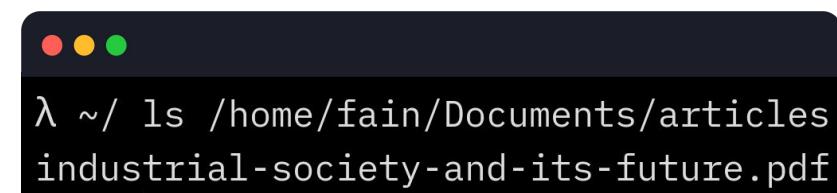
adalah sebuah path yang ditulis mulai dari root directory nya atau slash. Contoh :

/home/sabrina/Pictures/Tetris.png

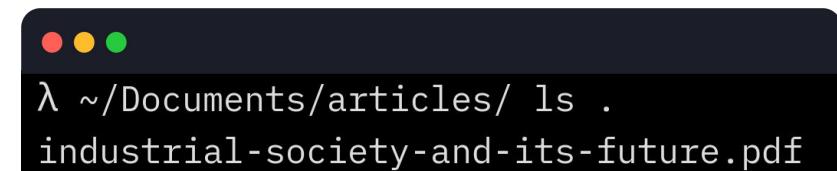
Relative Path

adalah path yang digunakan dengan menggunakan direktori yang sedang dibuka sebagai referensi. Sebagai contoh, saat ini kita berada di dalam direktori Pictures, maka kita bisa memanggil file Tetris.png secara relatif.

./Tetris.png



```
λ ~/ ls /home/fain/Documents/articles  
industrial-society-and-its-future.pdf
```



```
λ ~/Documents/articles/ ls .  
industrial-society-and-its-future.pdf
```

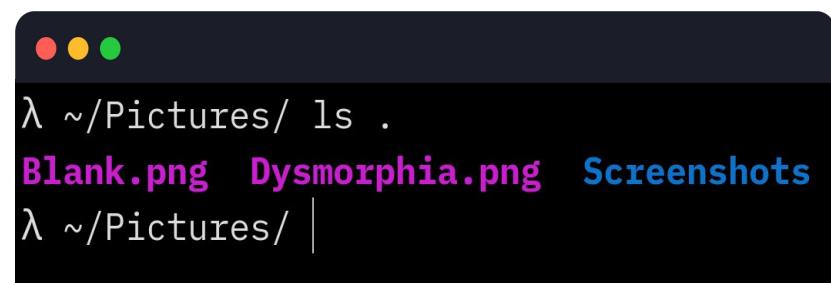


Ada dua jenis teks atau simbol yang dapat kita gunakan ketika kita ingin menulis sebuah **relative path**.

1. Current Directory (.)

Simbol satu titik digunakan untuk memanggil direktori yang sedang kita buka.

Misalnya, saat ini kita sedang berada di dalam direktori **Pictures**. Apabila kita menggunakan perintah **ls** dan diikuti dengan **titik**, artinya kita sedang menyuruh komputer kita untuk menampilkan file atau direktori apa aja yang ada di dalam direktori saat ini.



```
λ ~/Pictures/ ls .
Blank.png Dysmorphia.png Screenshots
λ ~/Pictures/ |
```



2. Parent Directory (..)

Simbol dua titik **digunakan untuk memanggil parent directory dari directory saat ini.**

Misalnya, kita sedang berada di dalam path

/home/sabrina/Pictures

Artinya, sabrina adalah parent directory dari direktori Pictures. Kita bisa gunakan perintah ls dan diikuti oleh argumen dua titik untuk menampilkan apa saja yang ada di dalam direktori sabrina.

```
λ ~/Pictures/ pwd  
/home/fain/Pictures  
λ ~/Pictures/ ls ..  
AUR      Documents    Pictures      'Stoplight Studio'  Works  
Desktop  Downloads    Repositories  Videos  
λ ~/Pictures/ |
```

Relative path ini bisa diteruskan path-nya. Contohnya, ketika kita sedang berada di dalam path

