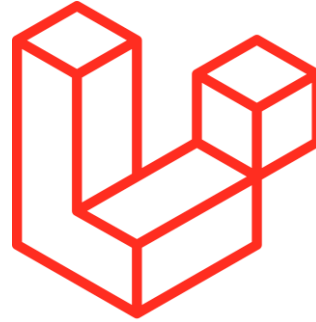


Pemrograman Backend



Pesantren PeTIK II YBM PLN

Jl. KH. Bisri Syansuri RT/01 RW/05, Plosogeneng,
Kec. Jombang, Kabupaten Jombang, Jawa Timur



10. Pagination, Filtering dan Export Data



Definisi Pagination

- Pagination merupakan adalah sebuah fitur web yang digunakan untuk membatasi tampilan data agar tidak terlalu panjang dan lebih rapi. Penggunaan pagination banyak dipergunakan digunakan untuk menampilkan data dalam jumlah banyak, sehingga dapat dipisah / dipilih berapa data yang akan ditampilkan terlebih dahulu sesuai dengan urutan akan kriteria pencarian tertentu.
- Pembuatan pagination seringkali merepotkan pengembangan web, terutama yang membangun situs webnya tanpa menggunakan framework web development. Banyak sekali proses manual yang harus dilakukan, serta melibatkan penggunaan HTML, CSS, PHP, dan JavaScript.
- Framework Laravel telah menyediakan fitur yang memungkinkan pengembang situs web yang menggunakan framework ini dapat dengan mudah membuat pagination dalam menampilkan data di situs webnya.



Membuat Pagination

Untuk membuat pagination pada sebuah halaman, contoh di halaman penerbit, panggil fungsi paginate() di sebuah fungsi, seperti tampak kode di bawah ini:

```
public function index()  
{  
    $ar_penerbit = DB::table('penerbit')->paginate(5);  
    return view('penerbit.index',compact('ar_penerbit'));  
}
```

paginate(5) berarti satu halaman menampung 5 baris data pada sebuah halaman.



