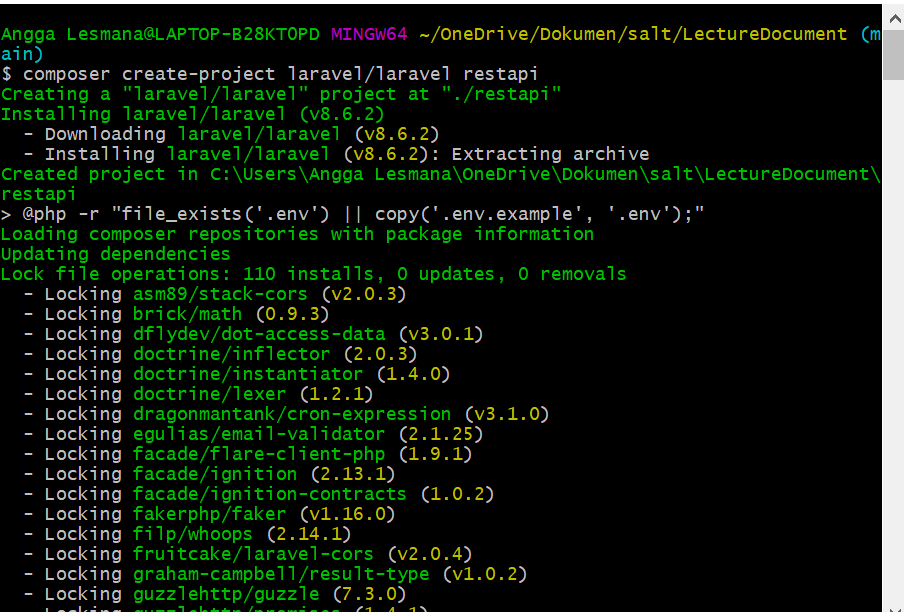
Membuat restAPI, mengunakan Laravel8

* Buat Laravel project
* Edit .env dan buat migration di Laravel
* Buat router
* Buat controller
* Test restAPI di Postman
* Buat documentasi restapi postman

1. **Memulai project pembuatan Laravel project menggunakan composer**

Cara membuat suatu project Laravel menggunakan composer

Sebelum membuat perintah dengan composer harus install composer terlebih dahulu

Nah jikalau sudah install composer , lalu ketikan perintah ini di cmd/terminal **

<https://laravel.com/docs/8.x#installation-via-composer>

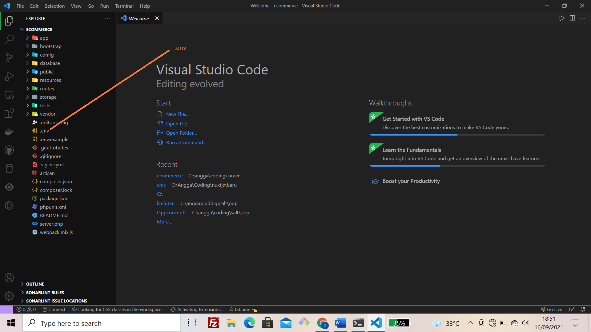
-. *composer create-project Laravel/Laravel*

Harus dengan ke adaan online agar bisa create project composer

Sekarang sudah jadi, terus tinggal masuk ke folder restapi yang barusan udah kita buat

1. **Edit .env**

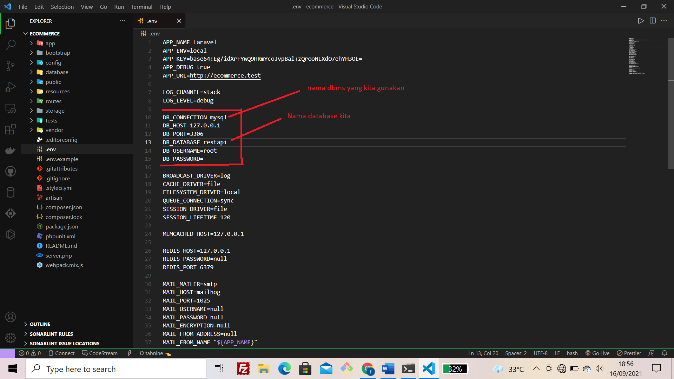
Ah oke setelah kita membuat project Laravel, kita akan coba mengedit .env hal yang pertama kita edit,

