

**CheatSheet + Penjelasan tentang *Laravel* dengan bahasa yg mudah di
mengerti (Gak pake bahasa teknis) Full Version**



DAFTAR ISI

	Nomor Halaman
DAFTAR ISI.....	2
 Pembuka.....	5
 <i>Sedikit refleksi</i>	5
Composer Command Cheat Sheet + Penjelasan dan Analogi (instalasi Laravel dulu dong!) 6	
1. composer create-project laravel/laravel nama-proyek.....	6
2. composer require vendor/paket.....	6
3. composer remove vendor/paket	7
4. composer update	7
5. composer update vendor/paket.....	7
6. composer install.....	7
7. composer dump-autoload	7
8. composer dump-autoload -o	7
9. composer show	7
10. composer depends vendor/paket	7
11. composer init	8
12. rm -rf vendor && composer install.....	8
13. composer diagnose	8
14. composer validate.....	8
Artisan Command Cheat Sheet + Penjelasan dengan Analogi (buat bisa bikin model, controller, dll).....	9
1. php artisan serve.....	9
2. php artisan make:controller NamaController	9
3. php artisan make:model NamaModel	9
4. php artisan make:migration create_namatabel_table.....	9
5. php artisan make:seeder NamaSeeder	9
6. php artisan db:seed	9
7. php artisan db:seed --class=NamaSeeder	9
8. php artisan migrate.....	10
9. php artisan migrate:rollback.....	10
10. php artisan migrate:refresh.....	10
11. php artisan route:list.....	10

12. php artisan cache:clear	10
13. php artisan config:clear	10
14. php artisan view:clear	10
15. php artisan make:request NamaRequest	10
16. php artisan make:controller NamaController --resource.....	11
17. php artisan make:policy NamaPolicy.....	11
18. php artisan make:event NamaEvent	11
19. php artisan make:listener NamaListener.....	11
20. php artisan make:command NamaCommand.....	11
 Laravel Blade Template Cheat Sheet.....	12
1. Menampilkan Data.....	12
2. Komentar	12
3. If Statement	12
4. Looping For	12
5. Looping Foreach.....	12
6. Extend Layout.....	13
7. Section & Yield	13
8. Include Partial File.....	13
 Laravel Routing Cheat Sheet	13
1. Route GET.....	13
2. Route POST	13
3. Route PUT / PATCH	13
4. Route DELETE	14
5. Route Resource	14
7. Route Parameter.....	14
8. Route Named	14
9. Redirect Route	14
 Laravel Validation Cheat Sheet.....	15
1. Validasi Manual di Controller.....	15
2. Menampilkan Error di Blade	15
3. Validasi Pakai Form Request.....	15
4. Validasi Conditional.....	16
 Laravel Middleware Cheat Sheet	16

1. Buat Middleware Baru	16
 Laravel Model & Database Cheat Sheet.....	17
✓ 1. Buat Model + Migration Sekaligus.....	17
✓ 2. Atur Kolom yang Bisa Diisi (fillable)	17
✓ 3. Relasi 1 to Many (One User punya Banyak Todo)	18
✓ 5. Ambil Data dari Relasi.....	18
✓ 6. Perintah Migration Dasar	19
✓ 7. Seeder & Factory Combo (Isi Database Dummy)	19
 Laravel Role & Permission Cheat Sheet.....	19
✓ 1. Pengantar: Apa Itu Role & Permission?.....	19
✓ 2. Cara Sederhana (Manual - tanpa package)	19
✓ 3. Pakai Package: Spatie Laravel-Permission (Rekomendasi)	20

Pembuka

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

"Dengan menyebut nama Allah Yang Maha Pengasih lagi Maha Penyayang."

أَفْرُأُ بِاسْمِ رَبِّكَ الَّذِي خَلَقَ

"Bacalah dengan (menyebut) nama Tuhanmu yang menciptakan."

(Surah Al-'Alaq: 1)

💡 "Kutipan ini bukan cuma sekadar ajakan membaca, tapi perintah ilahi untuk belajar, mengeksplorasi, dan memahami dunia dengan menyebut nama-Nya. **Modul CheatSheet** ini hadir untuk jadi salah satu jalan kecil dalam menjalankan perintah besar itu: *Iqra!*"

📘 "Bacalah, bukan cuma buku, tapi kode juga."

