

Modul Seri Praktikum

Membuat Application Programming Interface (API) Berbasis Laravel 13

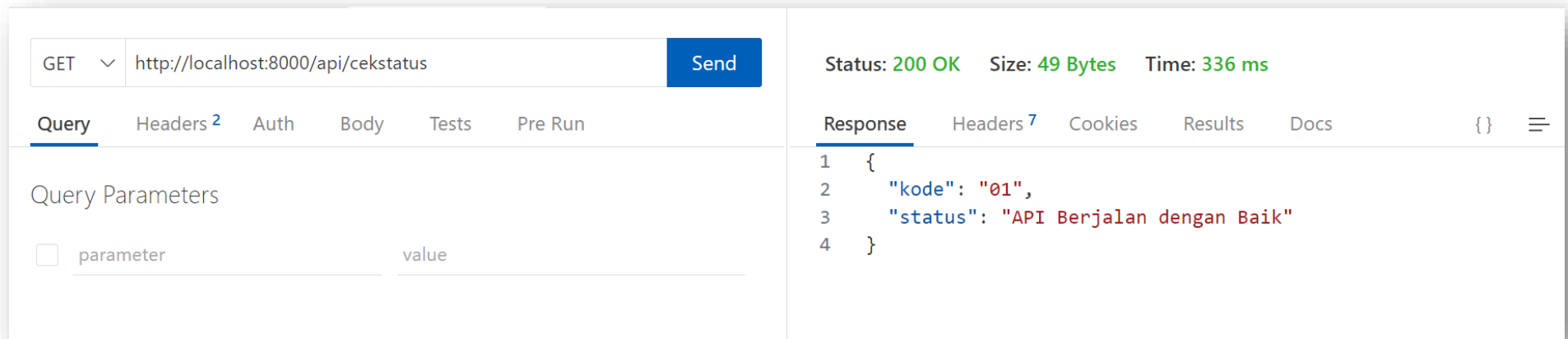
Study Kasus

Framework7 versi 9



Membuat Application Programming Interface (API) Berbasis Laravel 13

Pada praktikum kali ini kita akan **membangun sebuah Application Programming Interface (API) sederhana berbasis Laravel 13**. API yang dibangun hanya menampilkan sebuah **respon** sederhana berupa data berformat **Javascript Object Notation (JSON)** yang memiliki struktur statis seperti gambar dibawah ini.



Untuk merealisasikan hal tersebut, maka kita membutuhkan sebuah project **Laravel13**, beberapa **pengaturan dasar**, **pengaturan routing** serta sebuah **controller**. Untuk mendapatkan project Laravel 13 kita dapat menggunakan **Composer** serta minimal menggunakan **PHP 8.3**. **Composer** merupakan sebuah **tool manajer paket** untuk bahasa pemrograman PHP yang menyediakan format standar dalam mengelola **dependensi PHP** dan **library** atau **plugin** yang diperlukan dalam pengembangan project website. Berikut tabel contoh tools sejenisnya.

Jenis Tools

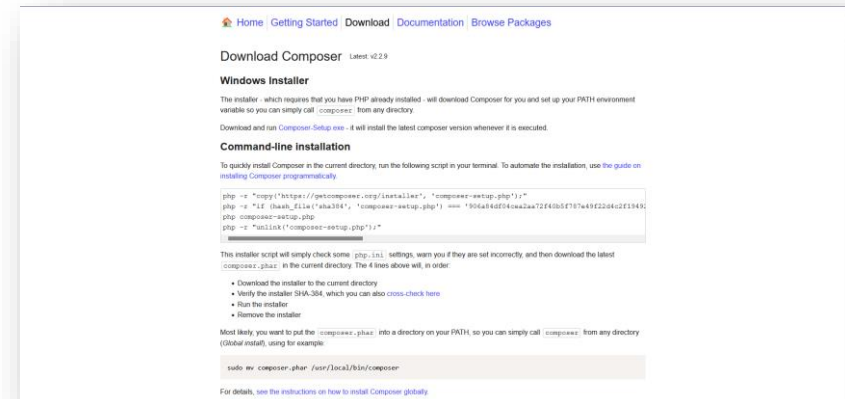
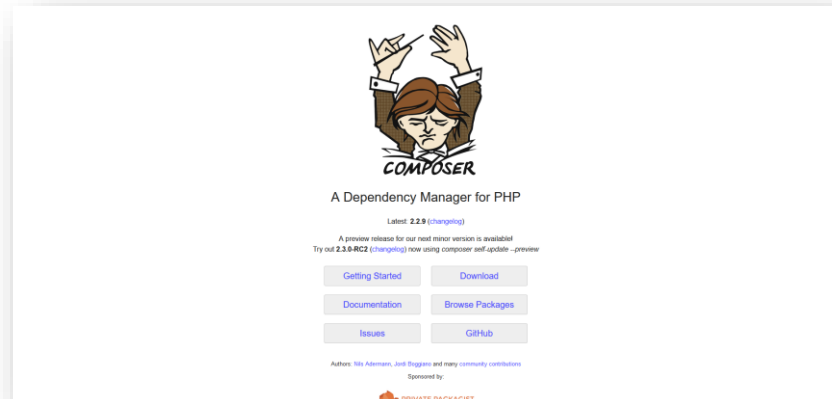
- Composer
- Node Package Manager (NPM)
- Preferred Installer Program (PIP)
- Maven
- dll

Untuk Bahasa Pemrograman

- PHP
- NodeJS / Javascript
- Python
- Java
- dll

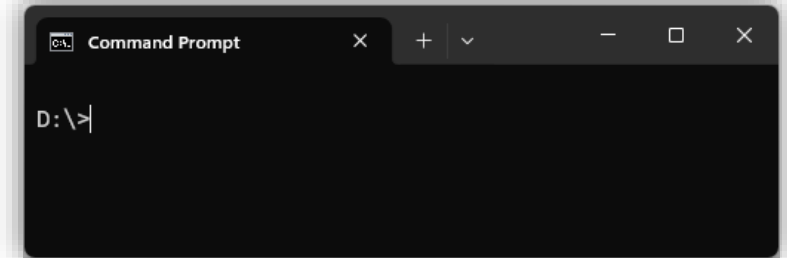
Untuk skenario penggunaan Web Server kita akan menggunakan **Local Development Server bawaan Laravel 13** yaitu **PHP Artisan**. Setelah memahami skenario yang akan dilakukan, maka langkah-langkahnya sebagai berikut

1. Langkah pertama ini kita akan mencoba untuk **mendapatkan project Laravel 13** menggunakan **Composer**. Cara mendapatkannya melalui **aplikasi berbasis terminal (Windows Command Prompt, Powershell, Bash, dll)**. Silahkan download terlebih dahulu tool bernama **Composer** yang bisa didapatkan di situs resminya dengan alamat <https://getcomposer.org/>. Jika telah memiliki Composer sebelumnya, silahkan lanjut ke 3.