**

kita setting untuk memasukan nama database dan jenis dbms yang di gunakan,

Tapi sebelumnya kita harus buat dlu databasenya yah,

Disini saya coba pake dbms mysql



Nah setelah kita sudah edit .env . sekarang kita pastikan dulu mau buat restAPI tentang apa,

Untuk contoh kasus saya saat ini. Saya akan membuat restapi mengenai student pringkat dan kelas

Berarti table apa aja nih yang cocok untuk database student, oke deh untuk contohnya saya akan buat beberapa table di antaranya

1. Table student
2. Table ranking
3. Table kelas
4. Table absensi

Nah jikalau kita sudah punya rencana untuk membuat suatu databasenya ,

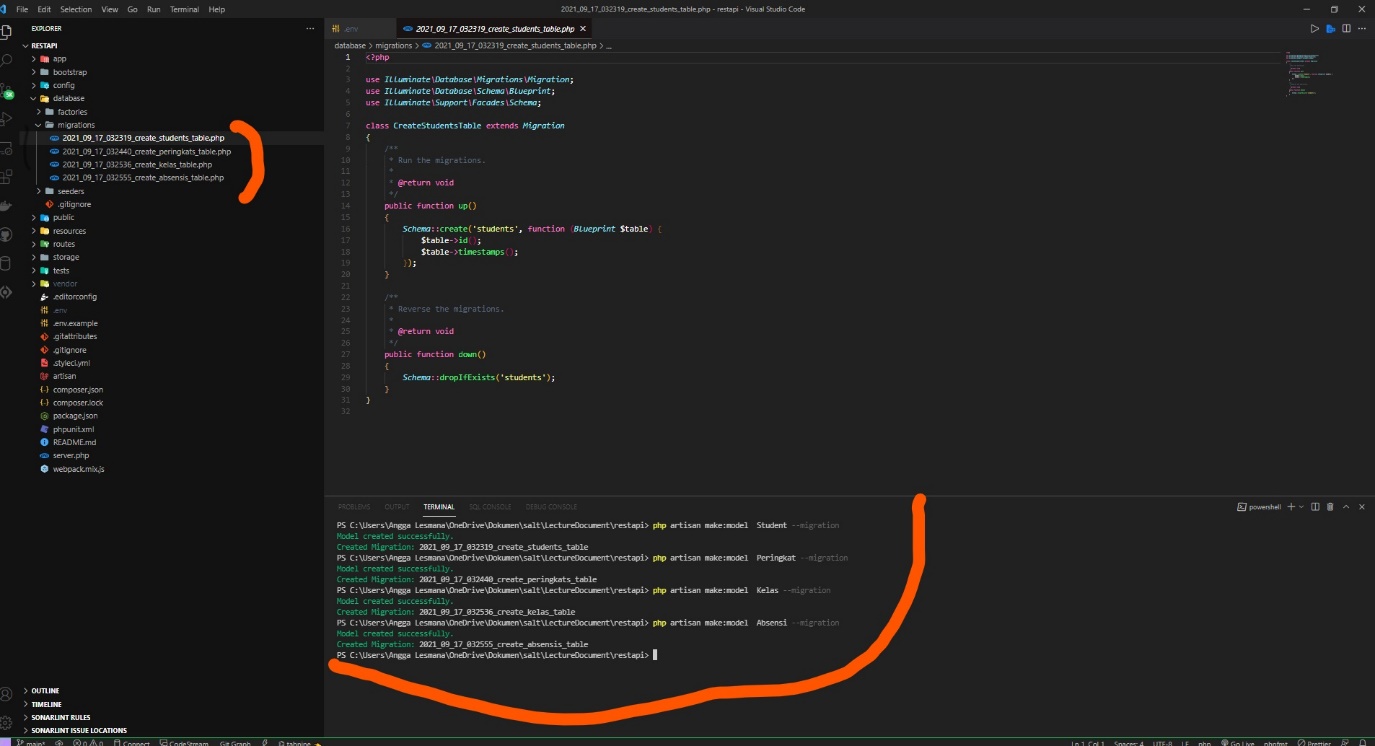
kita tidak harus lagi untuk membuat table manual dari mysql atau phpMyAdmin