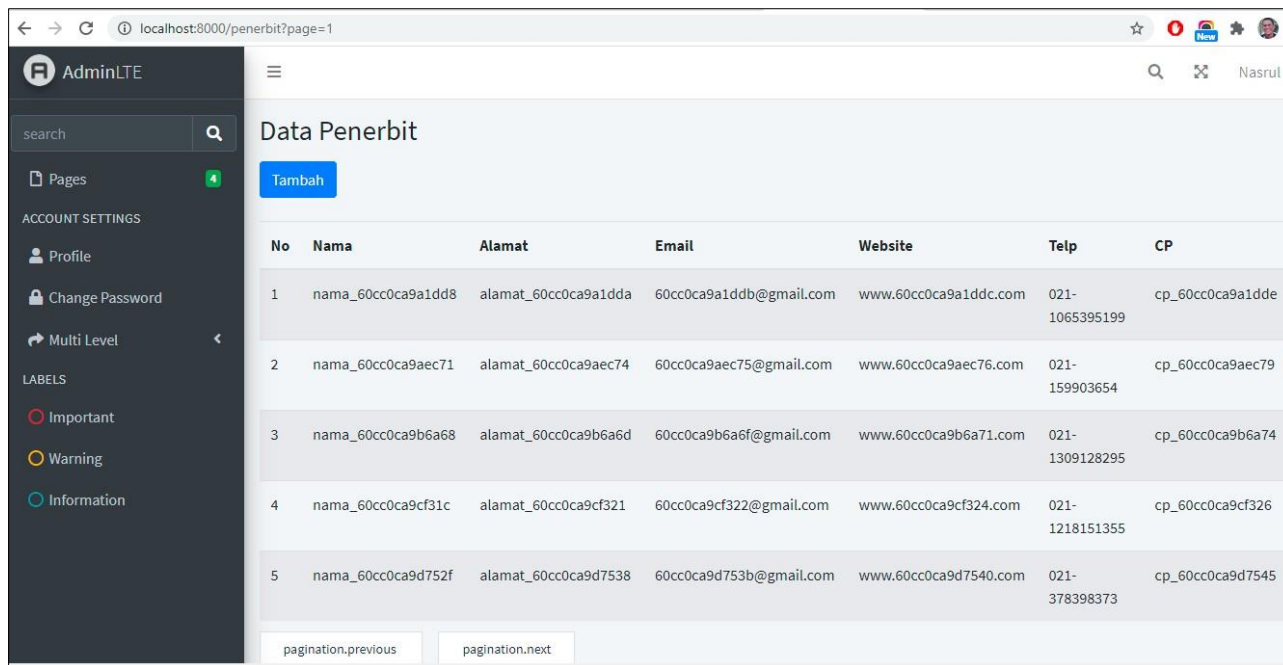
Membuat Link Pagination

Untuk membuat link pagination pada halaman penerbit, tambahkan fungsi `links()` di bawah data penerbit, seperti tampak kode di bawah ini:

```
resources > views > penerbit > index.blade.php > table.table.table-striped
15 <table class="table table-striped">|
16 <thead>
17 <tr>
18     @foreach($ar_judul as $jdl)
19         <th>{{ $jdl }}</th>
20     @endforeach
21 </tr>
22 </thead>
23 <tbody>
24     @foreach($ar_penerbit as $p)
25 <tr>
26     <td>{{ $no++ }}</td><td>{{ $p->nama }}</td><td>{{ $p->alamat }}</td>
27     <td>{{ $p->email }}</td><td>{{ $p->website }}</td><td>{{ $p->telp }}</td>
28     <td>{{ $p->cp }}</td>
29 </tr>
30 @endforeach
31 </tbody>
32 </table>
33 {{ $ar_penerbit->links() }} {{-- link untuk pagination --}}
```

Tampilan Pagination

Setelah melihat tampilan pagination, buka web browser, lalu pada url ketikkan `http://localhost:8000/penerbit`. Lihatlah di bawah data penerbit ada paginationnya seperti tampak gambar di bawah ini:



No	Nama	Alamat	Email	Website	Telp	CP
1	nama_60cc0ca9a1dd8	alamat_60cc0ca9a1dda	60cc0ca9a1ddb@gmail.com	www.60cc0ca9a1ddc.com	021-1065395199	cp_60cc0ca9a1dde
2	nama_60cc0ca9aec71	alamat_60cc0ca9aec74	60cc0ca9aec75@gmail.com	www.60cc0ca9aec76.com	021-159903654	cp_60cc0ca9aec79
3	nama_60cc0ca9b6a68	alamat_60cc0ca9b6a6d	60cc0ca9b6a6f@gmail.com	www.60cc0ca9b6a71.com	021-1309128295	cp_60cc0ca9b6a74
4	nama_60cc0ca9cf31c	alamat_60cc0ca9cf321	60cc0ca9cf322@gmail.com	www.60cc0ca9cf324.com	021-1218151355	cp_60cc0ca9cf326
5	nama_60cc0ca9d752f	alamat_60cc0ca9d7538	60cc0ca9d753b@gmail.com	www.60cc0ca9d7540.com	021-378398373	cp_60cc0ca9d7545

pagination.previous pagination.next

Filter Data

Fitur filter data adalah salah satu fitur yang terdapat pada aplikasi yang digunakan untuk menyaring data lalu menampilkan baris data tertentu sesuai filter yang kita terapkan dan menyembunyikan sisa baris data lainnya.

Fitur filter di Laravel biasanya tersedia di grid view data. Kali ini kita akan membuat filter berdasarkan judul buku, kita buat di action index di BukuController. Kita akan membuat action tersebut bisa membaca query string "keyword" dan menjadikannya kriteria saat query model Buku dengan contoh:

`http://localhost:8080/buku?keyword=sejarah`



Filter Data di Action Index BukuController

```
public function index(Request $request)
{
    $status = $request->get('status');
    $keyword = $request->get('keyword') ? $request->get('keyword') : '';
    if($status){
        $books = \App\Models\Buku::with('judul')->where('judul', "LIKE",
"%$keyword%")->where('status', strtoupper($status))->paginate(10);
    } else {
        $books = \App\Models\Buku::with('judul')->where('judul', "LIKE",
"%$keyword%")->paginate(10);
    }

    return view('buku.index', ['books' => $books]);
}
```



Form Filter Data pada views buku.index

Selanjutnya kita buat input untuk mengetikkan keyword pada view buku.index, tambahkan kode program di bawah ini:

```
<form action="{{route('buku.index')}}">
<div class="input-group">
  <input name="keyword" type="text" value="{{Request::get('keyword')}}"
    class="form-control" />
    <div class="input-group-append">
      <input type="submit" value="Filter" class="btn btn-primary">
    </div>
</div>
</form>
```



