

- Guarded untuk langsung fill tapi tidak bisa lihat di tabel ada apa saja
- Jika ada eror maka ketikkan `protected $table = ('items');`
- Note: items itu diambil dari nama tabel yang eror
- Biar tidak membingungkan codingan kita buat folder dengan nama sesuai dengan table
- Jika nanti gunakan Laravel maka yang sering dibuka yakni controller, view (blade.php) dan web.php
- Eloquent itu yang tulisan ::
- Class itemcontroller extend controller artinya itemcontroller class dari nama file extend controller artinya di controller
- Nama public function harus berbeda
- Item::all fungsi ambil data dari tabel secara keseluruhan
- Return view artinya kembalikan data kemana mau diletakkan datanya dlm hal ini ditaruh di folder view items di dalam index 'items.index', compact untuk melempar data pakai \$data (d disesuaikan dengan nama variable awal)
- Data yang dilempar itemcontroller akan ditangan di index.blade.php dengan forelse maksudnya jika data nya sudah ada lakukan foreach, namun jika data kosong tampilkan kata data tidak ada
- Data di php ada collection dan single
- `Dd($data);` untuk lihat data benar-benar ada atau tidak
- Pakai alias dengan ketikkan `$data as $item`
- `@php` sama dengan `?>php`, tiap dipanggil `@php` harus ditutup `@endphp`
- Manggil di index.php bisa pakai uri atau route

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

```

p User.php create.blade.php index.blade.php web.php x edit.blade.php
outes > web.php > ...
17 Route::get('/', function () {
18     return view('welcome');
19 });
20
21 Route::get('/items', [ItemController::class, 'index'])->name('items.index');
22 Route::get('items/show/{id}', [ItemController::class, 'show'])->name('items.show');
23 Route::get('/items/create', [ItemController::class, 'create'])->name('items.create');
24 Route::post('/items/store', [ItemController::class, 'store'])->name('items.store');
25 Route::get('items/edit/{id}', [ItemController::class, 'edit'])->name('items.edit');
26 Route::post('/items/update/{id}', [ItemController::class, 'update'])->name('items.update');
27 Route::get('items/delete/{id}', [ItemController::class, 'delete'])->name('items.delete');
```

Get untuk ambil data tapi tentukan dulu mau ambil dimana, /di dalam kurung boleh digunakan atau tidak. Fungsi / untuk tampilan di sini



Fungsi route untuk menjembatani.

```
Route::get('/items', [ItemController::class, 'index'])->
>name('items.index');

Items
```

```
@error('name')
    <span class="text-danger">{{ $message }}</span>
@enderror
```

Bisa digunakan bisa juga tidak untuk deteksi eror

Harus ditambahkan **@csrf** crossfield untuk amankan data tarok dibawah form khusus di form saja

EDIT

- Ada Parameter yakni id

Pada bagian web.php ada penambahan `Route::post('/items/update/{id}`

Id disitu boleh bebas penulisan yang penting di itemcontroller sama penulisannya
Ditampung dulu data biar idnya bisa dilempar

```
$data = Item::find($id);
```

BUAT TABEL BARU

\$ php artisan make:model Mahasiswa -mrc

Untuk refresh

\$php artisan migrate:fresh