

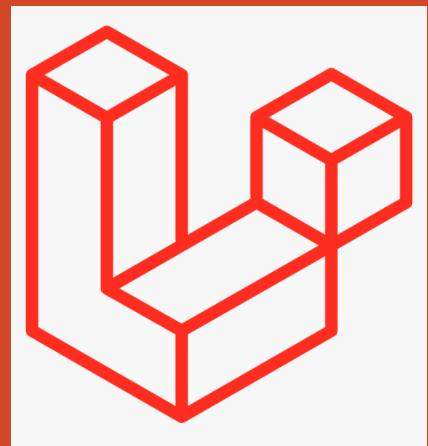


# Laravel 10

# Starting a

# PHP Framework

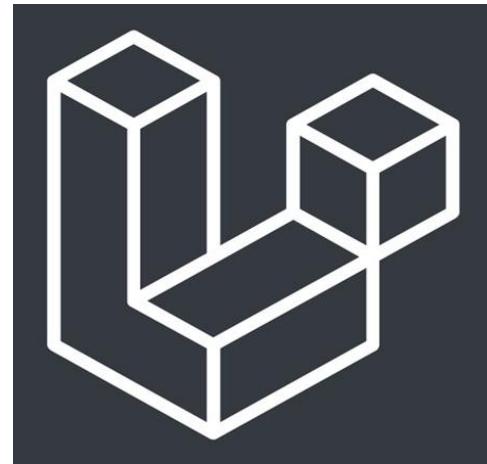
# Project in Laravel



# Laravel Project

1. Laravel 10 Project using Laragon
  - `composer create-project laravel/laravel projectname`
2. Laravel/UI package
  - `composer require laravel/breeze`
3. Auth scaffolding
  - `php artisan ui bootstrap`





**Laravel** Breeze

# Laravel Breeze



## 5. Login UI Laravel/Breeze

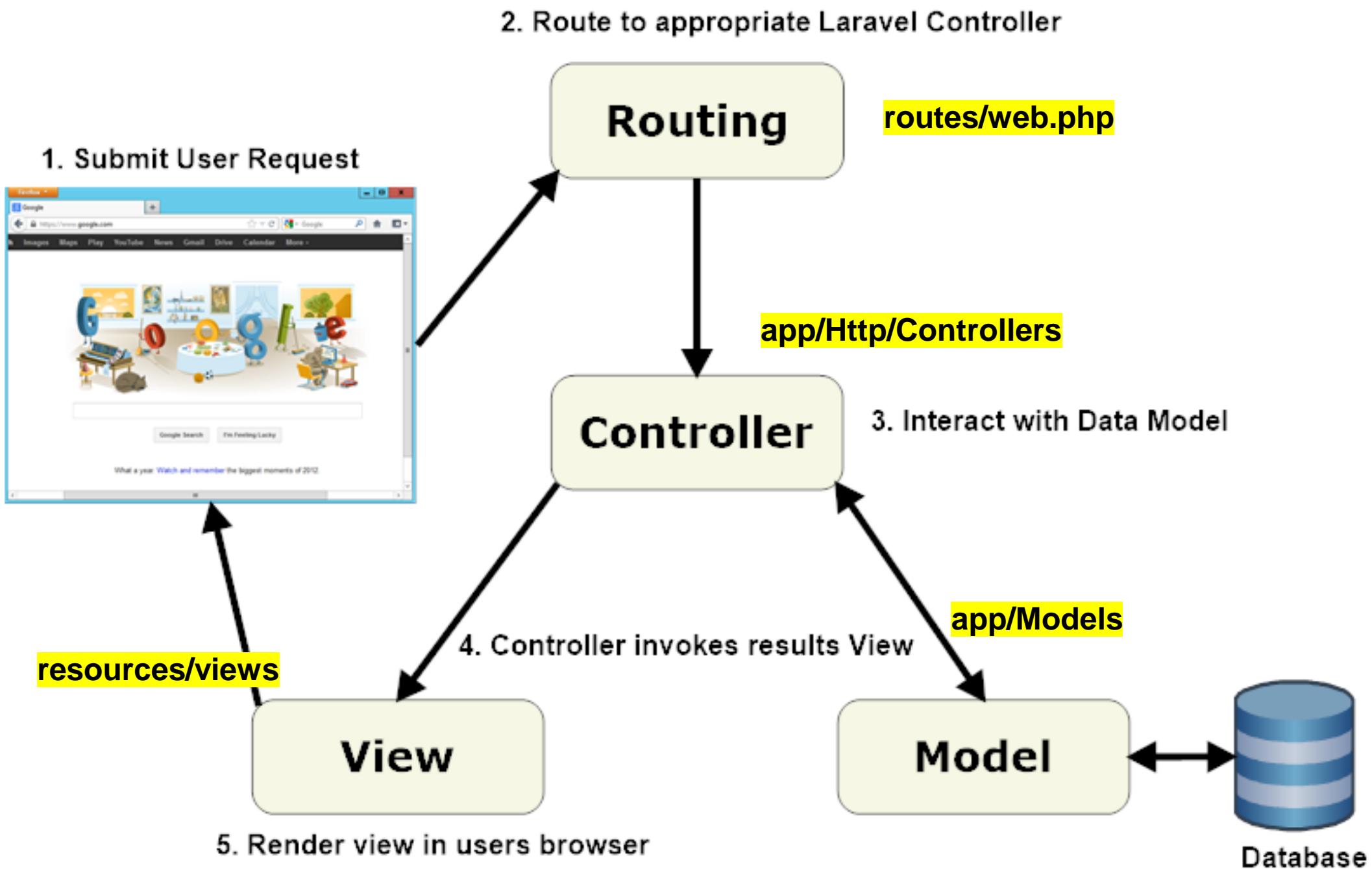
- <https://laravel.com/docs/10.x/starter-kits> (documentation)
- `composer require laravel/breeze -dev`
- `php artisan breeze:install`
- `npm install`
- `npm run dev`



## 7. Database – additional users table migrate

- `php artisan migrate`

## 8. Test again to register / login



# <https://github.com/khirulnizam/10tbtm/>

The screenshot shows a GitHub repository page for the project '10tbtm'. The repository is owned by 'khirulnizam' and is public. It has 1 branch and 0 tags. The main branch is 'main'. The repository was last updated yesterday by 'khirulnizam' with 2 commits. The commit details are as follows:

File/Folder	Commit Message	Date
app	create-xlayout	yesterday
bootstrap	Initial commit	yesterday
config	Initial commit	yesterday
database	Initial commit	yesterday
public	Initial commit	yesterday

At the top of the page, there are several tabs: Code, Issues, Pull requests, Actions, Projects, Security, Insights, and Settings. The 'Code' tab is currently selected. There is also a search bar at the top right.

# Module Outcome

## Day 1

- | <b>Day 1</b>               | <b>What will be needed</b> |
|----------------------------|----------------------------|
| • Create Laravel Project   | • Laragon                  |
| • Working with Database.   | • phpMyAdmin               |
| • Working with Model       | • VS Code                  |
| • Working with Controllers |                            |
| • Working with Views       |                            |

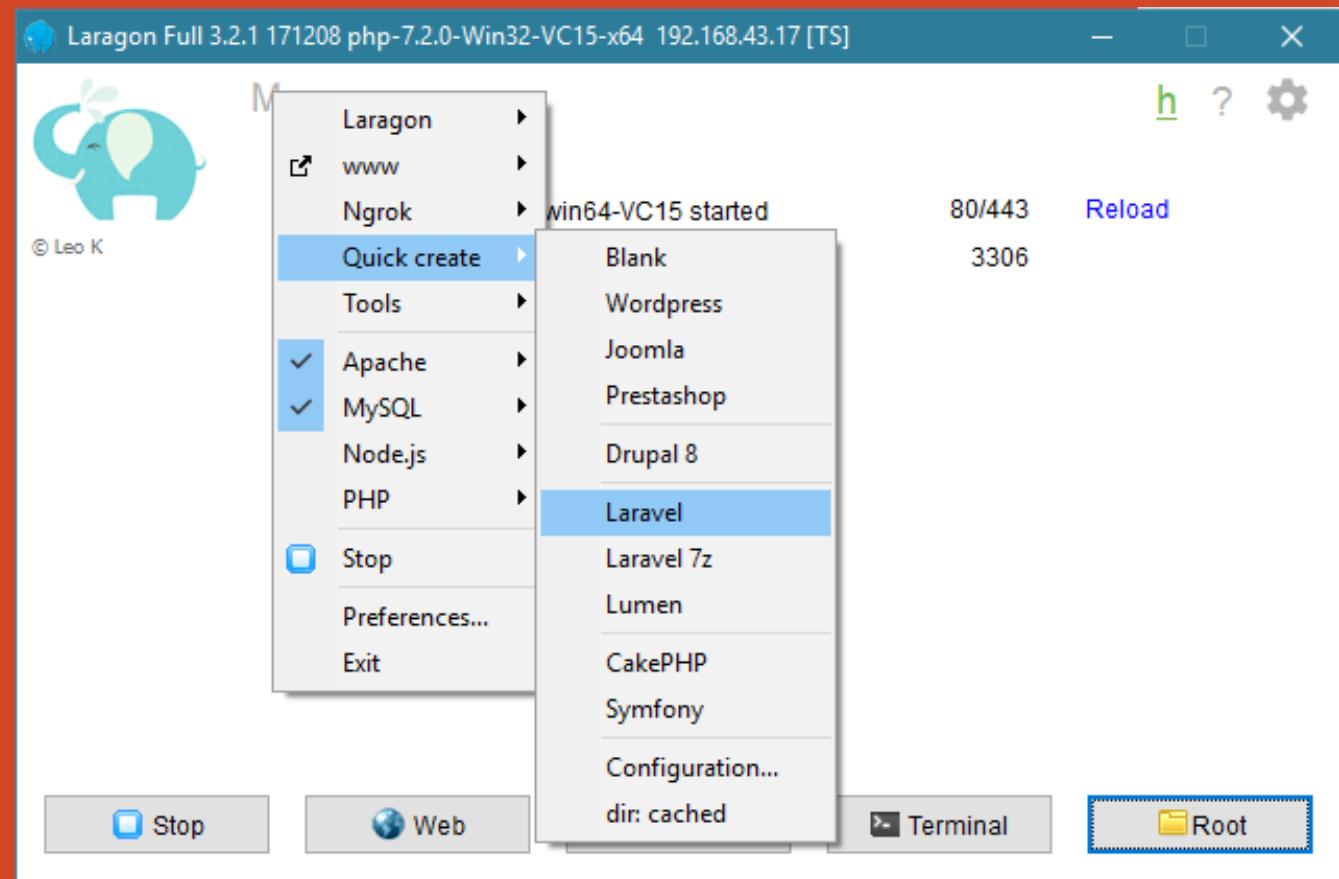
# Create Laravel First Project

Install Laravel by issuing the Composer create-project command in your terminal

```
composer create-project --prefer-dist laravel/laravel project-name
```

# laragon.org

With Laragon?  
Never been EASIER.



# Laragon

## What is Laragon?

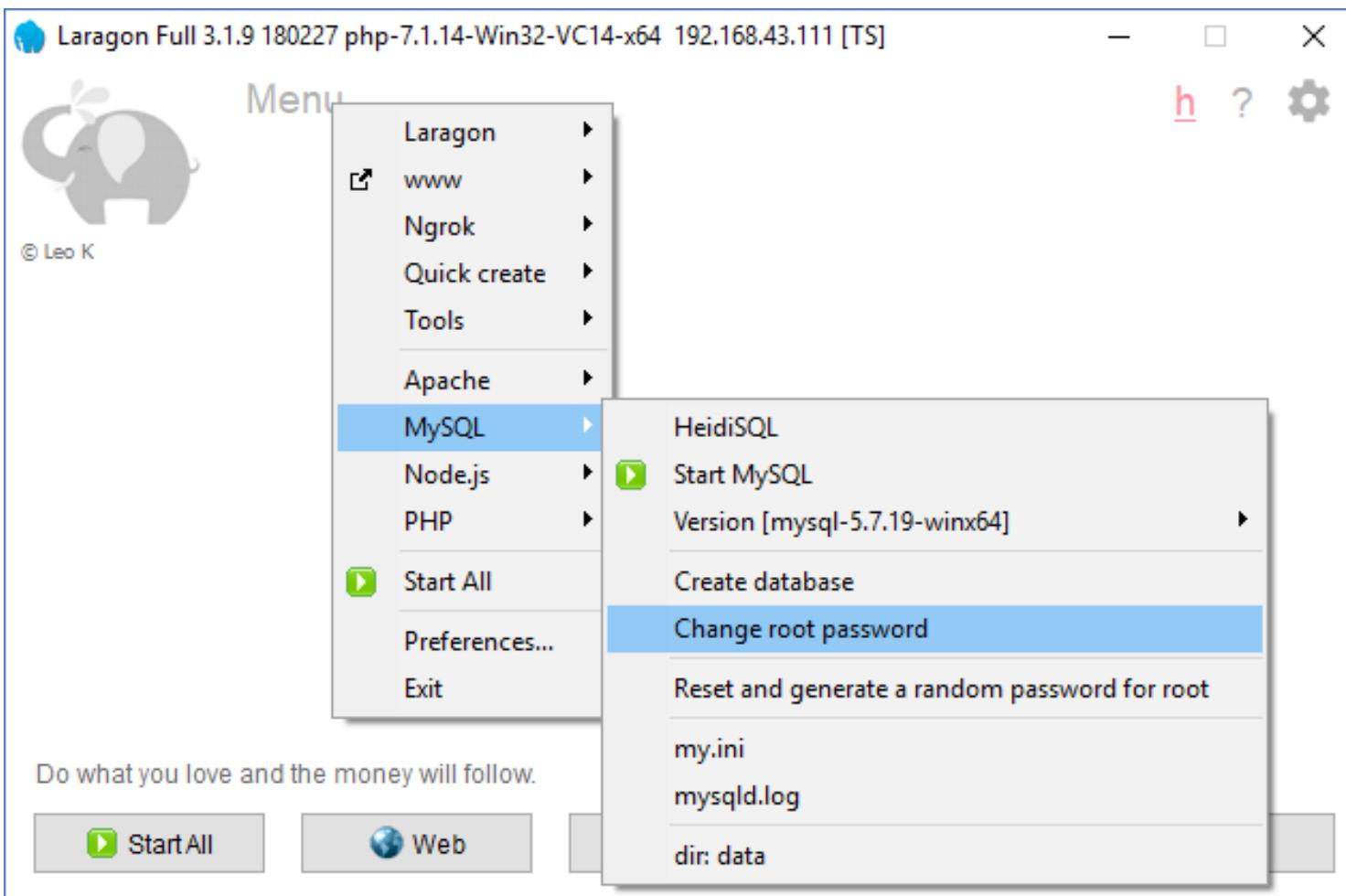
Laragon is a portable, isolated, fast & powerful universal development environment for PHP, Node.js, Python, Java, Go, Ruby. It is fast, lightweight, easy-to-use and easy-to-extend.

## Why Laragon?

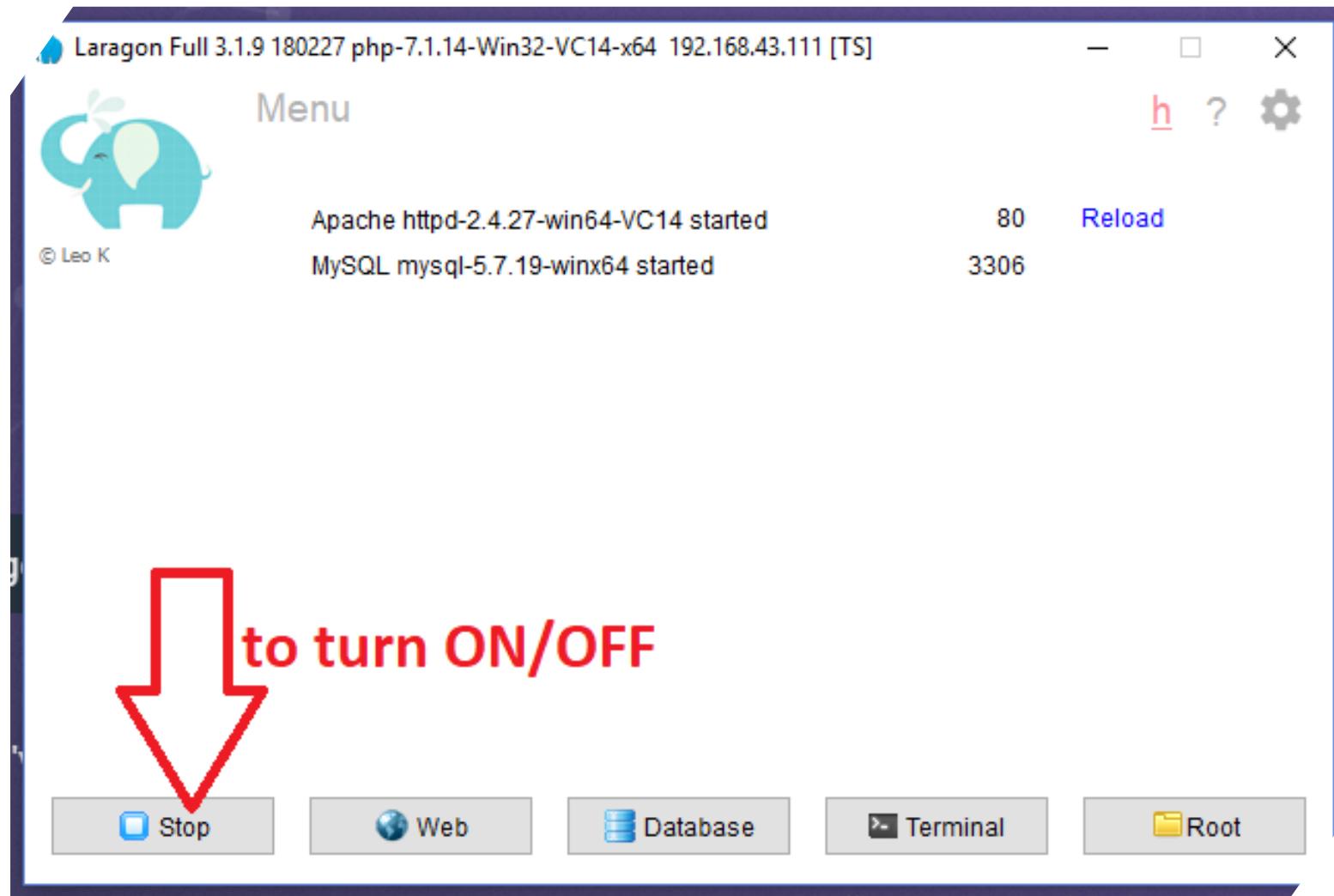
Laragon is great for building and managing modern web applications. It is focused on performance - designed around stability, simplicity, flexibility and freedom.

Laragon improves web development. Developers all over the world are using Laragon to make apps quickly and easily. It is used by thousands of developers with loves. You can check out the Testimonials to see how users think of Laragon and it's features page for more details.

# Change DB root password

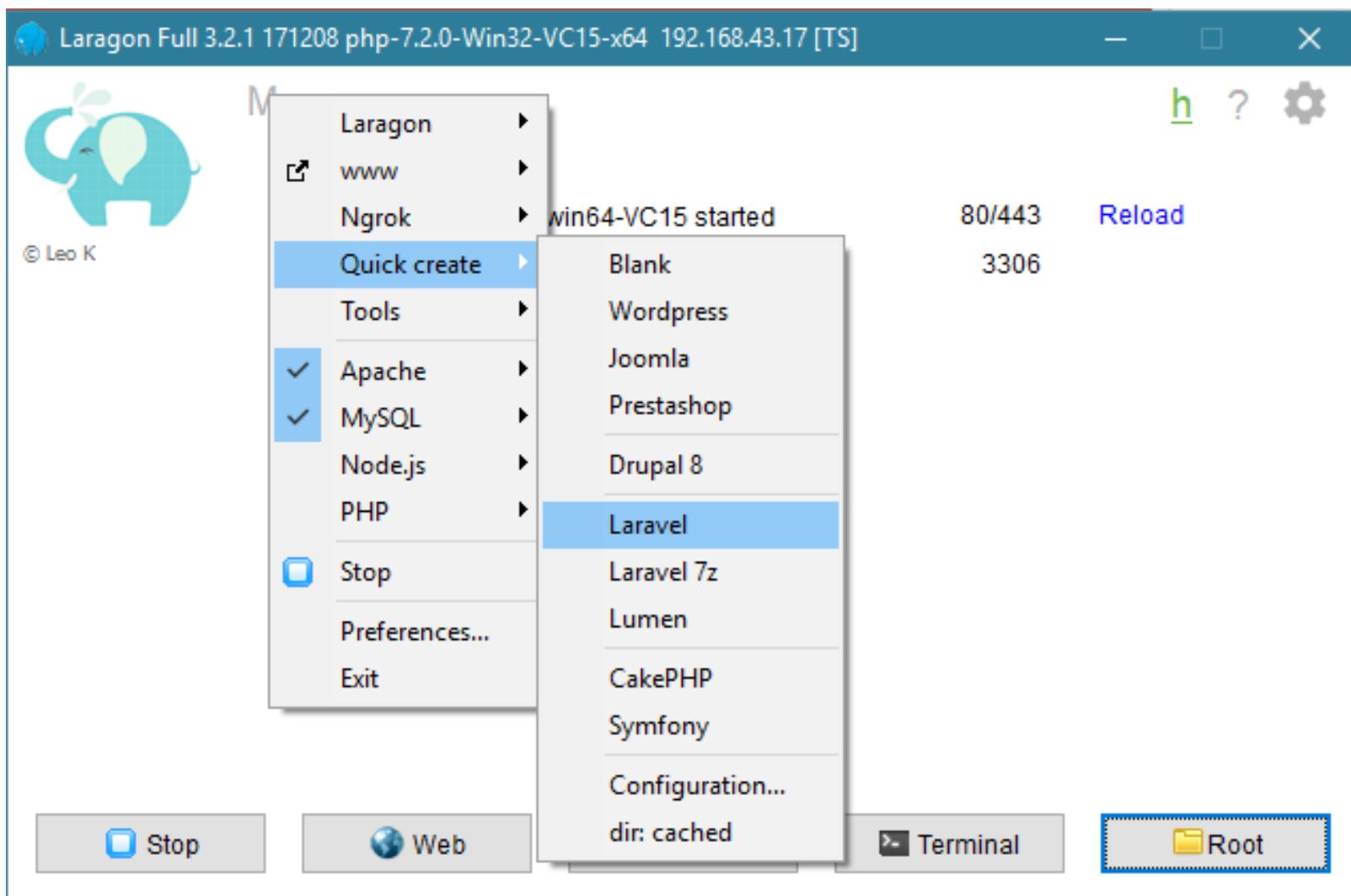


# Turn On Laragon

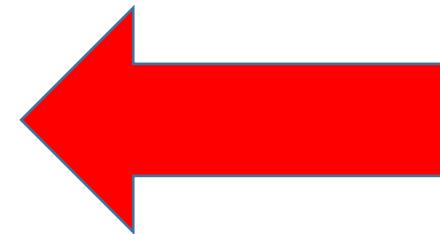


By default Laragon will serve port 80 for Apache and port 3306 for DB

# Create Project



# Laravel project with Pretty URL



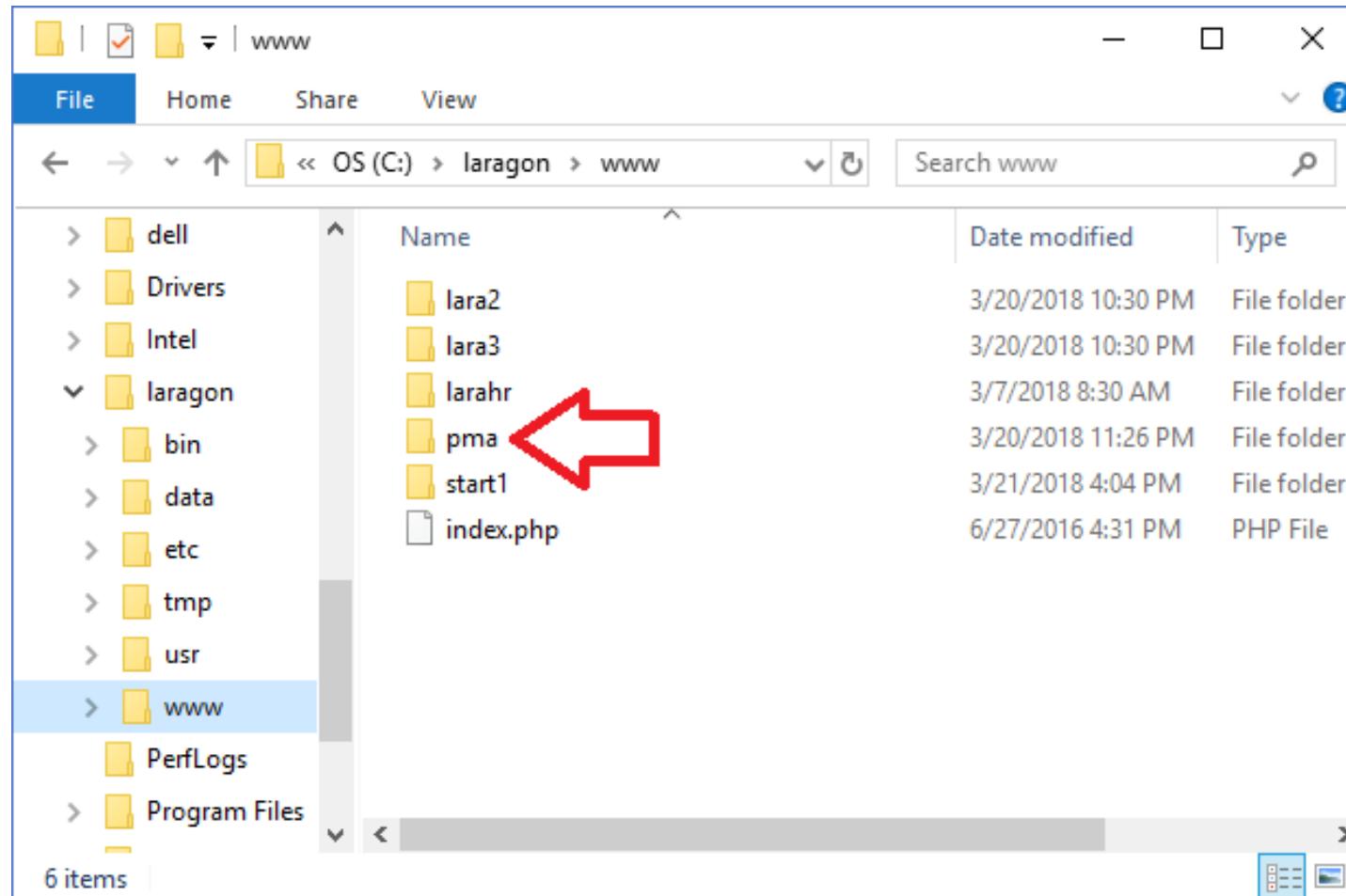
Test the Pretty url in browser

<http://fstm.kuis.edu.my/blog/laravel>

# Update DB credentials in .env

```
8  
9 DB_CONNECTION=mysql  
10 DB_HOST=127.0.0.1  
11 DB_PORT=3306  
12 DB_DATABASE=start1  
13 DB_USERNAME=root  
14 DB_PASSWORD=abc123  
15
```

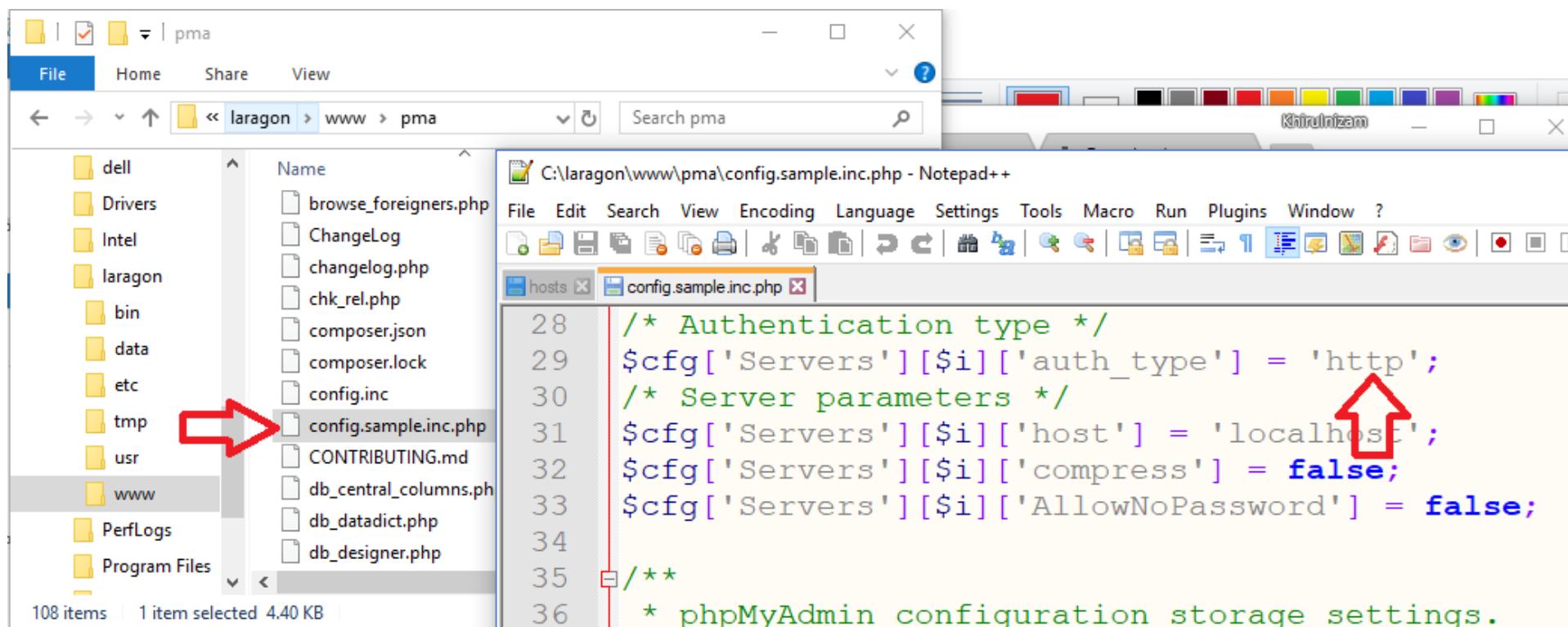
# Download and install phpmyadmin.Net



Try access

<http://localhost/pma>

# Phpmyadmin config.inc



# Authentication Module

Laravel makes implementing authentication very simple. In fact, almost everything is configured for you out of the box.

```
composer require laravel/ui --dev  
  
php artisan ui vue --auth  
  
npm install  
  
npm run dev
```

# Authentication Module

The screenshot shows a code editor with a file tree on the left and a code editor window on the right. The code editor window displays a PHP migration file for creating a 'users' table. The terminal window at the bottom shows the output of running the 'artisan migrate' command.