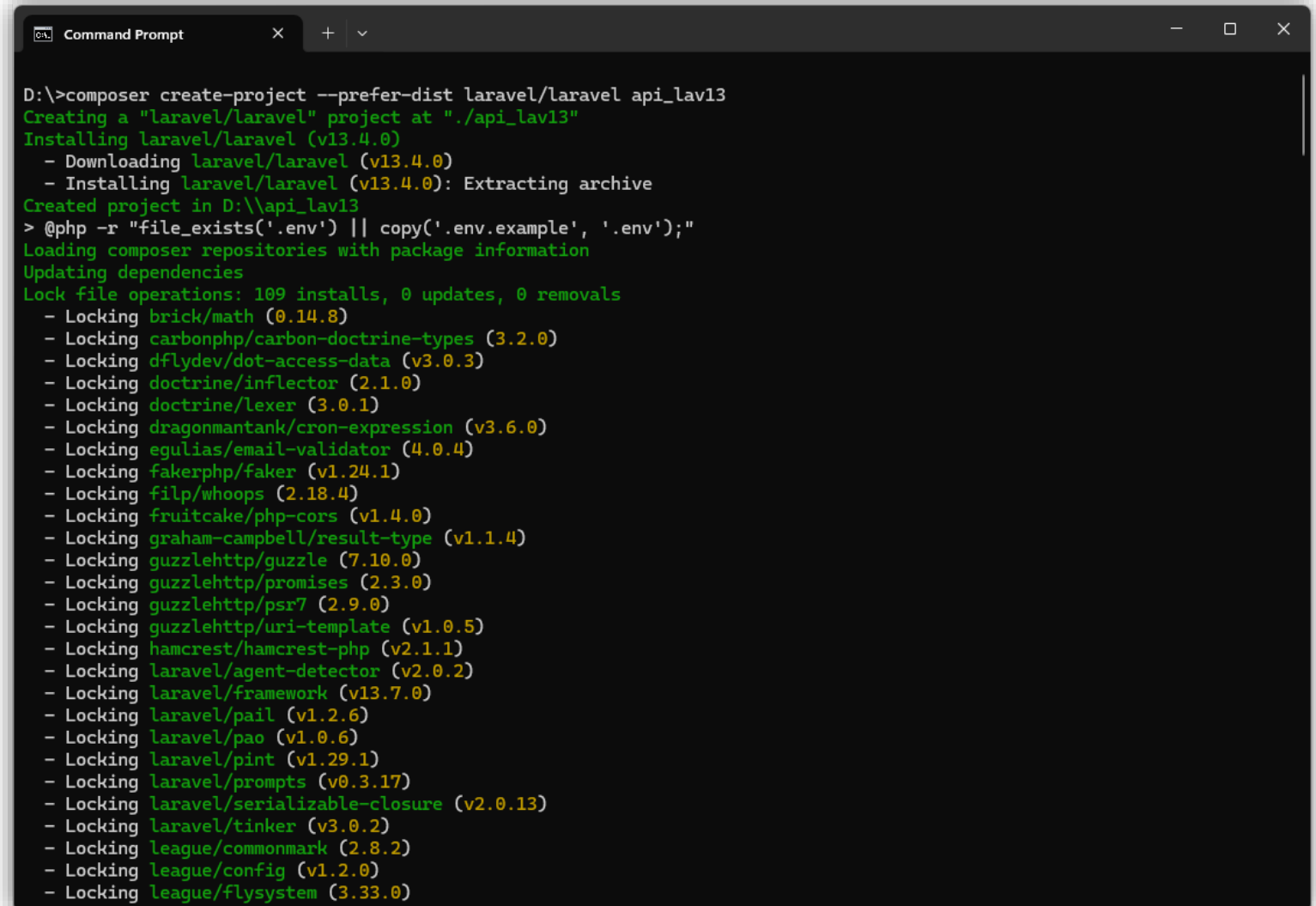


2. Setelah berhasil mendapatkan **file installer Composer**, silahkan **install Composer** terlebih dahulu.

3. Langkah berikutnya adalah **mendownload project Laravel 13** melalui **Composer**. Untuk merealisasikan-nya kita harus menggunakan **aplikasi berbasis terminal** (standarisasi menggunakan **Windows Command Prompt**). Skenario yang akan kita lakukan adalah membuat project API Laravel bernama **api_lav13** yang direncanakan diletakkan di **drive D**, sehingga kita harus mengakses drive D di **Windows Command Prompt** seperti gambar disamping



4. Langkah berikutnya adalah **mendownload project Laravel 13** dengan perintah **composer create-project --prefer-dist laravel/laravel api_lav13**. Perintah ini diketik pada **Windows Command Prompt** seperti gambar disamping dan seterusnya ke bawah (sebagai catatan agar proses lancar, matikan antivirus jika pada PC/laptop terdapat antivirus yang aktif).



```
D:\>composer create-project --prefer-dist laravel/laravel api_lav13
Creating a "laravel/laravel" project at "./api_lav13"
Installing laravel/laravel (v13.4.0)
- Downloading laravel/laravel (v13.4.0)
- Installing laravel/laravel (v13.4.0): Extracting archive
Created project in D:\api_lav13
> @php -r "file_exists('.env') || copy('.env.example', '.env');"
Loading composer repositories with package information
Updating dependencies
Lock file operations: 109 installs, 0 updates, 0 removals
- Locking brick/math (0.14.8)
- Locking carbonphp/carbon-doctrine-types (3.2.0)
- Locking dflydev/dot-access-data (v3.0.3)
- Locking doctrine/inflector (2.1.0)
- Locking doctrine/lexer (3.0.1)
- Locking dragonmantank/cron-expression (v3.6.0)
- Locking egulias/email-validator (4.0.4)
- Locking fakerphp/faker (v1.24.1)
- Locking filp/whoops (2.18.4)
- Locking fruitcake/php-cors (v1.4.0)
- Locking graham-campbell/result-type (v1.1.4)
- Locking guzzlehttp/guzzle (7.10.0)
- Locking guzzlehttp/promises (2.3.0)
- Locking guzzlehttp/psr7 (2.9.0)
- Locking guzzlehttp/uri-template (v1.0.5)
- Locking hamcrest/hamcrest-php (v2.1.1)
- Locking laravel/agent-detector (v2.0.2)
- Locking laravel/framework (v13.7.0)
- Locking laravel/pail (v1.2.6)
- Locking laravel/pao (v1.0.6)
- Locking laravel/pint (v1.29.1)
- Locking laravel/prompts (v0.3.17)
- Locking laravel/serializable-closure (v2.0.13)
- Locking laravel/tinker (v3.0.2)
- Locking league/commonmark (2.8.2)
- Locking league/config (v1.2.0)
- Locking league/flysystem (3.33.0)
```

