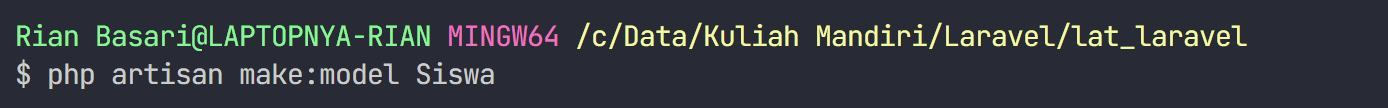
**MENAMPILKAN DATA DENGAN ELOQUENT LARAVEL9**

**Mengambil data dari database**

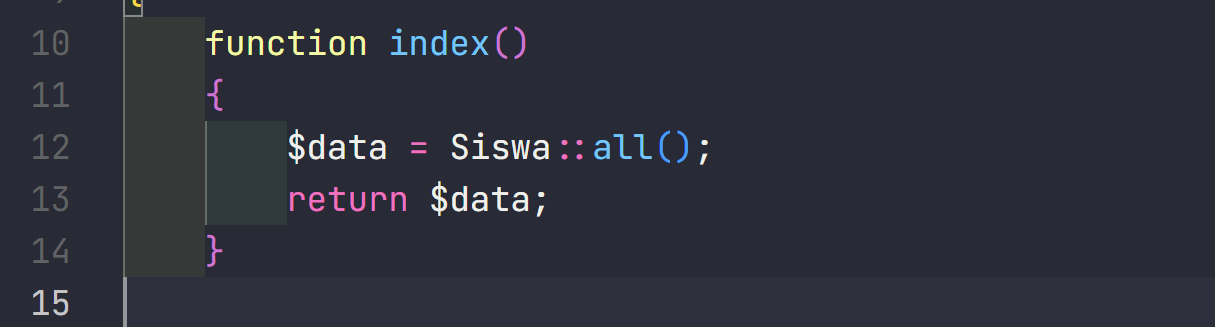
1. Buat model baru



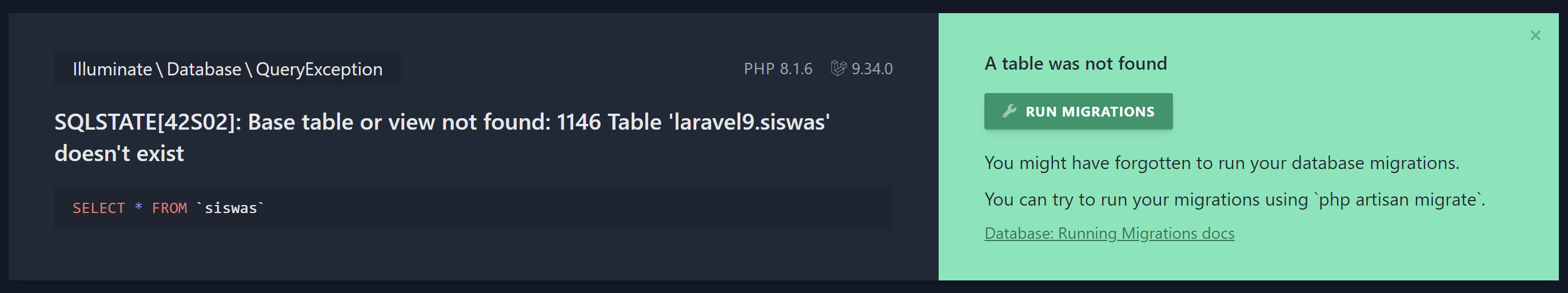
1. Pergi ke SiswaController, import model Siswa