File Tree:

- database
- factories
- migrations
  - 2014\_10\_12\_000000\_create\_users\_table
  - 2014\_10\_12\_100000\_create\_password\_resets\_table
- seeds
- .gitignore
- public
- resources
- routes
- storage
- tests
- vendor
- .env

Code Editor (Migration File Content):

```
14
15
16
17
18
19
20
21
22
23
```

```
public function up()
{
    Schema::create('users', function (Blueprint $table)
    {
        $table->increments('id');
        $table->string('name');
        $table->string('email')->unique();
        $table->string('password');
        $table->rememberToken();
        $table->timestamps();
    });
}
```

Terminal Window Output:

```
C:\laragon\www\start1>php artisan migrate
Migration table created successfully.
Migrating: 2014_10_12_000000_create_users_table
Migrated: 2014_10_12_000000_create_users_table
Migrating: 2014_10_12_100000_create_password_resets_table
Migrated: 2014_10_12_100000_create_password_resets_table
```

A red arrow points from the terminal window output to the 'artisan migrate' command in the code editor's terminal window.

Login

E-Mail Address

Password

Remember Me

**Login** [Forgot Your Password?](#)

Reset Password

E-Mail Address

**Send Password Reset Link**

Register

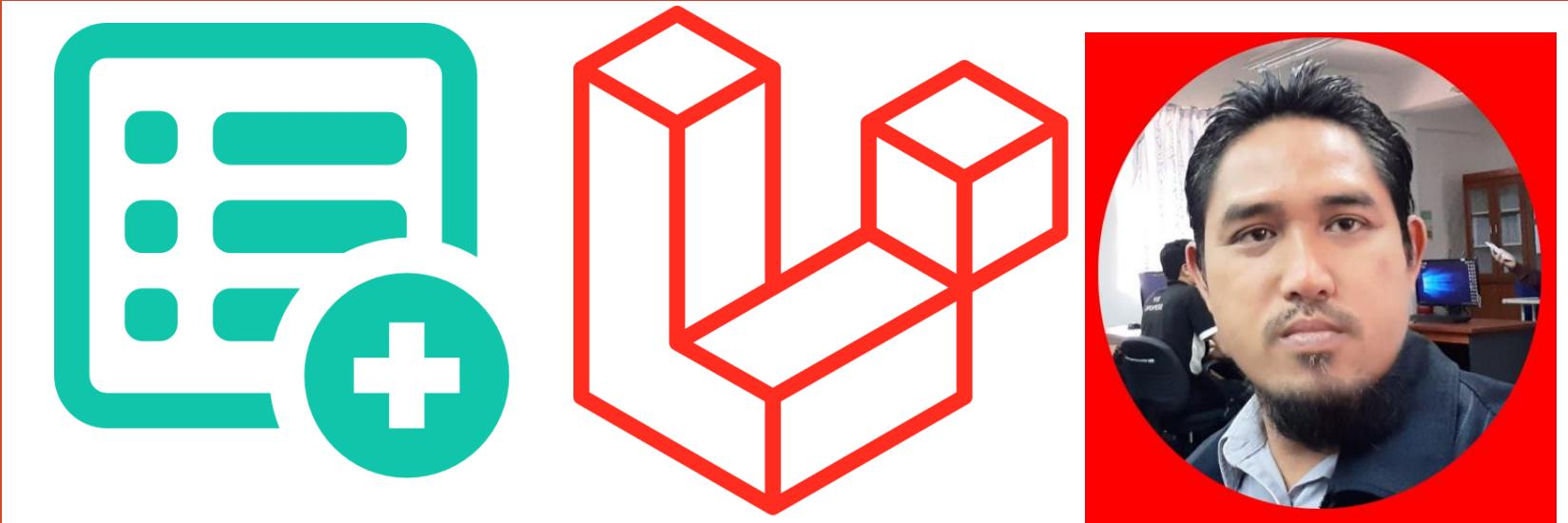
Name

E-Mail Address

Password

Confirm Password

**Register**



# Create a New Record Form

[github.com/khirulnizam](https://github.com/khirulnizam)



1. Submit User Request

2. Route to appropriate Laravel Controller

## Routing

`routes/web.php`

## Controller

`app/Http/Controllers`

3. Interact with Data Model

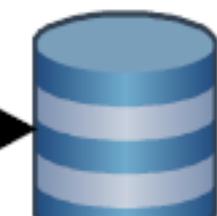
`app/Models`

## View

`resources/views`

4. Controller invokes results View

## Model



Database

5. Render view in users browser

# Create new record interface (create.blade.php)

## Create Training

Create Training

Title

Description

Trainer

[Create New Training](#) [Cancel](#)

# Views

Views contain the HTML served by your application and separate your controller / application logic from your presentation logic.

Views are stored in the resources/views directory

1. Go to resources/views
2. Create training folder
3. Inside training folder, create:
  1. create.blade.php
  2. index.blade.php

```
✓ resources
  > js
  > lang
  > sass
  ✓ views
    > admin
    > auth
    > layouts
    ✓ trainings
      └─ create.blade.php
      └─ index.blade.php
```

# Views – Create View

Views are stored in the resources/views directory

1. Go to

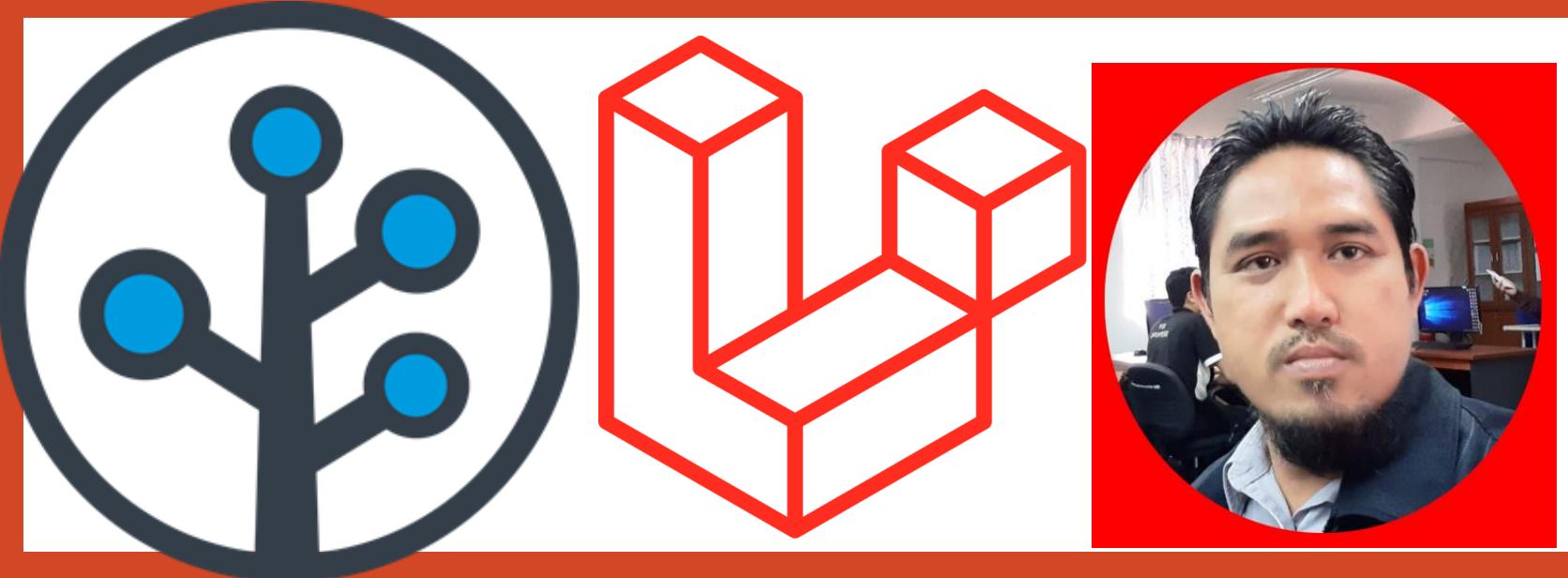
`resources/views/trainings/create.blade.php`

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

@section('content')
<div class="container">
    <div class="row justify-content-center">
        <div class="col-md-8">
            <div class="card">
                <div class="card-header">Create Training</div>

                <div class="card-body">
                    <form action="" method="POST">
                        @csrf
                        <div class="form-group">
                            <label for="title">Title</label>
                            <input type="text" class="form-control" name="title">
                        </div>
                        <div class="form-group">
                            <label for="description">Description</label>
                            <textarea name="description" cols="20" rows="10" class="form-control"></textarea>
                        </div>
                        <div class="form-group">
                            <label for="trainer">Trainer</label>
                            <input type="text" class="form-control" name="trainer">
                        </div>
                        <div class="form-group">
                            <button type="submit" class="btn btn-primary">Create New Training</button>
                            <a href="" class="btn btn-link">Cancel</a>
                        </div>
                    </form>
                </div>

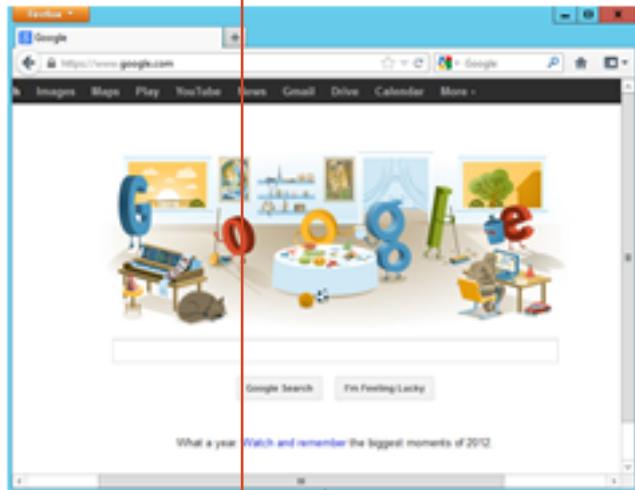
            </div>
        </div>
    </div>
</div>
@endsection
```



# Controller & Simple Route

[github.com/khirulnizam](https://github.com/khirulnizam)

1. Submit User Request



2. Route to appropriate Laravel Controller

**Routing**

`routes/web.php`

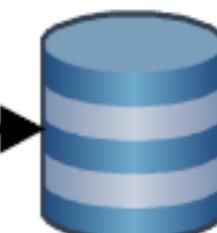
**Controller**

`app/Http/Controllers`

3. Interact with Data Model

`app/`

**Model**



Database

`resources/views`

4. Controller invokes results View

**View**

5. Render view in users browser

# Controller

Instead of defining all of your request handling logic as Closures in route files, you may wish to organize this behavior using Controller classes.

Controllers can group related request handling logic into a single class.

Controllers are stored in the app/Http/Controllers directory.

✓ app
> Console
> Exceptions
✓ Http
✓ Controllers
✓ Admin
🐘 BookingsController.php
🐘 CategoryController.php
🐘 CountriesController.php
🐘 CustomersController.php
🐘 FindRoomsController.php
🐘 RolesController.php
🐘 RoomsController.php

# Controller

7 method in controllers

## Make Controller

```
λ php artisan make:controller TrainingController --resource
```

### Action

index

create

store

show

edit

update

destroy

1. Submit User Request



2. Route to appropriate Laravel Controller

**Routing**

**routes/web.php**

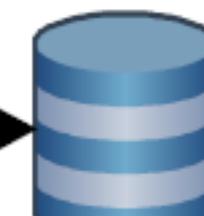
**app/Http/Controllers**

**Controller**

3. Interact with Data Model

**app/**

**Model**



**Database**

**resources/views**

**View**

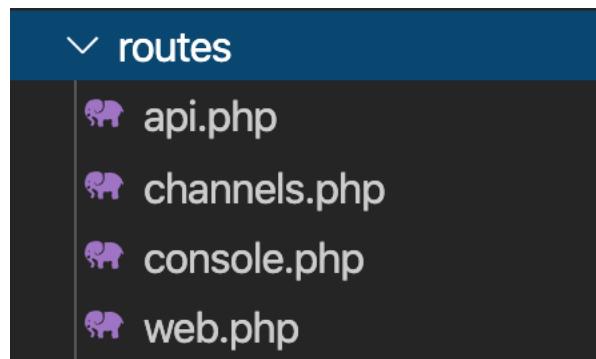
4. Controller invokes results View

5. Render view in users browser

# Routing

All Laravel routes are defined in your route files, which are located in the routes directory.

<https://laravel.com/docs/10.x/routing>

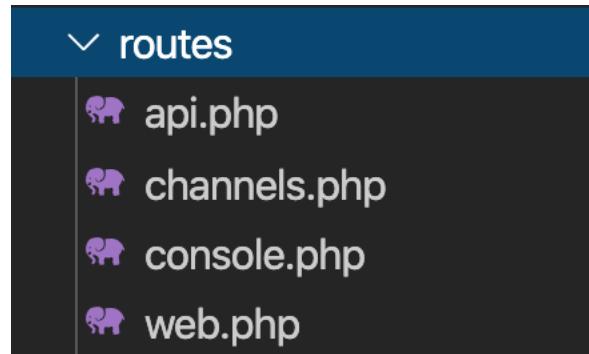


```
Route::get($uri, $callback);  
Route::post($uri, $callback);  
Route::put($uri, $callback);  
Route::patch($uri, $callback);  
Route::delete($uri, $callback);  
Route::options($uri, $callback);
```

# Simple Routing

All Laravel routes are defined in your route files, which are located in the routes directory.

<https://laravel.com/docs/10.x/routing>



# Simple routing ( routes/web.php )

## Laravel 10

```
web.php  X
routes > web.php

17
18 //route to call TrainingController@create
19 //laravel8 redo route declaration
20 Route::get('/training/create', [
21     TrainingController::class, 'create'
22 ]->name('training:create'));
23
24 Route::get('/', function () {
25     return view('welcome');
26 });
http://fstm.kuis.edu.my/blog/laravel
```

# Simple Routing

php artisan route:list

GET HEAD   trainings	training:index	App\Http\Controllers\TrainingController@index	web,auth
GET HEAD   trainings/create	training:create	App\Http\Controllers\TrainingController@create	web,auth
POST   trainings/create	training@store	App\Http\Controllers\TrainingController@store	web,auth

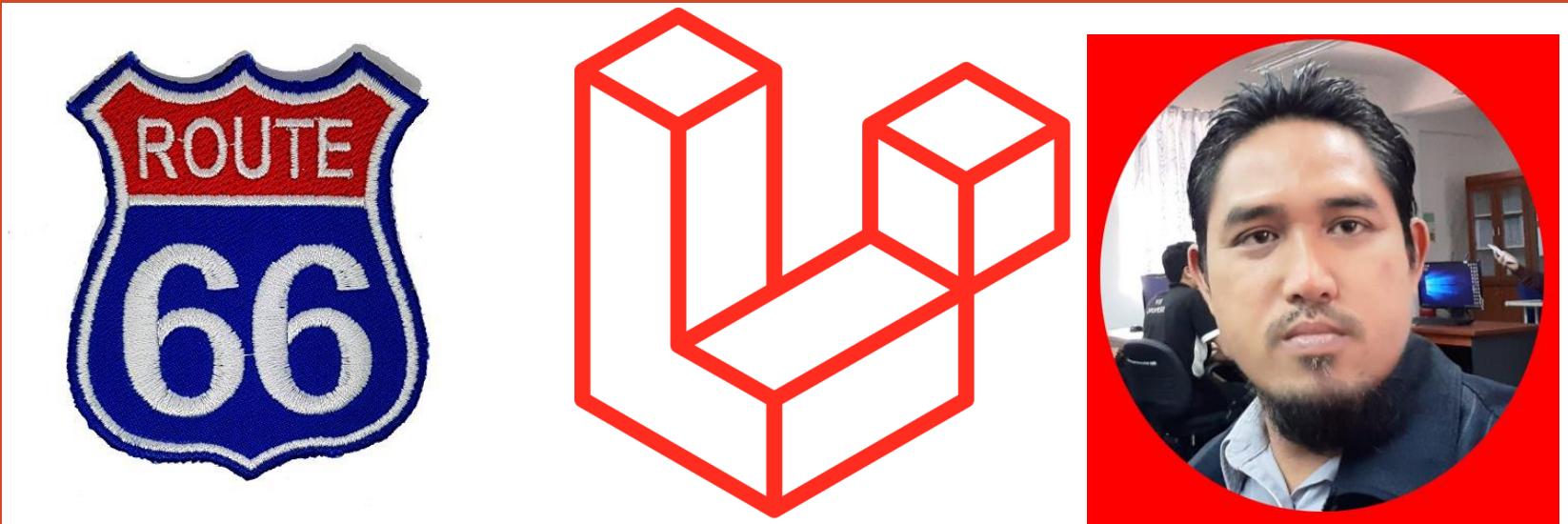
# php artisan route:list

Domain	Method	URI	Name	Action	Middleware
	GET HEAD	/		Closure	web
	GET HEAD	api/user		Closure	api,auth:api
	GET HEAD	home	home	App\Http\Controllers\HomeController@index	web,auth
	GET HEAD	login	login	App\Http\Controllers\Auth\LoginController@showLoginForm	web,guest
	POST	login		App\Http\Controllers\Auth\LoginController@login	web,guest
	POST	logout	logout	App\Http\Controllers\Auth\LoginController@logout	web
	POST	password/confirm		App\Http\Controllers\Auth\ConfirmPasswordController@confirm	web,auth
	GET HEAD	password/confirm	password.confirm	App\Http\Controllers\Auth\ConfirmPasswordController@showConfirmForm	web,auth
	POST	password/email	password.email	App\Http\Controllers\Auth\ForgotPasswordController@sendResetLinkEmail	web
	POST	password/reset	password.update	App\Http\Controllers\Auth\ResetPasswordController@reset	web
	GET HEAD	password/reset	password.request	App\Http\Controllers\Auth\ForgotPasswordController@showLinkRequestForm	web
	GET HEAD	password/reset/{token}	password.reset	App\Http\Controllers\Auth\ResetPasswordController@showResetForm	web
	POST	register		App\Http\Controllers\Auth\RegisterController@register	web,guest
	GET HEAD	register	register	App\Http\Controllers\Auth\RegisterController@showRegistrationForm	web,guest
	GET HEAD	trainings	training:index	App\Http\Controllers\TrainingController@index	web
	GET HEAD	trainings/create	training:create	App\Http\Controllers\TrainingController@create	web
	POST	trainings/create	training:store	App\Http\Controllers\TrainingController@store	web

# Advanced ROUTING: group & named

# php artisan route:list

```
C:\laragon\www\lara7
λ php artisan route:list
+---+---+---+---+---+---+
| Domain | Method | URI | Name | Action | Middleware |
+---+---+---+---+---+---+
| GET|HEAD | / | Closure | |
| GET|HEAD | api/user | Closure |
| GET|HEAD | home | home | App\Http\Controllers\HomeController@index |
| GET|HEAD | login | login | App\Http\Controllers\Auth\LoginController@showLoginForm |
| POST | login | login | App\Http\Controllers\Auth\LoginController@login |
| POST | logout | logout | App\Http\Controllers\Auth\LoginController@logout |
| POST | password/confirm | password.confirm | App\Http\Controllers\Auth\ConfirmPasswordController@confirm |
| GET|HEAD | password/confirm | password.confirm | App\Http\Controllers\Auth\ConfirmPasswordController@showConfirmForm |
| POST | password/email | password.email | App\Http\Controllers\Auth\ForgotPasswordController@sendResetLinkEmail |
| POST | password/reset | password.update | App\Http\Controllers\Auth\ResetPasswordController@reset |
| GET|HEAD | password/reset | password.request | App\Http\Controllers\Auth\ForgotPasswordController@showLinkRequestForm |
| GET|HEAD | password/reset/{token} | password.reset | App\Http\Controllers\Auth\ResetPasswordController@showResetForm |
| POST | register | register | App\Http\Controllers\Auth\RegisterController@register |
| GET|HEAD | register | register | App\Http\Controllers\Auth\RegisterController@showRegistrationForm |
| GET|HEAD | trainings | training:index | App\Http\Controllers\TrainingController@index |
| GET|HEAD | trainings/create | training:create | App\Http\Controllers\TrainingController@create |
| POST | trainings/create | training:store | App\Http\Controllers\TrainingController@store |
| GET|HEAD | trainings/{training} | training:show | App\Http\Controllers\TrainingController@store('create'); |
| POST | trainings/{training} | training:update | App\Http\Controllers\TrainingController@store('update'); |
| POST | trainings/{training}/delete | training:delete | App\Http\Controllers\TrainingController@store('show'); |
| GET|HEAD | trainings/{training}/edit | training:edit | App\Http\Controllers\TrainingController@store |
+---+---+---+---+---+---+
```



# Route resource & index

[github.com/khirulnizam](https://github.com/khirulnizam)

1. Submit User Request



2. Route to appropriate Laravel Controller

**Routing**

**routes/web.php**

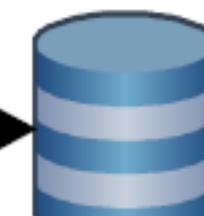
**app/Http/Controllers**

**Controller**

3. Interact with Data Model

**app/Models**

**Model**



**Database**

**View**

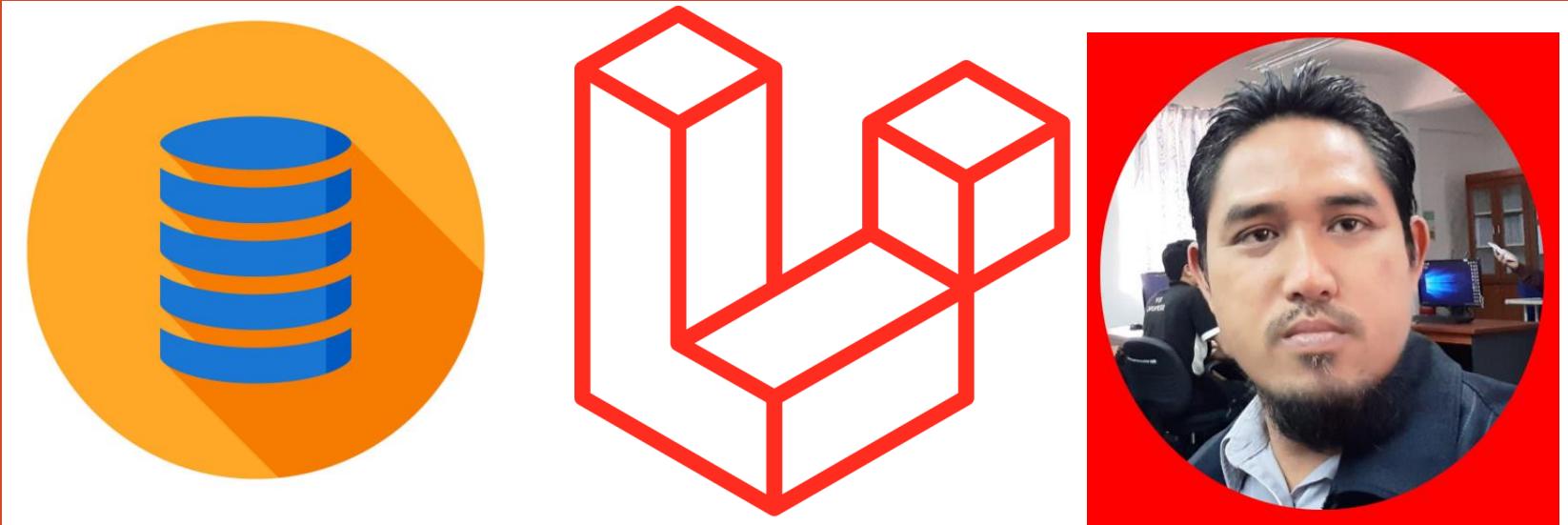
4. Controller invokes results View

**resources/views**

5. Render view in users browser

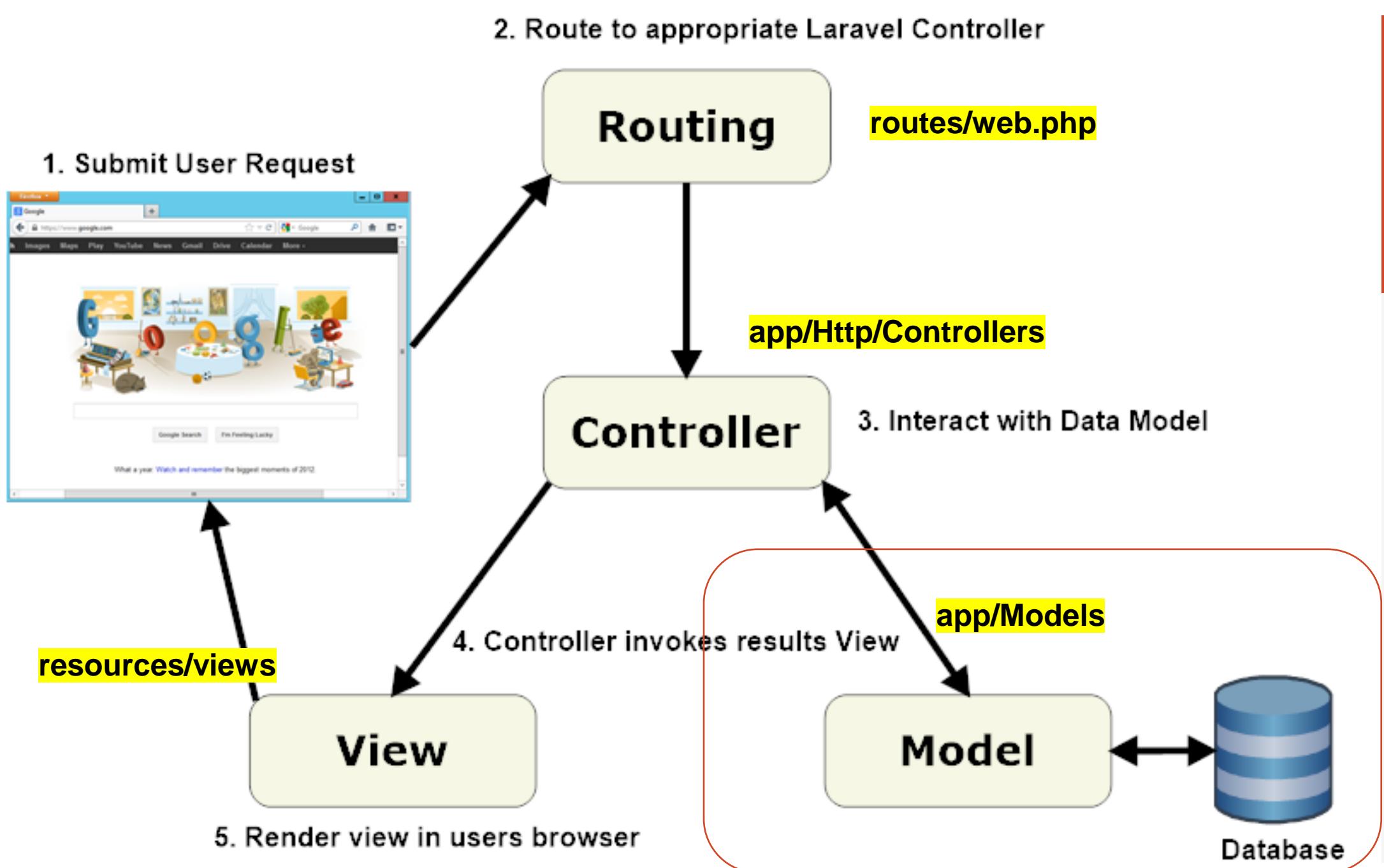
# Route resource