Kita cukup buatkan suatu migration studentnya

Dari Laravel sudah tersedia yang Namanya php artisan make / migration atau migrate

Cukup dengan perintah *php artisan make:migration nametable*

Example



Nah Kita sudah buat suatu data table ke dalam database kita , bisa di lihat yang dilingkari orange

Itu suatu perintah pembuatan **model && migration** di Laravel,

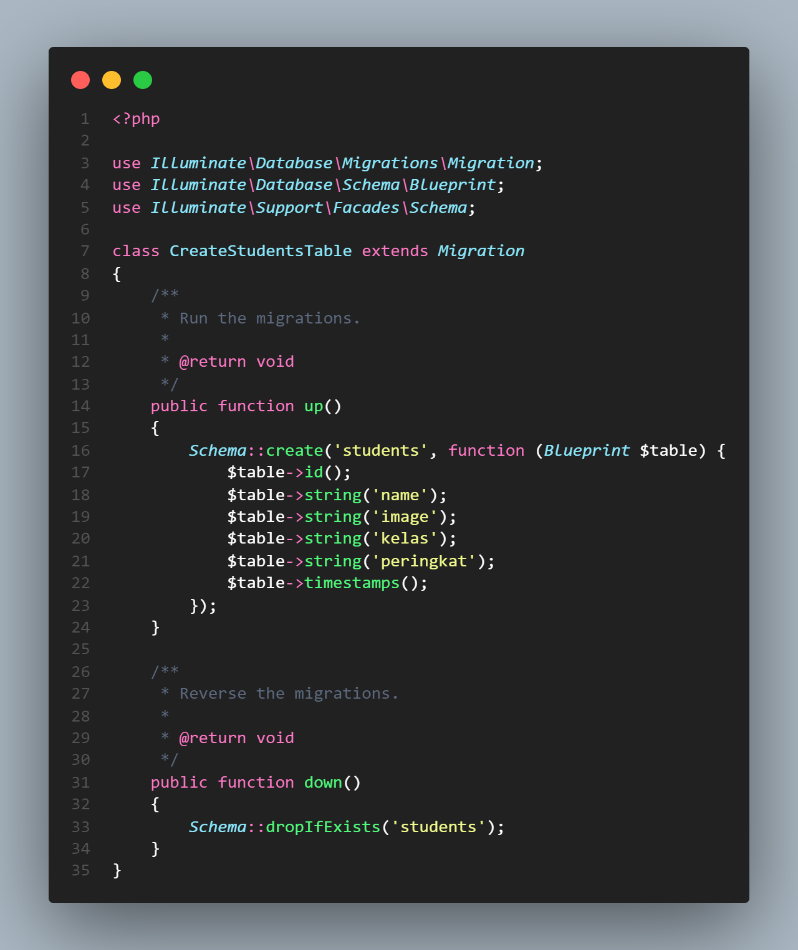
Maksudnya apa ? Maksudnya jadi kita tidak harus membuat dua kali, contoh, kita mau buat migration nih ke databases berarti kita harus buat dulu *php artisan make:migration namatable*

Nah terus setalah jadi kita mau menambahakan/input data kedalam database melalui controller yang akan kita buat nanti. Berarti kita butuh modelnya dong? Terus kita buat lagi model nya menggunakan , ***php artisan make:model namamodel***,

Nah jikalau seperti itu pasti ribet kan? Nah maka dari itu disini saya menggunakan perintah seperti di atas yang dimana langsung di gabungkan antara model dan migration

Nah setelah itu kita buat dulu table studentnya itu apa aja

Untuk contoh kita akan membuat table pada mysql kita melalui migration Laravel



Nama Field / Value

Nama Table

**3, Membuat router**

Nah oke setelah kita mengedit .env dan membuat suatu database menggunakan migration Laravel, kita akan membuat suatu Router,