Extensions

- Jika diterjemahkan secara harfiah, maka extension adalah tambahan atau perpanjangan. Namun yang dimaksudkan disini adalah paket kode terpisah yang didesain untuk digunakan pada aplikasi framework dan menyediakan fitur-fitur yang siap digunakan.
- Kata kuncinya adalah terpisah, artinya kode ini diluar kode utama atau core dari framework Laravel.
- Kita bisa lihat bahwa extension itu hanyalah istilah umum dimana bentuknya bisa berupa vendor, widget, components, module (sub aplikasi), helpers (fungsi tertentu), dan lain-lain.
- Kode extension bersifat reusable atau dapat digunakan kembali jika dibutuhkan. Dengan menggunakan extension tentu bisa mempercepat developer dalam mengembangkan aplikasinya. Ada banyak extension yang mendukung framework dengan berbagai fungsinya.





Install Extensions Export to PDF

```
Command Prompt - php artisan serve
C:\xampp\htdocs\laraveltm4>composer require barryvdh/laravel-dompdf
Using version ^0.8.6 for barryvdh/laravel-dompdf
./composer.json has been updated
Loading composer repositories with package information
Updating dependencies (including require-dev)
Package operations: 5 installs, 0 updates, 0 removals
  - Installing sabberworm/php-css-parser (8.3.0): Downloading (100%)
  - Installing phenx/php-svg-lib (v0.3.3): Downloading (100%)
  - Installing phenx/php-font-lib (0.5.2): Downloading (100%)
  - Installing dompdf/dompdf (v0.8.5): Downloading (100%)
  - Installing barryvdh/laravel-dompdf (v0.8.6): Downloading (100%)
dompdf/dompdf suggests installing ext-imagick (Improves image processing performance)
dompdf/dompdf suggests installing ext-gmagick (Improves image processing performance)
Writing lock file
Generating optimized autoload files
> Illuminate\Foundation\ComposerScripts::postAutoloadDump
> @php artisan package:discover --ansi
Discovered Package: barryvdh/laravel-dompdf
Discovered Package: facade/ignition
Discovered Package: fideloper/proxy
Discovered Package: fruitcake/laravel-cors
Discovered Package: laravel/tinker
Discovered Package: nesbot/carbon
Discovered Package: nunomaduro/collision
Package manifest generated successfully.
```



Konfigurasi Extensions Export to PDF

Setelah berhasil menginstall extension export ke dalam format pdf menggunakan vendor barryvdh melalui composer, langkah selanjutnya melakukan konfigurasi tambahan di config/app.php untuk menambahkan provider dan alias vendor seperti kode program berikut ini:

```
'providers' => [
    ....
    Barryvdh\DomPDF\ServiceProvider::class,
],

'aliases' => [
    ....
    'PDF' => Barryvdh\DomPDF\Facade::class,
]
```



Menambah Routing Baru

Selanjutnya tambahkanlah routing baru untuk akses fitur export to PDF, buka file **routes/web.php** lalu tambahkan kode di bawah ini:

```
use App\Http\Controllers\BukuController;
```

```
Route::get('bukupdf',  
    [BukuController::class, 'bukuPDF']);
```



Membuat Fungsi bukuPDF() di BukuController

```
use PDF;
class BukuController extends Controller
{
    public function bukuPDF()
    {
        $ar_buku = DB::table('buku')
        ->join('pengarang', 'pengarang.id', '=', 'buku.idpengarang')
        ->join('penerbit', 'penerbit.id', '=', 'buku.idpenerbit')
        ->join('kategori', 'kategori.id', '=', 'buku.kategori_id')
        ->select('buku.*', 'pengarang.nama', 'penerbit.nama AS pen',
            'kategori.nama AS kat')
        ->get();
        $pdf = PDF::loadView('buku/bukuPDF', ['ar_buku'=>$ar_buku]);
        return $pdf->download('dataBuku.pdf');
    }
}
```