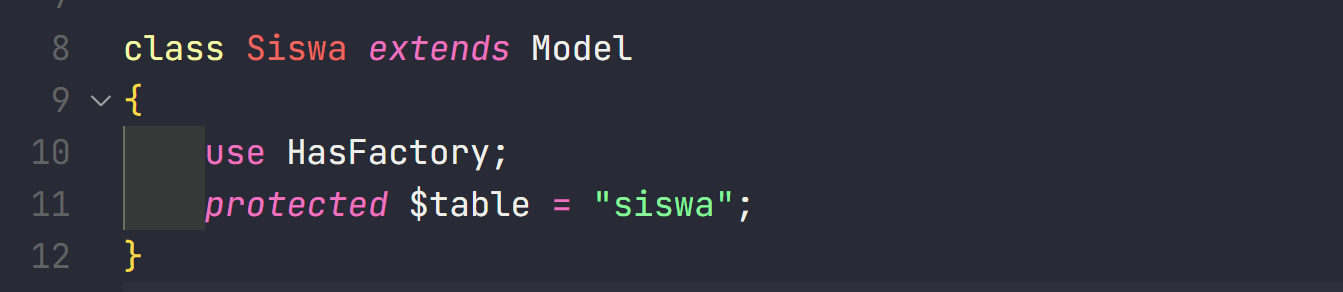
1. Buat variable di dalam method index untuk menampung data dari database.



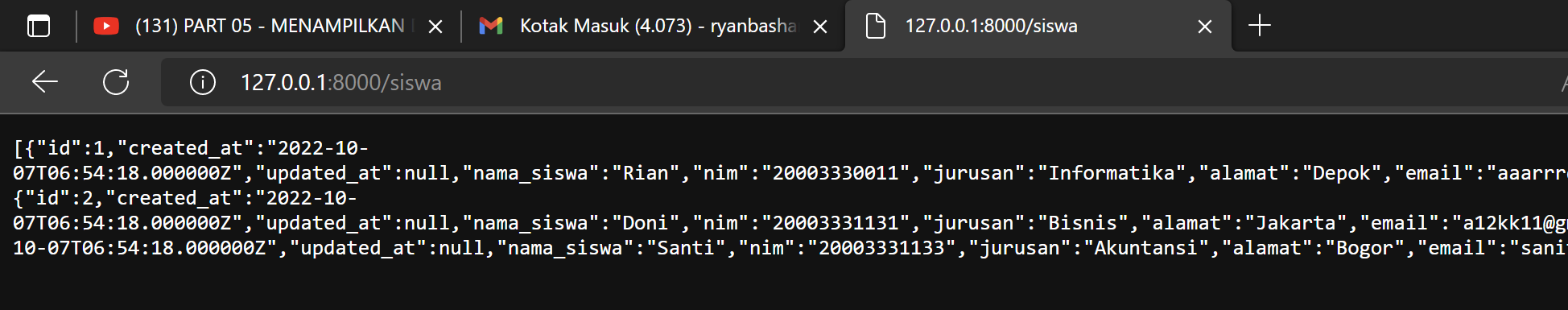
1. Jika dijalankan, akan muncul pesan error bahwa table tidak ditemukan. Ini karena table-nya berubah nama menjadi **“siswas”.**