```
routes > 🐾 web.php
12 | Here is where you can register web routes for your application. These
13 | routes are loaded by the RouteServiceProvider within a group which
14 | contains the "web" middleware group. Now create something great!
15 |
16 */
17
18 //route resource for TrainingController
19 Route::resource('training', '\App\Http\Controllers\TrainingController');
20
```



# Model & Database/Table

[github.com/khirulnizam](https://github.com/khirulnizam)



# Create new record interface (create.blade.php)

## Create Training

Create Training

Title

Description

Trainer

[Create New Training](#) [Cancel](#)

# Model

Models typically live in the app directory.

All Eloquent models  
extend Illuminate\Database\Eloquent\Model class.

## Model

```
php artisan make:model Training
```

```
class Training extends Model
{
    use HasFactory;

    protected $table="trainings";

    protected $fillable=['title','desc', 'trainer'];
}
```

# Model

```
protected $fillable = [  
    'title', 'description', 'trainer', 'user_id', 'attachment'  
];
```

```
public function user()  
{  
    return $this->belongsTo('App\User');
```

```
protected $table= 'articles';
```

```
//define $article->submitted_date  
public function getSubmittedDateAttribute(){  
    return $this->created_at->format('d/m/Y');  
}
```

```
public function scopePublished($query){  
    return $query->where('published',1);  
}
```

Model used to define fillable attributes.

Model also can be used to define relationship

Model also can be used to specify custom table.

Model also can be used to define mutator

Model also can be used to define local scope

# Working with Database

## Database Seeds

Laravel includes a simple method of seeding your database with test data using seed classes.

On [database/seeds/TrainingsTableSeeder.php](#)

```
public function run()
{
    DB::table('trainings')->insert([
        'title' => 'Laravel Training 6 Days(Advanced)',
        'description' => 'The 6 Days training focus on enhancement API to Restful Architecture',
        'trainer' => 'Khirulnizam',
        'user_id' => factory('App\User')->create()->id,
    ]);
}
```

## Make Seeder

```
php artisan make:seeder TrainingsTableSeeder
```

On [database/seeds/DatabaseSeeder.php](#)

```
public function run()
{
    $this->call(TrainingsTableSeeder::class);
}
```

# Working with Database

## Database Migrations

Migrations are like version control for your database.

```
class CreateTrainingsTable| extends Migration
{
    /**
     * Run the migrations.
     *
     * @return void
     */
    public function up()
    {
        Schema::create('trainings', function (Blueprint $table) {
            $table->id();
            $table->timestamps();

            $table->string('title');
            $table->string('description');
            $table->string('trainer');
        });
    }
}
```

<http://fstm.kuis.edu.my/blog/laravel>

# php artisan migrate

## Run Migrations

```
php artisan migrate
```

# HeidiSQL – heidisql.com

The screenshot shows the HeidiSQL Portable 10.2.0.5599 interface. On the left, a tree view displays databases: laravel7, lara7, lara8, laraspr6, laravel7, and mycompanyhr. The 'users' table under the 'lara8' database is selected, highlighted with a gray background. The main panel shows the 'Basic' tab of the 'Table: users' configuration window. The table name is 'users'. The columns section lists three columns:

#	Name	Datatype	Length/Set	Unsigned	Allow N...	Zerofill	Default	Con...
1	<b>id</b>	<b>BIGINT</b>	<b>20</b>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<b>AUTO_INCREMENT</b>	
2	name	VARCHAR	255	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	No default	
3	email	VARCHAR	255	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	No default	

At the bottom, the status bar shows the query number 40, the SQL command 'SHOW CREATE TABLE `lara8`.`users`;', the URL 'http://fstm.kuis.edu.my/blog/laravel', and connection information: Connected: 00 MariaDB or MySQL 5.7.2 Uptime: 00:01 h. The status bar also indicates the server time is 8:00 and it is idle.

# Working with Database : run seeder

## Run Seeder

```
php artisan db:seed
```

# Working with Database

## Database Factories

Laravel has a feature called model factories that allows you to build fake data for your models.

### Make Factory

```
php artisan make:factory TrainingFactory
```

```
use App\Training;
use Faker\Generator as Faker;

(factory->define(Training::class, function (Faker $faker) {
    return [
        'title' => $faker->sentence(5),
        'description' => $faker->text(),
        'trainer' => $faker->name,
        'user_id' => factory('App\User')->create()->id,
    ];
});
```

### Run Factory

```
php artisan tinker
factory('App\Training',10)->create();
```

## How to create and run migrations, seeder and factory?

Here are the example.

### Make Migrations

```
php artisan make:migration create_trainings_table  
php artisan make:migration alter_articles_table_add_attachment
```

### Run Migrations

```
php artisan migrate
```

### Make Seeder

```
php artisan make:seeder TrainingsTableSeeder
```

### Run Seeder

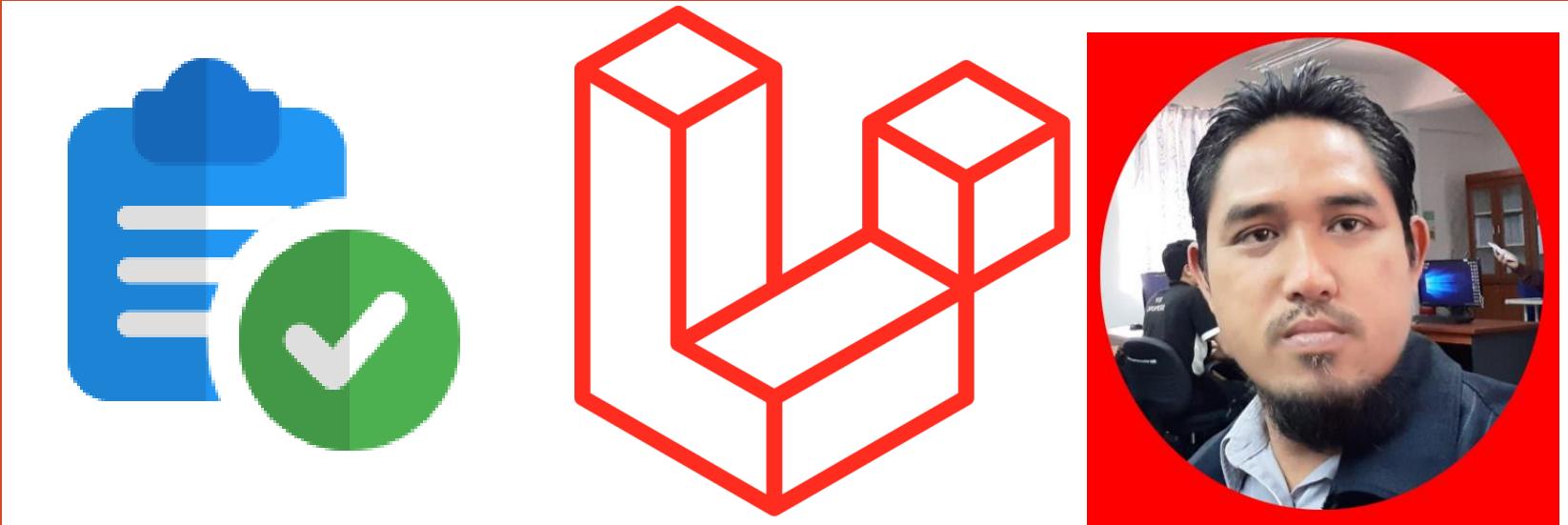
```
php artisan db:seed
```

### Make Factory

```
php artisan make:factory TrainingFactory
```

### Run Factory

```
php artisan tinker  
factory('App\Training',10)->create();
```



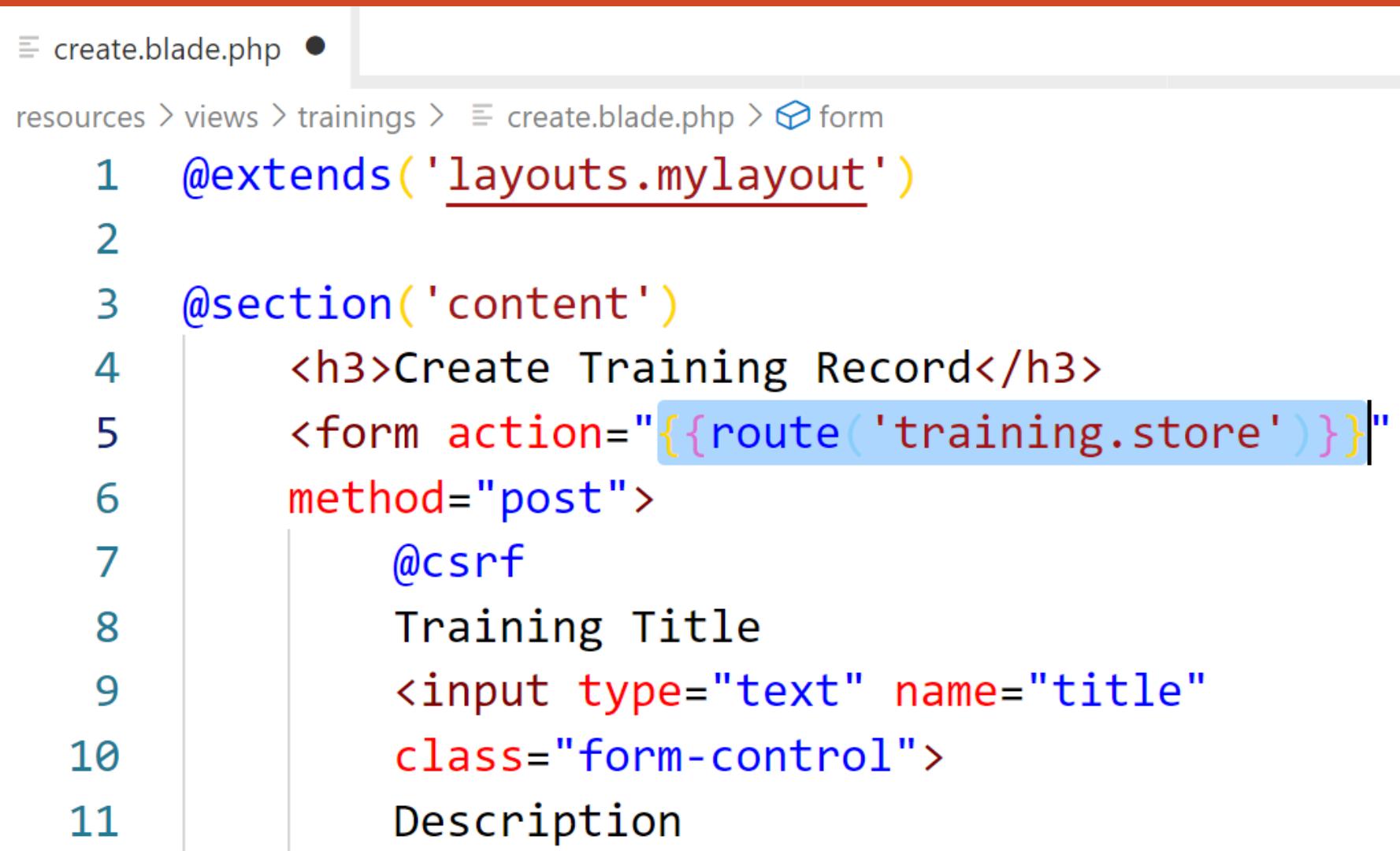
# Insert Form & Store New Record

[github.com/khirulnizam](https://github.com/khirulnizam)

# Controller - Create

Description	URL	Controller Function	View File
Show the form to create a new training.	GET example.com/trainings/create	create() <pre>public function create() {     return view('trainings.create'); }</pre>	app/views/trainings/create.blade.php

# Action path



The screenshot shows a code editor with a file named `create.blade.php` open. The file is located at `resources > views > trainings > create.blade.php`. The code itself is a Laravel blade template:

```
1 @extends('layouts.myLayout')
2
3 @section('content')
4     <h3>Create Training Record</h3>
5     <form action="{{ route('training.store') }}" method="post">
6         @csrf
7         Training Title
8         <input type="text" name="title" class="form-control">
9
10        Description
11
```

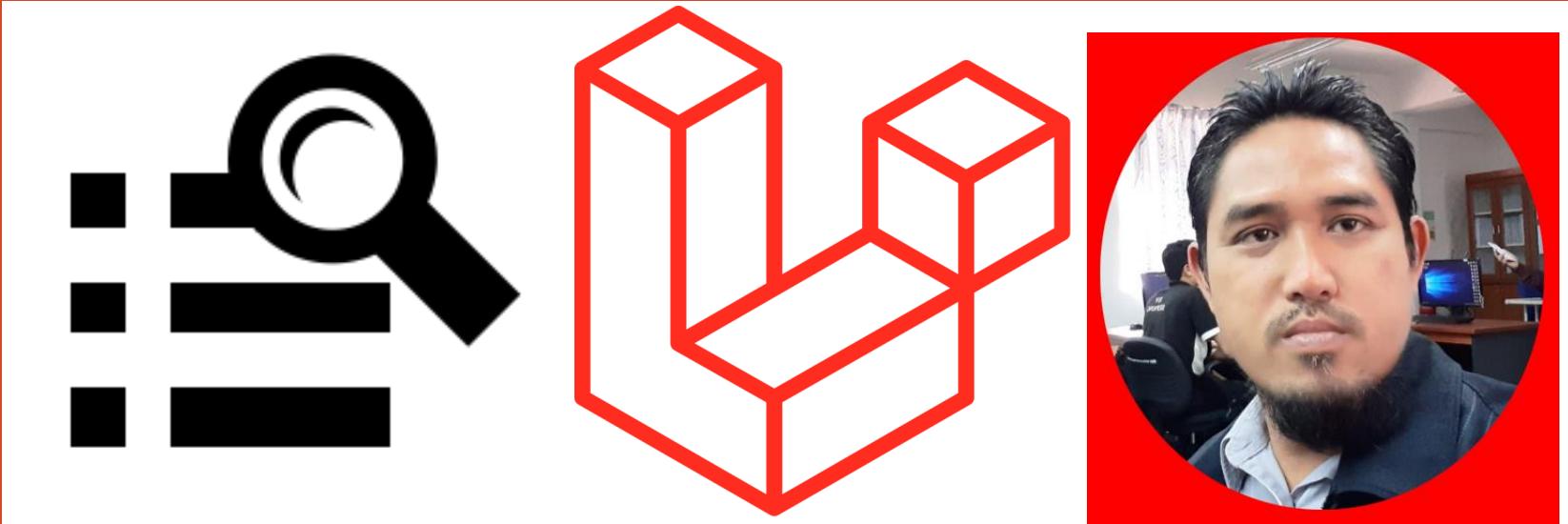
The line `action="{{ route('training.store') }}"` is highlighted with a light blue background.

# Controller - Store



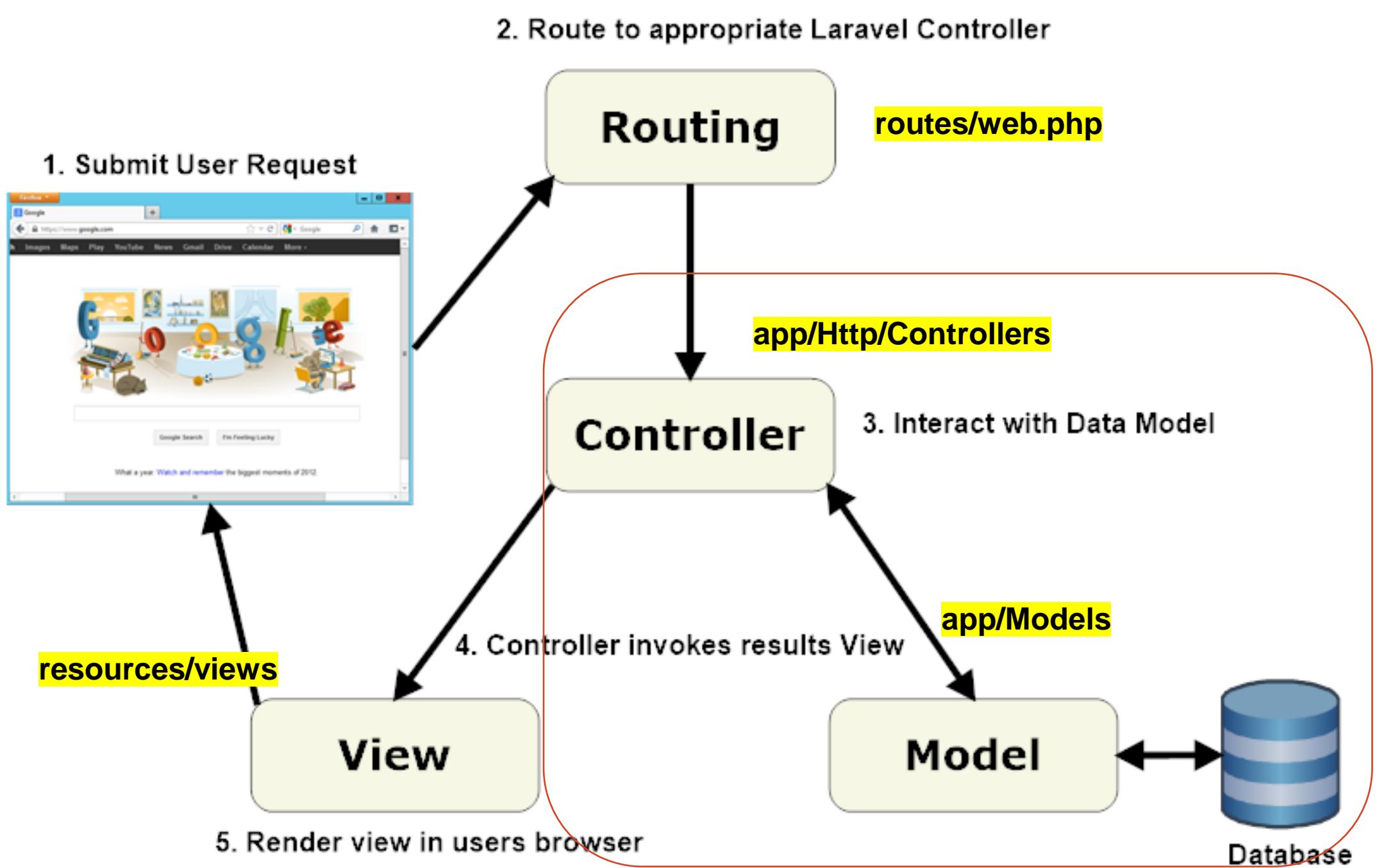
The image shows a screenshot of a code editor with a dark theme. The file being edited is `TrainingController.php`, located at `app > Http > Controllers > TrainingController.php`. The code is a Laravel controller for the `Training` model, specifically handling the `store` method. The code includes validation logic using the `validate` method and database saving logic using the `create` method.

```
36 * @param \Illuminate\Http\Request $request
37 * @return \Illuminate\Http\Response
38 */
39 public function store(Request $request)
40 {
41     //data validation/all field not nullable
42     $request->validate([
43         'title' => 'required',
44         'desc' => 'required',
45         'trainer' => 'required'
46     ]);
47     //execute saving record to database
48     Training::create($request->all());
49
50     return redirect()->route('training.index');
51     //return redirect('/training');
52 }
```



# Browse Records inside index

[github.com/khirulnizam](https://github.com/khirulnizam)



# Training index.blade.php

## Training Index

Training Index

Training Index

ID

Title

1

Laravel Beginner to Intermediate

2

PHP & Mysql Web System

Actions

Show

Edit

Delete

Show

Edit

Delete

Action

index

create

store

show

edit

update

destroy

# Controller - Index

Description	URL	Controller Function	View File
Default page for showing all the training.	GET example.com/trainings	index()	app/views/trainings/index.blade.php

```
use App\Training;

class TrainingController extends Controller
{
    //
    public function index()
    {
        $trainings = Training::all();
        return view('trainings.index')->with(compact('trainings'));
    }
}
```

# Views – Index View

Views are stored in the resources/views directory

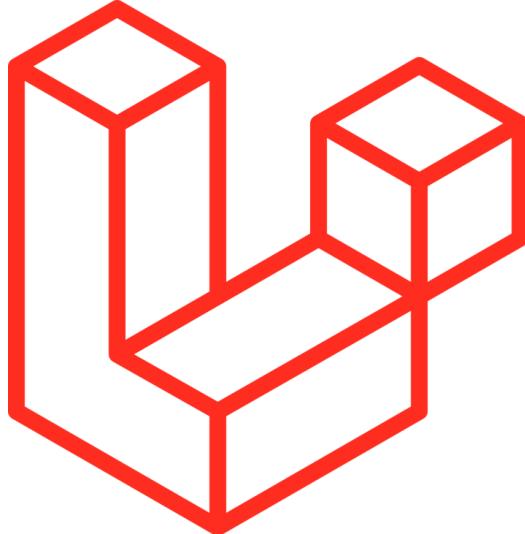
1. Go to resources/views/trainings/index.blade.php

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

@section('content')
<div class="container">
    <div class="row justify-content-center">
        <div class="col-md-8">
            <div class="card">
                <div class="card-header">Training Index</div>

                <div class="card-body">
                    <table class="table">
                        <thead>
                            <tr>
                                <th>ID</th>
                                <th>Title</th>
                                <th>Creator</th>
                                <th>Actions</th>
                            </tr>
                        </thead>
                        <tbody>
                            @foreach ($trainings as $training)
                                <tr>
                                    <td>{{ $training->id}}</td>
                                    <td>{{ $training->title}}</td>
                                    <td>{{ $training->user_id}}</td>
                                    <td>
                                        <a href="" class="btn btn-primary">Show</a>
                                        <a href="" class="btn btn-success">Edit</a>
                                        <a href="" class="btn btn-danger"
                                            onclick="return confirm('Are you sure?')">Delete</a>
                                    </td>
                                </tr>
                            @endforeach
                        </tbody>
                    </table>
                </div>
            </div>
        </div>
    </div>
</div>
@endsection
```





# Pagination & Search Records

[github.com/khirulnizam](https://github.com/khirulnizam)

# Search

1. Create form to input keyword
2. Query the keyword that match with title

Training Index			
ID	Title	Creator	Actions
13	Laravel Training 6 Days(Advanced)	Prof. Francesco Haag IV	<button>Show</button> <button>Edit</button> <button>Delete</button>
18	Laravel Medkad	Vincent Harding	<button>Show</button> <button>Edit</button> <button>Delete</button>

① [127.0.0.1:8000/trainings?keyword=laravel](http://127.0.0.1:8000/trainings?keyword=laravel)

# Search - Form

```
<div class="card-header">Training Index
<div class="float-right">
    <form
        class="d-none d-md-inline-block form-inline ml-auto mr-0 mr-md-3 my-2 my-md-0"
        method="GET"
        action="{{ route('training:index') }}"
    >
        <div class="input-group">
            <input
                class="form-control"
                type="text" placeholder="Search for..."
                aria-label="Search" aria-describedby="basic-addon2"
                name="keyword"
                value="{{ request()->get('keyword') }}"
            />
            <div class="input-group-append">
                <button class="btn btn-primary" type="submit"><i class="fas fa-search"></i></button>
            </div>
        </div>
    </form>
    <a href="{{ route('training:create') }}" class="btn btn-primary">+ Create New Training </a>
</div>
</div>
```

Using GET method to  
named route **training:index**

# Simple routing ( routes/web.php )

```
18 Auth::routes();  
19  
20 Route::get('/home', 'HomeController@index')->name('home');  
21  
22 //routes/web.php  
23 //route for training module  
24 Route::get('/trainings', 'TrainingController@index')->name('training');  
25 Route::get('/trainings/create', 'TrainingController@create')->name('training:create');  
26 Route::post('/trainings/create', 'TrainingController@store')->name('training:store');
```



The diagram illustrates the mapping between a route name and a controller method. A blue arrow points from the word "index" in the route definition "Route::get('/trainings', 'TrainingController@index')->name('training');" to the word "index" in the controller method signature "TrainingController@index". This visual cue helps in understanding how the route name is used to identify the specific controller action to be executed.