Membuat View bukuPDF di views/buku

```
@php
$no = 1;
$ar_judul = ['No', 'ISBN', 'Judul', 'Pengarang', 'Penerbit', 'Kategori'];
@endphp
<h3 align="center">Daftar Buku</h3>
<table border="1" width="100%" cellpadding="0" align="center">
  <thead>
    <tr bgcolor="pink">
      @foreach( $ar_judul as $jdl )
        <th>{{ $jdl }}</th>
      @endforeach
    </tr>
  </thead>
  <tbody>
    @foreach($ar_buku as $buku)
      <tr>
        <td>{{ $no++ }}</td><td>{{ $buku->isbn }}</td><td>{{ $buku->judul }}</td>
        <td>{{ $buku->nama }}</td> <td>{{ $buku->pen }}</td> <td>{{ $buku->kat }}</td>
      </tr>
    @endforeach
  </tbody>
</table>
```



Menambahkan Menu di views/buku/index

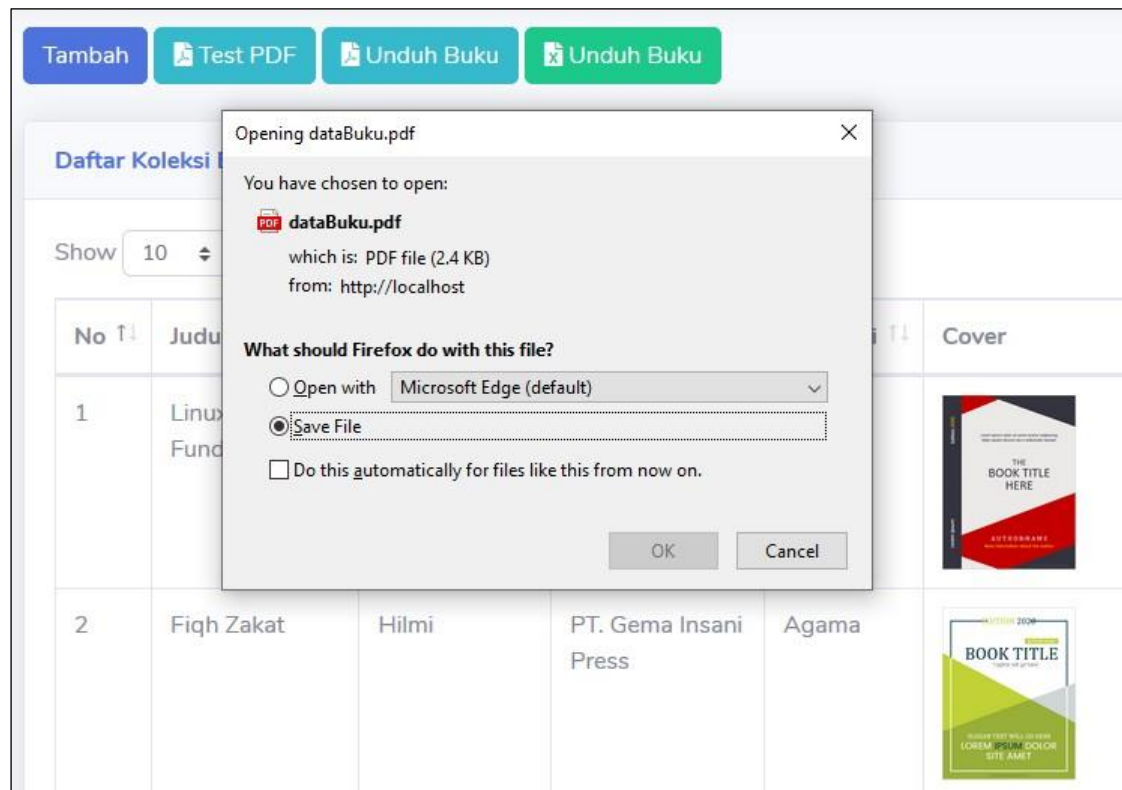
Setelah membuat fungsi bukuPDF di BukuController, langkah selanjutnya adalah membuat menu untuk hyperlink Unduh Buku dalam format PDF seperti kode program di dalam slide dengan penjelasan sebagai berikut:

```
<a href="{{ url('bukupdf') }}" class="btn btn-info">  
  <i class="fas fa-file-pdf"></i> Unduh Buku </a>
```





Tampilan Penggunaan Ext Export to PDF



Install Extensions Export to CSV

```
C:\xampp\htdocs\laraveltm4>composer require maatwebsite/excel
Using version ^3.1 for maatwebsite/excel
./composer.json has been updated
Loading composer repositories with package information
Updating dependencies (including require-dev)
Package operations: 4 installs, 0 updates, 0 removals
- Installing markbaker/matrix (1.2.0): Downloading (100%)
- Installing markbaker/complex (1.4.8): Downloading (100%)
- Installing phpoffice/phpspreadsheet (1.12.0): Downloading (100%)
- Installing maatwebsite/excel (3.1.19): Downloading (100%)
```