1. Untuk mengubahnya, pergi ke model Siswa, dan tambahkan ini.

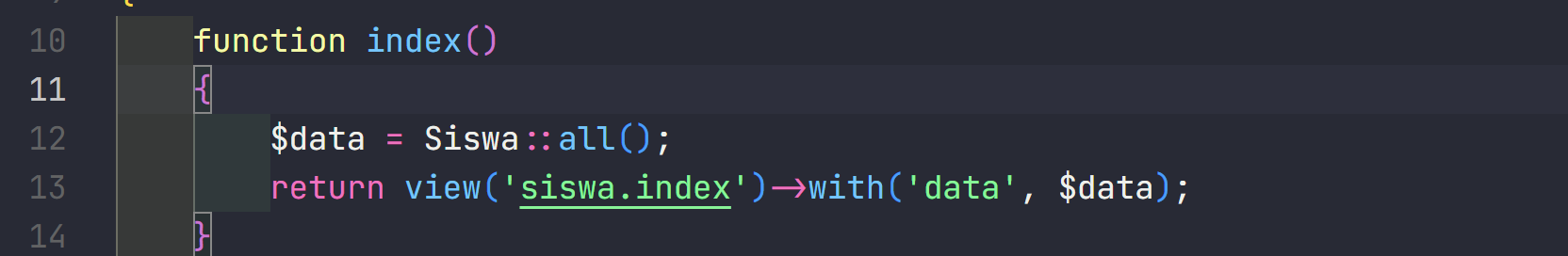


1. Jalankan aplikasi

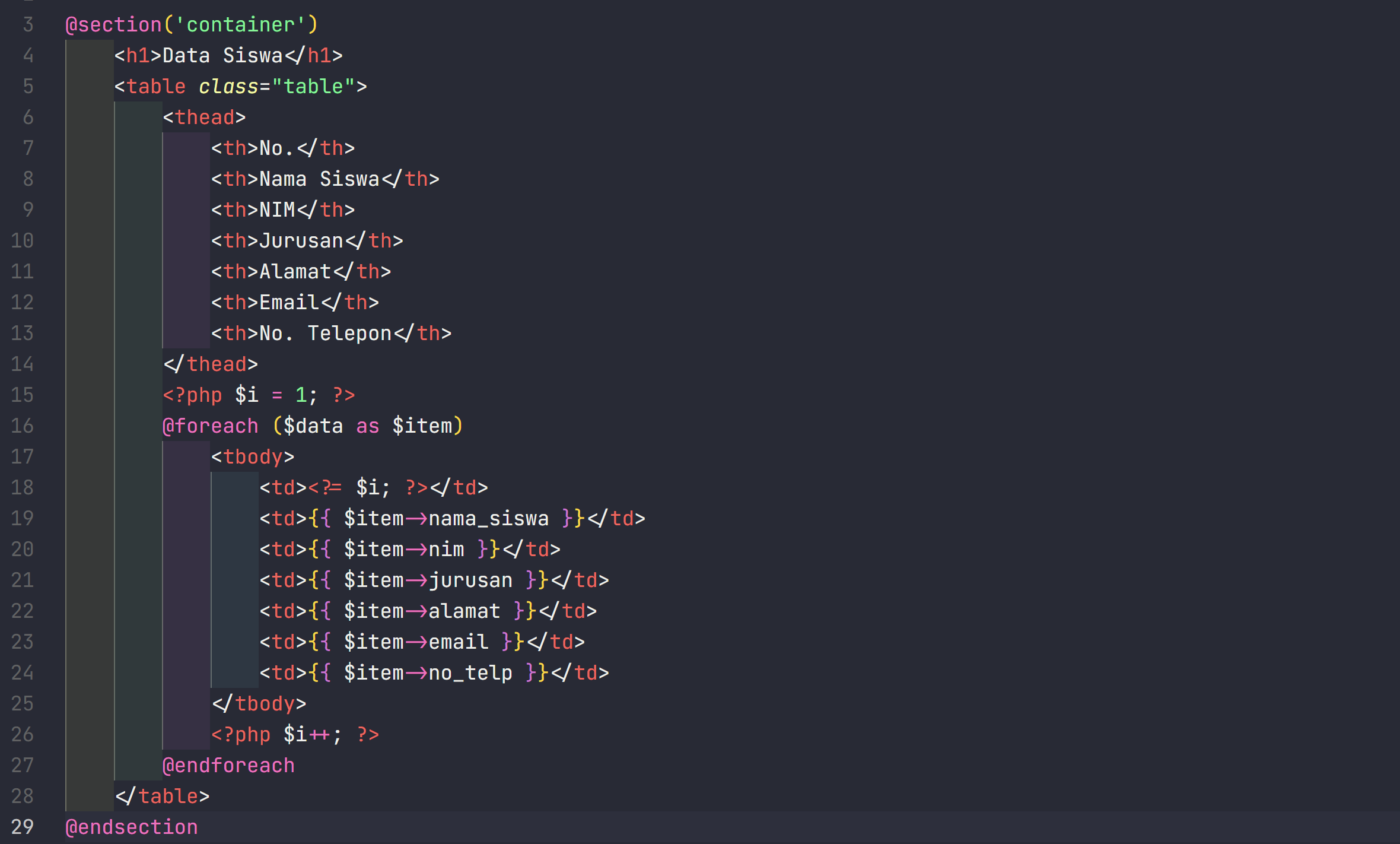


**Menampilkan Data dengan Eloquent**

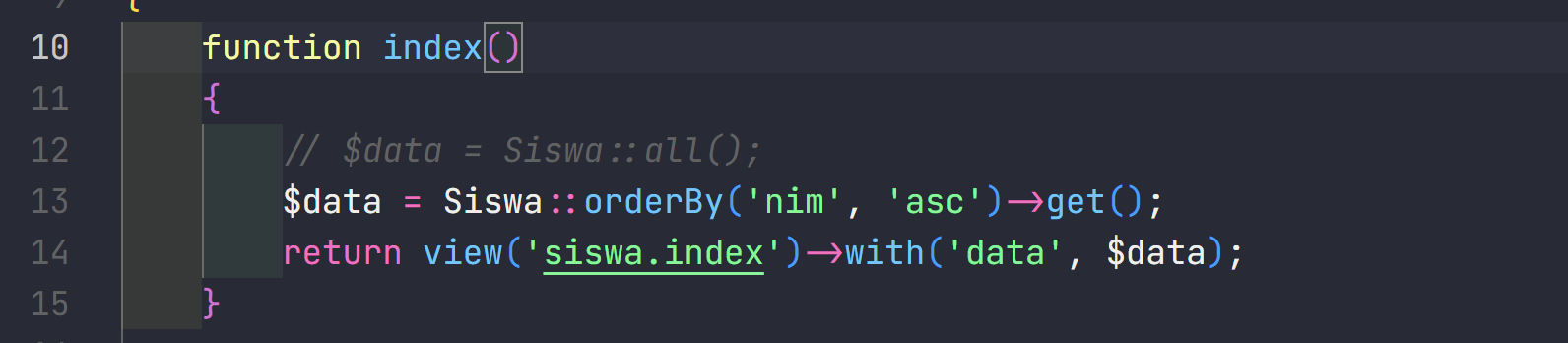
1. Buat view terlebih dahulu untuk menampilkan data dengan rapih (Menggunakan table)
2. Pergi ke SiswaController. Buat variable di dalam method index untuk mengambil data dari database, kemudian akan dikirim kan ke view yang sudah dibuat. ‘data’ artinya nama variable yang akan digunakan di view.