/home/sabrina/Pictures

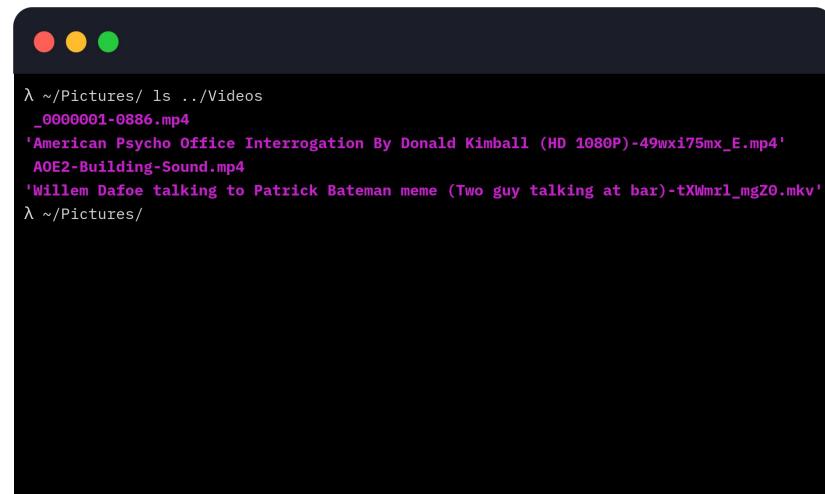
Tapi kita ingin melihat apa isi dari directory Download, yang ada di path berikut:

/home/sabrina/Videos

Maka kita bisa menuliskan path seperti dibawah ini :

../Videos

Path directory yang saat ini baru dibuka di terminal biasa disebut sebagai Current Working Directory (CWD). Nah relative path ini adalah path yang relatif terhadap CWD tadi.



```
λ ~/Pictures/ ls ../Videos
_0000001-0886.mp4
'American Psycho Office Interrogation By Donald Kimball (HD 1080P)-49wx175mx_E.mp4'
AOE2-Building-Sound.mp4
'Willem Dafoe talking to Patrick Bateman meme (Two guy talking at bar)-tXWmrl_mgZ0.mkv'
λ ~/Pictures/
```



Change Directory

Nah, setelah kalian tau apa itu path, sekarang saatnya kita cobain perintah yang digunakan untuk berpindah-pindah direktori di dalam terminal, layaknya lagi menggunakan File Explorer.

Untuk **berpindah direktori di dalam terminal, kita bisa menggunakan perintah bernama cd**. Dimana perintah cd ini meminta path sebagai argumen, baik itu relative path maupun absolute path.

Perintah cd ini akan merubah current working directory.

```
λ ~/ cd Downloads
λ ~/Downloads/ cd ../Documents
λ ~/Documents/ cd /home/fain.Repositories
λ ~/Repositories/
```

Eits, gak kok... fungsinya ga hanya pindah direktori aja. Ada **perintah-perintah di terminal** lainnya yang akan kita bahas. Dan pastinya bakal berguna banget. Cekidot.



Berikut perintah yang umum dipake terkait dengan terminal~

1. ls
2. cat
3. touch
4. rm
5. mkdir
6. echo

Kita cek satu persatu, yuk!





1. ls

ls merupakan **perintah terminal yang berguna untuk menampilkan konten dari sebuah directory**. Sama seperti cd perintah ini meminta **path** sebagai argumen.