Konfigurasi Extensions Export to CSV

Setelah berhasil menginstall extension export ke dalam format csv menggunakan vendor maatwebsite/excel melalui composer, langkah selanjutnya melakukan konfigurasi tambahan di config/app.php untuk menambahkan provider dan alias vendor seperti kode program di bawah ini:

```
'providers' => [
    ....
    Maatwebsite\Excel\ExcelServiceProvider::class,
],
'aliases' => [
    ....
    'Excel' => Maatwebsite\Excel\Facades\Excel::class,
],
```



Menambah Routing Baru

Selanjutnya tambahkanlah routing baru untuk akses fitur export to CSV, buka file **routes/web.php** lalu tambahkan kode di bawah ini:

```
use App\Http\Controllers\BukuController;
```

```
Route::get('bukucsv',  
    [BukuController::class, 'bukucsv']);
```



Koleksi Data(1)

Untuk mendapatkan data koleksi buku yang akan diexport ke dalam format csv, jalankan perintah di bawah ini melalui CMD:

```
php artisan make:import BukuImport --model=Buku
```

Dengan menjalankan perintah tersebut, maka secara otomatis akan dibuatkan class BukuExport dan di dalamnya juga secara otomatis akan dibuatkan fungsi collection untuk mendapatkan data koleksi buku yang akan diexport ke dalam format csv.





Koleksi Data(2)

```
<?php
//Pada file BukuExport.php yang berada di folder app/Exports isi fungsi collection()
namespace App\Exports;
use App\Buku;
use Maatwebsite\Excel\Concerns\FromCollection;
use Maatwebsite\Excel\Concerns\WithHeadings;
use DB;
class BukuExport implements FromCollection, WithHeadings
{
    public function collection()
    {
        //return Buku::all();
        $ar_buku = DB::table('buku')
        ->join('pengarang', 'pengarang.id', '=', 'buku.idpengarang')
        ->join('penerbit', 'penerbit.id', '=', 'buku.idpenerbit')
        ->join('kategori', 'kategori.id', '=', 'buku.kategori_id')
        ->select('buku.isbn', 'buku.judul', 'buku.tahun_cetak', 'buku.stok',
            'pengarang.nama', 'penerbit.nama AS pen', 'kategori.nama AS kat')
        ->get();
        return $ar_buku;
    }
}
```

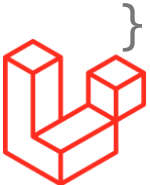




Koleksi Data(3)

Untuk mencetak judul kolom pada tabel di csv, buatlah fungsi headings yang menampung array judul kolom seperti kode program di dalam slide dengan penjelasan sebagai berikut:

```
public function headings(): array
{
    return [
        'ISBN', 'Judul', 'Tahun Cetak', 'Stok',
        'Pengarang', 'Penerbit', 'Kategori'
    ];
}
```





Membuat Fungsi bukuPDF() di BukuController

```
<?php
namespace App\Http\Controllers;
use Illuminate\Http\Request;
use DB;
use App\Buku;
use App\Exports\BukuExport;
use Maatwebsite\Excel\Facades\Excel;

class BukuController extends Controller
{
    public function bukucsv()
    {
        return Excel::download(new BukuExport, 'buku.csv');
    }
}
```





Menambahkan Menu di views/buku/index

```
<h3>Daftar Buku</h3>
```

```
<a class="btn btn-primary" href="{{ route('buku.create') }}"  
  role="button">Tambah</a>
```