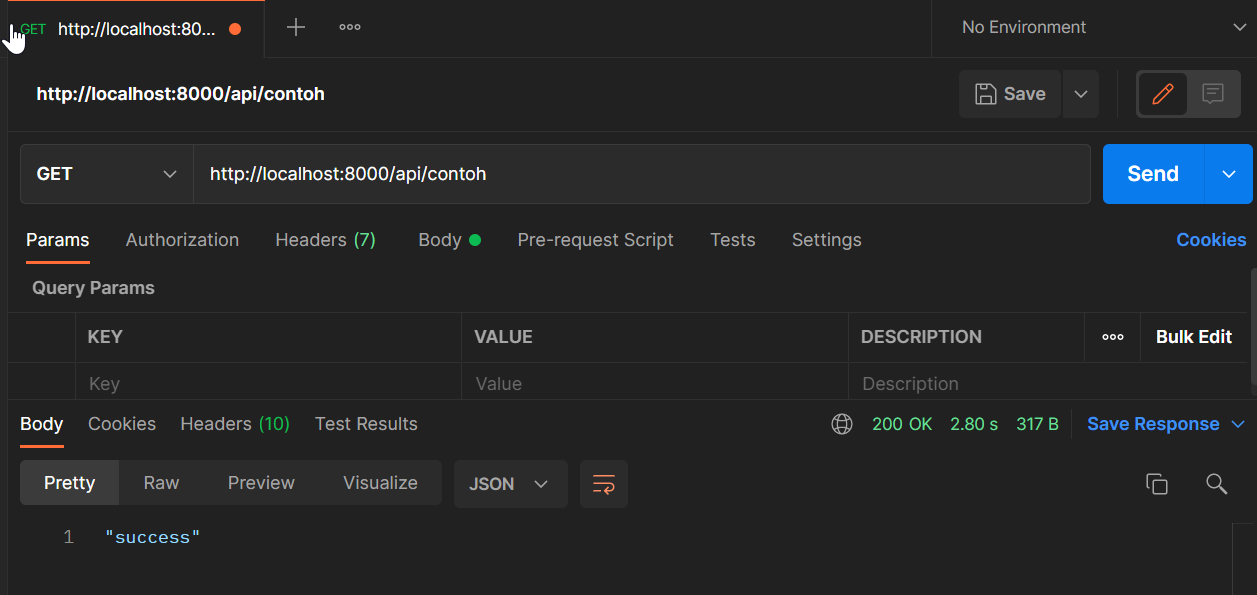
Untuk contoh saya buatkan suatu router sederhana untuk menampilkan hasil get yang kita buat contohnya

Dan mencobanya di postman

*Route*::get('/contoh', function(){

    return response()->json('success'); Nama Router

});

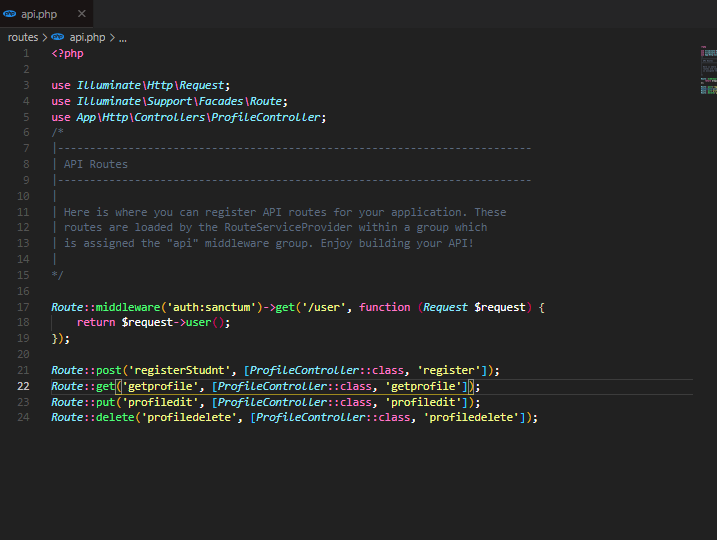


Hasil response yang udah kita buat

Router yang di atas yang sudah saya buat itu, di satu file router saja, tidak menggunakan metode mvc di Laravel, nah untuk menggunakan mvc kita akan pisah router dan controller,

Dengan router kita akan tau restAPI seperti apa yang akan kita buat,

oke untuk contoh saya buatkan Router seperti di bawah ini yang menggunakan metode mvc