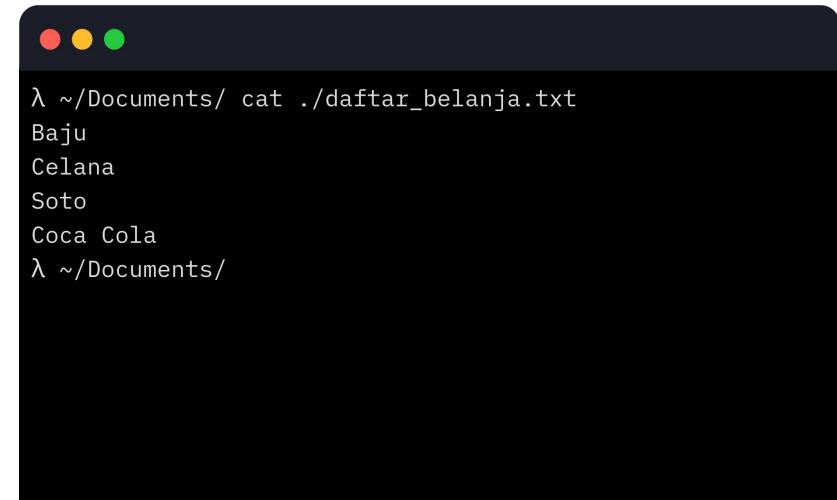
```
λ ~/Repositories/ ls .
bookstoresvc          GO
buf                   grpc-gateway-boilerplate
coffeesvc             next-elements
dotfiles              serat
express-public        stoplight-gatsby
fikrirnurhidayat     stoplight-next
gatsby-starter-elements stoplight-react
λ ~/Repositories/
```

2. cat

Perintah ini bukan perintah untuk ngelus-ngelus kucing ya! Tapi perintah ini digunakan untuk **menampilkan konten dari sebuah file**.

Misal, kalian punya daftar belanja yang kalian simpan di sebuah file bernama **daftar_belanja.txt**. Nah, kalian bisa melihat isi dari file tersebut melalui terminal dengan menggunakan perintah cat.

Sama seperti ls dan cd, cat meminta path sebagai argumen, namun path tersebut haruslah merujuk ke sebuah file.



```
λ ~/Documents/ cat ./daftar_belanja.txt
Baju
Celana
Soto
Coca Cola
λ ~/Documents/
```

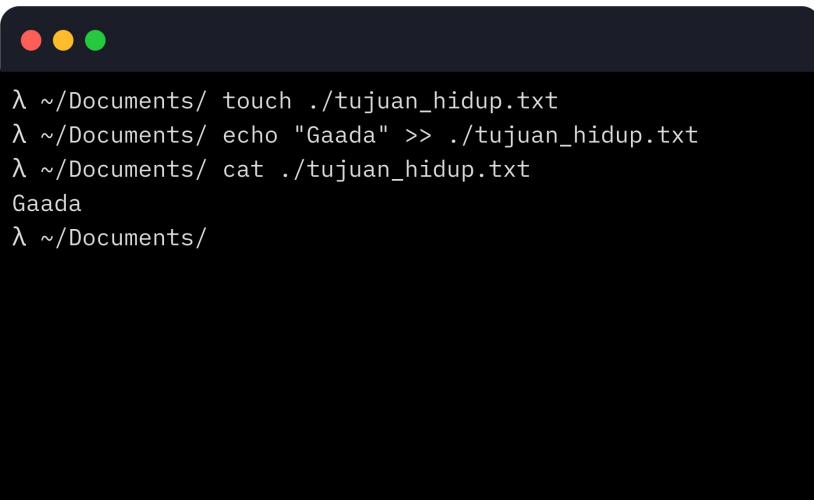


3. touch

Perintah ini bisa kita pakai untuk **membuat sebuah file di dalam terminal**. Nah, perintah ini meminta path juga sebagai argumen.

Biasanya perintah ini dikombinasikan dengan echo, yang digunakan untuk memasukkan teks ke dalam file tersebut.

You know, di dalam Unix-like system, semua hal itu adalah file.