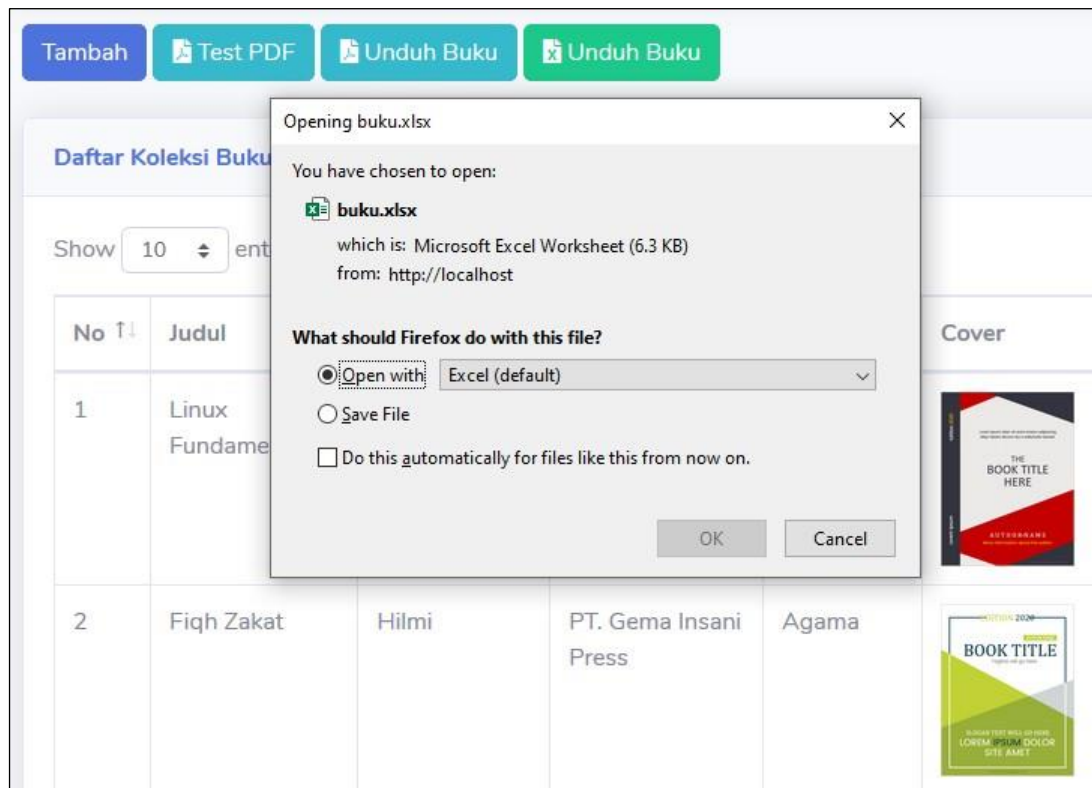
```
<a class="btn btn-info" href="{{ url('bukupdf') }}"  
  role="button">Export to PDF</a>
```

```
<a class="btn btn-warning" href="{{ url('bukucsv') }}"  
  role="button">Export to CSV</a>
```





Tampilan Penggunaan Ext Export to CSV



JSON

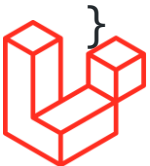
- JSON (JavaScript Object Notation) adalah sebuah format data yang digunakan untuk pertukaran dan penyimpanan data.
- JSON merupakan bagian (subset) dari Javascript. JSON bisa dibaca dengan berbagai macam bahasa pemrograman seperti PHP, Java, Javascript, Perl, Python, dan banyak lagi.
- Hal ini membuat JSON menjadi bahasa yang ideal untuk pertukaran data antar aplikasi.
- Berbeda dengan XML (extensive markup language) dan format lainnya yang memiliki fungsi serupa, JSON memiliki struktur data yang sederhana dan mudah dipahami. Itulah mengapa JSON sering digunakan pada API(Application Programming Interface).



Export Data to JSON

Untuk mengekspor data buku ke dalam format JSON, buatlah fungsi baru misalnya JSONBuku() di BukuController.php seperti kode program di bawah ini:

```
public function JSONBuku(){  
    return DB::table('buku')  
        ->join('pengarang', 'pengarang.id', '=', 'buku.idpengarang')  
        ->join('penerbit', 'penerbit.id', '=', 'buku.idpenerbit')  
        ->join('kategori', 'kategori.id', '=', 'buku.idkategori')  
        ->select('buku.*', 'pengarang.nama', 'penerbit.nama AS pen',  
            'kategori.nama AS kat')  
        ->get();  
}
```