# Simple Routing

```
php artisan route:list
```

GET HEAD   trainings	training:index	App\Http\Controllers\TrainingController@index	web,auth
GET HEAD   trainings/create	training:create	App\Http\Controllers\TrainingController@create	web,auth
POST   trainings/create	training@store	App\Http\Controllers\TrainingController@store	web,auth

# Search – Index Method

Check if there is keyword, if there is keyword on query string, run query to search column where like title.

If no keyword, the method will return collection rows from trainings table.

```
public function index(Request $request)
{
    if($request->get('keyword') != null)
    {
        $keyword = $request->get('keyword');
        $trainings = Training::where('title', 'LIKE', '%' . $keyword . '%')->get();
    }
    else
    {
        $trainings = Training::all();
    }
    return view('trainings.index')->with(compact('trainings'));
}
```



# Laravel t08 Pagination

Tarmizi Sanusi  
Khirulnizam Abd Rahman  
[fstm.kuis.edu.my/blog/laravel](http://fstm.kuis.edu.my/blog/laravel)

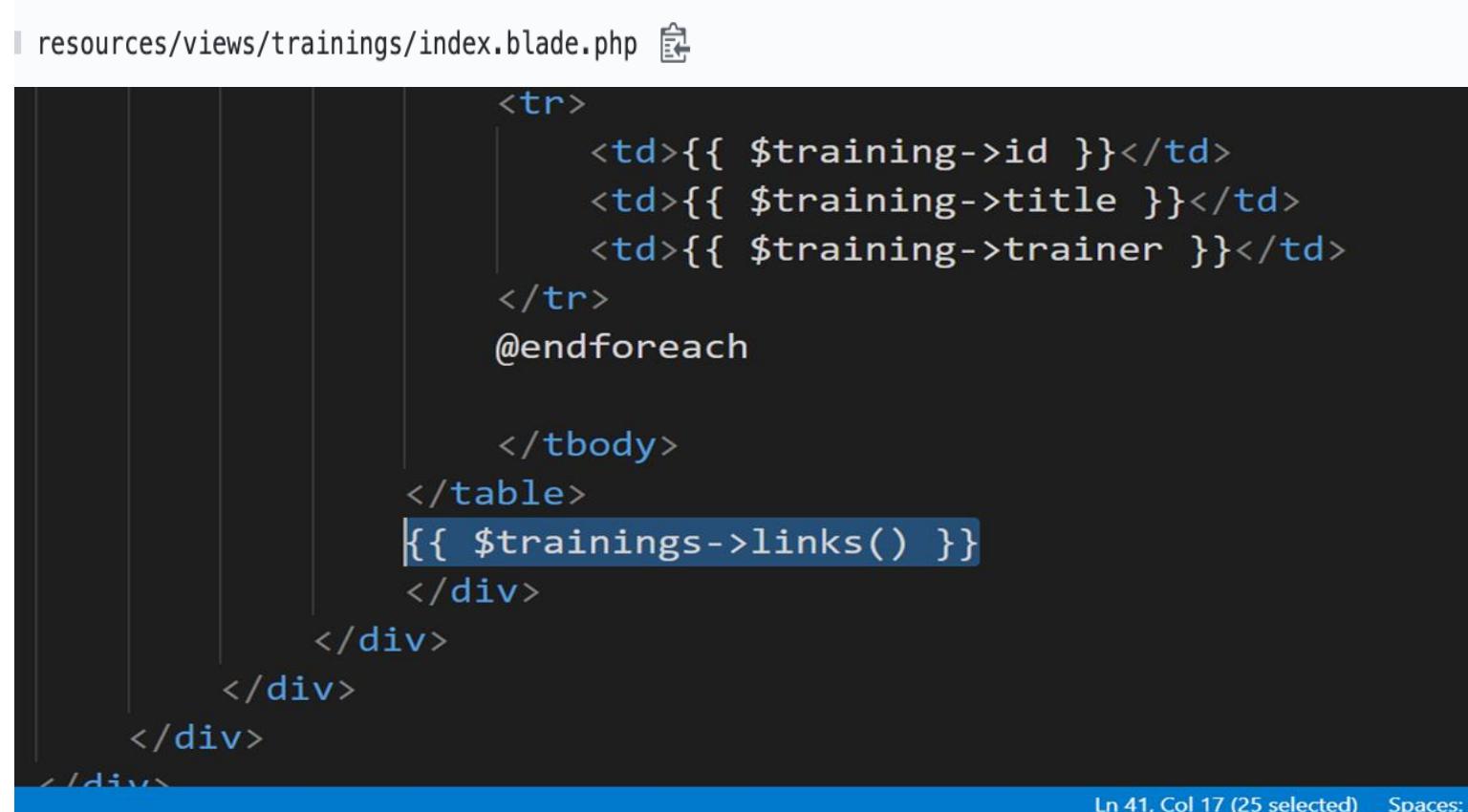


Training Index		
<input type="text"/>		<button>Search</button>
<b>id</b>	<b>title</b>	<b>trainer</b>
1	Laravel Basics	Khirulnizam Abd Rahman
2	Laravel Intermediate	Tarmizi Sanusi
3	Laravel Advanced with Unit Test	Nasrul Hazim
4	Ipsa molestiae enim atque voluptatem.	Dr. Imani Feil
5	Tempore dicta eos ut sint.	Frida White

[«](#) [1](#) [2](#) [3](#) [4](#) [5](#) [6](#) [»](#)

# Search - Pagination

The paginate method automatically takes care of setting the proper limit and offset based on the current page being viewed by the user and append with keywords.



A screenshot of a code editor displaying a Blade template file named `index.blade.php`. The code is part of a table structure, likely for displaying a list of trainings. It includes a loop (`@foreach`) to generate table rows (`<tr>`), individual table cells (`<td>{{ $training->id }}</td>`, `<td>{{ $training->title }}</td>`, `<td>{{ $training->trainer }}</td>`), and the closing loop tag (`</tr>`). The loop is preceded by the opening tag (`<tbody>`) and followed by the closing tag (`</tbody>`). Below the loop, there is a call to a method (`links()`) on the `$trainings` variable, which is enclosed in a `<div>` tag. The entire table structure is enclosed in another `<div>` tag. The code editor shows line 41, column 17, with 25 selected characters and 4 spaces used.

```
resources/views/trainings/index.blade.php   
| {{ $training->id }} | {{ $training->title }} | {{ $training->trainer }} |
  
 @endforeach  
  
 </tbody>  
 </table>  
 {{ $trainings->links() }}  
 </div>  
 </div>  
 </div>
```

# TrainingController@index

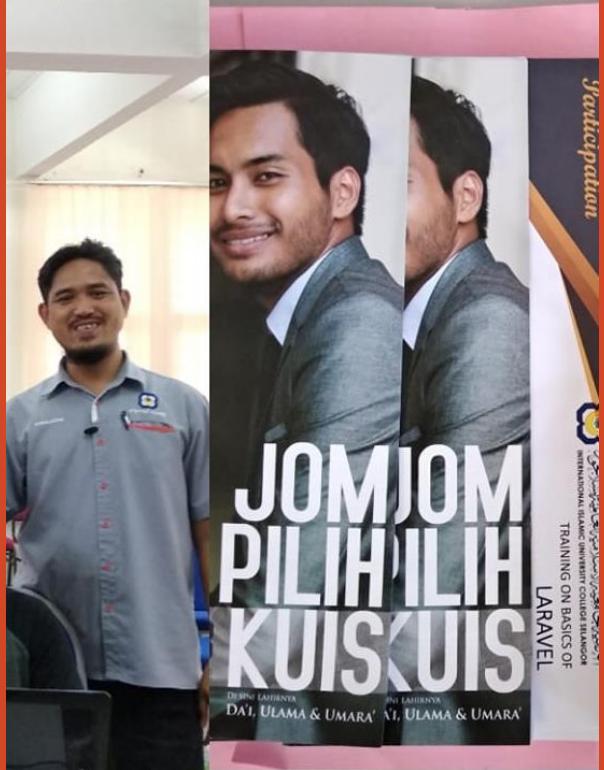
```
if($request->get('keyword') != null)
{
    $keyword = $request->get('keyword');
    $trainings = Training::where('title','LIKE','%'.$keyword.'%')->get();
    $trainings = Training::where('title','LIKE','%'.$keyword.'%')->paginate(5);
}
else
{
    $trainings = Training::all();
    $trainings = Training::paginate(5);
}
```

# Hacks

Creating Index - <https://github.com/samtarmizi/laravel-spr/commit/1c925853f61da081f0e3ce298e58a4a423b66040>

Creating Create - <https://github.com/samtarmizi/laravel-spr/commit/7a5b5f3b643bcb8a265af3bb9f4e94e3aef95aa1>

Creating Store - <https://github.com/samtarmizi/laravel-spr/commit/2513b55c65bc4d27afa5004144ee2c7d41b3cbd7>



# Laravel t10 npm not found

[fstm.kuis.edu.my/blog/laravel](http://fstm.kuis.edu.my/blog/laravel)



# npm not found



```
Command Prompt

D:\>npm -v
'npm' is not recognized as an internal or external command,
operable program or batch file.

D:\>npm install
'npm' is not recognized as an internal or external command,
operable program or batch file.

D:\>
```

# Authentication Module

Laravel makes implementing authentication very simple. In fact, almost everything is configured for you out of the box.

```
composer require laravel/ui --dev
```

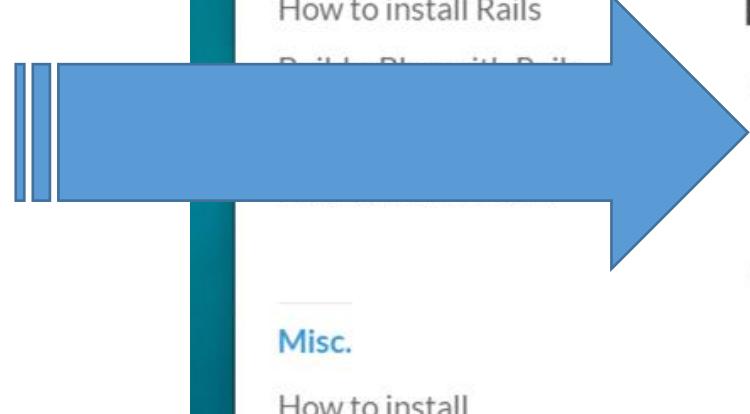
```
php artisan ui vue --auth
```

```
npm install
```

```
npm run dev
```

# Laragon

Includes NODEJS



Download | Laragon - portable, is + [laragon.org/download/index.html](http://laragon.org/download/index.html)

# Download

Laragon is a universal development environment. It has many features to make you more productive:

**Benefits of Laragon**

After downloading, You can add **git, phpmyadmin, Node.js/MongoDB, Python/Django/Flask/Postgres, Ruby, Java, Go** using “Tools > Quick add”

Note: [You can also download from GitHub](#)

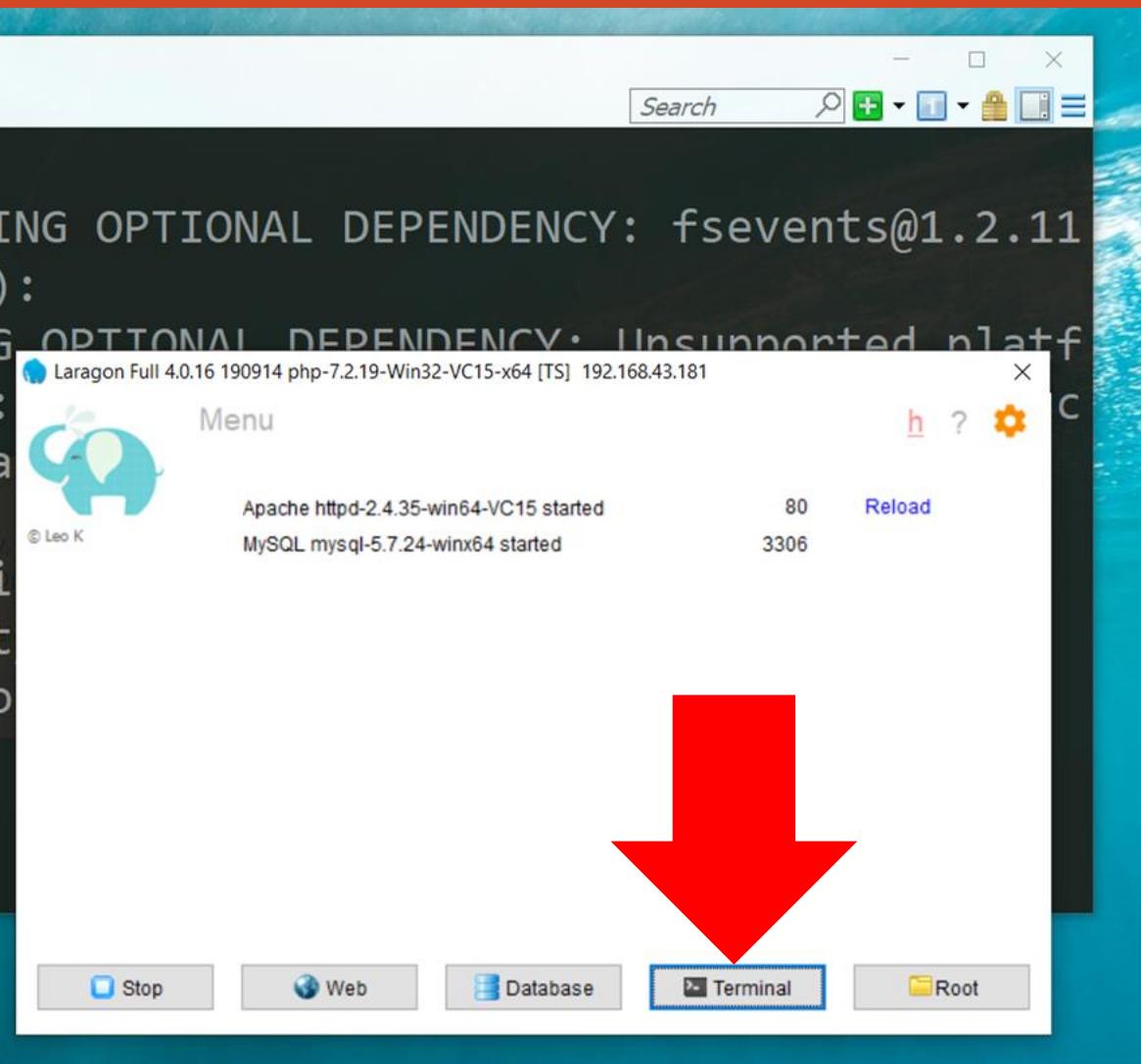
## Edition

- Laragon Full:** Apache 2.4, Nginx, MySQL 5.7, PHP 7.2, Redis, Memcached, Node.js 11, npm, yarn, git, ...  
[Download Laragon - Full \(130 MB\)](#)
- Laragon Lite:** Don't include Node.js 11, npm, yarn, git but you can add them easily using “Tools > Quick add”  
[Download Laragon - Lite \(85 MB\)](#)

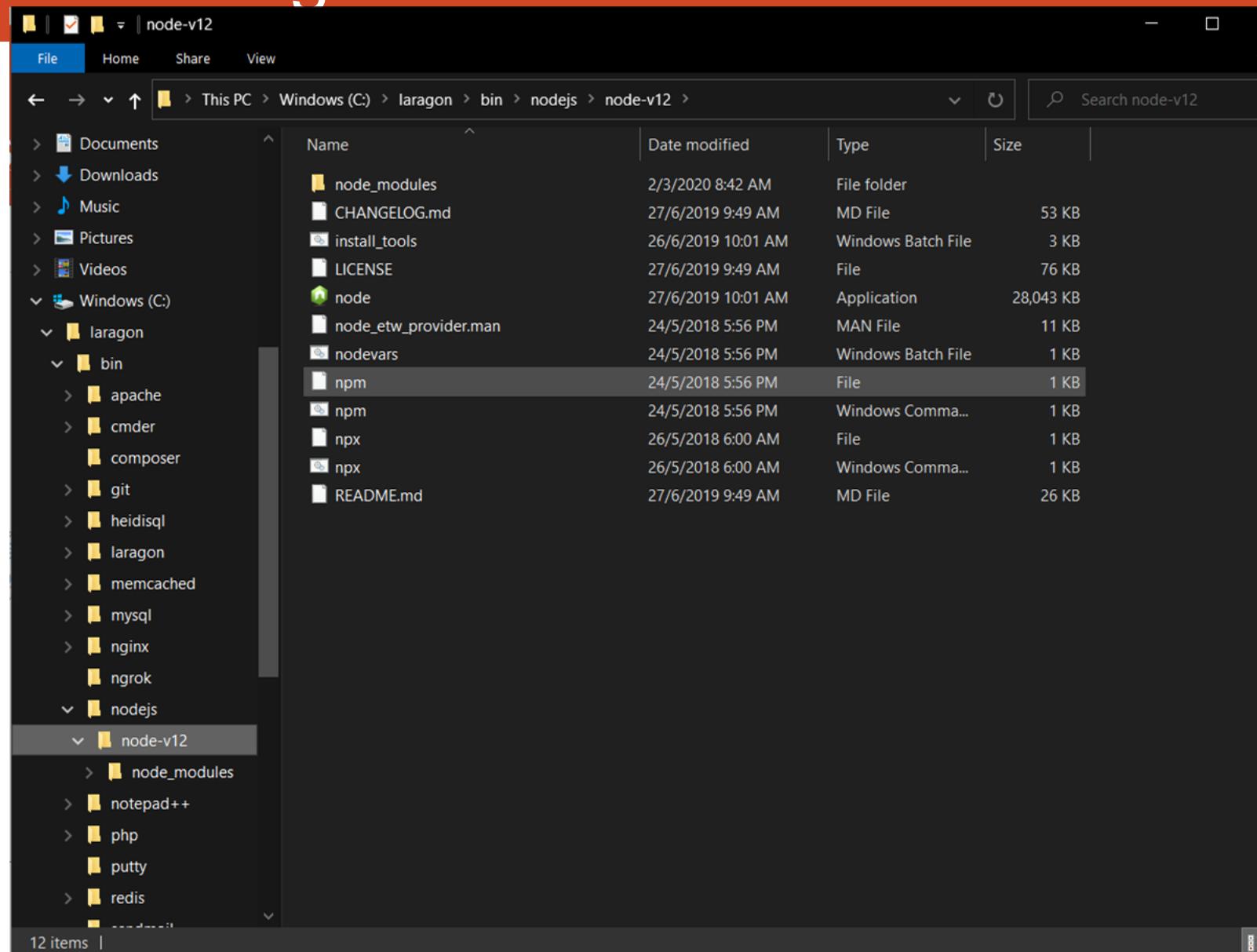
# Terminal in Laragon

Laravel makes lf.

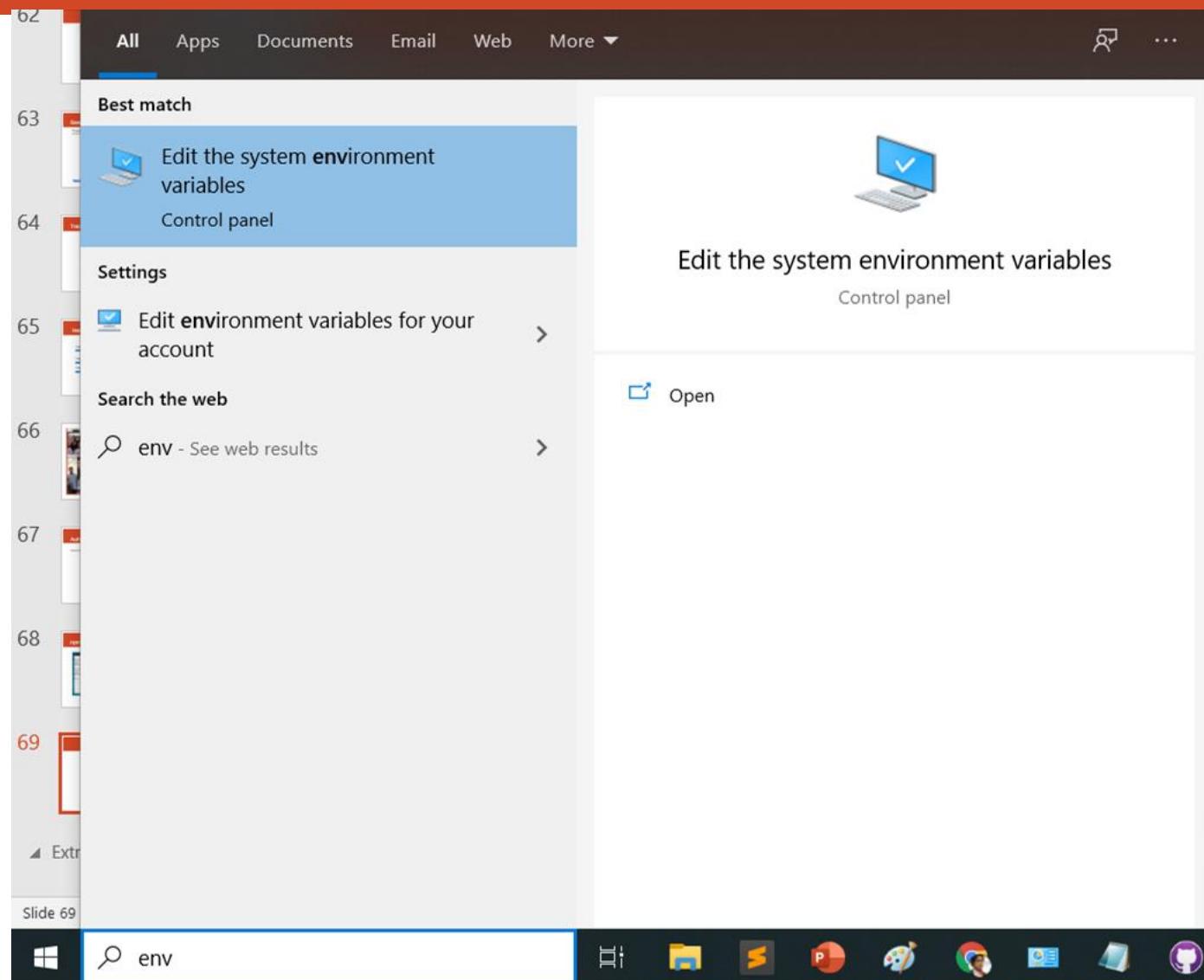
```
npm WARN optional SKIPPING OPTIONAL DEPENDENCY: fsevents@1.2.11  
  (node_modules\fsevents):  
npm WARN notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform  
  for fsevents@1.2.11:  
  current: {"os": "win32", "arch": "x64"}  
  
Sorry, something went wrong.  
audited 17591 packages in 1.03s  
found 1 moderate severity  
  run `npm audit fix` to  
  
C:\laragon\www\lara7  
λ npm install
```



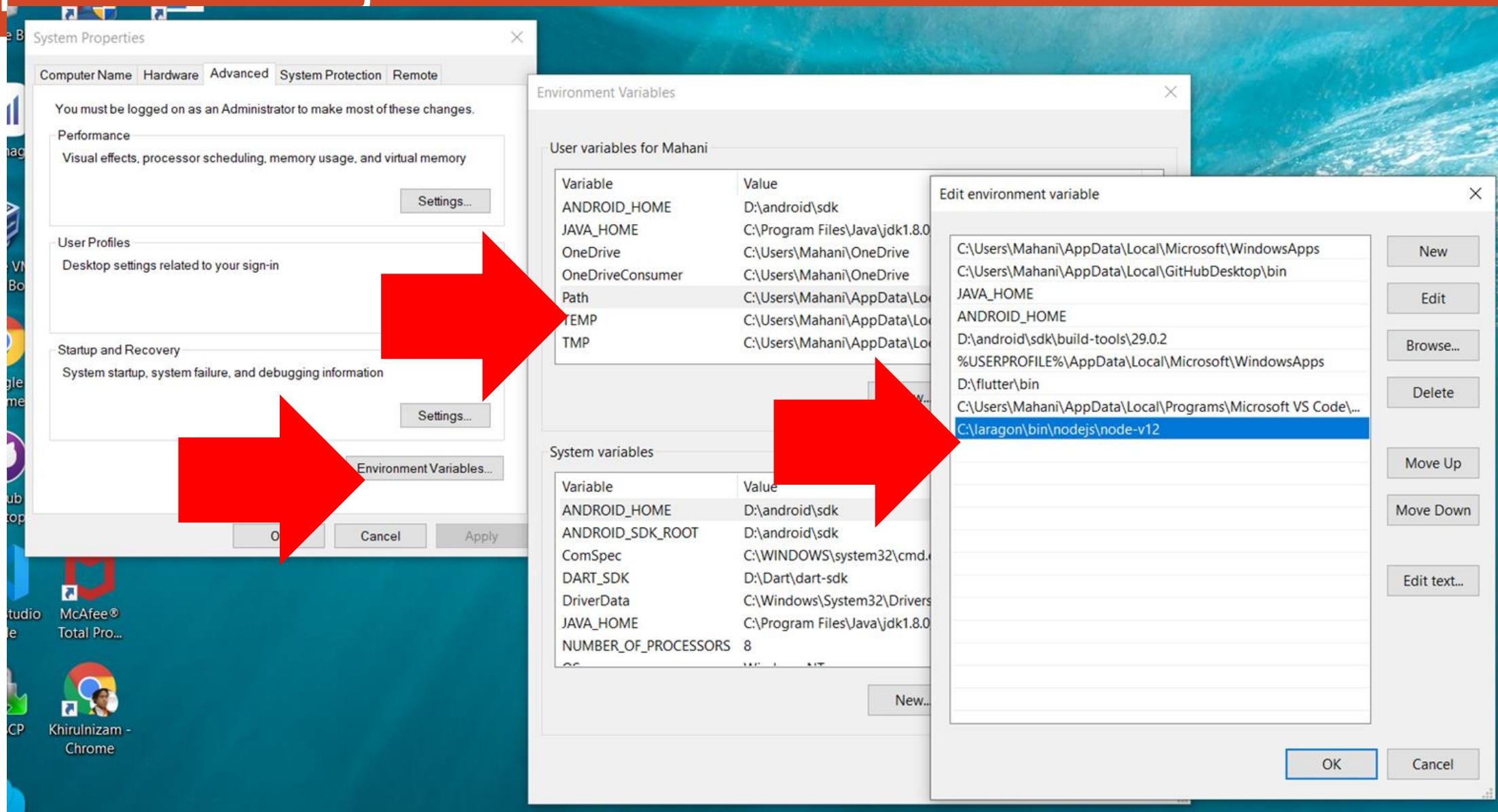
# Locate nodejs in Laragon



# Environment variables



# Set path to nodejs



# NODEJS.ORG

The screenshot shows a web browser window displaying the Node.js homepage at [nodejs.org/en/](https://nodejs.org/en/). The page features a dark header with the Node.js logo and navigation links for HOME, ABOUT, DOWNLOADS, DOCS, GET INVOLVED, SECURITY, and NEWS. A large green button labeled "Download for Windows (x64)" is highlighted with a red box. This button contains two options: "12.16.1 LTS" (labeled "Recommended For Most Users") and "13.11.0 Current" (labeled "Latest Features"). Below these buttons, links for "Other Downloads", "Changelog", and "API Docs" are provided for each version. A note at the bottom encourages users to look at the "Long Term Support (LTS) schedule".

Node.js® is a JavaScript runtime built on Chrome's V8 JavaScript engine.