⌚ Sedikit refleksi...

Di zaman sekarang, banyak dari kita (termasuk aku dan kamu) yang dengan mudahnya bisa buka Reel IG, Yt Short, Tiktok, dll selama **3 jam non-stop**, tapi ketika disuruh buka dokumen belajar atau buka terminal coding, langsung merasa "berat".

Kita kebanyakan **scroll**... tapi jarang **mempelajari**.

Kita kebanyakan **menonton**... tapi lupa **mendalami**.

Padahal, **Iqra** bukan cuma baca teks—tapi juga perintah untuk *mikir, eksplor, dan bangun diri*.

Modul ini dibuat untuk itu.

Bukan sekadar kumpulan cheat sheet, tapi ajakan untuk **kembali menyalaikan semangat belajar**, walaupun pelan.

🧠 Jadi... kalau kamu bisa buka Reel IG, Yt Short, Tiktok 3 jam tanpa ngeluh, kamu pasti juga bisa buka Laravel 30 menit sehari — dengan **niat yang sama, tapi arah yang beda**.

Mari mulai dari **Iqra**. Mari belajar, Mari bangkit

Composer Command Cheat Sheet + Penjelasan dan Analogi (instalasi Laravel dulu dong!)

1. composer create-project laravel/laravel nama-proyek

Analogi: Kayak lo beli bahan bangunan satu paket buat bikin rumah Laravel.

Penjelasan: Ini buat bikin project Laravel baru dari template resmi.

1.1 Buat Laravel dengan Versi Tertentu misal ingin laravel 9,10,11

```
composer create-project laravel/laravel:^10.0 nama-proyek
```

Penjelasan: Lo mau bikin rumah Laravel tapi pengen desain arsitek tahun tertentu, misalnya versi 10.0—nah ini caranya!

1.2 Laravel + Vue

```
composer require laravel/breeze --dev
php artisan breeze:install vue
npm install && npm run dev
```

Analogi: Lo masang dapur Vue ke rumah Laravel, lengkap dengan kompor Vite buat masak cepat!

1.3 Laravel + React

```
composer require laravel/breeze --dev
php artisan breeze:install react
npm install && npm run dev
```

Analogi: Rumah Laravel + jendela interaktif React = rumah modern minimalis!

1.4 Laravel + Inertia + Vue/React (SPA Style)

```
composer require laravel/breeze --dev
php artisan breeze:install vue --inertia
npm install && npm run dev
```

Analogi: Laravel jadi backend solid, Inertia jadi jembatan, Vue/React jadi interiornya. Rumah jadi satu kesatuan SPA (Single Page App)!

2. composer require vendor/paket

Analogi: Tambah alat dapur atau perabot baru ke rumah.

Penjelasan: Install package/pustaka tambahan ke project lo, misal Laravel Breeze, DomPDF, dsb.

3. composer remove vendor/paket

Analogi: Buang alat dapur yang udah gak dipakai.

Penjelasan: Hapus package dari project biar gak numpuk & berat.

4. composer update

Analogi: Upgrade semua peralatan ke versi terbaru.

Penjelasan: Update semua package ke versi terbaru sesuai aturan di composer.json.

5. composer update vendor/paket

Analogi: Upgrade satu alat doang, misalnya kompor doang.

Penjelasan: Update satu package tertentu tanpa ganggu yang lain.

6. composer install

Analogi: Bikin ulang isi rumah lo sesuai daftar belanja di kertas (composer.lock).

Penjelasan: Pas lo cloning project dari orang lain, ini buat install semua dependency-nya.

7. composer dump-autoload

Analogi: Bikin Laravel lebih hafal isi rumahnya setelah lo tambahin ruangan baru.

Penjelasan: Buat Laravel nge-refresh semua class yang bisa di-load, apalagi kalau lo buat file baru manual.

8. composer dump-autoload -o

Analogi: Sama kayak di atas tapi pakai turbo biar lebih efisien.

Penjelasan: Optimalkan autoload, cocok buat project besar.

9. composer show

Analogi: Lihat daftar isi lemari dapur lo.

Penjelasan: Tampilkan semua package yang udah ke-install di project lo.