Menambah Routing Baru

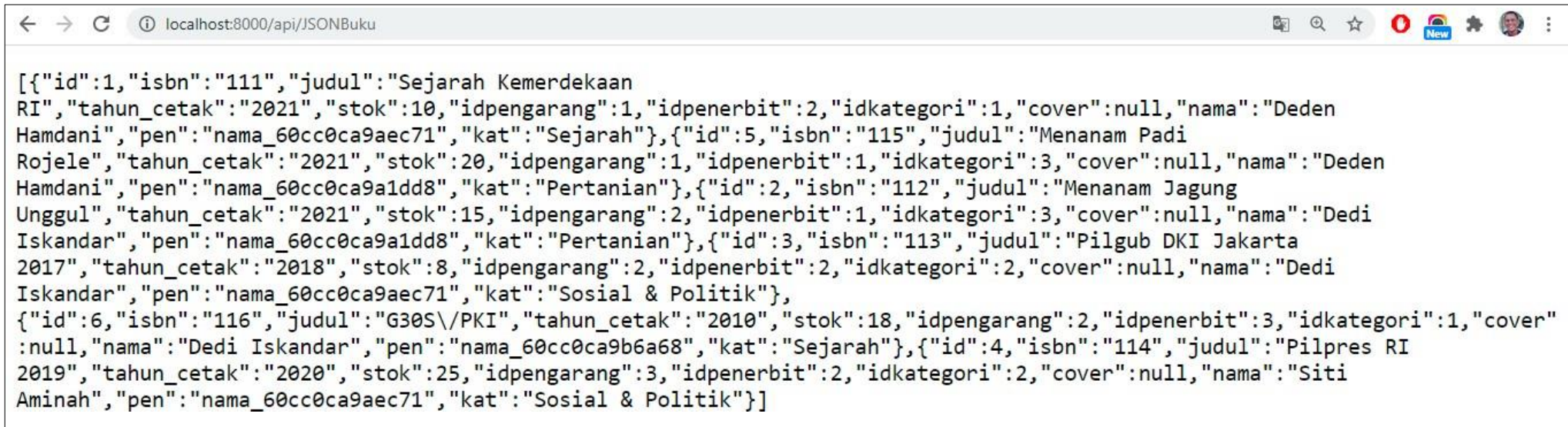
Selanjutnya tambahkanlah routing baru untuk akses fitur export to JSON, buka file **routes/api.php** lalu tambahkan kode di bawah ini:

```
Route::get('JSONBuku',  
    'App\Http\Controllers\BukuController@JSONBuku');
```



Tampilan Penggunaan Export Data to JSON

Setelah membuat fungsi baru JSONBuku() di BukuController dan menambahkan routing baru di routes/api.php, selanjutnya lihatlah tampilan data JSON-nya pada web browser dengan url <http://localhost:8000/api/JSONBuku> seperti tampak gambar di bawah ini:



```
[{"id":1,"isbn":"111","judul":"Sejarah Kemerdekaan RI","tahun_cetak":"2021","stok":10,"idpengarang":1,"idpenerbit":2,"idkategori":1,"cover":null,"nama":"Deden Hamdani","pen":"nama_60cc0ca9aec71","kat":"Sejarah"}, {"id":5,"isbn":"115","judul":"Menanam Padi Rojele","tahun_cetak":"2021","stok":20,"idpengarang":1,"idpenerbit":1,"idkategori":3,"cover":null,"nama":"Deden Hamdani","pen":"nama_60cc0ca9a1dd8","kat":"Pertanian"}, {"id":2,"isbn":"112","judul":"Menanam Jagung Unggul","tahun_cetak":"2021","stok":15,"idpengarang":2,"idpenerbit":1,"idkategori":3,"cover":null,"nama":"Dedi Iskandar","pen":"nama_60cc0ca9a1dd8","kat":"Pertanian"}, {"id":3,"isbn":"113","judul":"Pilgub DKI Jakarta 2017","tahun_cetak":"2018","stok":8,"idpengarang":2,"idpenerbit":2,"idkategori":2,"cover":null,"nama":"Dedi Iskandar","pen":"nama_60cc0ca9aec71","kat":"Sosial & Politik"}, {"id":6,"isbn":"116","judul":"G30S/PKI","tahun_cetak":"2010","stok":18,"idpengarang":2,"idpenerbit":3,"idkategori":1,"cover":null,"nama":"Dedi Iskandar","pen":"nama_60cc0ca9b6a68","kat":"Sejarah"}, {"id":4,"isbn":"114","judul":"Pilpres RI 2019","tahun_cetak":"2020","stok":25,"idpengarang":3,"idpenerbit":2,"idkategori":2,"cover":null,"nama":"Siti Aminah","pen":"nama_60cc0ca9aec71","kat":"Sosial & Politik"}]
```

**TERIMA KASIH
ATAS SEGALA PERHATIAN
SEMOGA BERMANFAAT...**