Nama metode RestAPI yang akan kita gunakan nanti disini delete untuk mengapus

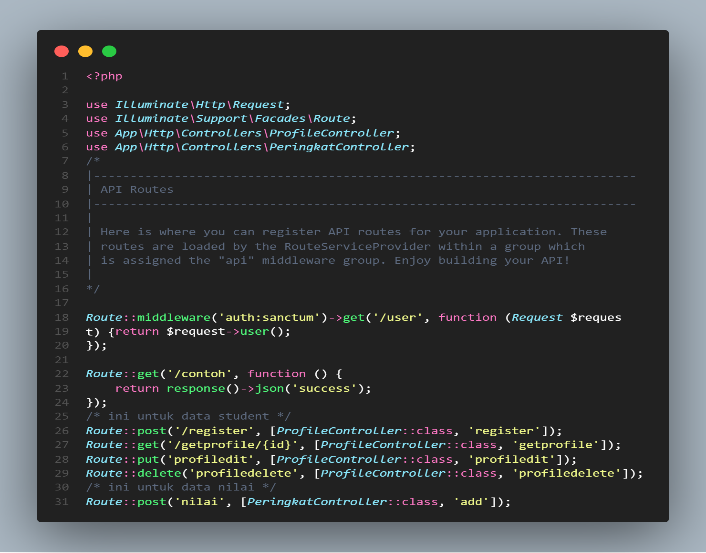
Nama Router RestAPI yang akan kita gunakan nanti

Nama function Controller yang akan kita buat nanti

Nama Class Controller yang akan kita buat nanti

Disini saya membuat router restAPI student terlebih dahulu,

Dan seterusnya pembuatan router sama aja yah guys kecuali nanti saya akan buatkan khusus middleware



Oke Kita suda buat suatu router yang dimana ada router khusus Profile dan data nilai

Dengan itu kita akan segera buatkan controller sesuai dengan apa yang ada di router seperti ProfileController dan PeringkatController

Untuk membuat suatu controller di Laravel kita bisa menggunakan artisan make

Cukup perintahkan di terminal kita dengan perintah

*Php artisan make:controller ProfileController*

Dan harus kita ketahui untuk penamaan controller di haruskan menggunakan CamelCase yah

Oke setelah kita buat ProfileController kita, kita langsung aja buatkan suatu logic untuk create dan read data kita

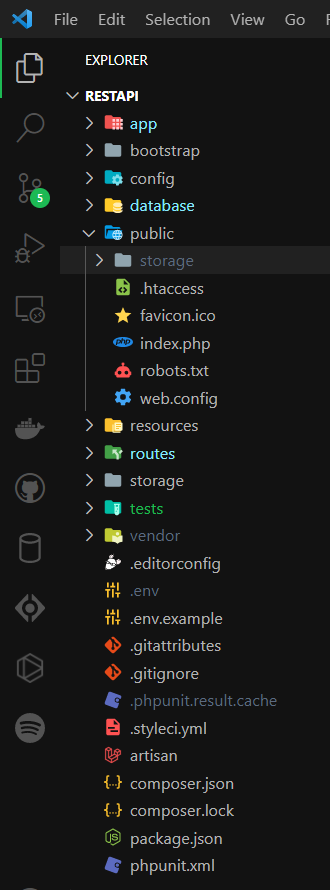


Penggunaan model (eloquent )