10. composer depends vendor/paket

Analogi: Cari tahu si alat dapur itu butuh alat lain gak buat kerja.

Penjelasan: Liat package mana aja yang tergantung ke package tersebut.

11. composer init

Analogi: Mulai nyusun daftar belanja lo sendiri dari nol.

Penjelasan: Bikin file composer.json buat project kosong (biasanya buat lib atau tool custom).

12. rm -rf vendor && composer install

Analogi: Bongkar semua isi rumah dan isi ulang dari kertas belanjaan.

Penjelasan: Kadang dipakai pas error, buat reset semua dependency biar fresh.

13. composer diagnose

Analogi: Ngecek kondisi rumah dan peralatan—ada yang rusak atau ngaco gak?

Penjelasan: Cek kemungkinan error atau konfigurasi yang salah.

14. composer validate

Analogi: Periksa ulang daftar belanjaan lo bener gak formatnya.

Penjelasan: Validasi file composer.json supaya gak ada typo/error.

Artisan Command Cheat Sheet + Penjelasan dengan Analogi (buat bisa bikin model, controller, dll)

1. php artisan serve

Analogi: Kayak lo nyalain kompor biar bisa mulai masak.

Penjelasan: Ini buat nyalain server lokal. Setelah ini, lo bisa buka web-nya di browser lewat localhost:8000.

2. php artisan make:controller NamaController

Analogi: Bikin chef di dapur. Dia yang ngatur alur masak.

Penjelasan: Ini bikin file controller, tempat logika dan aksi dijalankan pas user klik atau submit form.

3. php artisan make:model NamaModel

Analogi: Bikin rak tempat nyimpen bahan dapur.

Penjelasan: Ini file yang ngehubungin Laravel ke database. Di sini lo bisa atur gimana data disimpan atau diambil.

4. php artisan make:migration create_namatabel_table

Analogi: Nyusun cetak biru buat bangun rak tadi.

Penjelasan: Migration itu kayak peta buat bikin tabel di database. Bisa tambah kolom, ubah, atau hapus.

5. php artisan make:seeder NamaSeeder

Analogi: Nyediain bahan dapur supaya rak gak kosong pas dicoba.

Penjelasan: Seeder dipakai buat ngisi data awal ke tabel di database, misalnya data dummy.

6. php artisan db:seed

Analogi: Masukin bahan ke rak berdasarkan daftar yang udah dibuat.

Penjelasan: Eksekusi semua seeder yang udah lo buat.

7. php artisan db:seed --class=NamaSeeder

Analogi: Masukin bahan spesifik doang, bukan semua.

Penjelasan: Jalankan satu seeder tertentu, bukan semua seeder.

8. php artisan migrate

Analogi: Bangun rak sesuai cetak birunya.

Penjelasan: Buat tabel-tabel di database sesuai dengan file migration.

9. php artisan migrate:rollback

Analogi: Bongkar rak terakhir yang tadi dibangun.

Penjelasan: Hapus migration terakhir (rollback satu langkah mundur).

10. php artisan migrate:refresh

Analogi: Bongkar semua rak, bangun ulang dari awal.

Penjelasan: Hapus semua migrasi dan jalanin lagi dari awal—bersih total.

11. php artisan route:list

Analogi: Lihat daftar menu dan jalur dapur.

Penjelasan: Nampilin semua route (jalan/jalur) yang dipake Laravel dan controller-nya.

12. php artisan cache:clear

Analogi: Bersihin kulkas biar gak ada bahan kadaluarsa.

Penjelasan: Hapus cache aplikasi Laravel biar gak stuck di data lama.

13. php artisan config:clear

Analogi: Reset pengaturan dapur ke keadaan baru.

Penjelasan: Hapus konfigurasi yang disimpan sementara (biar .env terbaru dibaca).

14. php artisan view:clear

Analogi: Bersihin jendela biar tampilan gak ngaco.

Penjelasan: Hapus cache tampilan (view) yang udah dikompilasi, penting saat edit Blade.

15. php artisan make:request NamaRequest

Analogi: Bikin form resmi buat order bahan.

Penjelasan: Buat class khusus untuk validasi inputan dari user.