Command Prompt

- Locking league/flysystem-local (3.31.0)
- Locking league/mime-type-detection (1.16.0)
- Locking league/uri (7.8.1)
- Locking league/uri-interfaces (7.8.1)
- Locking mockery/mockery (1.6.12)
- Locking monolog/monolog (3.10.0)
- Locking myclabs/deep-copy (1.13.4)
- Locking nesbot/carbon (3.11.4)
- Locking nette/schema (v1.3.5)
- Locking nette/utils (v4.1.3)
- Locking nikic/php-parser (v5.7.0)
- Locking nunomaduro/collision (v8.9.4)
- Locking nunomaduro/termwind (v2.4.0)
- Locking phar-io/manifest (2.0.4)
- Locking phar-io/version (3.2.1)
- Locking phpoption/phpoption (1.9.5)
- Locking phpunit/php-code-coverage (12.5.6)
- Locking phpunit/php-file-iterator (6.0.1)
- Locking phpunit/php-invoker (6.0.0)
- Locking phpunit/php-text-template (5.0.0)
- Locking phpunit/php-timer (8.0.0)
- Locking phpunit/phpunit (12.5.24)
- Locking psr/clock (1.0.0)
- Locking psr/container (2.0.2)
- Locking psr/event-dispatcher (1.0.0)
- Locking psr/http-client (1.0.3)
- Locking psr/http-factory (1.1.0)
- Locking psr/http-message (2.0)
- Locking psr/log (3.0.2)
- Locking psr/simple-cache (3.0.0)
- Locking psy/psysh (v0.12.22)
- Locking ralouphie/getallheaders (3.0.3)
- Locking ramsey/collection (2.1.1)
- Locking ramsey/uuid (4.9.2)
- Locking sebastian/cli-parser (4.2.0)
- Locking sebastian/comparator (7.1.6)
- Locking sebastian/complexity (5.0.0)
- Locking sebastian/diff (7.0.0)

```
- Locking sebastian/environment (8.1.0)
- Locking sebastian/exporter (7.0.2)
- Locking sebastian/global-state (8.0.2)
- Locking sebastian/lines-of-code (4.0.0)
- Locking sebastian/object-enumerator (7.0.0)
- Locking sebastian/object-reflector (5.0.0)
- Locking sebastian/recursion-context (7.0.1)
- Locking sebastian/type (6.0.3)
- Locking sebastian/version (6.0.0)
- Locking staabm/side-effects-detector (1.0.5)
- Locking symfony/clock (v8.0.8)
- Locking symfony/console (v8.0.9)
- Locking symfony/css-selector (v8.0.9)
- Locking symfony/deprecation-contracts (v3.6.0)
- Locking symfony/error-handler (v8.0.8)
- Locking symfony/event-dispatcher (v8.0.9)
- Locking symfony/event-dispatcher-contracts (v3.6.0)
- Locking symfony/finder (v8.0.8)
- Locking symfony/http-foundation (v8.0.8)
- Locking symfony/http-kernel (v8.0.8)
- Locking symfony/mailer (v8.0.8)
- Locking symfony/mime (v8.0.9)
- Locking symfony/polyfill-ctype (v1.37.0)
- Locking symfony/polyfill-intl-grapheme (v1.37.0)
- Locking symfony/polyfill-intl-idn (v1.37.0)
- Locking symfony/polyfill-intl-normalizer (v1.37.0)
- Locking symfony/polyfill-mbstring (v1.37.0)
- Locking symfony/polyfill-php80 (v1.37.0)
- Locking symfony/polyfill-php84 (v1.37.0)
- Locking symfony/polyfill-php85 (v1.37.0)
- Locking symfony/polyfill-php86 (v1.37.0)
- Locking symfony/polyfill-uuid (v1.37.0)
- Locking symfony/process (v8.0.8)
- Locking symfony/routing (v8.0.9)
- Locking symfony/service-contracts (v3.6.1)
- Locking symfony/string (v8.0.8)
- Locking symfony/translation (v8.0.8)
- Locking symfony/translation-contracts (v3.6.1)
```

Command Prompt

```
- Locking symfony/translation-contracts (v3.6.1)
- Locking symfony/uid (v8.0.9)
- Locking symfony/var-dumper (v8.0.8)
- Locking theseer/tokenizer (2.0.1)
- Locking tijsverkoyen/css-to-inline-styles (v2.4.0)
- Locking vlucas/phpdotenv (v5.6.3)
- Locking voku/portable-ascii (2.1.1)
Writing lock file
Installing dependencies from lock file (including require-dev)
Package operations: 109 installs, 0 updates, 0 removals
- Downloading doctrine/inflector (2.1.0)
- Downloading symfony/deprecation-contracts (v3.6.0)
- Downloading symfony/polyfill-mbstring (v1.37.0)
- Downloading symfony/http-foundation (v8.0.8)
- Downloading guzzlehttp/psr7 (2.9.0)
- Downloading guzzlehttp/promises (2.3.0)
- Downloading guzzlehttp/guzzle (7.10.0)
- Downloading symfony/polyfill-php80 (v1.37.0)
- Downloading guzzlehttp/uri-template (v1.0.5)
- Downloading symfony/polyfill-intl-grapheme (v1.37.0)
- Downloading symfony/polyfill-ctype (v1.37.0)
- Downloading symfony/string (v8.0.8)
- Downloading symfony/console (v8.0.9)
- Downloading nunomaduro/termwind (v2.4.0)
- Downloading voku/portable-ascii (2.1.1)
- Downloading symfony/css-selector (v8.0.9)
- Downloading symfony/var-dumper (v8.0.8)
- Downloading symfony/polyfill-uuid (v1.37.0)
- Downloading symfony/uid (v8.0.9)
- Downloading symfony/routing (v8.0.9)
- Downloading symfony/process (v8.0.8)
- Downloading symfony/polyfill-php86 (v1.37.0)
- Downloading symfony/polyfill-php85 (v1.37.0)
- Downloading symfony/polyfill-php84 (v1.37.0)
- Downloading symfony/polyfill-intl-idn (v1.37.0)
- Downloading symfony/mime (v8.0.9)
- Downloading symfony/event-dispatcher-contracts (v3.6.0)
- Downloading symfony/event-dispatcher (v8.0.9)
```