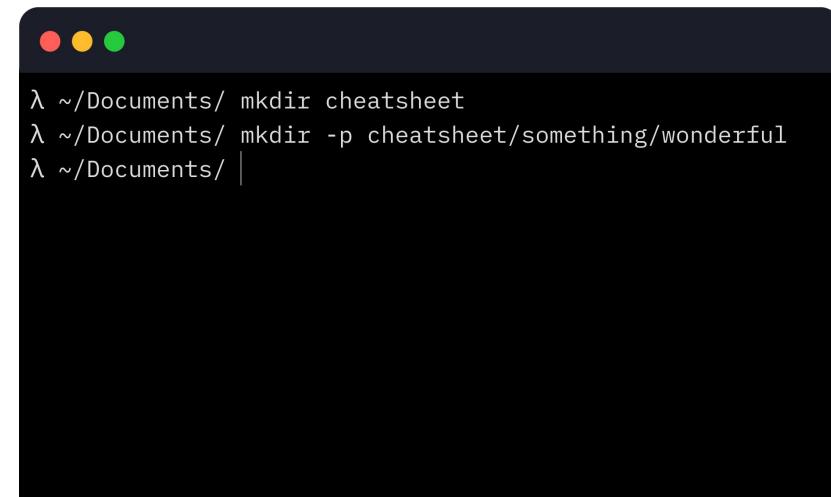
```
λ ~/Documents/ touch ./tujuan_hidup.txt
λ ~/Documents/ echo "Gaada" >> ./tujuan_hidup.txt
λ ~/Documents/ cat ./tujuan_hidup.txt
Gaada
λ ~/Documents/
```



4. mkdir

Perintah ini **digunakan untuk membuat direktori**. Mirip dengan cd, perintah ini meminta path sebagai argumen.

Apabila kita ingin membuat direktori yang bersarang, kita bisa menambahkan flag **-p** agar semua direktori yang belum ada pada path tersebut akan otomatis terbuat.



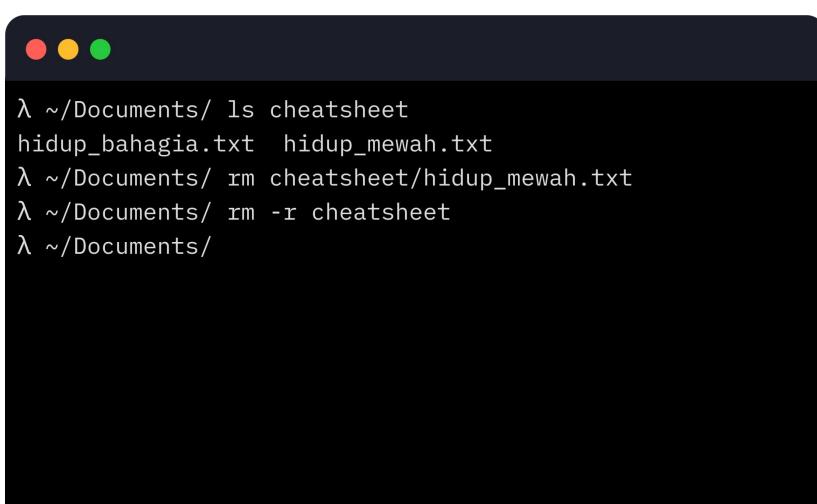
```
λ ~/Documents/ mkdir cheatsheet
λ ~/Documents/ mkdir -p cheatsheet/something/wonderful
λ ~/Documents/ |
```



5. rm

Kalo ada yang dibuat, pasti ada yang dihapus juga. Nah, perintah rm ini **digunakan untuk menghapus sesuatu, baik itu file maupun direktori.**