Get data by Id

Pemanggilan model

Disini kita akan coba belajar untuk membuat upload gambar,

**

Tempat penyimpanan file yang akan kita upload

Cara upload file/gambar

Untuk menggunakan file upload di Laravel

kita bisa create storage link use php

artisan,

storage ini tempat penyimpanan file

kita nanti , cukup perintahkan di terminal

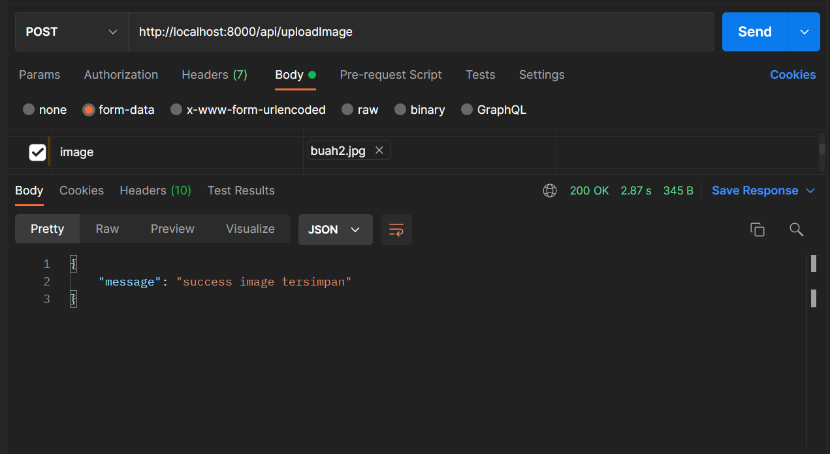
dengan perintah

*php artisan storage:link*

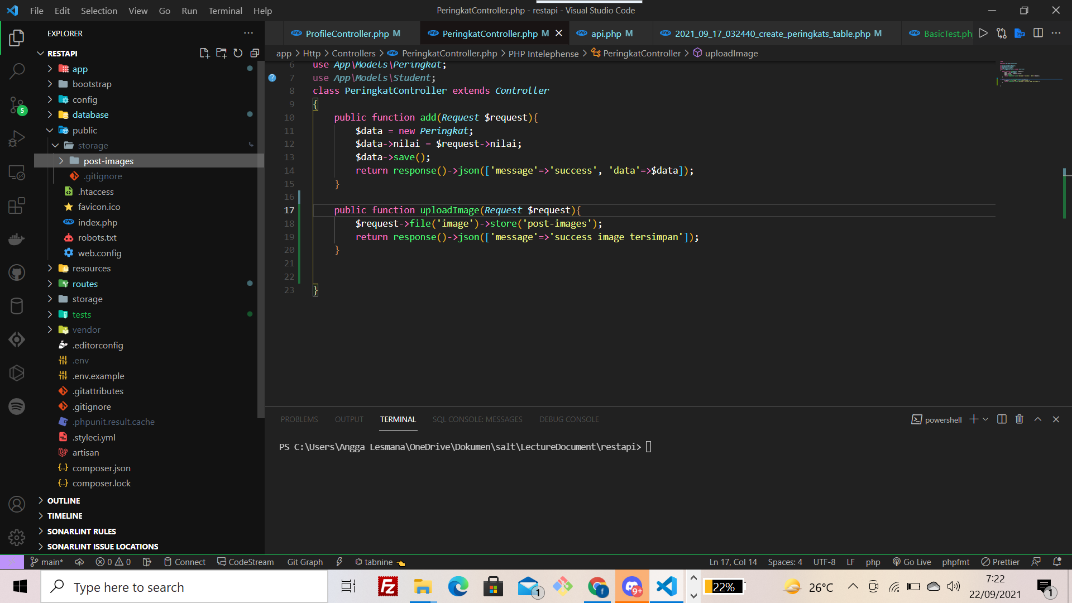
Setelah kita sudag ng generate folder storage ke public kita, sekarang kita akan coba buat upload image sederhana di controllernya



Nah Setelah kita buat controller seperti di atas kita akan coba di postman terlebih dahulu dengan value image dan format input body file



Nah setelah kita coba run di postman, otomatis akan kebuatkan file yang bernama post-images dan di dalamnya tesimpan image yang udah kita upload sebelumnya



Setelah kita coba upload file/image bisa, berarti kita tinggal masukan gambar ke database,, eh inget database gak nerima file yah, nah jadi disini database kita akan hanya nerima nama image nya saja dan imagenya akan tersimpan di local kita, yaitu di folder localstorage tadi,

Oke deh kita langsung aja buatkan full untuk input data ke databasenya beserta gambar juga



Nama function yang di panggil di router

Field / Value database

Variable untuk meng upload file

Variable untuk meng upload file

Variable untuk meng upload file

Nama Model yang udah kita buat

Nah kita sudah nih buat input data student ke database beserta gambarnya, selebihnya sama yah guys pembuatan controller, coba kalian buatkan aja seperti ini akan tetapi beda controller yah guys,

Sekarang kita tinggal coba pake di postman yah guys