Command Prompt

- Downloading egulias/email-validator (4.0.4)
- Downloading symfony/mailer (v8.0.8)
- Downloading symfony/error-handler (v8.0.8)
- Downloading symfony/http-kernel (v8.0.8)
- Downloading symfony/finder (v8.0.8)
- Downloading ramsey/collection (2.1.1)
- Downloading brick/math (0.14.8)
- Downloading symfony/translation (v8.0.8)
- Downloading symfony/clock (v8.0.8)
- Downloading nesbot/carbon (3.11.4)
- Downloading monolog/monolog (3.10.0)
- Downloading league/uri-interfaces (7.8.1)
- Downloading league/uri (7.8.1)
- Downloading league/flysystem-local (3.31.0)
- Downloading league/flysystem (3.33.0)
- Downloading nette/utils (v4.1.3)
- Downloading nette/schema (v1.3.5)
- Downloading league/commonmark (2.8.2)
- Downloading laravel/serializable-closure (v2.0.13)
- Downloading laravel/prompts (v0.3.17)
- Downloading laravel/framework (v13.7.0)
- Downloading laravel/pail (v1.2.6)
- Downloading laravel/agent-detector (v2.0.2)
- Downloading laravel/pao (v1.0.6)
- Downloading laravel/pint (v1.29.1)
- Downloading psy/psysh (v0.12.22)
- Downloading laravel/tinker (v3.0.2)
- Downloading hamcrest/hamcrest-php (v2.1.1)
- Downloading filp/whoops (2.18.4)
- Downloading nunomaduro/collision (v8.9.4)
- Downloading sebastian/version (6.0.0)
- Downloading sebastian/type (6.0.3)
- Downloading sebastian/recursion-context (7.0.1)
- Downloading sebastian/object-reflector (5.0.0)
- Downloading sebastian/object-enumerator (7.0.0)
- Downloading sebastian/global-state (8.0.2)
- Downloading sebastian/exporter (7.0.2)
- Downloading sebastian/environment (8.1.0)

Command Prompt

- Downloading sebastian/diff (7.0.0)
- Downloading sebastian/comparator (7.1.6)
- Downloading sebastian/cli-parser (4.2.0)
- Downloading phpunit/php-timer (8.0.0)
- Downloading phpunit/php-text-template (5.0.0)
- Downloading phpunit/php-invoker (6.0.0)
- Downloading phpunit/php-file-iterator (6.0.1)
- Downloading theseer/tokenizer (2.0.1)
- Downloading sebastian/lines-of-code (4.0.0)
- Downloading sebastian/complexity (5.0.0)
- Downloading phpunit/php-code-coverage (12.5.6)
- Downloading myclabs/deep-copy (1.13.4)
- Downloading phpunit/phpunit (12.5.24)
- Installing doctrine/inflector (2.1.0): Extracting archive
- Installing doctrine/lexer (3.0.1): Extracting archive
- Installing dragonmantank/cron-expression (v3.6.0): Extracting archive
- Installing symfony/deprecation-contracts (v3.6.0): Extracting archive
- Installing psr/container (2.0.2): Extracting archive
- Installing fakerphp/faker (v1.24.1): Extracting archive
- Installing symfony/polyfill-mbstring (v1.37.0): Extracting archive
- Installing symfony/http-foundation (v8.0.8): Extracting archive
- Installing fruitcake/php-cors (v1.4.0): Extracting archive
- Installing psr/http-message (2.0): Extracting archive
- Installing psr/http-client (1.0.3): Extracting archive
- Installing ralouphie/getallheaders (3.0.3): Extracting archive
- Installing psr/http-factory (1.1.0): Extracting archive
- Installing guzzlehttp/psr7 (2.9.0): Extracting archive
- Installing guzzlehttp/promises (2.3.0): Extracting archive
- Installing guzzlehttp/guzzle (7.10.0): Extracting archive
- Installing symfony/polyfill-php80 (v1.37.0): Extracting archive
- Installing guzzlehttp/uri-template (v1.0.5): Extracting archive
- Installing symfony/polyfill-intl-normalizer (v1.37.0): Extracting archive
- Installing symfony/polyfill-intl-grapheme (v1.37.0): Extracting archive
- Installing symfony/polyfill-ctype (v1.37.0): Extracting archive
- Installing symfony/string (v8.0.8): Extracting archive
- Installing symfony/service-contracts (v3.6.1): Extracting archive
- Installing symfony/console (v8.0.9): Extracting archive
- Installing nunomaduro/termwind (v2.4.0): Extracting archive

Command Prompt

- Installing voku/portable-ascii (2.1.1): Extracting archive
- Installing phpopcion/phpopcion (1.9.5): Extracting archive
- Installing graham-campbell/result-type (v1.1.4): Extracting archive
- Installing vlucas/phpdotenv (v5.6.3): Extracting archive
- Installing symfony/css-selector (v8.0.9): Extracting archive
- Installing tijsverkoyen/css-to-inline-styles (v2.4.0): Extracting archive
- Installing symfony/var-dumper (v8.0.8): Extracting archive
- Installing symfony/polyfill-uuid (v1.37.0): Extracting archive
- Installing symfony/uid (v8.0.9): Extracting archive
- Installing symfony/routing (v8.0.9): Extracting archive
- Installing symfony/process (v8.0.8): Extracting archive
- Installing symfony/polyfill-php86 (v1.37.0): Extracting archive
- Installing symfony/polyfill-php85 (v1.37.0): Extracting archive
- Installing symfony/polyfill-php84 (v1.37.0): Extracting archive
- Installing symfony/polyfill-intl-idn (v1.37.0): Extracting archive
- Installing symfony/mime (v8.0.9): Extracting archive
- Installing psr/event-dispatcher (1.0.0): Extracting archive
- Installing symfony/event-dispatcher-contracts (v3.6.0): Extracting archive
- Installing symfony/event-dispatcher (v8.0.9): Extracting archive
- Installing psr/log (3.0.2): Extracting archive
- Installing egulias/email-validator (4.0.4): Extracting archive
- Installing symfony/mailer (v8.0.8): Extracting archive
- Installing symfony/error-handler (v8.0.8): Extracting archive
- Installing symfony/http-kernel (v8.0.8): Extracting archive
- Installing symfony/finder (v8.0.8): Extracting archive
- Installing ramsey/collection (2.1.1): Extracting archive
- Installing brick/math (0.14.8): Extracting archive
- Installing ramsey/uuid (4.9.2): Extracting archive
- Installing psr/simple-cache (3.0.0): Extracting archive
- Installing symfony/translation-contracts (v3.6.1): Extracting archive
- Installing symfony/translation (v8.0.8): Extracting archive
- Installing psr/clock (1.0.0): Extracting archive
- Installing symfony/clock (v8.0.8): Extracting archive
- Installing carbonphp/carbon-doctrine-types (3.2.0): Extracting archive
- Installing nesbot/carbon (3.11.4): Extracting archive
- Installing monolog/monolog (3.10.0): Extracting archive
- Installing league/uri-interfaces (7.8.1): Extracting archive
- Installing league/uri (7.8.1): Extracting archive