Download for Windows (x64)

**12.16.1 LTS**  
Recommended For Most Users

**13.11.0 Current**  
Latest Features

Other Downloads | Changelog | API Docs    Other Downloads | Changelog | API Docs

Or have a look at the [Long Term Support \(LTS\) schedule](#).

Sign up for [Node.js Everywhere](#), the official Node.js Monthly Newsletter.



LaravelFSTM  
@KVBalikPulau

# Laravel t11

## Middleware

## enforced Auth

Khirulnizam.com



# Middleware

Middleware provide a convenient mechanism for filtering HTTP requests entering your application.

For example, Laravel includes a middleware that verifies the user of your application is authenticated. If the user is not authenticated, the middleware will redirect the user to the login screen.

```
Route::get('/home', 'HomeController@index')->name('home')->middleware('auth');
```

```
public function __construct()  
{  
    $this->middleware('auth');  
}
```

# Add to dashboard menu

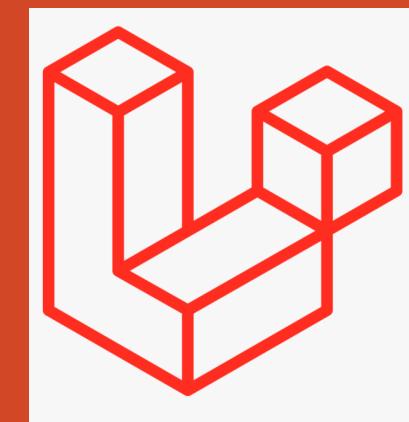
The screenshot shows a Sublime Text editor window with several tabs open. The active tab is 'app.blade.php' which contains the following blade template code:

```
22 <body>
23   <div id="app">
24     <nav class="navbar navbar-expand-md navbar-light
25       bg-white shadow-sm">
26       <div class="container">
27         <a class="navbar-brand" href="{{ url('/') }}">
28           {{ config('app.name', 'FSTM Laravel') }}
29         </a>
30         <!-- menu -->
31         <a href="{{ url('trainings/create') }}">Insert
32         </a>
33         ||
34         <a href="{{ url('trainings') }}">Listing/Search
35         </a>
36       <!-- end menu -->
```

The code is part of a navigation bar, specifically the 'menu' section. It includes links for 'Insert' and 'Listing/Search'. The file path in the title bar is 'C:\laragon\www\lara7\resources\views\layouts\app.blade.php'.

Laravel t12

# Test Laravel project from Github repo



# NODEJS.ORG

The screenshot shows a web browser window displaying the Node.js homepage at [nodejs.org/en/](http://nodejs.org/en/). The page features a dark header with the Node.js logo and navigation links for HOME, ABOUT, DOWNLOADS, DOCS, GET INVOLVED, SECURITY, and NEWS. Below the header, a large green button labeled "Download for Windows (x64)" is highlighted with a red rounded rectangle. This button contains two options: "12.16.1 LTS" (labeled "Recommended For Most Users") and "13.11.0 Current" (labeled "Latest Features"). At the bottom of the page, there is a link to the "Long Term Support (LTS) schedule".

Node.js® is a JavaScript runtime built on Chrome's V8 JavaScript engine.

Download for Windows (x64)

**12.16.1 LTS**  
Recommended For Most Users

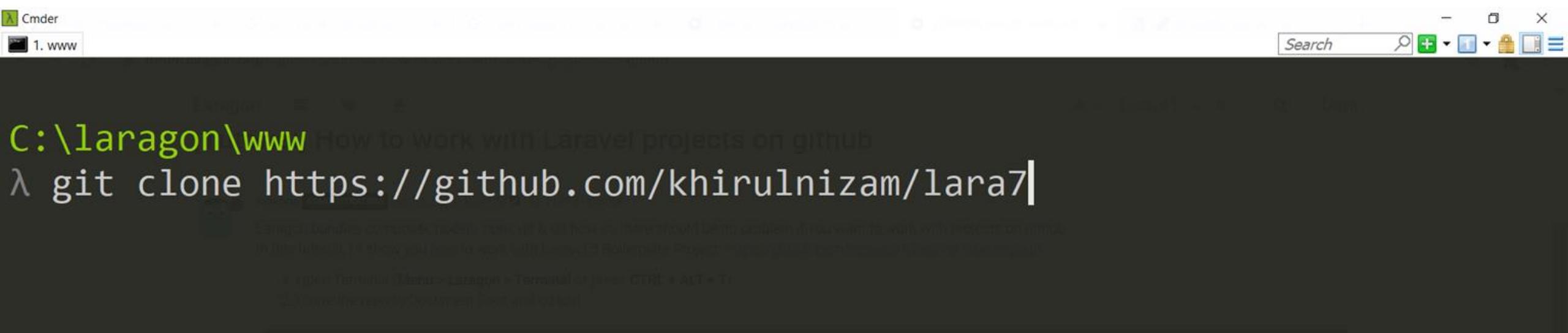
**13.11.0 Current**  
Latest Features

Other Downloads | Changelog | API Docs   Other Downloads | Changelog | API Docs

Or have a look at the [Long Term Support \(LTS\) schedule](#).

Sign up for [Node.js Everywhere](#), the official Node.js Monthly Newsletter.

# Git clone



A screenshot of a Cmder terminal window. The title bar says "Cmder". The tab bar shows "1. www". The search bar has "Search" and other icons. The main area shows a command line with the path "C:\laragon\www" and the command "λ git clone https://github.com/khirulnizam/lara7|". Below the command, there is some small, semi-transparent text about Laragon and GitHub.

```
C:\laragon\www How to work with Laravel projects on github
λ git clone https://github.com/khirulnizam/lara7|
```

# composer & npm install

The screenshot shows a terminal window titled '1. larat' in a Cmder interface. The window displays the following command sequence:

```
C:\laragon\www  
λ cd larat  
  
C:\laragon\www\larat  
λ composer install  
  
C:\laragon\www\larat  
λ npm install
```

The background of the terminal window features a dark theme with a grid pattern and various icons for file operations like Cut, Copy, Paste, Select, Rotate, Crop, Resize, Brushes, Shapes, and Colors.

## .env file

```
6  
7 LOG_CHANNEL=stack  
8  
9 DB_CONNECTION=mysql  
10 DB_HOST=127.0.0.1  
11 DB_PORT=3306  
12 DB_DATABASE=lara7  
13 DB_USERNAME=root  
14 DB_PASSWORD=abcd1234  
15
```

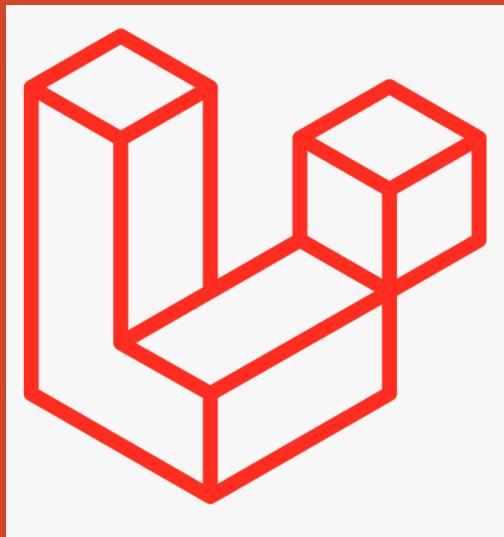
# php artisan migrate & seed

Inside Laravel project folder

php artisan key:generate

php artisan migrate

php artisan db:seed



# Laravel t13

## Show records

### (Paparan semua data dalam rekod)

# Table with lots of column

The screenshot shows a database interface in HeidiSQL Portable version 10.2.0.5599. The left sidebar displays a tree view of the database structure under 'Laragon'. The 'mycompanyhr' database is selected, and its 'employee' table is currently viewed. The table has 29 rows and 14 columns. The columns are: EMPNO, FIRSTNAME, MIDINT, LASTNAME, WORKDEPT, PHONENO, HIREDATE, JOB, EDLEV..., GENDER, BIRTHDATE, SALARY, BONUS, and COMM. The first row, with EMPNO 000341 and FIRSTNAME KHIRULNIZAM, is highlighted. The interface includes standard database navigation tools like 'Host: localhost', 'Database: mycompanyhr', 'Table: employee', 'Data', 'Query\*', and 'mycompanyhr.sql'. A 'Columns (14/14)' button is visible at the top right of the table area.

EMPNO	FIRSTNAME	MIDINT	LASTNAME	WORKDEPT	PHONENO	HIREDATE	JOB	EDLEV...	GENDER	BIRTHDATE	SALARY	BONUS	COMM
000341	KHIRULNIZAM	B	ABD RAHMAN	C01	1111	2000-01-29	PROGRAMM	17	M	1976-08-18	36,000.00	6,000.00	2,000.00
000020	MICHAEL	L	THOMPSONS	C01	2222	1973-10-10	MANAGER	18	M	1948-02-02	41,250.00	800.00	3,300.00
000050	JOHN	B	GEYER	E01	6789	1949-08-17	MANAGER	16	M	1925-09-15	40,175.00	800.00	3,214.00
000070	EVA	D	PULASKI	D21	7831	1980-09-30	MANAGER	16	F	1953-05-26	36,170.00	700.00	2,893.00
000090	EILEEN	W	HENDERSON	E11	5498	1970-08-15	MANAGER	16	F	1941-05-15	29,750.00	600.00	2,380.00
000100	THEODORE	Q	SPENSER	E21	0972	1980-06-19	MANAGER	14	M	1956-12-18	26,150.00	500.00	2,092.00
000110	VINCENZO	G	LUCCHESI	A00	3490	1958-05-16	SALESREP	19	M	1929-11-05	46,500.00	900.00	3,720.00
000120	SEAN		O'CONNELL	A00	2167	1963-12-05	CLERK	14	M	1942-10-18	29,250.00	600.00	2,340.00
000130	DOLORES	M	QUINTANA	C01	4578	1971-07-28	ANALYST	16	F	1925-09-15	23,800.00	500.00	1,904.00
000140	HEATHER	A	NICHOLLS	C01	1793	1976-12-15	ANALYST	18	F	1946-01-19	28,420.00	600.00	2,274.00
000150	BRUCE		ADAMSON	D11	4510	1972-02-12	DESIGNER	16	M	1947-05-17	25,280.00	500.00	2,022.00
000160	ELIZABETH	R	PIANKA	D11	3782	1977-10-11	DESIGNER	17	F	1955-04-12	22,250.00	400.00	1,780.00
000180	MARILYN	S	SCOUTTEN	D11	1682	1973-07-07	DESIGNER	17	F	1949-02-21	21,340.00	500.00	1,707.00
000190	JAMES	H	WALKER	D11	2986	1974-07-26	DESIGNER	16	M	1952-06-25	20,450.00	400.00	1,636.00
000200	DAVID		BROWN	D11	4501	1966-03-03	DESIGNER	16	M	1941-05-29	27,740.00	600.00	2,217.00
000210	WILLIAM	T	JONES	D11	0942	1979-04-11	DESIGNER	17	M	1953-02-23	18,270.00	400.00	1,462.00
000220	JENNIFER	K	LUTZ	D11	0672	1968-08-29	DESIGNER	18	F	1948-03-19	29,840.00	600.00	2,387.00
000230	JAMES	J	JEFFERSON	D21	2094	1966-11-21	CLERK	14	M	1935-05-30	22,180.00	400.00	1,774.00
000240	SALVATORE	M	MARINO	D21	3780	1979-12-05	CLERK	17	M	1954-03-31	28,760.00	600.00	2,301.00
000250	DANIEL	S	SMITH	D21	0961	1969-10-30	CLERK	15	M	1939-11-12	19,180.00	400.00	1,534.00
000260	SYBIL	P	JOHNSON	D21	8953	1975-09-11	CLERK	16	F	1936-10-05	17,250.00	300.00	1,380.00
000270	MARIA	L	PEREZ	D21	9001	1980-09-30	CLERK	15	F	1953-05-26	27,380.00	500.00	2,190.00
000280	ETHEL	R	SCHNEIDER	E11	8997	1967-03-24	OPERATOR	17	F	1936-03-28	26,250.00	500.00	2,100.00
000290	JOHN	R	PARKER	E11	4502	1980-05-30	OPERATOR	12	M	1946-07-09	15,340.00	300.00	1,227.00
000300	PHILIP	X	SMITH	E11	2095	1972-06-19	OPERATOR	14	M	1936-10-27	17,750.00	400.00	1,420.00
000310	MAUDE	F	SETRIGHT	E11	3332	1964-09-12	OPERATOR	12	F	1931-04-21	15,900.00	300.00	1,272.00
000320	RAMLAL	V	MEHTA	E21	9990	1965-07-07	FIELDREP	16	M	1932-08-11	19,950.00	400.00	1,596.00

# DISCLAIMER

Codes are for demo purposes. Examples showed here are only to validate the proof of concepts. Security aspects the demonstrated examples are beyond the coverage of these video tutorials. You are advised to explore / search after advanced/security topics of PHP web programming as these tutorials are only for beginner level.

Suitable to view in DESKTOP

# Views

Views are stored in the resources/views directory

1. Go to resources/views
2. Go to trainings folder, create:
  1. show.blade.php
  2. edit.blade.php

```
resources
  > js
  > lang
  > sass
  < views
    > admin
    > auth
    > layouts
    < trainings
      ≡ create.blade.php
      ≡ edit.blade.php
      ≡ index.blade.php
      ≡ show.blade.php
```

# Controller

7 method in controllers

## Make Controller

```
php artisan make:controller TrainingController
```

### Action

---

-index

---

-create

---

-store

---

show

---

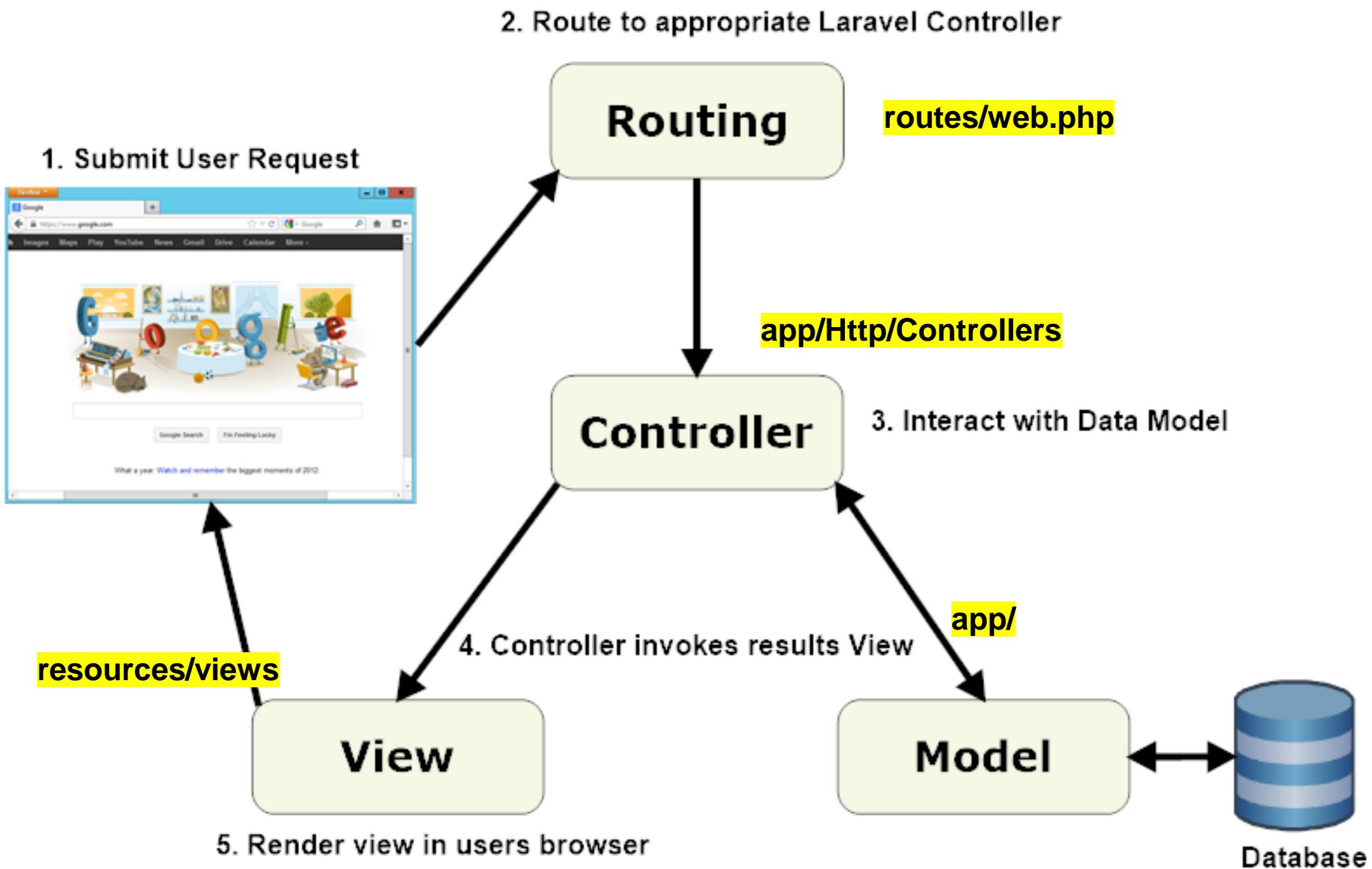
edit

---

update

---

destroy



# Controller - Show

Description	URL	Controller	View File
		Function	
Show one of the trainings.	GET example.com/trainings/{id}	show()	app/views/trainings/show.blade.php

```
public function show($id)
{
    $training=Training::find($id);
    return view('trainings.show')->with(compact('training'));
}
```

# Views – Show View

Views are stored in the resources/views directory

1. Go to resources/views/trainings/show.blade.php

# Github.com/khirulnizam/lara7

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

@section('content')


Show Training



Title



Description
{{ $training->title }}

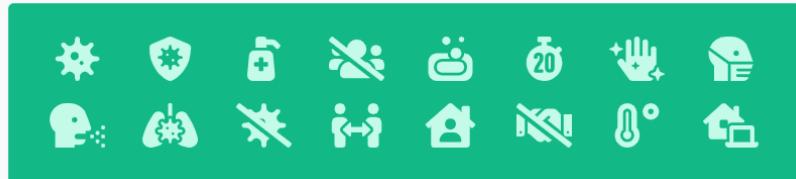


Trainer



Back to Training Index


```



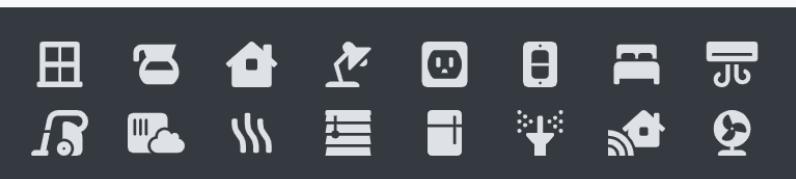
COVID-19: Help spread awareness and not the virus with these free icons!



Our new Science Fiction icons are ready to beam up.



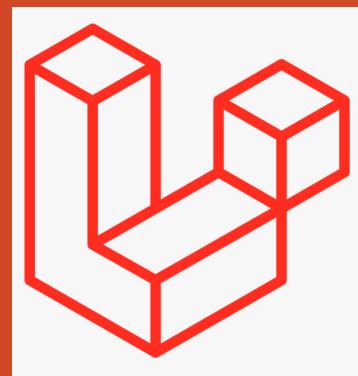
Ready for a close encounter? Check out our new space icons.



Hey Siri, turn up the heat on our new household icons.

# Laravel t14

## Font-awesome thru CDN (Pasang icon font-awesome guna CDN)



# CDN – CONTENT DELIVERY NETWORK

- CDN – load library/ framework from public host server
- to provide high availability and performance
- **<https://www.bootstrapcdn.com/fontawesome/>**

## Font Awesome

**Font Awesome CSS**

```
https://stackpath.bootstrapcdn.com/fontawesome/4.7.0/css/font-awesome.min.css
```

Click to copy ▾

**HTML**

```
<link href="https://stackpath.bootstrapcdn.com/fontawesome/4.7.0/css/font-awesome.min.css" rel="stylesheet"/>
```

Copied text to clipboard

**Pug**

```
link(href="https://stackpath.bootstrapcdn.com/fontawesome/4.7.0/css/font-awesome.min.css", rel="stylesheet")
```

Click to copy

**Haml**

```
%link{href: "https://stackpath.bootstrapcdn.com/fontawesome/4.7.0/css/font-awesome.min.css", rel: "stylesheet"}
```

Click to copy

# FONTAWESOME IN SIMPLE HTML

## Icons Chart

[Icons Currency](#)

[Icons Directional](#)

[Icons File Type](#)

[Icons Form](#)

[Icons Gender](#)

[Icons Hand](#)

[Icons Medical](#)

[Icons Payment](#)

[Icons Spinner](#)

[Icons Text](#)

[Icons Transportation](#)

[Icons Video](#)

[Icons Web Application](#)

[Bootstrap](#)

[Icons BS Glyphicons](#)

[Google](#)

[Google Icons Intro](#)

[Icons Action](#)

[Icons Alert](#)

## Text Icons

The table below shows all Font Awesome Text icons:

Icon	Description	Example
	fa fa-align-center	<a href="#">Try it</a>
	fa fa-align-justify	<a href="#">Try it</a>
	fa fa-align-left	<a href="#">Try it</a>
	fa fa-align-right	<a href="#">Try it</a>
	fa fa-bold	<a href="#">Try it</a>
	fa fa-chain	<a href="#">Try it</a>
	fa fa-chain-broken	<a href="#">Try it</a>
	fa fa-clipboard	<a href="#">Try it</a>
	fa fa-columns	<a href="#">Try it</a>
	fa fa-copy	<a href="#">Try it</a>

# Import font-awesome from CDN

```
<html>
<head>
<title>Font Awesome Icons</title>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/4.7.0/css/font-awesome.min.css">
</head>
<body>

<h1>fa fa-file-text</h1>

<i class="fa fa-file-text"></i>
<i class="fa fa-file-text" style="font-size:24px"></i>
<i class="fa fa-file-text" style="font-size:36px;"></i>
<i class="fa fa-file-text" style="font-size:48px;color:red"></i>
<br>

<p>Used on a button:</p>
<button style="font-size:24px">Show <i class="fa fa-file-text"></i>
</button>

</body>
</html>
```

## Tag: <i></i>

```
<i class="fa fa-file-text"></i>
```

<p>Used on a button:</p>

```
<button>
```

Show <**i class="fa fa-file-text"**></i>

```
</button>
```

**fa fa-file-text**



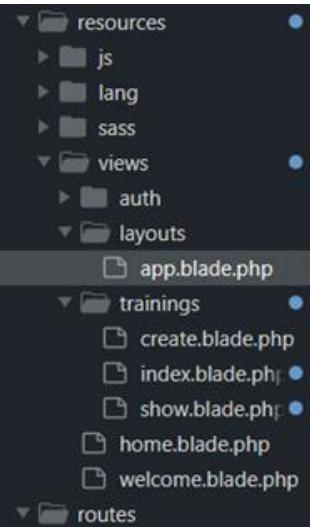
Used on a button:



Unicode:



# Import font-awesome CDN in Laravel layout



18

19

20

- <https://www.bootstrapcdn.com/fontawesome/>

## Font Awesome

### Font Awesome CSS

```
https://stackpath.bootstrapcdn.com/fontawesome/4.7.0/css/font-awesome.min.css
```

Click to copy

### HTML

```
<link href="https://stackpath.bootstrapcdn.com/fontawesome/4.7.0/css/font-awesome.min.css" rel="styl
```

Copied text to clipboard

### Pug

```
link(href="https://stackpath.bootstrapcdn.com/fontawesome/4.7.0/css/font-awesome.min.css", rel="style
```

Click to copy

### Haml

```
%link{href: "https://stackpath.bootstrapcdn.com/fontawesome/4.7.0/css/font-awesome.min.css", rel: "styl
```

Click to copy

# Implement in buttons

Training Index

🔍

<b>id</b>	<b>Title</b>	<b>Task</b>
1	Laravel Basics	
2	Laravel Intermediate	
3	Laravel Advanced with Unit Test	

<https://github.com/khirulnizam/10btm/>

The screenshot shows a GitHub repository page for '10btm'. The repository is owned by 'khirulnizam' and is public. It has 1 branch and 0 tags. The most recent commit was made yesterday by 'khirulnizam' and is titled 'create-xlayout'. The commit message includes the URL <http://test.kuis.eau.my/blog/laravel>. The repository contains five main directories: app, bootstrap, config, database, and public, each with an initial commit.

Sign in | khirulnizam/10btm: Laravel 10 B | WhatsApp | Managing visibility of your project | +

https://github.com/khirulnizam/10btm/

khirulnizam / 10btm

Type ⌘ to search

Code Issues Pull requests Actions Projects Security Insights Settings

10btm Public Pin Unwatch 1

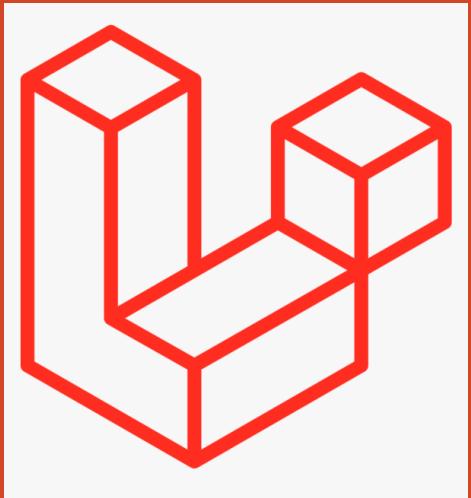
main ▾ 1 branch 0 tags Go to file Add file ▾ Code ▾

khirulnizam create-xlayout 7c08961 yesterday 2 commits

File	Commit Message	Date	Commits
app	create-xlayout	yesterday	2
bootstrap	Initial commit	yesterday	1
config	Initial commit	yesterday	1
database	Initial commit	yesterday	1
public	Initial commit	yesterday	1

http://test.kuis.eau.my/blog/laravel

109

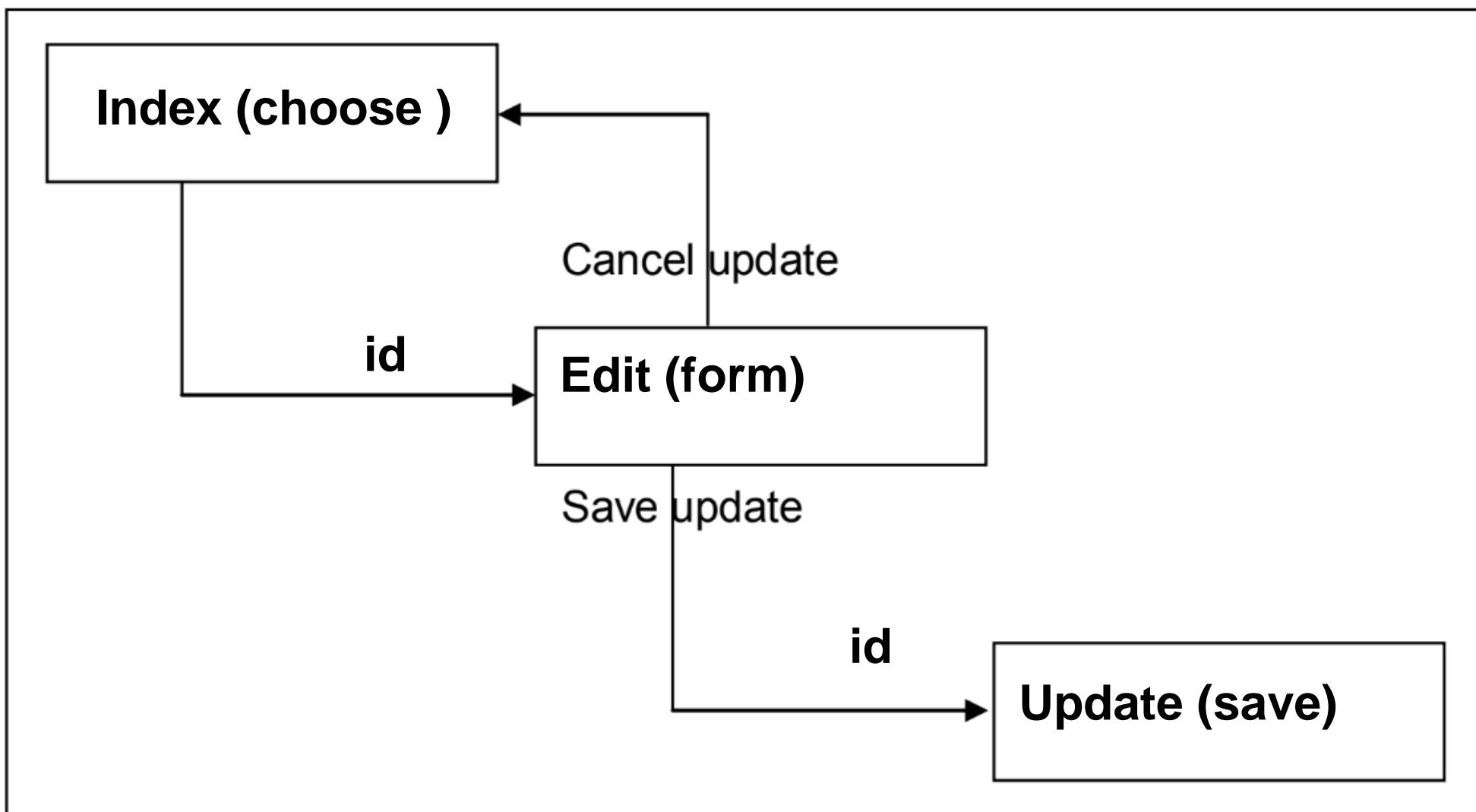


# Laravel t15

# Edit/Update

(Kemaskini data  
dalam rekod)

# UPDATE RECORD PROCESS



# DISCLAIMER

Codes are for demo purposes. Examples showed here are only to validate the proof of concepts. Security aspects the demonstrated examples are beyond the coverage of these video tutorials. You are advised to explore / search after advanced/security topics of PHP web programming as these tutorials are only for beginner level.

Suitable to view in DESKTOP

# <https://github.com/khirulnizam/10tbtm/>

## Slides & codes

The screenshot shows a GitHub repository page for 'khirulnizam / 10tbtm'. The repository is public. The main branch is 'main'. There is 1 branch and 0 tags. The commit history for 'main' shows two commits from 'khirulnizam' yesterday. The commits are:

Commit	Message	Date	Commits
7c08961	create-xlayout	yesterday	2
app	create-xlayout	yesterday	
bootstrap	Initial commit	yesterday	
config	Initial commit	yesterday	
database	Initial commit	yesterday	
public	Initial commit	yesterday	

# Controller - Edit

Description	URL	Controller	View File
			Function
Pull a training from the database and allow editing.	GET example.com/trainings/{id}/edit	edit()	app/views/trainings/edit.blade.php