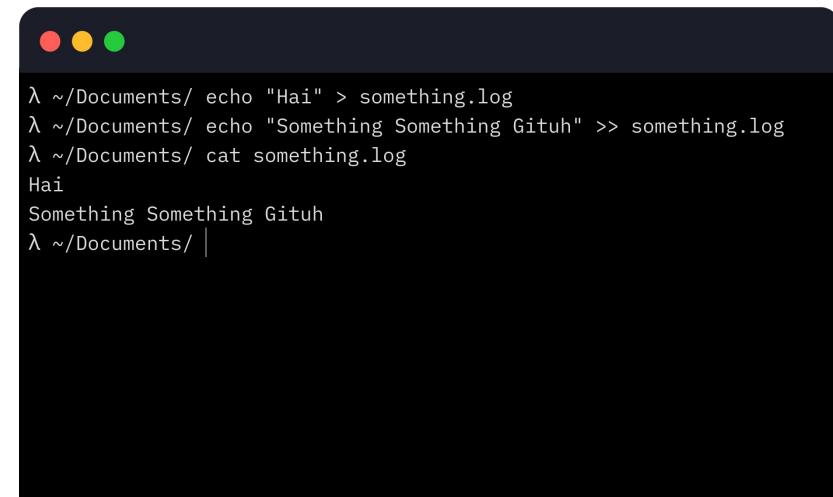
Perintah ini meminta path sebagai argumen, dan jika path tersebut berupa direktori, maka kita wajib menyertakan flag **-r** yang berarti hapus secara rekursif.



```
λ ~/Documents/ ls cheatsheet  
hidup_bahagia.txt hidup_mewah.txt  
λ ~/Documents/ rm cheatsheet/hidup_mewah.txt  
λ ~/Documents/ rm -r cheatsheet  
λ ~/Documents/
```

6. echo

Perintah ini **digunakan untuk mengeluarkan sebuah output di terminal**. Tidak hanya untuk logging, perintah ini bisa memodifikasi output agar dapat disimpan pada file lain.



```
λ ~/Documents/ echo "Hai" > something.log
λ ~/Documents/ echo "Something Something Gituh" >> something.log
λ ~/Documents/ cat something.log
Hai
Something Something Gituh
λ ~/Documents/ |
```

Nah, tadi itu kegunaan dasar terminal yang wajib diketahui terlebih dahulu.

Nantinya sebagai developer, akan banyak perintah-perintah baru yang akan digunakan untuk menjalankan websitemu masing-masing.

Bash from Scratch bisa bantu kamu untuk eksplorasi lebih jauh. Silakan dicoba ~



Terminal hanyalah salah satu tools yang digunakan dalam mengembangkan sebuah website. Ada tools lain yang sangat powerful bernama **IDE**.

Benda apaan tuh? Yuk langsung kita bahas!





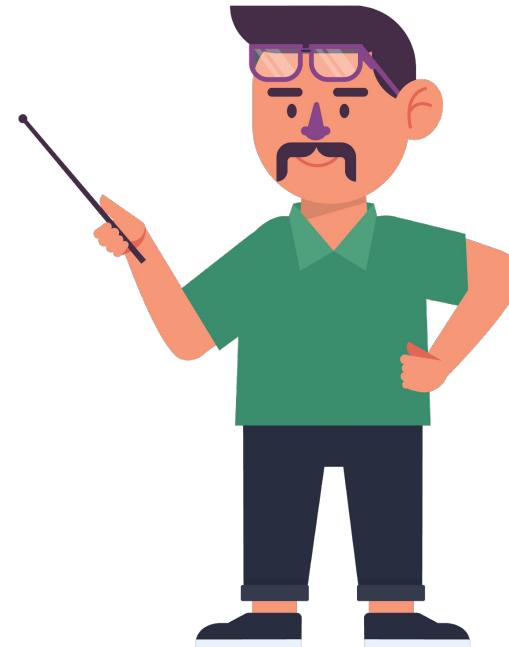
Ketika melakukan software development, sangat dianjurkan untuk menggunakan IDE. Nggak mungkin dong kamu cuma berbekal notepad++ aja karena notepad++ belum dilengkapi fitur untuk compile atau running kode.

Nah IDE di dalam software development sendiri biasa terdiri dari beberapa hal, antara lain:

- **Text editor**
- **Terminal Emulator**

Integrated Development Environment

Merupakan aplikasi yang digunakan menulis kode untuk website yang mau kita buat. Dan di dalam IDE terdapat Intergrated Terminal, jadi kalian bisa jalankan website kalian langsung di dalam IDE.