Command Prompt

- Installing league/mime-type-detection (1.16.0): Extracting archive
- Installing league/flysystem-local (3.31.0): Extracting archive
- Installing league/flysystem (3.33.0): Extracting archive
- Installing nette/utils (v4.1.3): Extracting archive
- Installing nette/schema (v1.3.5): Extracting archive
- Installing dflydev/dot-access-data (v3.0.3): Extracting archive
- Installing league/config (v1.2.0): Extracting archive
- Installing league/commonmark (2.8.2): Extracting archive
- Installing laravel/serializable-closure (v2.0.13): Extracting archive
- Installing laravel/prompts (v0.3.17): Extracting archive
- Installing laravel/framework (v13.7.0): Extracting archive
- Installing laravel/pail (v1.2.6): Extracting archive
- Installing laravel/agent-detector (v2.0.2): Extracting archive
- Installing laravel/pao (v1.0.6): Extracting archive
- Installing laravel/pint (v1.29.1): Extracting archive
- Installing nikic/php-parser (v5.7.0): Extracting archive
- Installing psy/psysh (v0.12.22): Extracting archive
- Installing laravel/tinker (v3.0.2): Extracting archive
- Installing hamcrest/hamcrest-php (v2.1.1): Extracting archive
- Installing mockery/mockery (1.6.12): Extracting archive
- Installing filp/whoops (2.18.4): Extracting archive
- Installing nunomaduro/collision (v8.9.4): Extracting archive
- Installing staabm/side-effects-detector (1.0.5): Extracting archive
- Installing sebastian/version (6.0.0): Extracting archive
- Installing sebastian/type (6.0.3): Extracting archive
- Installing sebastian/recursion-context (7.0.1): Extracting archive
- Installing sebastian/object-reflector (5.0.0): Extracting archive
- Installing sebastian/object-enumerator (7.0.0): Extracting archive
- Installing sebastian/global-state (8.0.2): Extracting archive
- Installing sebastian/exporter (7.0.2): Extracting archive
- Installing sebastian/environment (8.1.0): Extracting archive
- Installing sebastian/diff (7.0.0): Extracting archive
- Installing sebastian/comparator (7.1.6): Extracting archive
- Installing sebastian/cli-parser (4.2.0): Extracting archive
- Installing phpunit/php-timer (8.0.0): Extracting archive
- Installing phpunit/php-text-template (5.0.0): Extracting archive
- Installing phpunit/php-invoker (6.0.0): Extracting archive
- Installing phpunit/php-file-iterator (6.0.1): Extracting archive

```

Command Prompt
- Installing theseer/tokenizer (2.0.1): Extracting archive
- Installing sebastian/lines-of-code (4.0.0): Extracting archive
- Installing sebastian/complexity (5.0.0): Extracting archive
- Installing phpunit/php-code-coverage (12.5.6): Extracting archive
- Installing phar-io/version (3.2.1): Extracting archive
- Installing phar-io/manifest (2.0.4): Extracting archive
- Installing myclabs/deep-copy (1.13.4): Extracting archive
- Installing phpunit/phpunit (12.5.24): Extracting archive
57 package suggestions were added by new dependencies, use `composer suggest` to see details.
Generating optimized autoload files
> Illuminate\Foundation\ComposerScripts::postAutoloadDump
> @php artisan package:discover --ansi

  INFO  Discovering packages.

laravel/pail ..... DONE
laravel/pao ..... DONE
laravel/tinker ..... DONE
nesbot/carbon ..... DONE
nunomaduro/collision ..... DONE
nunomaduro/termwind ..... DONE

78 packages you are using are looking for funding.
Use the `composer fund` command to find out more!
> @php artisan vendor:publish --tag=laravel-assets --ansi --force

  INFO  No publishable resources for tag [laravel-assets].

No security vulnerability advisories found.
> @php artisan key:generate --ansi

  INFO  Application key set successfully.

> @php -r "file_exists('database/database.sqlite') || touch('database/database.sqlite');"
> @php artisan migrate --graceful --ansi

  INFO  Preparing database.

Creating migration table ..... 25.36ms DONE

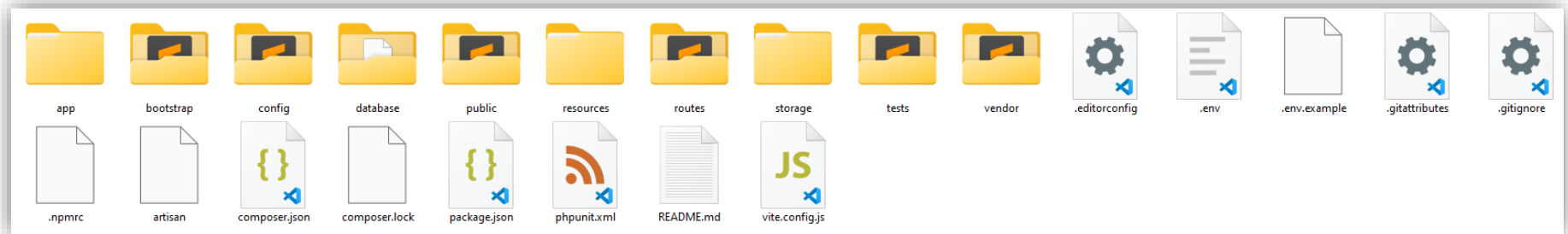
  INFO  Running migrations.

0001_01_01_000000_create_users_table ..... 36.65ms DONE
0001_01_01_000001_create_cache_table ..... 21.01ms DONE
0001_01_01_000002_create_jobs_table ..... 23.91ms DONE

D:\>