1. Isi data pada tabel pada view dengan data-data yang dibutuhkan dari database. Looping pada variable $data untuk menampilkan datanya.



1. Atau bisa juga menampilkan data dengan diurutkan berdasarkan field tertentu. (Contoh : Order By NIM)
2. Pergi ke SiswaController untuk mengubahnya

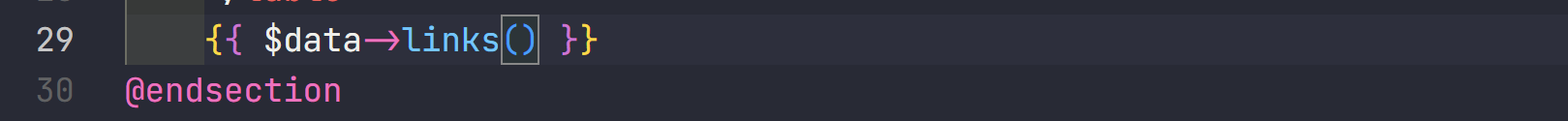


**Membuat Pagination**

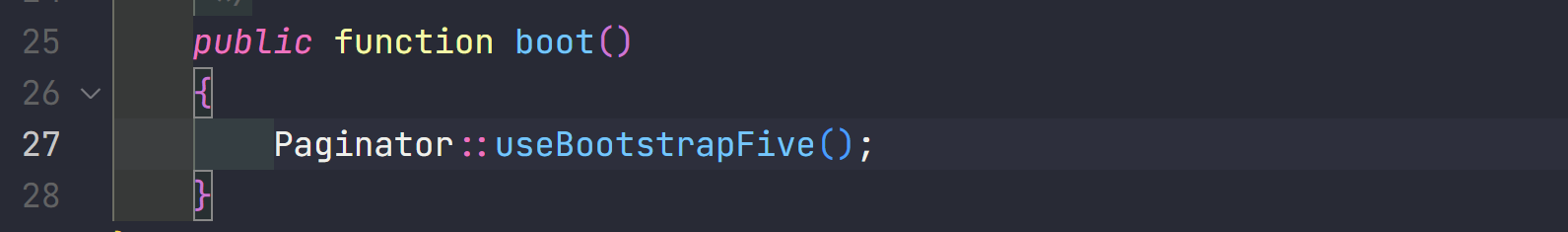
1. Untuk membuat pagination, ubah get() pada method index di SiswaController menjadi paginate(jumlah). Jumlah disini adalah seberapa banyak baris data akan ditampilkan dalam satu halaman.