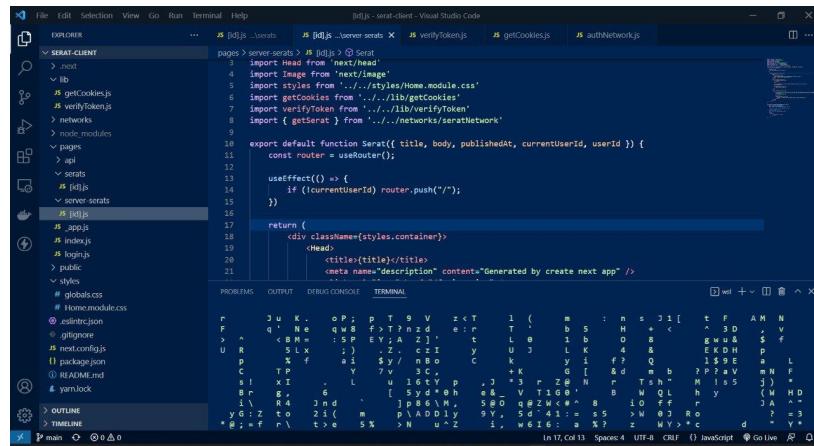


Hmmm.. kita udah punya terminal emulator kan? Ya, kita sudah punya, maka dari itu kita hanya perlu menginstal Text Editor yang proper buat kita.

Berikut contoh-contoh text editor yang populer dipakai:

- [Visual Studio Code](#)
- [Atom](#)
- [Sublime Text](#)
- Eclipse (Java and Kotlin only)
- Vim

Nah, Sabrina sangat merekomendasikan untuk menggunakan **VSCode** sebagai IDE, karena sangat populer dan overpower hehe.



The screenshot shows the Visual Studio Code interface with the following details:

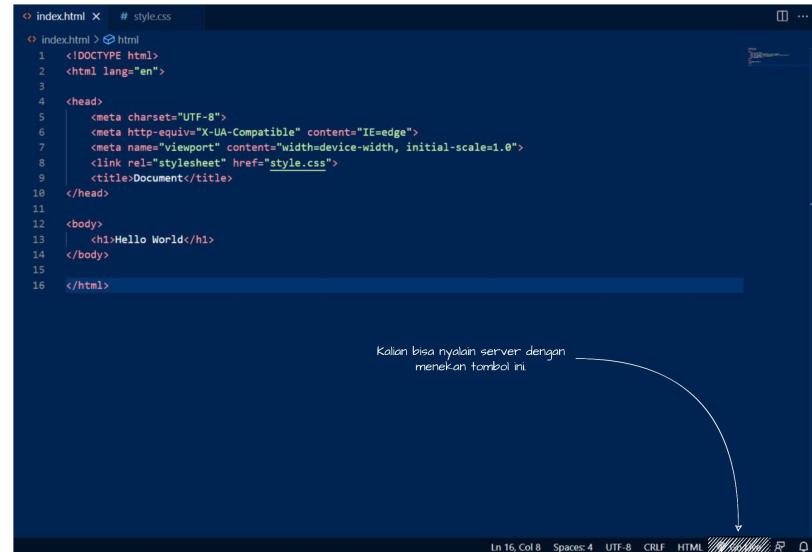
- File Explorer:** Shows a file tree with a folder named "SERAT-CLIENT" containing subfolders ".next", ".lib", "pages", "public", "styles", and "server-serats". Inside "server-serats", there are files "index.js", "login.js", and "app.js".
- Code Editor:** The main editor area displays the content of the "index.js" file. The code is a Next.js application component:

```
pages.getServerSerats = (id) => {
  const router = useRouter();
  useEffect(() => {
    if (!currentUserId) router.push("/");
  })
  return (
    <div className="container">
      <Head>
        <title>{title}</title>
        <meta name="description" content="Generated by create next app" />
      </Head>
      <main>
        <h1>Hello from Serat!</h1>
        <p>This page is generated by the server-side rendering API.</p>
      </main>
    </div>
  )
}
```

- Terminal:** A terminal window at the bottom shows the command "npx create-next-app serat-client" being run.
- Bottom Bar:** Includes tabs for PROBLEMS, OUTPUT, DEBUG CONSOLE, and TERMINAL, along with status indicators for 117 files, 13 spaces, and 10 tabs.