```

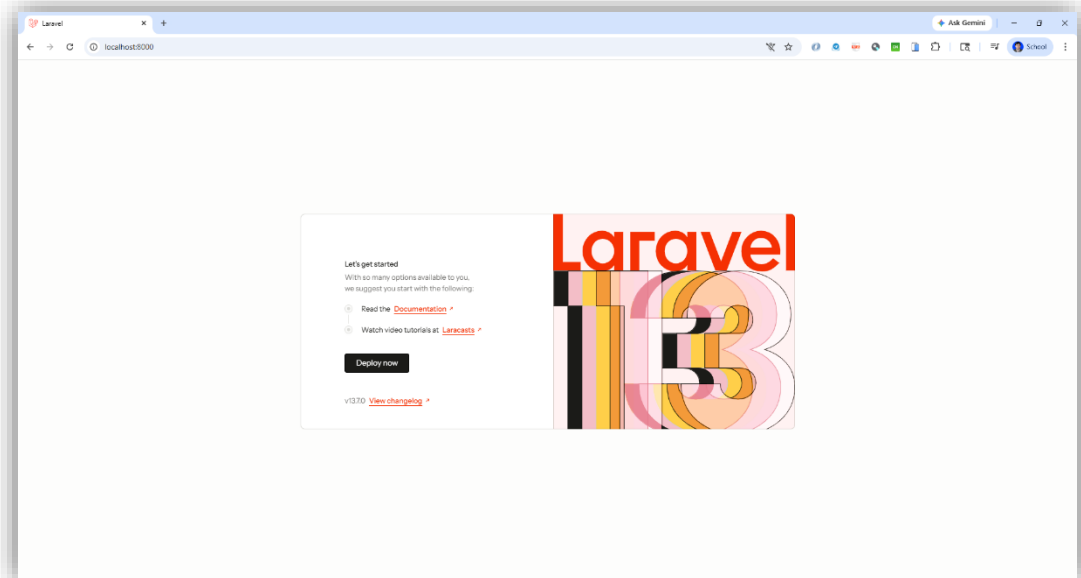
Gambar diatas menggambarkan proses download project **Laravel 13**. Jika berhasil maka terdapat sebuah folder yang bernama **api_lav13** pada **drive D** dan didalam folder tersebut akan terdapat struktur **folder** dan **file** seperti gambar dibawah ini



- Langkah berikutnya adalah **mencoba menjalankan project laravel dengan Local Development Server bawaan Laravel 13**. Silahkan buka **Windows Command Prompt** (cara seperti Langkah 3) pada path **D:\api_lav13**. Kemudian ketik perintah **php artisan serve**. Jika berhasil, maka akan menghasilkan seperti gambar disamping. Gambar disamping menunjukkan bahwa **Local Development Server** sedang aktif (running) dengan url akses **localhost** atau **127.0.0.1** dengan port **8000**. Jika gagal, **biasanya terdapat peringatan blocked oleh antivirus**, silahkan untuk membuat **pengaturan pengecualian (Exception)** pada antivirusnya. Jika berhasil untuk mengaksesnya silahkan buka browser kemudian ketik alamat **localhost:8000** atau **127.0.0.1:8000**. Jika berhasil akan muncul seperti gambar disamping pada browser. Sampai titik ini, maka project **Laravel 13** telah berhasil dijalankan dengan Baik.

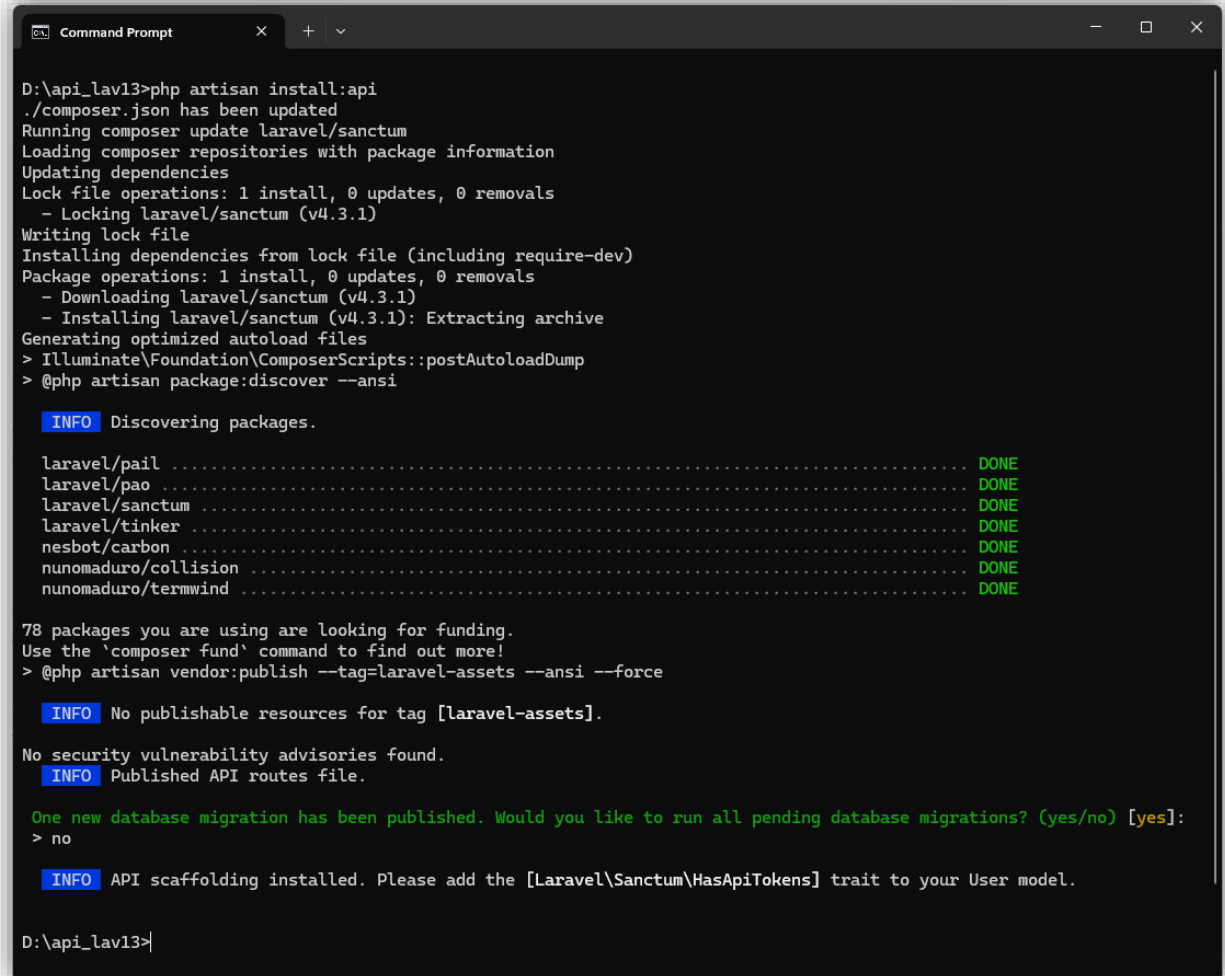
```
Command Prompt - php artis x + - □ x
D:\api_lav13>php artisan serve
INFO Server running on [http://127.0.0.1:8000].
Press Ctrl+C to stop the server
```