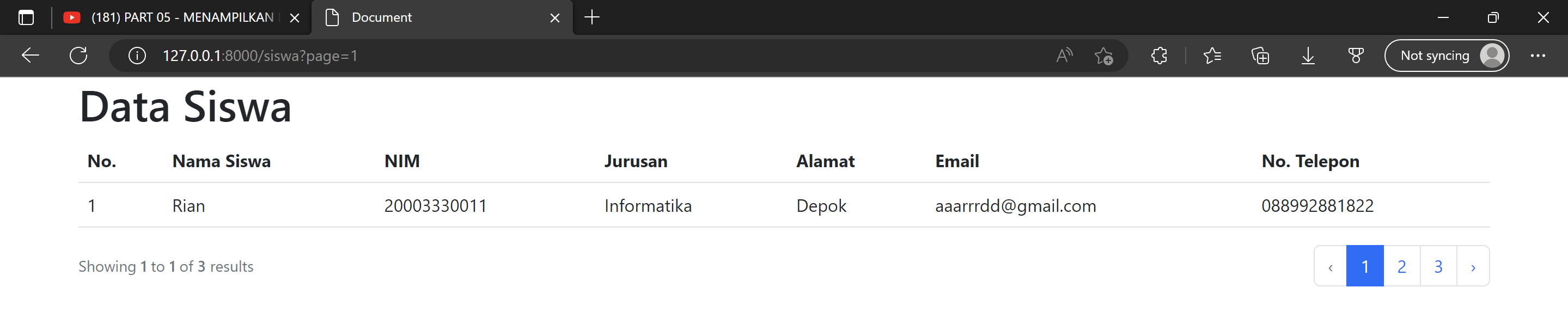
1. Tambahkan navigasi **(data->links())** untuk paging di view index siswa.



1. Jika dijalankan, akan muncul navigasi paging-nya dan berjalan dengan baik. Namun, tampilannya cukup berantakan. Kita bisa menggunakan navigasi paging dari Bootstrap.
2. Untuk menambahkan navigasi bootstrap, pergi ke **app/Services/AppServiceProvider.php**.
3. Isi pada bagian method **boot()**.



1. Buka kembali aplikasi, dan tampilan pagination berubah.

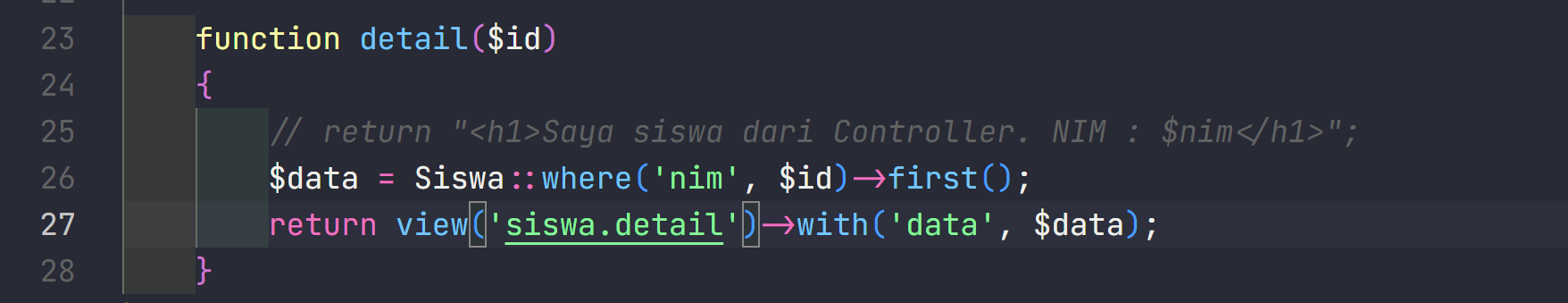


**Menambahkan fungsi Detail**

1. Pada halaman index siswa, tambahkan tombol untuk mengarahkan ke halaman detail.
2. Url pada halaman detail akan tergantung pada nim dari masing-masing siswa. (Contoh URL : 127.0.0.1:8000/siswa/20033118801)



1. Pergi ke SiswaController, kemudian ubah isi dari method detail().



Pada method detail, nantinya akan mengarahkan ke view detail

1. Buat View untuk detail siswa di dalam folder siswa.
2. Isi setiap data-data yang akan di tampilkan.