Kalian bisa lihat video singkat ini untuk mengetahui tentang apa itu VSCode dan apa saja yang bisa kalian lakukan.

VS Code in 100 Seconds



A screenshot of the Visual Studio Code (VS Code) interface. The main editor window displays the following HTML code:

```
index.html X # style.css
index.html > html
1  <!DOCTYPE html>
2  <html lang="en">
3
4  <head>
5      <meta charset="UTF-8">
6      <meta http-equiv="X-UA-Compatible" content="IE=edge">
7      <meta name="viewport" content="width=device-width, initial-scale=1.0">
8      <link rel="stylesheet" href="style.css">
9      <title>Document</title>
10 </head>
11
12 <body>
13     <h1>Hello World</h1>
14 </body>
15
16 </html>
```

In the bottom right corner of the code editor, there is a small blue rectangular callout box containing the text: "Kalian bisa nyalon server dengan menekan tombol ini". A white curved arrow points from this text towards the green "Run" button located at the bottom right of the code editor interface.

Ln 16, Col 8 Spaces: 4 UTF-8 CRLF HTML

Saatnya kita Quiz!





1. Manakah dari pernyataan dibawah ini yang kurang tepat ?

- A. Terminal adalah sebuah aplikasi yang digunakan untuk menjalankan perintah ke komputer dalam bentuk teks.
- B. Terminal adalah sebuah aplikasi yang digunakan untuk mendesain UI dari sebuah aplikasi.
- C. Terminal adalah sebuah aplikasi yang dapat digunakan untuk menulis kode program kita.

1. Manakah dari pernyataan dibawah ini yang kurang tepat ?

- A. Terminal adalah sebuah aplikasi yang digunakan untuk menjalankan perintah ke komputer dalam bentuk teks.
- B. **Terminal adalah sebuah aplikasi yang digunakan untuk mendesain UI dari sebuah aplikasi.**
- C. Terminal adalah sebuah aplikasi yang dapat digunakan untuk menulis kode program kita.

Terminal tidak dapat digunakan dan sangat tidak efektif untuk mendesain UI dari sebuah aplikasi.



2. Dari baris perintah berikut, manakah pernyataan yang benar mengenai baris perintah echo "Banana" "Orange" "Apple"

- A. echo adalah sebuah perintah, dan "Banana" "Orange" "Apple" adalah perintah juga.
- B. echo adalah sebuah perintah, dan "Banana" "Orange" "Apple" adalah sub perintah dari perintah echo.
- C. echo adalah sebuah perintah, dan "Banana" "Orange" "Apple" adalah argumen dari perintah echo.

2. Dari baris perintah berikut, manakah pernyataan yang benar mengenai baris perintah tersebut?

echo "Banana" "Orange" "Apple"

- A. echo adalah sebuah perintah, dan "Banana" "Orange" "Apple" adalah perintah juga.
- B. echo adalah sebuah perintah, dan "Banana" "Orange" "Apple" adalah sub perintah dari perintah echo.
- C. **echo adalah sebuah perintah, dan "Banana" "Orange" "Apple" adalah argumen dari perintah echo.**

echo adalah sebuah perintah di dalam terminal, yang mana apa yang ditulis setelah perintah terus akan menjadi argumen dari perintah tersebut.

3. Jika saat ini kamu berada di direktori A, maka bagaimana caramu biar bisa berpindah ke direktori B?

- A. ls .. /B
- B. cd .. /B
- C. cd /home/sabrina/B



3. Jika saat ini kamu berada di direktori A, maka bagaimana caramu biar bisa berpindah ke direktori B?

- A. ls .. /B
- B. cd .. /B
- C. cd /home/sabrina/B



Karena kita tidak tau path seluruhnya dari direktori A, maka dari itu kita bisa gunakan relative path saja, dan perintah untuk berpindah direktori adalah dengan menggunakan cd.