```
public function edit(Training $training)
{
    return view('trainings.edit')->with(compact('training'));
}
```

# Views – Edit View

Views are stored in the  
resources/views directory

Go to

resources/views/trainings/edit.blade.php

```

@extends('layouts.app')

@section('content')


Edit Training



<form action="" method="POST">
    @csrf
    <div class="form-group">
        <label for="title">Title</label>
        <input type="text" class="form-control" name="title" value="{{ $training->title }}>
    </div>
    <div class="form-group">
        <label for="description">Description</label>
        <textarea name="description" cols="20" rows="10" class="form-control">{{ $training->title }}</textarea>
    </div>
    <div class="form-group">
        <label for="trainer">Trainer</label>
        <input type="text" class="form-control" name="trainer" value="{{ $training->trainer }}>
    </div>
    <div class="form-group">
        <button type="submit" class="btn btn-primary">Update Training</button>
        <a href="{{ route('training:index') }}" class="btn btn-link">Cancel</a>
    </div>


```

# Controller - Update

Description	URL	Controller Function	View File
Process the create form submit and save the training to the database.	POST example.com/trainings/{id}	update()	NONE



The screenshot shows a code editor with a file tree on the left and the code for the `TrainingController.php` file on the right. The file tree includes `Controllers`, `Auth`, `Middleware`, `Providers`, `bootstrap`, `config`, `database`, `node_modules`, `public`, and `resources` directories. The `TrainingController.php` file is selected. The code editor displays the following PHP code:

```
public function update(Request $request, $id)
{
    //save edited record
    $training = Training::find($id);
    $training->update($request->only(
        'title', 'description', 'trainer'
    ));

    return redirect('/trainings')->with('success',
        'Training info has been updated!!');
}
```

The cursor is positioned at the start of the line `//save edited record`. The line numbers 89 through 99 are visible on the left side of the code editor.

# Output screen

Update Record

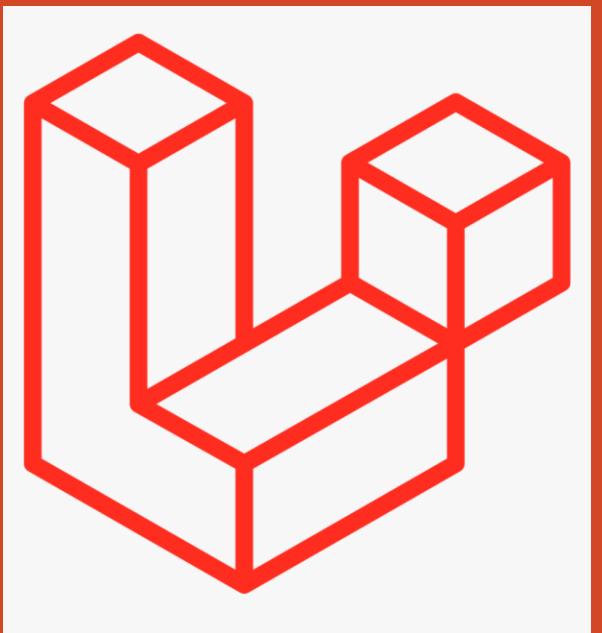
Title  
Laravel Basics 1

Description  
3 days Laravel course with PHP OOP

Trainer  
En Kerul

**Save Update**

Training Index		
Training record has been updated		
Title	Task	
Laravel Basics 1	  	
Laravel Intermediate	  	
Laravel Too Advanced	  	



w09

# Delete records

(Padam rekod)

# DISCLAIMER

Codes are for demo purposes. Examples showed here are only to validate the proof of concepts. Security aspects the demonstrated examples are beyond the coverage of these video tutorials. You are advised to explore / search after advanced/security topics of PHP web programming as these tutorials are only for beginner level.

Suitable to view in DESKTOP

# <https://github.com/khirulnizam/10tbtm/>

## Slides & codes

The screenshot shows a GitHub repository page for 'khirulnizam / 10tbtm'. The repository is public and has 1 branch and 0 tags. The commit history for the 'main' branch shows two commits from 'khirulnizam' yesterday, creating a new layout. The commits are:

Commit	Message	Date	Commits
7c08961	create-xlayout	yesterday	2
app	create-xlayout	yesterday	
bootstrap	Initial commit	yesterday	
config	Initial commit	yesterday	
database	Initial commit	yesterday	
public	Initial commit	yesterday	

# Overview

Insert||Listing/Search

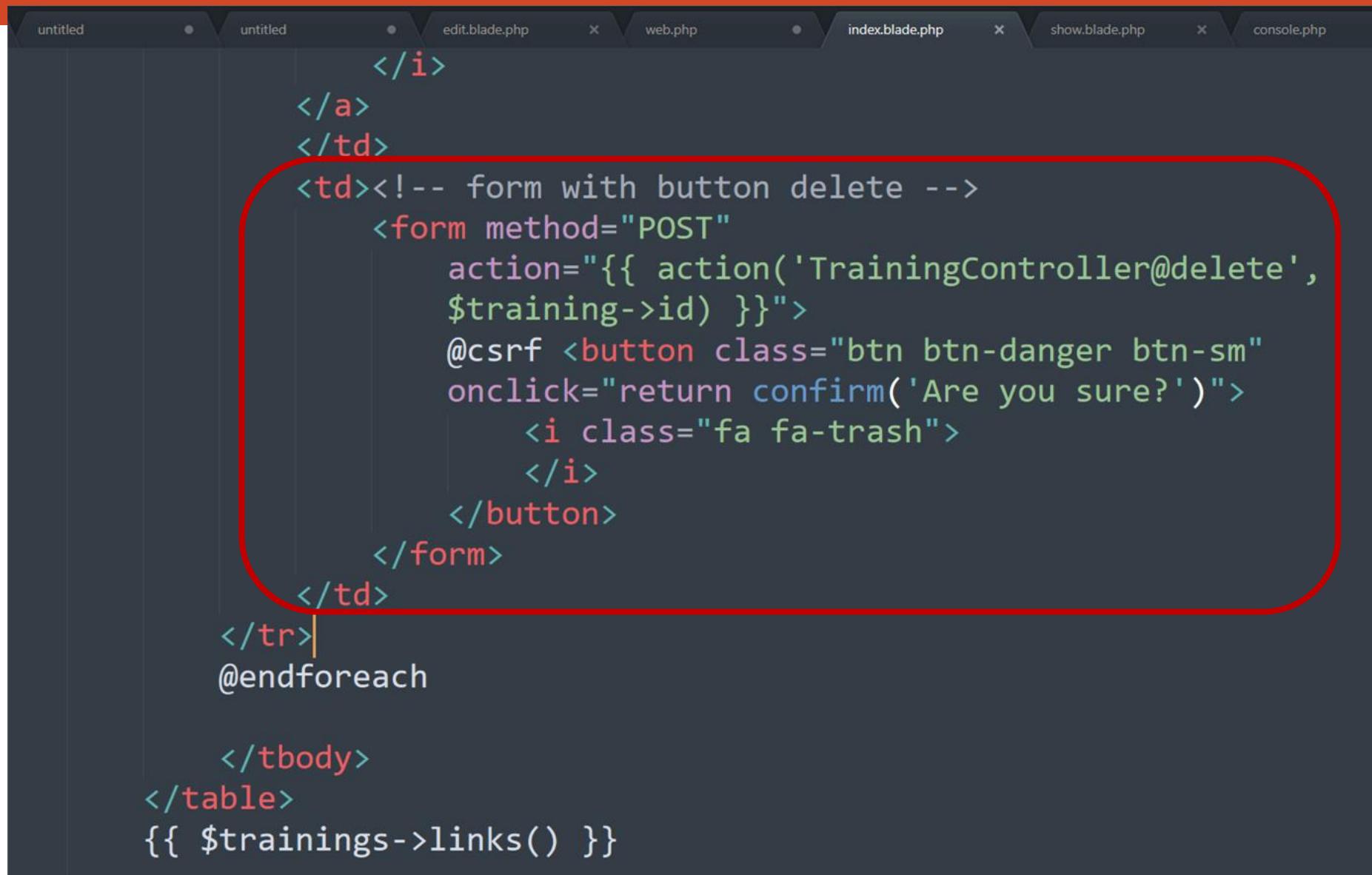
lara7.test says  
Are you sure to delete the record (id:1) ?

OK Cancel

Training Index  

id	Title	Task
1	Laravel Basics 1	  
2	Laravel Intermediate	  
3	Laravel Too Advanced	  

# Delete button inside FORM



The screenshot shows a code editor with multiple tabs at the top: 'untitled', 'untitled', 'edit.blade.php', 'web.php', 'index.blade.php' (which is the active tab), 'show.blade.php', and 'console.php'. The main content area displays the following Blade template code:

```
</i>
</a>
</td>
<td><!-- form with button delete -->
<form method="POST"
      action="{{ action('TrainingController@delete',
      $training->id) }}"
      @csrf <button class="btn btn-danger btn-sm"
      onclick="return confirm('Are you sure?')">
        <i class="fa fa-trash">
        </i>
      </button>
    </form>
</td>
</tr>
@endforeach

</tbody>
</table>
{{ $trainings->links() }}
```

A red oval highlights the entire section of code within the first table row that contains the delete button. The code is enclosed in a `<td>` element, which is itself within a `<tr>` element. The delete button is generated by a `<form>` tag with a POST method and a CSRF token, and it triggers a confirmation dialog before performing the delete action.

# Routing – Group and Named Route

All Laravel routes are defined in your route files,  
which are located in the routes directory.

<https://laravel.com/docs/7.x/routing>.

## routes

- api.php
- channels.php
- console.php
- web.php

```
//group routing
//for training module
Route::group([
    'middleware'=>'auth',//lock this route group to only authenticated users
    'prefix'=>'trainings',
    'as'=>'training:',
],function(){
    Route::get('/', 'TrainingController@index')->name('index');
    Route::get('/create', 'TrainingController@create')->name('create');
    Route::post('/create', 'TrainingController@store')->name('store');
    Route::get('/{training}', 'TrainingController@show')->name('show');
    Route::get('/{training}/edit', 'TrainingController@edit')->name('edit');
    Route::post('/{training}', 'TrainingController@update')->name('update');
    Route::post('/{training}/delete', 'TrainingController@delete')->name('delete');
});
```

# Routing – Group and Named Route

```
php artisan route:list
```

[https://laravel.com/docs/7.x/routing.](https://laravel.com/docs/7.x/routing)

GET HEAD	trainings	training:index	App\Http\Controllers\TrainingController@index	web,auth
GET HEAD	trainings/create	training:create	App\Http\Controllers\TrainingController@create	web,auth
POST	trainings/create	training:store	App\Http\Controllers\TrainingController@store	web,auth
GET HEAD	trainings/{training}	training:show	App\Http\Controllers\TrainingController@show	web,auth
POST	trainings/{training}	training:update	App\Http\Controllers\TrainingController@update	web,auth
POST	trainings/{training}/delete	training:delete	App\Http\Controllers\TrainingController@delete	web,auth
GET HEAD	trainings/{training}/edit	training:edit	App\Http\Controllers\TrainingController@edit	web,auth

```
D:\laragon\www\lara7 (master)
λ php artisan route:list
```

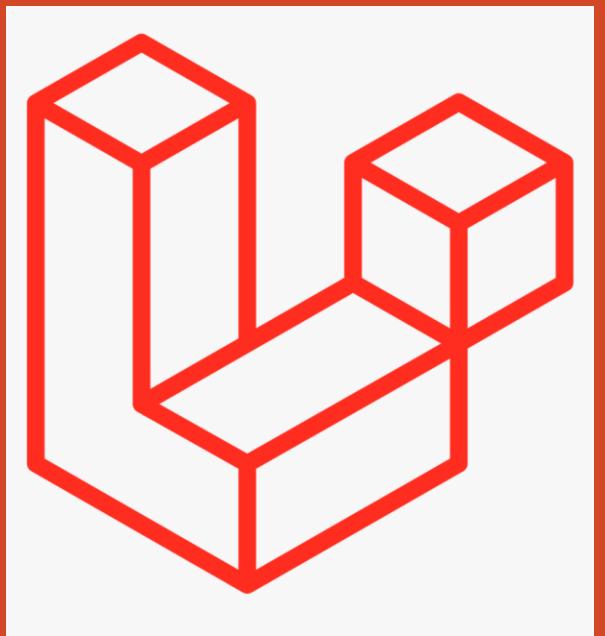
http://laravel.com/docs/7.x/routing

# Controller – Destroy/Delete

Description	URL	Controller Function	View File
Delete the training from DB.	GET example.com/trainings{id}/delete	delete()	NONE

```
public function delete(Training $training)
{
    $training->delete();

    return redirect('/trainings');
}
```



# Laravel t17 @csrf

(Built-in security token  
for post method)  
[bit.ly/fstmtubelaravel](http://bit.ly/fstmtubelaravel)

# DISCLAIMER

Codes are for demo purposes. Examples showed here are only to validate the proof of concepts. Security aspects the demonstrated examples are beyond the coverage of these video tutorials. You are advised to explore / search after advanced/security topics of PHP web programming as these tutorials are only for beginner level.

Suitable to view in DESKTOP

# <https://github.com/khirulnizam/10tbtm/>

## Slides & codes

The screenshot shows a GitHub repository page for 'khirulnizam / 10tbtm'. The repository is public and has 1 branch and 0 tags. The code tab is selected. The commit history shows two commits from 'khirulnizam' yesterday, creating a new repository structure.

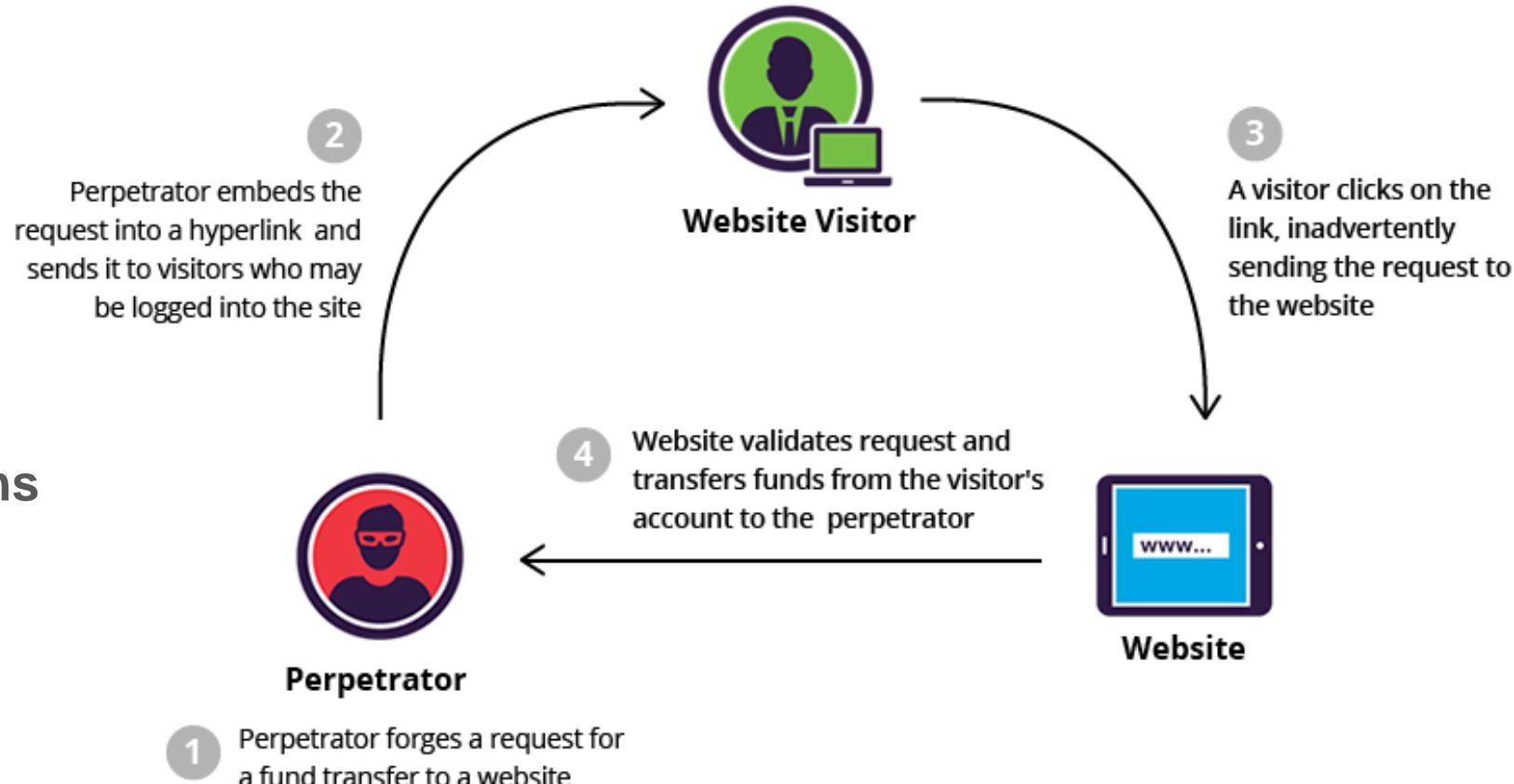
Commit	Message	Date	Commits
create-xlayout	khirulnizam create-xlayout	yesterday	2 commits
app	create-xlayout	yesterday	
bootstrap	Initial commit	yesterday	
config	Initial commit	yesterday	
database	Initial commit	yesterday	
public	Initial commit	yesterday	

# What is CSRF?

OWASP.org

## Cross-Site Request Forgery

- is an attack that forces an end user to **execute unwanted actions** on a web application in which they're currently authenticated.



# Implementation in Laravel blade

```
<!-- form start -->
<form action="{{action('TrainingController@update',
$training->id )}}" method="POST">
```

```
@csrf
```

```
Title
```

```
<input name="title" type="text"
class="form-control"
```

```
← → C ⓘ Not secure | view-source:lara7.test/trainings/1/edit
```

```
81
82 <!-- form start -->
83 <form action="http://lara7.test/trainings/1" method="POST">
84   <input type="hidden" name="_token"
85     value="r4SPb6vSY1tu03UHiZUphq2HsvogIJZYJjlBq2PX">
86   Title
87   <input name="title" type="text"
88     class="form-control"
89     value="Laravel Basics 1"
90   >
```

# Location of csrf app\_key

Use to hash security token

```
D:\laragon\www\lara7 (master)
λ php artisan key:generate
Application key set successfully.
```

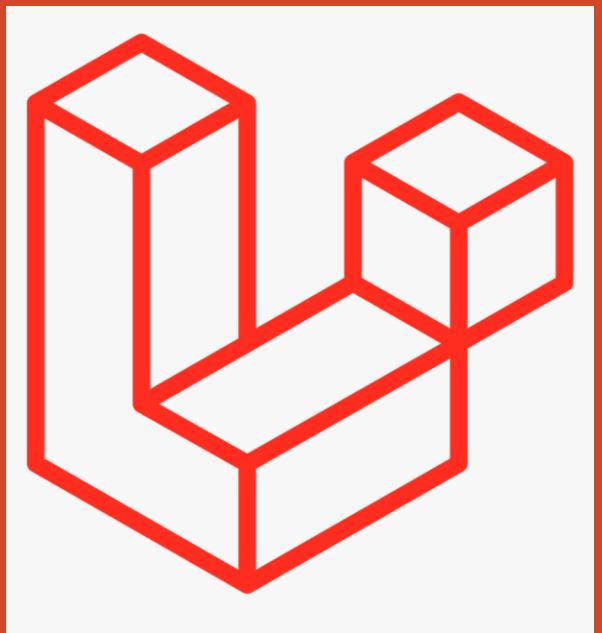
```
D:\laragon\www\lara7 (master)
λ php artisan key:generate --show
base64:64tnzwWAKX9tvEIrYeOK+aQ+3A06SUTvdzh0T3Uq1d0=
```



```
.env ✘ TrainingController.php ✘ untitled • untitled • edit.blade.php ✘ app.php ✘ auth.php ✘ web.php • index.blade.php ✘
```