- Langkah berikutnya adalah **meng-install fitur API di Laravel 13**. **Fitur API** merupakan salah satu fitur pada Laravel yang digunakan untuk menangani konsep integrasi sistem/data dengan menggunakan API. Pada



versi **Laravel 10**, fitur ini telah terinstall secara otomatis akan tetapi pada versi **Laravel 11** keatas fitur ini harus diinstall secara terpisah. Untuk menginstall-nya kita menggunakan perintah **PHP artisan**. Silahkan tulis perintah **php artisan install:api** jika berhasil akan menghasilkan tampilan seperti gambar disamping (sebagai catatan Windows Command Prompt berada pada posisi path **D:\api_lav13**). Saat proses instalasi API, jika ada pertanyaan terkait **migrasi database** dijawab **no** saja. Jika berhasil, maka didalam folder **routes** terdapat sebuah file bernama **api.php**. File **api.php** ini digunakan untuk **menyimpan konfigurasi routing** terkait **API** yang nantinya akan dibangun.

- Langkah berikutnya adalah **membuat routing untuk API sederhana**. Jika Route API ini diakses, akan menghasilkan **respon berupa JSON** atau respon lain sesuai pengaturan yang diinginkan. Silahkan modifikasi file **api.php** yang ada di folder **routes** hingga menjadi seperti gambar berikut



```
D:\api_lav13>php artisan install:api
./composer.json has been updated
Running composer update laravel/sanctum
Loading composer repositories with package information
Updating dependencies
Lock file operations: 1 install, 0 updates, 0 removals
  - Locking laravel/sanctum (v4.3.1)
Writing lock file
Installing dependencies from lock file (including require-dev)
Package operations: 1 install, 0 updates, 0 removals
  - Downloading laravel/sanctum (v4.3.1)
  - Installing laravel/sanctum (v4.3.1): Extracting archive
Generating optimized autoload files
> Illuminate\Foundation\ComposerScripts::postAutoloadDump
> @php artisan package:discover --ansi

 INFO  Discovering packages.

laravel/pail ..... DONE
laravel/pao ..... DONE
laravel/sanctum ..... DONE
laravel/tinker ..... DONE
nesbot/carbon ..... DONE
nunomaduro/collision ..... DONE
nunomaduro/termwind ..... DONE

78 packages you are using are looking for funding.
Use the 'composer fund' command to find out more!
> @php artisan vendor:publish --tag=laravel-assets --ansi --force

 INFO  No publishable resources for tag [laravel-assets].

No security vulnerability advisories found.
 INFO  Published API routes file.

One new database migration has been published. Would you like to run all pending database migrations? (yes/no) [yes]:
> no

 INFO  API scaffolding installed. Please add the [Laravel\Sanctum\HasApiTokens] trait to your User model.

D:\api_lav13>
```

```
<?php
use Illuminate\Http\Request;
use Illuminate\Support\Facades\Route;
use App\Http\Controllers\Utama;

Route::get('/user', function (Request $request) {
    return $request->user();
})->middleware('auth:sanctum');

Route::get("cekstatus", [Utama::class, "statusAPI"]);
```

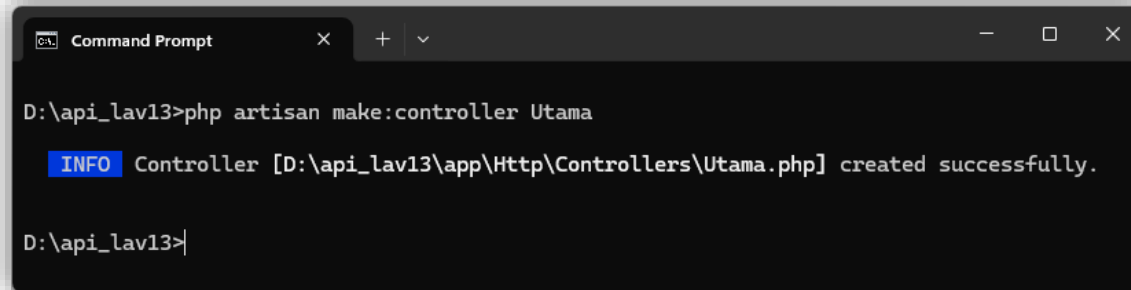
Berdasarkan gambar diatas, terdapat 2 syntax tambahan yang ditempatkan pada file **api.php** yaitu

- **Use App\Http\Controllers\Utama;**
Syntax ini digunakan agar file **api.php** dapat mengakses controller yang bernama **Utama.php**
- **Route::get("cekstatus", [Utama::class, "statusAPI"]);**
Syntax ini merupakan pengaturan routing API dilaravel. Jika ada yang menuliskan keyword **cekstatus** dengan metode **get** ketika mengakses melalui url, tools API, syntax API Handler, dll maka Laravel akan mengakses controller **Utama** pada function **statusAPI**.

8. Langkah berikutnya adalah **membuat controller yang digunakan untuk menangani respon API yang telah dibuat pada langkah sebelumnya**. Untuk membuat controller dapat menggunakan 2 cara yaitu

- Cara Manual
Dengan cara ini kita membuat file controller dengan ekstensi **.php** yang ditempatkan di folder **app – Http – Controllers**. Untuk syntax didalamnya kita harus menulisnya secara konvensional.
- Menggunakan perintah PHP Artisan
Dengan cara ini kita menggunakan **aplikasi berbasis Terminal** pada path **root folder** project **Laravel 13** dengan menuliskan perintah **php artisan make:controller nama_controller_yang_diinginkan**. Jika berhasil, maka akan terbuat sebuah **file php** yang berfungsi sebagai controller di folder **app – Http – Controllers**. Adapun syntax controller-nya akan tertulis secara otomatis akan tetapi sebatas syntax dasar controller saja sehingga kita tetap harus menyesuaikannya kembali.