16. `php artisan make:controller NamaController --resource`

Analogi: Chef yang lengkap, bisa handle semua menu (tambah, edit, hapus, dll).

Penjelasan: Buat controller dengan fungsi lengkap otomatis: index, store, update, destroy, dll.

17. `php artisan make:policy NamaPolicy`

Analogi: Aturan siapa boleh pakai dapur mana.

Penjelasan: File buat ngatur siapa yang boleh akses fitur tertentu (authorization).

18. `php artisan make:event NamaEvent`

Analogi: Bikin alarm otomatis di dapur, misal nasi matang.

Penjelasan: Event dipake buat trigger otomatis saat sesuatu terjadi (misal user daftar).

19. `php artisan make:listener NamaListener`

Analogi: Orang yang langsung gerak pas alarm bunyi.

Penjelasan: Listener ngerespon Event, misalnya kirim email kalau user baru daftar.

20. `php artisan make:command NamaCommand`

Analogi: Bikin robot bantu dapur jalankan tugas otomatis.

Penjelasan: Custom command yang bisa lo jalankan via terminal buat tugas-tugas rutin.

💡 Laravel Blade Template Cheat Sheet

1. Menampilkan Data

```
{{ $nama }}
```

🔍 **Analogi:** Kayak naro nama lo di kartu nama—tinggal {{ ... }} langsung keluar.

2. Komentar

```
{{-- Ini komentar --}}
```

🔍 **Analogi:** Catatan buat developer, tapi gak bakal kelihatan user kayak post-it di belakang layar.

3. If Statement

```
@if($nilai > 75)
```

```
Lulus
```

```
@endif
```

🔍 **Analogi:** Kalau lo dapet nilai bagus, baru dapet hadiah. Ini kondisi bersyarat.

4. Looping For

```
@for($i = 0; $i < 10; $i++)
```

```
    {{ $i }}
```

```
@endfor
```

🔍 **Analogi:** Ulangi sepuluh kali, kayak push-up pagi.

5. Looping Foreach

```
@foreach($tugas as $todo)
```

```
    {{ $todo }}
```

```
@endforeach
```

🔍 **Analogi:** Lo muterin isi rak satu per satu buat ditampilkan ke layar.

6. Extend Layout

```
@extends('layouts.app')
```

🔍 **Analogi:** Ini kayak lo ngewarisin tampilan dasar, jadi halaman lain gak perlu bikin ulang header/footer.

7. Section & Yield

```
@section('content')
```

```
<h1>Halo Dunia</h1>
```

```
@endsection
```

🔍 **Analogi:** Ngisi kotak kosong yang udah disediakan di layout utama, biar semua halaman rapi.

8. Include Partial File

```
@include('partials.navbar')
```

🔍 **Analogi:** Panggil potongan template kecil, kayak bagian pintu atau jendela rumah yang bisa dipasang di mana aja.



Laravel Routing Cheat Sheet

1. Route GET

```
Route::get('/home', [HomeController::class, 'index']);
```



Analogi: Jalan tol satu arah. Bisa masuk, tapi gak bisa kirim data balik.

2. Route POST

```
Route::post('/form', [FormController::class, 'submit']);
```

🔍 **Analogi:** Kayak ngirim formulir ke resepsionis. Lo kirim data, backend terima dan proses.

3. Route PUT / PATCH

```
Route::put('/update/{id}', [UserController::class, 'update']);
```



Analogi: Ganti data lama dengan data baru. Mirip lo benerin biodata di KTP.

4. Route DELETE

```
Route::delete('/hapus/{id}', [TodoController::class, 'destroy']);
```

🔍 **Analogi:** Minta petugas buat nyapu bersih data dari database.

5. Route Resource

```
Route::resource('posts', PostController::class);
```

🔍 **Analogi:** Sekali bikin jalur, langsung dapet semua rute lengkap (index, create, store, show, edit, update, destroy).

6. Route dengan Middleware

```
Route::middleware(['auth'])->group(function () {  
    Route::get('/dashboard', [UserController::class, 'index']);  
});
```

🔍 **Analogi:** Ibarat Ada **Gerbang** nya jadi Hanya yang punya **kartu akses (login)** yang bisa masuk ruangan dashboard.