```
1 APP_NAME=Laravel
2 APP_ENV=local
3 APP_KEY=base64:1sxP7anB8mPwR0m3oNUCHQiiGbngiCrrWNWPx0fD3mI=
4 APP_DEBUG=true
5 APP_URL=http://localhost
6
```

A screenshot of a code editor showing the .env file. The first three lines of the file are highlighted with a red box. The first line contains the variable APP\_NAME with the value Laravel. The second line contains APP\_ENV with the value local. The third line contains APP\_KEY with a long base64 encoded string. The code editor interface includes tabs for other files like TrainingController.php, edit.blade.php, app.php, auth.php, web.php, and index.blade.php.



# Laravel t18

## Apply Bootstrap Layout (Pasang layout Bootstrap baru)

# Output overview

Search your prefered training...

Search

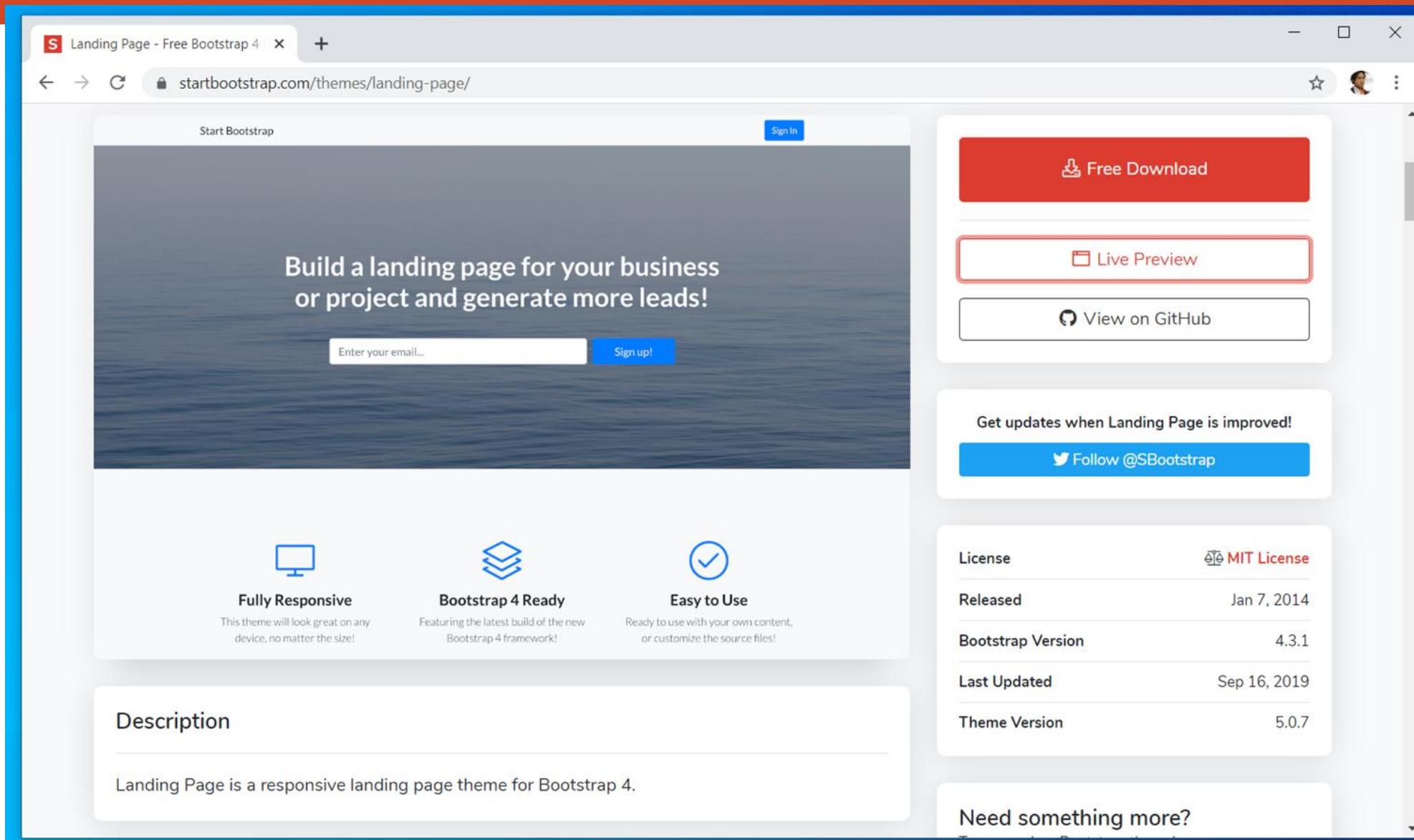
id	title	trainer	
1	Laravel Basics	Khirulnizam Abd Rahman	Show
2	Laravel Intermediate	Tarmizi Sanusi	Show
3	Laravel Advanced with Unit Test	Nasrul Hazim	Show

# DISCLAIMER

Codes are for demo purposes. Examples showed here are only to validate the proof of concepts. Security aspects the demonstrated examples are beyond the coverage of these video tutorials. You are advised to explore / search after advanced/security topics of PHP web programming as these tutorials are only for beginner level.

Suitable to view in DESKTOP

# Theme from <https://startbootstrap.com/themes/landing-page/>



The screenshot shows a web browser displaying the "Landing Page - Free Bootstrap 4" theme from Start Bootstrap. The page has a dark blue header with the "Start Bootstrap" logo and a "Sign In" button. The main content area features a large blue gradient background with white text: "Build a landing page for your business or project and generate more leads!". Below this is a form with a placeholder "Enter your email..." and a "Sign up!" button. To the right, there's a sidebar with a "Free Download" button, a "Live Preview" button, and a "View on GitHub" button. Further down, there's a section for updates with a "Follow @SBootstrap" button, and a table of metadata including license (MIT), release date (Jan 7, 2014), bootstrap version (4.3.1), last update (Sep 16, 2019), and theme version (5.0.7). At the bottom, there's a "Description" section and a "Need something more?" link.

Landing Page - Free Bootstrap 4

startbootstrap.com/themes/landing-page/

Start Bootstrap

Sign In

Build a landing page for your business or project and generate more leads!

Enter your email... Sign up!

Free Download

Live Preview

View on GitHub

Get updates when Landing Page is improved!

Follow @SBootstrap

License	MIT License
Released	Jan 7, 2014
Bootstrap Version	4.3.1
Last Updated	Sep 16, 2019
Theme Version	5.0.7

Description

Landing Page is a responsive landing page theme for Bootstrap 4.

Need something more?

# Asset/resources for template

File		Home	Share	View
←	→	↑	↓	☰
This PC	>	Windows (C:) > laragon > www > lara7 > public		
Name	Date			
assets	3/4/2020 7:19 PM			
css	28/3/2020 11:09 AM			
js	28/3/2020 11:09 AM			
landpage	3/4/2020 3:29 PM			
.htaccess	28/3/2020 11:09 AM			
browserconfig	3/4/2020 9:19 AM			
favicon	28/3/2020 11:09 AM			
14 items	1 item selected			

File		Home	Share	View
←	→	↑	↓	☰
This PC	>	Windows (C:) > laragon > www > startbootstrap-landing-page-gh-pages		
Name	Date modified			
Quick access	17/9/2019 5			
Desktop	3/4/2020 8:00			
Downloads	17/9/2019 5			
Documents	17/9/2019 5			
Pictures	17/9/2019 5			
Data (D:)	17/9/2019 5			
0000-FE-sesi-2-2019-20	17/9/2019 5			
Laravel	17/9/2019 5			
PHP	17/9/2019 5			
ScreencastingOBS	17/9/2019 5			
OneDrive	17/9/2019 5			
This PC	17/9/2019 5			
3D Object	17/9/2019 5			
Desktop	17/9/2019 5			
Documents	17/9/2019 5			
Downloads	17/9/2019 5			
Music	17/9/2019 5			
Pictures	17/9/2019 5			
Videos	17/9/2019 5			
Windows (C:)	17/9/2019 5			
Data (D:)	17/9/2019 5			
Network	17/9/2019 5			
index	17/9/2019 5			
LICENSE	17/9/2019 5			
package.json	17/9/2019 5			
package-lock.json	17/9/2019 5			
README.md	17/9/2019 5			
13 items				

File		Home	Share	View
←	→	↑	↓	☰
This PC	>	Windows (C:) > laragon > www > lara7 > resources > views > layouts		
Name	Date modified	Type		
app.blade.php	28/3/2020 11:09 AM	PHP		
mobitemp.blade.php	3/4/2020 7:28 PM	PHP		
template_landingpage.blade.php	3/4/2020 7:55 PM	PHP		
3 items				

# Blade asset( ) function – apply on both css & js

The image shows a code editor interface with a sidebar containing a file tree and the main area displaying a blade template file.

**File Tree:**

- app.blade.php
- mobitemp.blade.php
- template\_landingpage.blade.php
- trainings
  - create.blade.php
  - index.blade.php
  - Dapatkan percuma 20+ playlist tut...
  - home.blade.php
  - landingpage.blade.php
  - welcome.blade.php
- routes
  - api.php
  - channels.php
  - console.php
  - web.php
- storage
- tests
- vendor
  - .editorconfig
  - .env
  - .gitattributes
  - .gitignore
  - .styleci.yml
  - artisan
  - composer.json
  - composer.lock
- Laravel Training Youtube Tutorial Set2.pptx
- Laravel Training Youtube Tutorial.pptx
- package-lock.json
- package.json
- phpunit.xml
- README.md
- server.php
- webpack.mix.js

**Code Editor Area:**

```
21 <!-- Custom fonts for this template -->
22 <link href="{{asset('landpage/vendor/
23 fontawesome-free/css/all.min.css')}}" rel="stylesheet">
24 <link href="{{asset('landpage/vendor/
25 simple-line-icons/css/simple-line-icons.css')}}" rel="stylesheet" type="text/css">
26 <link href="https://fonts.googleapis.com/css?family=Lato:300,400,700,300italic,400italic,700italic" rel="stylesheet" type="text/css">
27 <!-- Custom styles for this template -->
28 <link href="{{asset('landpage/css/
29 landing-page.min.css')}}" rel="stylesheet">
30 </head>
```

# @yield( )

The screenshot shows a code editor with several tabs at the top: welcome.blade.php, template\_landingpage.blade.php (which is the active tab), web.php, landingpage.blade.php, index.blade.php, TrainingController.php, and home. On the left, there's a sidebar with project files: sass, views (auth, layouts - app.blade.php, mobitemp.blade.php, template\_landingpage.blade.php), trainings (create.blade.php, index.blade.php), routes (api.php, channels.php, console.php, web.php), storage, tests, vendor, .editorconfig, .env, .gitattributes, .gitignore, .styleci.yml, artisan, composer.json, and composer.lock.

The main content area displays the following code:

```
77 </header>
78
79 <!-- Main content - display welcome and search
training here -->
80
81 @yield('public_content')
82
83 <! end main content -->
84
85 <!-- Footer -->
86 <footer class="footer bg-light">
87   <div class="container">
88     <div class="row">
```

The line `@yield('public_content')` is highlighted with a red rounded rectangle. The word `search` in the line above it is also highlighted with a green rounded rectangle.

# Your ICT Training partner

[Search](#)

id	title	trainer	
1	Laravel Basics	Khirulnizam Abd Rahman	<a href="#">Show</a>
2	Laravel Intermediate	Tarmizi Sanusi	<a href="#">Show</a>
3	Laravel Advanced with Unit Test	Nasrul Hazim	<a href="#">Show</a>

@yield( )

[About](#) · [Contact](#) · [Terms of Use](#) · [Privacy Policy](#)

© FSTM KUIS 2020. All Rights Reserved.



# Output overview

Search your prefered training...

Search

id	title	trainer	
1	Laravel Basics	Khirulnizam Abd Rahman	Show
2	Laravel Intermediate	Tarmizi Sanusi	Show
3	Laravel Advanced with Unit Test	Nasrul Hazim	Show

# landingpage.blade.php

The screenshot shows the Sublime Text editor with the file `landingpage.blade.php` open. The file path is `C:\laragon\www\lara7\resources\views\landingpage.blade.php`. The code is a Blade template:

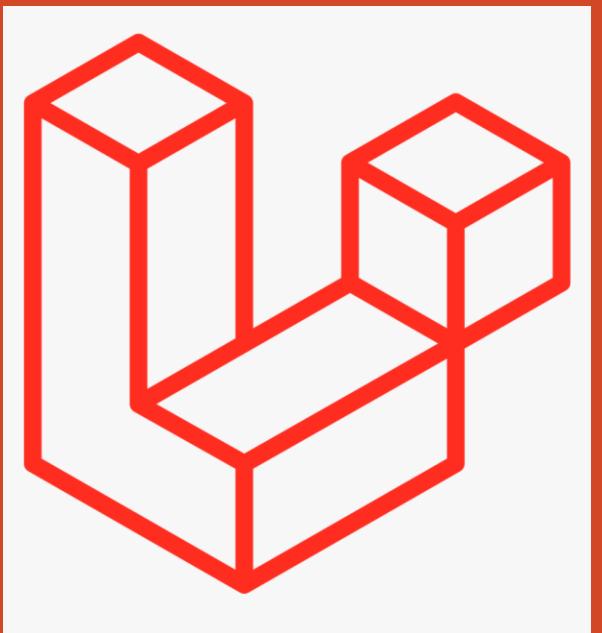
```
1 @extends('layouts.template_landingpage')
2
3 @section('public_content')
4     <div class="container">
5         <div class="row justify-content-center">
6             <div class="col-md-12">
7                 <div class="card">
8                     <div class="card-body">
9
10                    <table class="table table-striped">
11                        <thead><tr>
12                            <th>id</th>
13                            <th>title</th>
14                            <th>trainer</th>
15                            <th></th>
16                        </tr></thead>
```

The sidebar on the left shows the project structure with folders like `views`, `layouts`, `trainings`, and `routes`, and files like `api.php`, `channels.php`, `console.php`, `web.php`, and `.env`.

# route/web.php

```
• | 15 //the landing page route
• | 16 Route::get('/', function () {
• | 17     //welcome /public search
• | 18     $trainings = Training::all();
• | 19     return view('landingpage')->with(compact(
• | 20         'trainings'));
• | 21     });

routes
api.php
channels.php
console.php
web.php
storage
tests
vendor
.editorconfig
.env
.gitattributes
.gitignore
/* .styleci.yml
artisan
/* composer.json
```



# Laravel t19

## Apply Bootstrap on Admin Dashboard

### (Pasang layout Bootstrap baru pada Panel Admin)

# Output overview

The screenshot shows a web browser window titled "FSTM KUIS - template Sbadmin2" displaying the "trainings" page. The URL in the address bar is "Not secure | lara7.test/trainings". The page has a blue header with the title "FSTM ADMIN 2" and a sidebar on the left containing links for "Dashboard", "Insert new record", and "Listing/Search". The main content area is titled "FSTM KUIS Admin Panel" and "Training Index". It features a table with three rows, each representing a training course. The columns are "id", "title", and "tasks". The "id" column contains values 1, 2, and 3. The "title" column contains the course names: "Laravel Basics", "Laravel Intermediate", and "Laravel Advanced with Unit Test". The "tasks" column contains three icons per row: a green document icon, an orange edit icon, and a red trash bin icon. At the bottom of the page, there is a copyright notice: "Copyright © FSTM KUIS 2020".

id	title	tasks
1	Laravel Basics	
2	Laravel Intermediate	
3	Laravel Advanced with Unit Test	

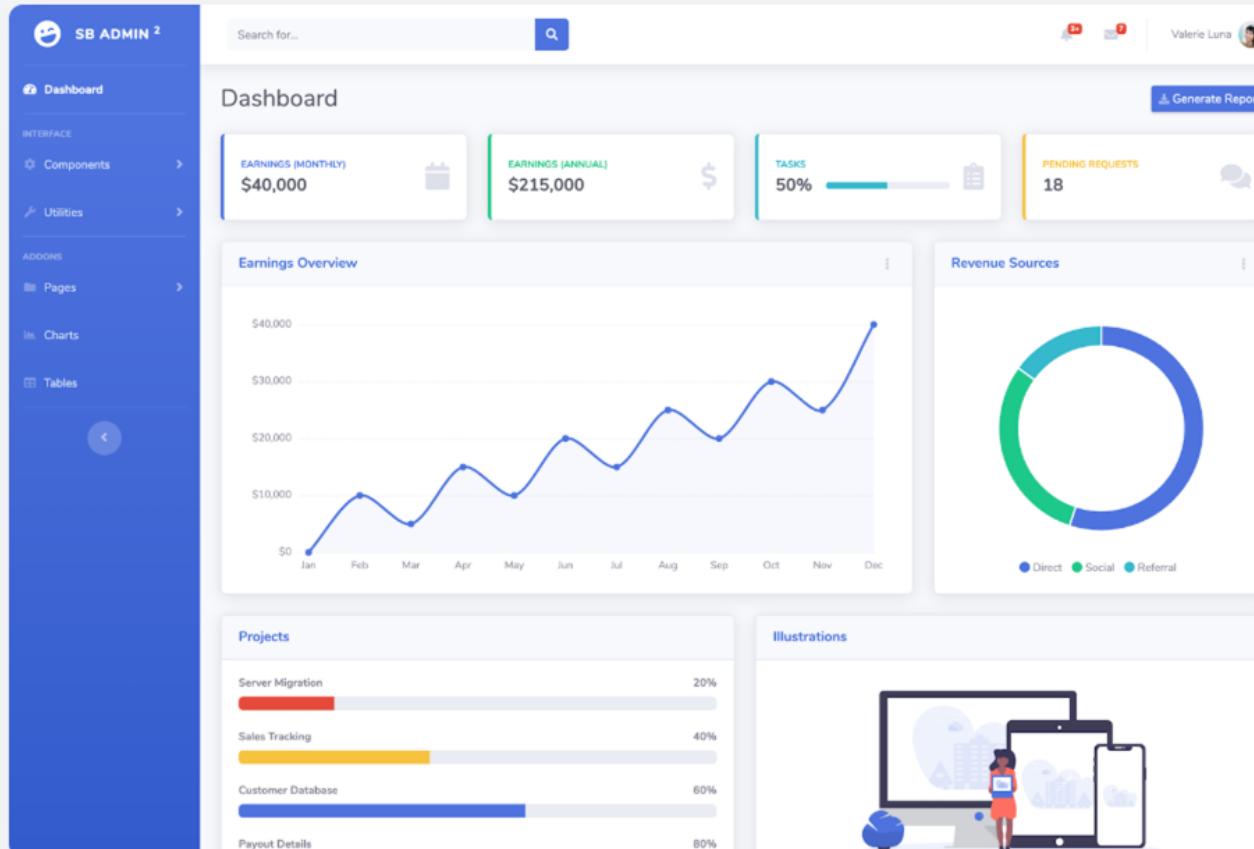
Copyright © FSTM KUIS 2020

# DISCLAIMER

Codes are for demo purposes. Examples showed here are only to validate the proof of concepts. Security aspects the demonstrated examples are beyond the coverage of these video tutorials. You are advised to explore / search after advanced/security topics of PHP web programming as these tutorials are only for beginner level.

Suitable to view in DESKTOP

# Theme from <https://startbootstrap.com/themes/sb-admin-2/>



The dashboard features a left sidebar with navigation links for Dashboard, Components, Utilities, Pages, Charts, and Tables. The main area includes a search bar, user profile, and a 'Generate Report' button. Key metrics displayed are Earnings (Monthly) at \$40,000, Earnings (Annual) at \$215,000, Tasks at 50% completion, and Pending Requests at 18. It also includes a line chart for Earnings Overview and a donut chart for Revenue Sources (Direct, Social, Referral).

### Description

SB Admin 2 is a free, open source, Bootstrap 4 based admin theme perfect for quickly creating dashboards and web applications. It's modern design style with subtle shadows and a card-

 [Free Download](#)

 [Live Preview](#)

 [View on GitHub](#)

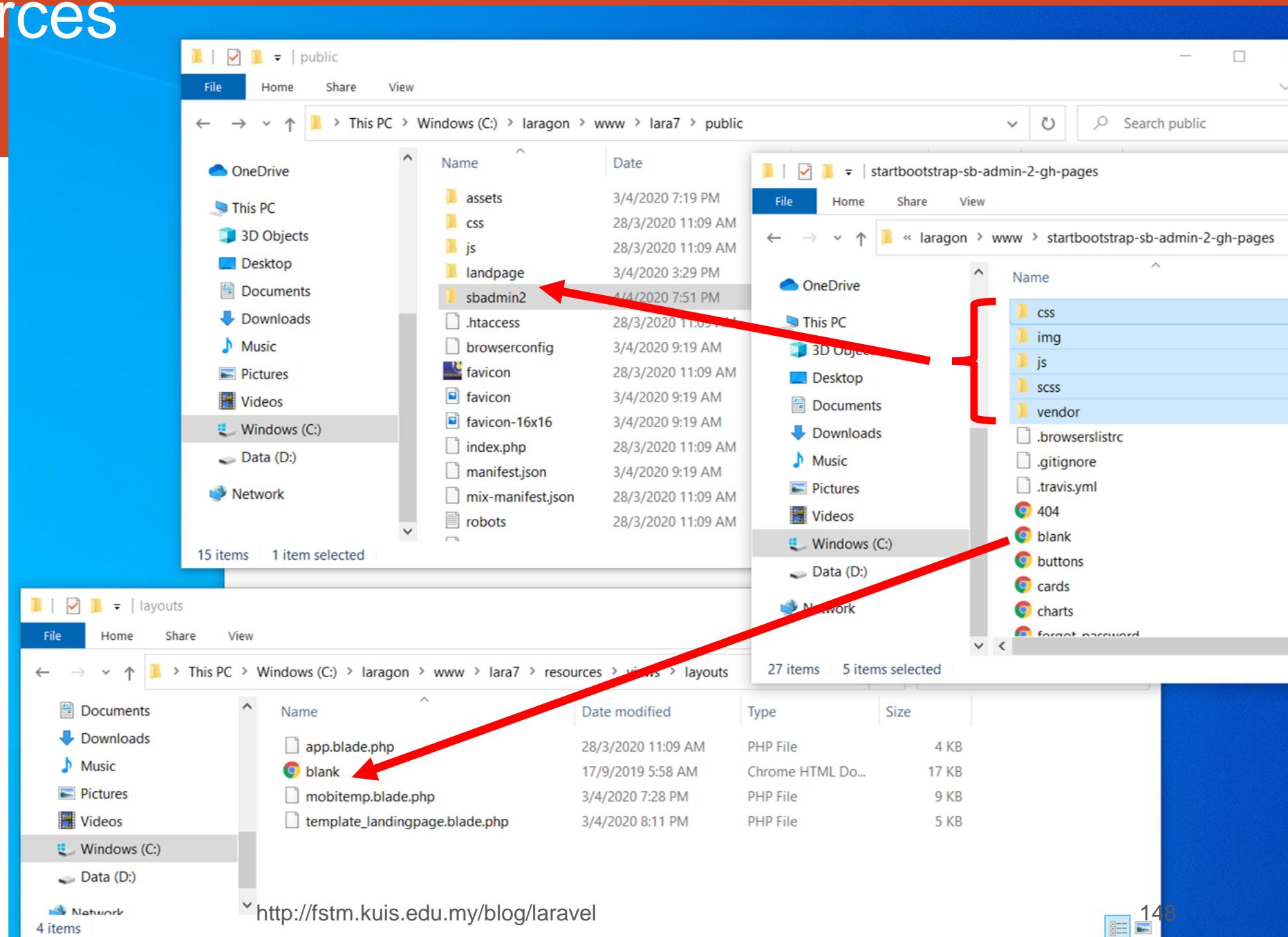
Get updates when SB Admin 2 is improved!

 [Follow @SBootstrap](#)

License	 <a href="#">MIT License</a>
Released	Jan 14, 2014
Bootstrap Version	4.3.1
Last Updated	Sep 16, 2019
Theme Version	4.0.7

[Read Documentation](#)

# Asset/resources for template



# First time display the new template

[SB Admin 2](#)

- 
- [Dashboard](#)

Interface

- [Components](#)

Custom Components:

[Buttons](#) [Cards](#)

- [Utilities](#)

Custom Utilities:

[Colors](#) [Borders](#) [Animations](#) [Other](#)

---

Addons

- [Pages](#)

Login Screens:

[Login](#) [Register](#) [Forgot Password](#)

Other Pages:

[404 Page](#) [Blank Page](#)

- [Charts](#)

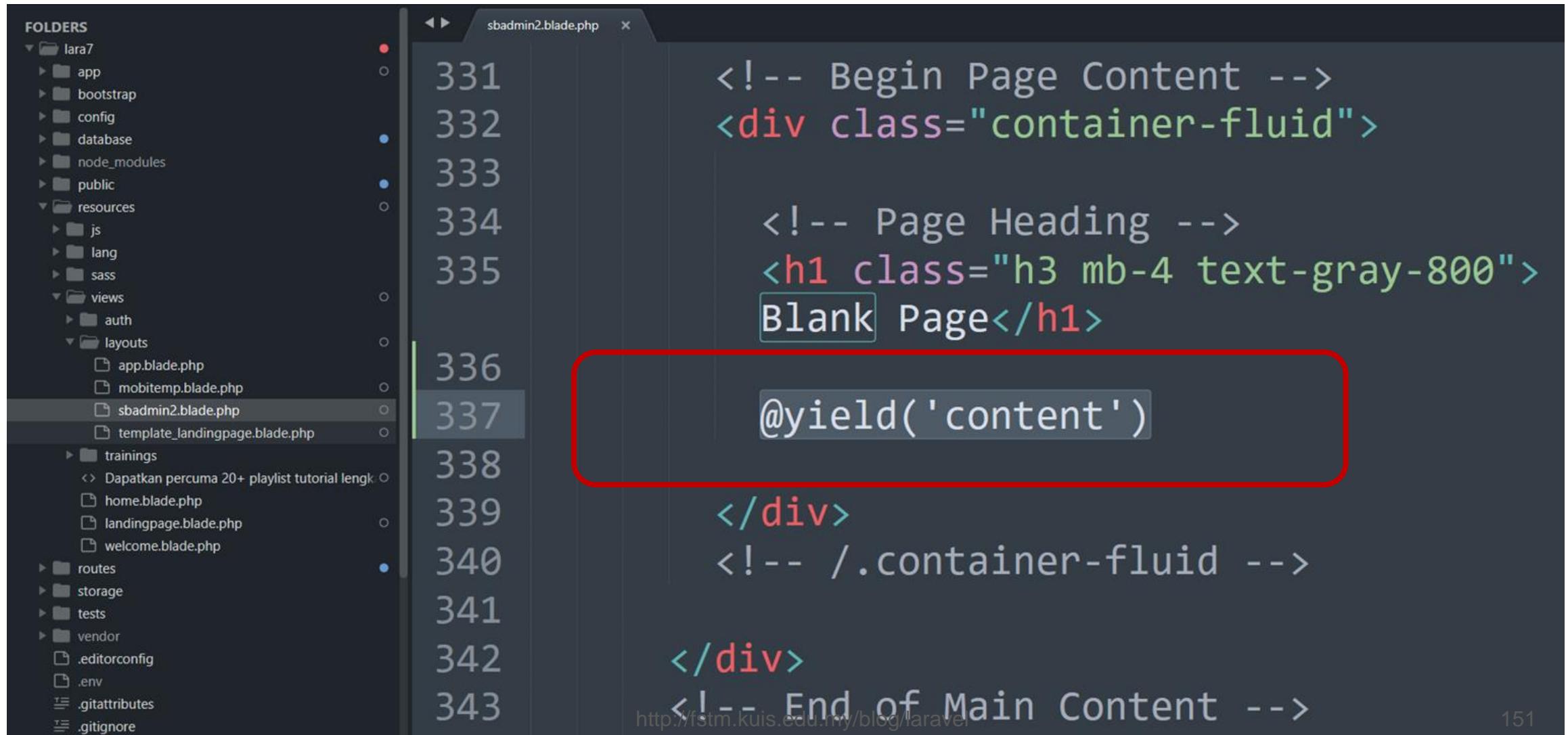
# Blade asset( ) function – apply on both css & js

The screenshot shows a code editor interface with two tabs: 'sbadmin2.blade.php' and 'index.blade.php'. The 'index.blade.php' tab is active, displaying the following code:

```
13
14
15    <!-- Custom fonts for this template-->
16    <link href="{{asset('sbadmin2/vendor/
17        fontawesome-free/css/all.min.css')}}" rel="stylesheet" type="text/css">
18    <link href="https://fonts.googleapis.com/css?family=Nunito:200,200i,300,300i,400,400i,600,600i
19        ,700,700i,800,800i,900,900i" rel="stylesheet">
20
21    <!-- Custom styles for this template-->
22    <link href="{{asset('sbadmin2/css/
23        sb-admin-2.min.css')}}" rel="stylesheet">
24
25    </head>
```

The file structure on the left side of the editor shows various Laravel components like 'lang', 'sass', 'views', 'auth', 'layouts', 'trainings', 'routes', 'storage', 'tests', 'vendor', and several configuration files.

# @yield( )



The image shows a code editor interface with a dark theme. On the left is a file tree labeled "FOLDERS" containing project files like lara7, app, bootstrap, config, database, node\_modules, public, resources (js, lang, sass), views (auth, layouts), trainings, routes, storage, tests, vendor, .editorconfig, .env, .gitattributes, and .gitignore. The main pane displays a file named "sbadmin2.blade.php". The code is as follows:

```
<!-- Begin Page Content -->
<div class="container-fluid">

    <!-- Page Heading -->
    <h1 class="h3 mb-4 text-gray-800">
        Blank Page</h1>

    @yield('content')

</div>
<!-- /.container-fluid -->

</div>
<!-- End of Main Content -->
```

A red rectangular box highlights the line of code "@yield('content')". The line numbers on the left range from 31 to 43.

# @yield( )

The screenshot shows the FSTM KUIS Admin Panel interface. On the left, a sidebar titled "FSTM ADMIN 2" contains links for "Dashboard", "Insert new record", and "Listing/Search". The main content area is titled "FSTM KUIS Admin Panel" and displays a table titled "Training Index". The table has columns for "id", "title", and "tasks". Three rows are listed:

id	title	tasks
1	Laravel Basics	
2	Laravel Intermediate	
3	Laravel Advanced with Unit Test	

A large red arrow points from the text "@yield( )" at the top left of the main content area towards the table, indicating its use in rendering the page content.

http://fstm.kuis.edu.my/blog/laravel

152

# Output overview

The screenshot shows a web browser window titled "FSTM KUIS - template Sbadmin2" with the URL "lara7.test/trainings". The page is titled "FSTM KUIS Admin Panel" and displays a "Training Index". On the left, there is a sidebar with icons for Dashboard, Insert new record, and Listing/Search. The main content area has a search bar with a "Search" button. A table lists three training entries:

ID	Title	Tasks
1	Laravel Basics	
2	Laravel Intermediate	
3	Laravel Advanced with Unit Test	

At the bottom, there is a copyright notice: "Copyright © FSTM KUIS 2020".

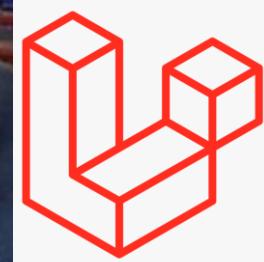
# index.blade.php

C:\laragon\www\lara7\resources\views\trainings\index.blade.php (lara7) - Sublime Text (UNREGISTERED)

File Edit Selection Find View Goto Tools Project Preferences Help

The screenshot shows a Sublime Text editor window with the file 'index.blade.php' open. The file content is a Blade template for a training index page. The code includes an extends directive to inherit from 'layouts.sbadmin2', a section for the main content area, and a card header with the text 'Training Index'. The file is part of a Laravel project structure, with other blade files like 'create.blade.php' and 'welcome.blade.php' visible in the sidebar.