Sebagai contoh pada praktikum kali ini, cara yang digunakan adalah **menggunakan PHP Artisan**. Silahkan mengakses **root folder** project **Laravel 13** menggunakan **Windows Command Prompt**. Kemudian ketik perintah **php artisan make:controller Utama** seperti gambar dibawah.



```
Command Prompt
D:\api_lav13>php artisan make:controller Utama

INFO Controller [D:\api_lav13\app\Http\Controllers\Utama.php] created successfully.

D:\api_lav13>
```

```
<?php

namespace App\Http\Controllers;

use Illuminate\Http\Request;

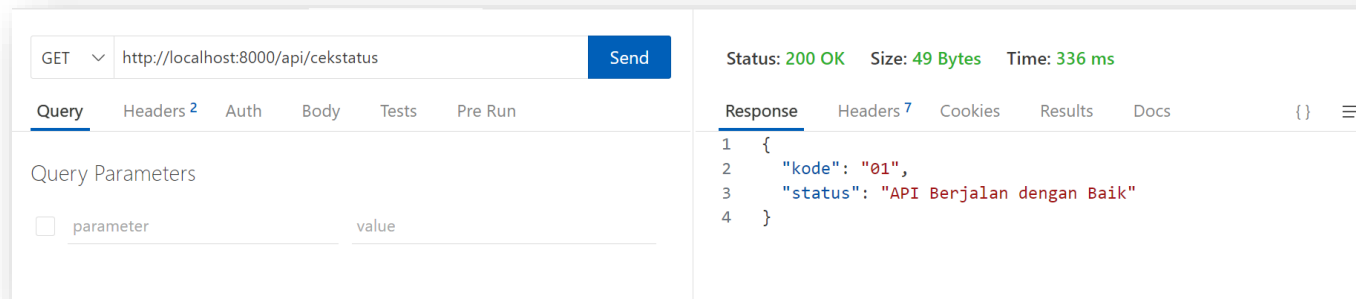
class Utama extends Controller
{
    //
}
```

Gambar diatas sebelah kiri menandakan bahwa file controller yang bernama **Utama.php** telah berhasil terbuat di folder **app – Http – Controllers**. Syntax standar controller **Utama.php** yang berhasil dibuat adalah seperti gambar diatas sebelah kanan.

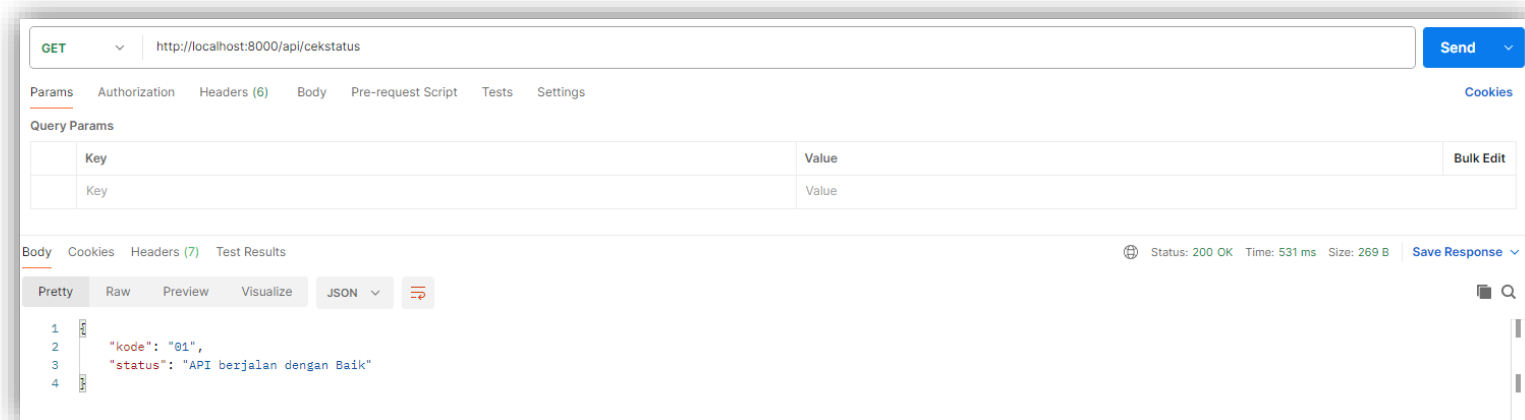
- Langkah berikutnya adalah **memodifikasi controller Utama.php**. Modifikasi dilakukan dengan menambah **function** bernama **statusAPI** (sesuai yang telah dikonfigurasi-kan pada pengaturan routing API pada langkah 7). Function tersebut digunakan untuk **mengembalikan respon berupa JSON dengan 2 key yaitu kode dan status**. Silahkan modifikasi file controller **Utama.php** hingga syntax-nya menjadi sebagai berikut

```
<?php
namespace App\Http\Controllers;
use Illuminate\Http\Request;
class Utama extends Controller{
    public function statusAPI(){
        $dtx = ["kode"=>"01", "status"=>"API berjalan dengan Baik"];
        return response()->json($dtx, 200);
    }
}
```

10. Untuk melakukan **testing terhadap API**, dibutuhkan software khusus seperti **Postman**, **Restman** (extensi dari Google Chrome), **Thunder Client** (extensi dari Visual Studio Code), dll. Silahkan instal terlebih dahulu salah satu software tersebut. Sebagai informasi tambahan pengujian API juga dipengaruhi oleh **metode dalam me-request ke API**. Jika metode yang digunakan dikonfigurasi sebagai **GET** maka dapat menggunakan browser secara langsung. Tetapi jika metode yang digunakan dikonfigurasi selain **GET** seperti **POST**, **PUT**, dll maka dapat menggunakan tools khusus yang telah dijelaskan sebelumnya.
11. Langkah berikutnya adalah menjalankan **Local Development Server** seperti langkah ke 5.
12. Langkah berikutnya adalah **menguji API yang menggunakan konfigurasi routing cekstatus**. Jika kita melihat **API cekstatus** menggunakan method **GET** untuk mengaksesnya. Sehingga kita bisa mengujinya seperti gambar berikut
 - Menggunakan **Thunder Client**



- Menggunakan **Postman**



- Menggunakan Browser **Google Chrome**



Beberapa percobaan diatas menunjukkan kondisi **respon API** dengan kategori berhasil (**status 200 OK** serta **JSON** tampil sebagai respon yang diharapkan).

13. Sampai langkah ini, praktikum dianggap berhasil karena sudah bisa untuk mengakses **API** secara normal. Silahkan tanya pengajar jika terdapat masalah atau hal lainnya.