7. Route Parameter

```
Route::get('/user/{id}', [UserController::class, 'show']);
```

🔍 **Analogi:** Lo mau liat info user ke-5, tinggal masukin angka itu di URL.

8. Route Named

```
Route::get('/profile', [UserController::class, 'profile'])->name('profile');
```

🔍 **Analogi:** Ngasih nama jalan biar gampang dipanggil di mana-mana, kayak route('profile').

9. Redirect Route

```
Route::redirect('/lama', '/baru');
```

🔍 **Analogi:** Ada papan penunjuk “Arah Baru” kalau jalan lama udah gak dipakai.

✓ Laravel Validation Cheat Sheet

1. Validasi Manual di Controller

```
$request->validate([
    'nama' => 'required|string|max:255',
    'email' => 'required|email|unique:users',
    'password' => 'required|min:6',
]);
```

🔍 **Analogi:** Ini kayak ngecek form pendaftaran sebelum dikasih masuk ke acara. Kalau salah isi, suruh perbaiki dulu.

2. Menampilkan Error di Blade

```
@if ($errors->any())
    <div>
        @foreach ($errors->all() as $error)
            <p>{{ $error }}</p>
        @endforeach
    </div>
@endif
```

🔍 **Analogi:** Ngasih tahu ke peserta form mana yang belum bener. Kayak panitia nyamperin dan bilang: “Eh, isi email-mu salah tuh!”

3. Validasi Pakai Form Request

```
php artisan make:request StoreUserRequest
```

Di file `StoreUserRequest.php`:

```
public function rules()
{
    return [
        'nama' => 'required',
        'email' => 'required|email',
    ];
}
```

4. Validasi Conditional

```
$request->validate([
    'telepon' => $request->metode == 'telepon' ? 'required' : 'nullable',
]);
```

🔍 **Analogi:** Cek isi formulir sesuai pilihan user. Kalau pilih "kirim via telepon", ya harus isi nomor. Kalau enggak, santai aja.

🔒 Laravel Middleware Cheat Sheet

1. Buat Middleware Baru

```
php artisan make:middleware CekUmur
```

🔍 **Analogi:** Ini kayak bikin satpam baru yang punya aturan khusus, misalnya: "Harus umur 18+ buat masuk."

1.1 Contoh Code Middleware nya:

```
public function handle($request, Closure $next)
{
    if ($request->umur < 18) {
        return redirect('gagal');
    }

    return $next($request);
}
```

🔍 **Analogi:** Satpam ngecek dulu sebelum izinkan orang masuk ke ruangan. Kalau gak lolos, disuruh balik.

1.2 Daftarin Middleware di Kernel

```
// app/Http/Kernel.php (ini adalah letak path file kernel.php)
```

```
'cek.umur' => \App\Http\Middleware\CekUmur::class,
```

🔍 **Analogi:** Lo masukin nama satpam itu ke daftar penjaga resmi gedung Laravel.

1.3 Pakai Middleware di Route

```
Route::get('/dewasa', function () {
    return 'Selamat datang, orang dewasa!';
})->middleware('cek.umur');
```

🔍 **Analogi:** Pasang satpam di pintu masuk rute tertentu. Cuma yang lolos bisa lewat.

1.4 Middleware Bawaan Laravel

- `auth` → Cek apakah user sudah login
- `guest` → Cek apakah user belum login
- `verified` → Cek apakah email sudah diverifikasi

 **Analogi:** Ini kayak satpam default dari Laravel yang udah terlatih—gak usah ngoding ulang.

 **Laravel Model & Database Cheat Sheet**

 **1. Buat Model + Migration Sekaligus**
`php artisan make:model NamaModel -m`

 **Analogi:** Sekali jalan, lo bikin rak + cetak biru buat raknya juga.

 **2. Atur Kolom yang Bisa Diisi (fillable)**
`protected $fillable = ['nama', 'email', 'password'];`

 **Analogi:** Lo kasih daftar kolom mana aja yang boleh diisi otomatis. Yang gak ada di sini, Laravel tolak.

✓ 3. Relasi 1 to Many (One User punya Banyak Todo)

Di model User.php

```
php
SalinEdit
public function todos()
{
    return $this->hasMany(Todo::class);
}
```

Di model Todo.php

```
public function user()
{
    return $this->belongsTo(User::class);
}
```

🔍 **Analogi:** Satu orang bisa punya banyak tugas. Tapi satu tugas cuma milik satu orang.