- 4. Dari deretan perintah di bawah ini, manakah perintah yang digunakan untuk membuat sebuah file?**
- A. touch .
 - B. echo "Hello" > hello.txt
 - C. mkdir hello.txt

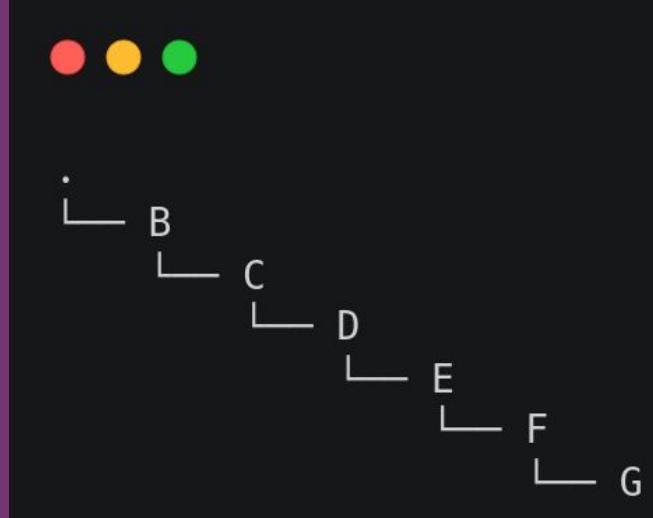
- 4. Dari deretan perintah di bawah ini, manakah perintah yang digunakan untuk membuat sebuah file?**
- A. touch .
 - B. echo "Hello" > hello.txt
 - C. mkdir hello.txt

touch . adalah perintah yang tidak valid, karena target path-nya adalah current working directory, maka dari itu, ketika kita menjalankan perintah itu, tidak akan ada file baru.



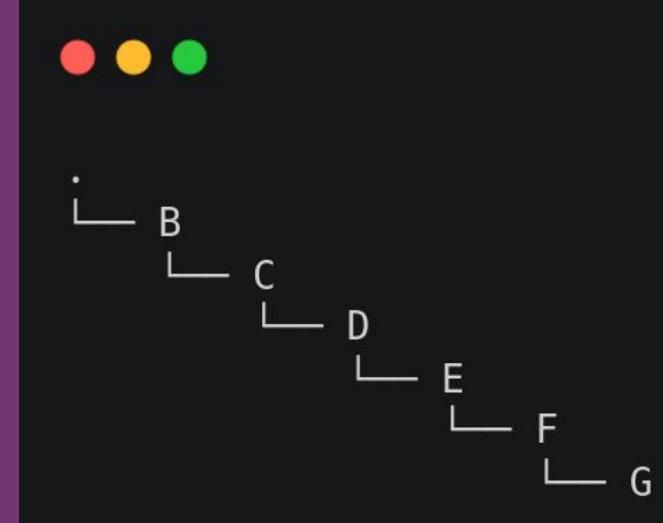
5. Terdapat struktur direktori seperti gambar disamping. Bagaimana cara menghapus direktori E sampai ke G?

- A. rm -f ./B/C/D/E
- B. rm -r B/C/D/E/F/G
- C. rm -r B/C/D/E



5. Terdapat struktur direktori seperti gambar disamping. Bagaimana cara menghapus direktori E sampai ke G?

- A. rm -f ./B/C/D/E
- B. rm -r B/C/D/E/F/G
- C. rm -r B/C/D/E



Dengan menggunakan -r artinya kita ingin menghapus sebuah path secara rekursif. Maka dari itu opsi C sangat tepat, karena kita ingin menghapus direktori E sampai ke G.

Yuk eksplor lebih jauh!

- [Bash From Scratch](#)
- [Beginner's Guide To The Linux Terminal](#)



Nah, selesai sudah pembahasan kita di Chapter 3 Topic 1 ini.

Selanjutnya, kita bakal bahas **GIT**.

Penasaran kayak gimana? Cus langsung ke topik selanjutnya~



Terima Kasih!



Next Topic

loading...