✓ 4. Relasi Many to Many (User & Role, Misalnya)

Di model User.php

```
public function roles()
{
    return $this->belongsToMany(Role::class);
}
```

Di model Role.php

```
public function users()
{
    return $this->belongsToMany(User::class);
}
```

Jangan lupa: harus ada tabel pivot `role_user`!

🔍 **Analogi:** User bisa punya banyak role, dan role bisa dimiliki banyak user — kayak guru bisa ngajarin banyak kelas dan sebaliknya.

✓ 5. Ambil Data dari Relasi

```
$user = User::with('todos')->find(1);
```

⌚ **Analogi:** Lo sekalian bawa daftar tugas si user saat ngambil datanya, biar gak balik-balik ke dapur.

✓ 6. Perintah Migration Dasar

```
php artisan migrate      # Jalankan semua migration  
php artisan migrate:rollback # Mundur satu langkah  
php artisan migrate:refresh # Reset & ulang semua migrasi  
php artisan migrate:fresh  # Hapus semua tabel & migrasi ulang
```

✓ 7. Seeder & Factory Combo (Isi Database Dummy)

```
php artisan make:seeder UserSeeder  
php artisan make:factory UserFactory --model=User
```

Lalu panggil di DatabaseSeeder:

```
User::factory()->count(10)->create();
```

⌚ **Analogi:** Lo bikin penghuni palsu buat ngetes rumah Laravel sebelum dipake beneran.

🔒 Laravel Role & Permission Cheat Sheet

✓ 1. Pengantar: Apa Itu Role & Permission?

- **Role = jabatan**

Contoh: SuperAdmin, Admin, Editor, Owner, Customer, Pelanggan, User Biasa, dsb.

- **Permission = izin kerja**

Contoh: Bisa tambah data, bisa hapus data, bisa lihat data doang.

✓ 2. Cara Sederhana (Manual - tanpa package)

🛠 Tambah kolom `role` di tabel `users`

```
$table->string('role')->default('user');
```

🌐 Di Controller:

```
if (auth()->user()->role == 'admin') {  
    // kasih akses khusus  
}
```

🔍 **Analogi:** Lo cek kartu identitas dulu sebelum kasih akses ke ruang VIP.

✓ 3. Pakai Package: Spatie Laravel-Permission (Rekomendasi)

📦 Install: `composer require spatie/laravel-permission`

🧠 Publish & migrate:

```
php artisan vendor:publish --  
provider="Spatie\Permission\PermissionServiceProvider"  
php artisan migrate
```

✓ 4. Tambah Trait ke Model User

```
use Spatie\Permission\Traits\HasRoles;  
  
class User extends Authenticatable  
{  
    use HasRoles;  
}
```

✓ 5. Bikin Role & Assign ke User

```
use Spatie\Permission\Models\Role;  
  
$role = Role::create(['name' => 'admin']);  
  
$user->assignRole('admin');
```

✓ 6. Bikin Permission & Assign ke Role

```
use Spatie\Permission\Models\Permission;  
  
Permission::create(['name' => 'edit post']);  
$role->givePermissionTo('edit post');
```

✓ 7. Cek Role & Permission

```
$user->hasRole('admin'); // true/false  
$user->can('edit post'); // true/false
```

🔍 **Analogi:** Cek ID card dan cek SIM sekalian. Punya jabatan & punya izin ngelakuin aksi.

✓ 8. Middleware Berdasarkan Role

```
Route::group(['middleware' => ['role:admin']], function () {  
    Route::get('/admin', [AdminController::class, 'index']);  
});
```

🔍 **Analogi:** Pintu ruangan admin cuma kebuka kalau si user punya jabatan admin.

✓ 9. Assign Multiple Roles

```
$user->assignRole('editor', 'moderator');
```

✓ 10. Role & Permission Check di Blade

```
@role('admin')  
    <a href="/admin">Dashboard Admin</a>  
@endrole  
  
@can('edit post')  
    <button>Edit</button>  
@endcan
```

🔍 **Analogi:** Tampilkan tombol atau fitur cuma kalau orang itu punya izin.