```
1 @extends('layouts.sbadmin2')
2
3 @section('content')
4 <div class="container">
5   <div class="row justify-content-center">
6     <div class="col-md-10">
7       <div class="card">
8         <div class="card-header">Training
9           Index
10        </div>
11        <div class="float-right">
12          <form class="form-inline" method="get"
13            action="{{ route('training.index') }}>
```



**Laravel t20**  
**Muat naik fail gambar**

# Upload image

FSTM KUIS - template Sbadmin2 x +

Not secure | lara7.test/trainings/10

FSTM ADMIN 2

Dashboard

Insert new record

Listing/Search

FSTM ADMIN 2

Insert New Record

Title  
Flutter Basics to Intermediate

Description  
Develop Multiplatform Apps using Flutter & Android Studio

Trainer  
EnKerul

Poster image filename  
2020-04-07-Flutter Basics to Intermediate.png

Upload training poster (png, jpg, img file only with max 2MB filesize)

Choose File bannerBMI.png

Save new Record

Copyright © FSTM KUIS 2020

http://fstm.kuis.edu.my/blog/laravel

FSTM KUIS - template Sbadmin2 x +

Not secure | lara7.test/trainings/10

FSTM ADMIN 2

Dashboard

Insert new record

Listing/Search

Show all data Record

Title  
Flutter Basics to Intermediate

Description  
Develop Multiplatform Apps using Flutter & Android Studio

Trainer  
EnKerul

Poster image filename  
2020-04-07-Flutter Basics to Intermediate.png



Show uploaded image

Back

Upload image

Show uploaded image

# DISCLAIMER

Codes are for demo purposes. Examples showed here are only to validate the proof of concepts. Security aspects the demonstrated examples are beyond the coverage of these video tutorials. You are advised to explore / search after advanced/security topics of PHP web programming as these tutorials are only for beginner level.

Suitable to view in DESKTOP

# Add column to table trainings

Laragon\lara7test\trainings - HeidiSQL Portable 10.2.0.5599

File Edit Search Tools Go to Help

Database filter Table filter Host: localhost Database: lara7test Table: trainings Data Query

Laragon

- information\_schema 160.0 KiB
- bengkelpdo
- lara7
- lara7test
  - failed\_jobs 16.0 KiB
  - migrations 16.0 KiB
  - new 16.0 KiB
  - password\_resets 16.0 KiB
  - subjects 16.0 KiB
  - trainings 16.0 KiB
  - users 16.0 KiB
- mysql
- newsystem
- performance\_schema

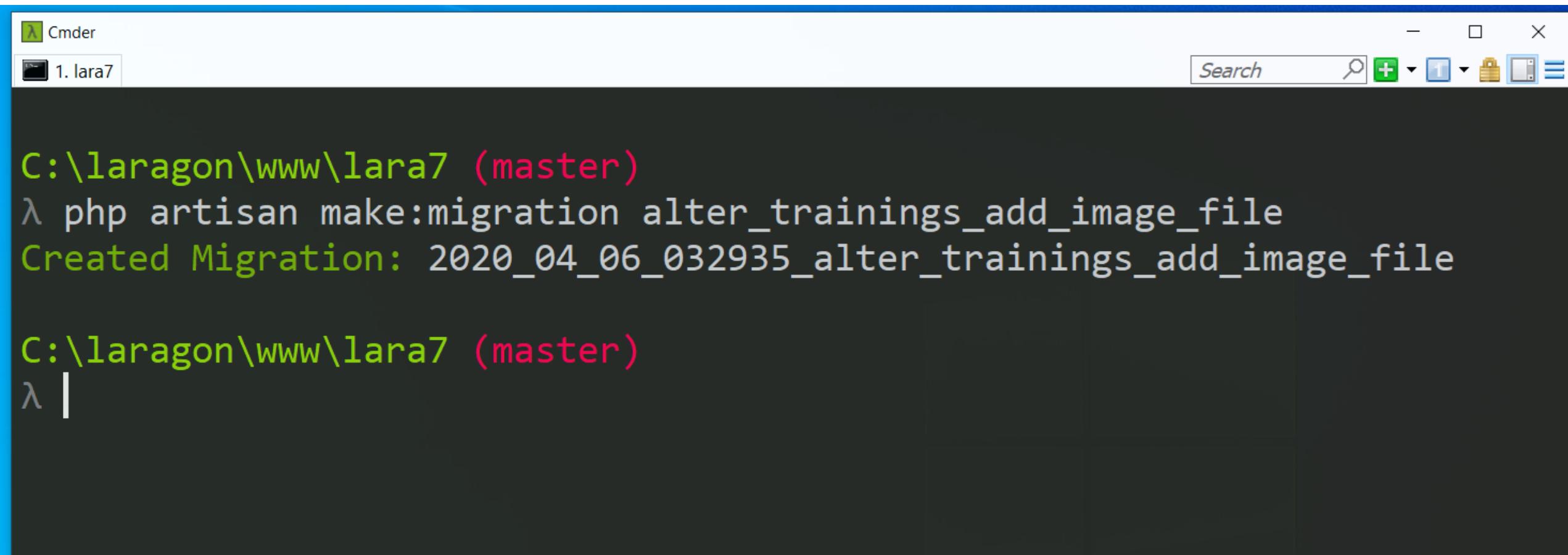
lara7test.trainings: 8 rows total (approximately) > Next ◆ Show all ▼ Sorting

	id	title	description	trainer	filename
	1	Laravel Basics	2 days Laravel course	Khirulnizam Abd ...	(NULL)
	2	Laravel Intermediate	2 days Laravel course...	Tarmizi Sanusi	(NULL)
	3	Laravel Advanced wit...	2 days Laravel course...	Nasrul Hazim	(NULL)
	4	Laravel API 2	Training to generate ...	Kerul Rahman2	2020-04-06-Laravel API 2...
	5	Open Broadcasting S...	Open Broadcasting S...	En Kerul	2020-04-06-Open Broadca...
	6	Test 2 WO Pic	Upload record WO Pic	Mr WHO	(NULL)
	8	Flutter 101	Flutter to develop mu...	Mr Mano	2020-04-06-Flutter 101.jpeg
	9	Ionic 101	Basic Ionic training to...	enKerul	2020-04-06-Ionic 101.png



Field to hold image's filename

# Additional field to table trainings



The screenshot shows a terminal window titled 'Cmder' with a dark theme. The window title bar includes the application name 'Cmder' and a tab labeled '1. lara7'. The interface features a search bar and various icons for file operations. The terminal itself displays the following command and its output:

```
C:\laragon\www\lara7 (master)
λ php artisan make:migration alter_trainings_add_image_file
Created Migration: 2020_04_06_032935_alter_trainings_add_image_file

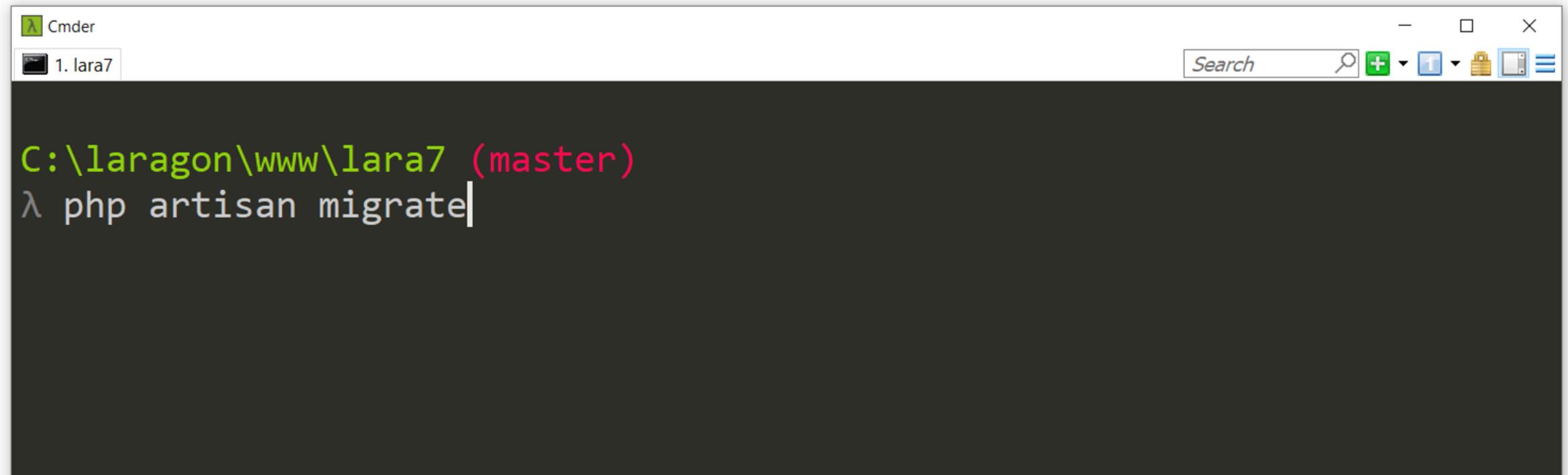
C:\laragon\www\lara7 (master)
λ |
```

# function up() – additional migration file

The screenshot shows a Sublime Text interface with two tabs open. The left tab is titled '2020\_04\_06\_024502\_add\_image\_trainings.php' and the right tab is titled '2020\_04\_06\_032935\_alter\_trainings\_add\_image\_file.php'. The file structure on the left includes 'lara7', 'app', 'bootstrap', 'config', 'database' (with 'factories' and 'migrations' subfolders), 'migrations' (containing several migration files like '2014\_10\_12\_000000\_create\_users\_table.php'), 'seeds', '.gitignore', 'node\_modules', 'public', 'resources', 'routes', 'storage', 'tests', 'vendor', '.editorconfig', and 'env'. A red circle highlights the 'migrations' folder. The code in the right tab is a PHP migration file:

```
15
14     public function up()
15     {
16         //add fields to hold image file
17         Schema::table('trainings',
18             function (Blueprint $table) {
19                 $table->string('filename')->nullable();
20                 $table->string('mime')->nullable();
21                 $table->string('original_fname')->nullable();
22
23                 //for next training table relationship
24                 $table->integer('idadmin')->nullable();
25             });
26     }
```

# php artisan migrate



A screenshot of a Cmder terminal window. The title bar says "Cmder" and the tab bar shows "1. lara7". The search bar contains "Search". The main area of the terminal shows the command line:

```
C:\laragon\www\lara7 (master)
λ php artisan migrate|
```

# create.blade.php – to select and upload image

Insert New Record

Title  
Ionic 101

Description  
Basic Ionic training to develop Hybrid apps

Trainer  
enKerul

Upload training poster (png, jpg, img file only with max 2MB filesize)

bannerFlutter.png

 **Upload image**

# Add enctype in create.blade.php

```
<!-- create.blade.php form start -->
<form action="{{ action('TrainingController@store') }}"
      method="POST"
      enctype="multipart/form-data">
    @csrf
    Title
    <input name="title" type="text"
          class="form-control">
    Description
    <input name="description" type="text"
          class="form-control">
```

# Add input type="file" in create.blade.php

```
▶ config
▶ database
▶ node_modules
▶ public
└ resources
  └ views
    └ trainings
      └ create.blade.php
      └ index.blade.php
      └ landingpage.blade.php
      └ welcome.blade.php
  └ routes
  └ storage
  └ tests
  └ vendor
    └ .editorconfig
    └ .env
    └ .gitattributes
    └ .gitignore
    └ .styleci.yml
    └ artisan
    └ composer.json
```

```
23
24 Trainer
25 <input name="trainer" type="text"
26 class="form-control">
27
28 <!-- create.blade.php add file input -->
29 Upload training poster
30 <input type="file"
31 name="attachment">
32 <button type="submit" class="btn
33 btn-primary">
34   Submit Record</button>
35 </form>
36
37 <!-- form ends -->
```

# app/Training.php - Add fillable for the fields

The screenshot shows a code editor interface with a sidebar labeled "FOLDERS" on the left. The "app" folder is expanded, showing subfolders like "Console", "Exceptions", "Http", "Controllers", "Middleware", "Providers", and "User". Inside the "Controllers" folder, "TrainingController.php" is visible. The main editor area displays the "Training.php" file. The code is as follows:

```
1 <?php
2
3 namespace App;
4
5 use Illuminate\Database\Eloquent\Model;
6
7 class Training extends Model
8 {
9     //Training Model at app/Training.php
10    //add the new fields
11    protected $fillable = [
12        'title', 'description', 'trainer',
13        'filename','idadmin'//add the new fields
14    ];
15
16    protected $table='trainings';
17 }
18
```

The line number 10 is highlighted with a yellow background.

# Validate uploaded file type

### Insert New Record

• Only jpeg, png, bmp image file are allowed. Do make sure file size is not larger than 2MB

Title

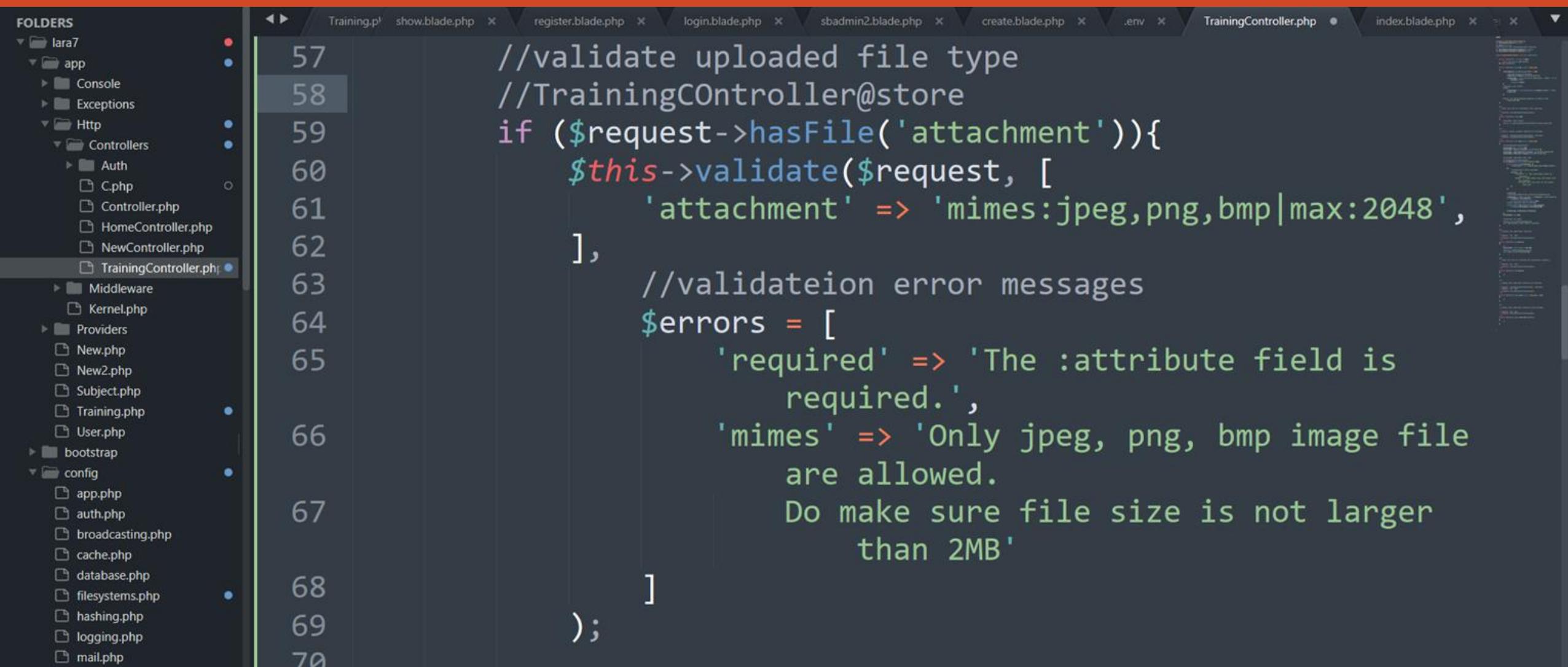
Description

Trainer

Upload training poster (png, jpg, img file only with max 2MB filesize)  
 Choose File No file chosen

**Save new Record**

# TrainingController@store – validate file type



The screenshot shows a code editor interface with a dark theme. On the left is a sidebar titled "FOLDERS" containing a tree view of a Laravel project structure. The "Http\Controllers" folder is expanded, showing files like Auth, C.php, Controller.php, HomeController.php, NewController.php, and TrainingController.php. The "TrainingController.php" file is selected and shown in the main editor area.

```
57     //validate uploaded file type
58     //TrainingCOnroller@store
59     if ($request->hasFile('attachment')){
60         $this->validate($request, [
61             'attachment' => 'mimes:jpeg,png,bmp|max:2048',
62         ],
63             //validateion error messages
64             $errors = [
65                 'required' => 'The :attribute field is
66                     required.',
67                 'mimes' => 'Only jpeg, png, bmp image file
68                     are allowed.
69                     Do make sure file size is not larger
70                     than 2MB'
71             ]
72         );
73     }
74 }
```

# TrainingController@store – continued...

The screenshot shows a code editor interface with a dark theme. On the left, there's a sidebar titled "FOLDERS" containing a file tree for a Laravel project named "lara7". The "Http\Controllers" folder is expanded, showing files like Auth, C.php, Controller.php, HomeController.php, NewController.php, and TrainingController.php (which is currently selected). Other visible files include Middleware, Kernel.php, Providers, New.php, New2.php, Subject.php, Training.php, User.php, bootstrap, config, and services.php.

The main editor area displays PHP code for a controller action. The code is as follows:

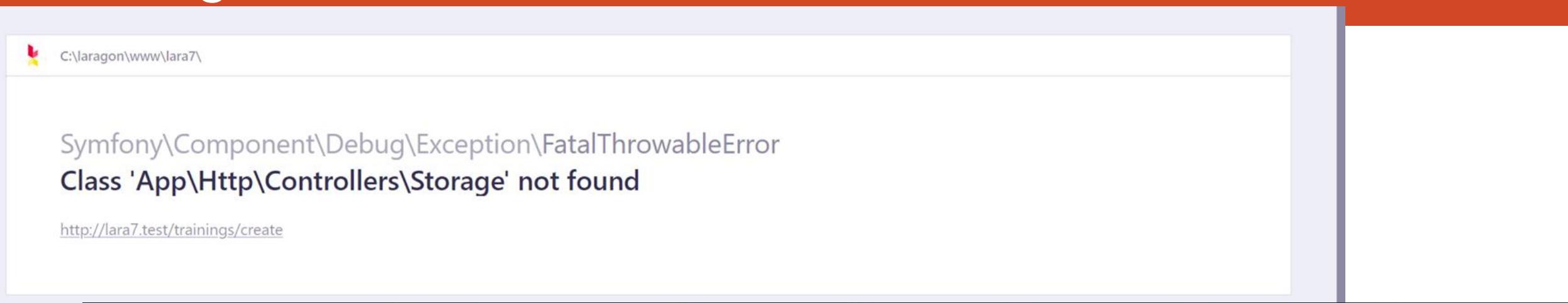
```
//example
//filename=2020-04-06-LaravelTraining101.png
$filename=date('Y-m-d').'-'.'$request->get('title').
'.'.$request->attachment->
    getClientOriginalExtension();
//save attaced file to storage
Storage::disk('public')->put($filename,
    File::get($request->attachment));

$training->filename=$filename;
}
$training->save();

//redirect to index
return redirect('/trainings')
    ->with('success','New record saved');
}//end TrainingController@store
```

The code uses Laravel's Eloquent ORM to save a new record to the database and store the uploaded file in the public directory. The file extension is determined from the original client file.

# Storage not found



```
1 <?php  
2  
3 namespace App\Http\Controllers;  
4 use Illuminate\Http\Request;  
5 use App\Training;  
6 //add this to use Storage::disk() function  
7 use Illuminate\Support\Facades\Storage;  
8 use Illuminate\Support\Facades\File;
```

# Configuring Filesystem for Storage

## config/filesystems.php



The screenshot shows a code editor with several tabs at the top: Kernel.php, filesystems.php (which is the active tab), Training.php, create.blade.php, TrainingController.php, User.php, and 2020\_04\_06\_032. The left sidebar displays the project's directory structure, including middleware, Providers, bootstrap, config (with files like app.php, auth.php, broadcasting.php, cache.php, database.php, and filesystems.php), database, node\_modules, public, resources (with js, lang, sass, and views), and auth. The filesystems.php file contains the following configuration:

```
//file location config/filesystems.php
44
45
46     'disks' => [
47
48         'local' => [
49             'driver' => 'local',
50             'root' => storage_path('app'),
51         ],
52
53         'public' => [
54             'driver' => 'local',
55             'root' => storage_path('app/public'),
56             'url' => env('APP_URL').'/storage',
57             'visibility' => 'public',
58         ],
59     ],
60
61     'media' => [
62         'driver' => 'local',
63         'root' => storage_path('app/storage'),
64         'url' => env('APP_URL').'/storage',
65         'visibility' => 'public',
66     ],
67
68     's3' => [
69         'driver' => 'aws',
70         'key' => env('AWS_ACCESS_KEY_ID'),
71         'secret' => env('AWS_SECRET_ACCESS_KEY'),
72         'region' => env('AWS_DEFAULT_REGION'),
73         'bucket' => env('AWS_BUCKET'),
74     ],
75
76     'sftp' => [
77         'driver' => 'sftp',
78         'host' => env('SFTP_HOST'),
79         'username' => env('SFTP_USERNAME'),
80         'password' => env('SFTP_PASSWORD'),
81         'root' => env('SFTP_ROOT'),
82     ],
83
84     'rackspace' => [
85         'driver' => 'rackspace',
86         'username' => env('RACKSPACE_USERNAME'),
87         'key' => env('RACKSPACE_API_KEY'),
88         'container' => env('RACKSPACE_CONTAINER'),
89     ],
90
91     'dropbox' => [
92         'driver' => 'dropbox',
93         'token' => env('DROPBOX_TOKEN'),
94     ],
95
96     'cloudinary' => [
97         'driver' => 'cloudinary',
98         'cloud_name' => env('CLOUDINARY_CLOUD_NAME'),
99         'api_key' => env('CLOUDINARY_API_KEY'),
100        'api_secret' => env('CLOUDINARY_API_SECRET'),
101    ],
102
103    'v3' => [
104        'driver' => 'v3',
105        'key' => env('V3_KEY'),
106        'secret' => env('V3_SECRET'),
107        'region' => env('V3_REGION'),
108        'bucket' => env('V3_BUCKET'),
109    ],
110
111    'gridFS' => [
112        'driver' => 'gridFS',
113        'connection' => env('MONGO_CONNECTION'),
114        'bucket' => env('MONGO_BUCKET'),
115    ],
116
117    'mogodb' => [
118        'driver' => 'mogodb',
119        'connection' => env('MONGO_CONNECTION'),
120        'bucket' => env('MONGO_BUCKET'),
121    ],
122
123    's3' => [
124        'driver' => 'aws',
125        'key' => env('AWS_ACCESS_KEY_ID'),
126        'secret' => env('AWS_SECRET_ACCESS_KEY'),
127        'region' => env('AWS_DEFAULT_REGION'),
128        'bucket' => env('AWS_BUCKET'),
129    ],
130
131    'sftp' => [
132        'driver' => 'sftp',
133        'host' => env('SFTP_HOST'),
134        'username' => env('SFTP_USERNAME'),
135        'password' => env('SFTP_PASSWORD'),
136        'root' => env('SFTP_ROOT'),
137    ],
138
139    'rackspace' => [
140        'driver' => 'rackspace',
141        'username' => env('RACKSPACE_USERNAME'),
142        'key' => env('RACKSPACE_API_KEY'),
143        'container' => env('RACKSPACE_CONTAINER'),
144    ],
145
146    'dropbox' => [
147        'driver' => 'dropbox',
148        'token' => env('DROPBOX_TOKEN'),
149    ],
150
151    'cloudinary' => [
152        'driver' => 'cloudinary',
153        'cloud_name' => env('CLOUDINARY_CLOUD_NAME'),
154        'api_key' => env('CLOUDINARY_API_KEY'),
155        'api_secret' => env('CLOUDINARY_API_SECRET'),
156    ],
157
158    'v3' => [
159        'driver' => 'v3',
160        'key' => env('V3_KEY'),
161        'secret' => env('V3_SECRET'),
162        'region' => env('V3_REGION'),
163        'bucket' => env('V3_BUCKET'),
164    ],
165
166    'gridFS' => [
167        'driver' => 'gridFS',
168        'connection' => env('MONGO_CONNECTION'),
169        'bucket' => env('MONGO_BUCKET'),
170    ],
171
172    'mogodb' => [
173        'driver' => 'mogodb',
174        'connection' => env('MONGO_CONNECTION'),
175        'bucket' => env('MONGO_BUCKET'),
176    ],
177
178    's3' => [
179        'driver' => 'aws',
180        'key' => env('AWS_ACCESS_KEY_ID'),
181        'secret' => env('AWS_SECRET_ACCESS_KEY'),
182        'region' => env('AWS_DEFAULT_REGION'),
183        'bucket' => env('AWS_BUCKET'),
184    ],
185
186    'sftp' => [
187        'driver' => 'sftp',
188        'host' => env('SFTP_HOST'),
189        'username' => env('SFTP_USERNAME'),
190        'password' => env('SFTP_PASSWORD'),
191        'root' => env('SFTP_ROOT'),
192    ],
193
194    'rackspace' => [
195        'driver' => 'rackspace',
196        'username' => env('RACKSPACE_USERNAME'),
197        'key' => env('RACKSPACE_API_KEY'),
198        'container' => env('RACKSPACE_CONTAINER'),
199    ],
200
201    'dropbox' => [
202        'driver' => 'dropbox',
203        'token' => env('DROPBOX_TOKEN'),
204    ],
205
206    'cloudinary' => [
207        'driver' => 'cloudinary',
208        'cloud_name' => env('CLOUDINARY_CLOUD_NAME'),
209        'api_key' => env('CLOUDINARY_API_KEY'),
210        'api_secret' => env('CLOUDINARY_API_SECRET'),
211    ],
212
213    'v3' => [
214        'driver' => 'v3',
215        'key' => env('V3_KEY'),
216        'secret' => env('V3_SECRET'),
217        'region' => env('V3_REGION'),
218        'bucket' => env('V3_BUCKET'),
219    ],
220
221    'gridFS' => [
222        'driver' => 'gridFS',
223        'connection' => env('MONGO_CONNECTION'),
224        'bucket' => env('MONGO_BUCKET'),
225    ],
226
227    'mogodb' => [
228        'driver' => 'mogodb',
229        'connection' => env('MONGO_CONNECTION'),
230        'bucket' => env('MONGO_BUCKET'),
231    ],
232
233    's3' => [
234        'driver' => 'aws',
235        'key' => env('AWS_ACCESS_KEY_ID'),
236        'secret' => env('AWS_SECRET_ACCESS_KEY'),
237        'region' => env('AWS_DEFAULT_REGION'),
238        'bucket' => env('AWS_BUCKET'),
239    ],
240
241    'sftp' => [
242        'driver' => 'sftp',
243        'host' => env('SFTP_HOST'),
244        'username' => env('SFTP_USERNAME'),
245        'password' => env('SFTP_PASSWORD'),
246        'root' => env('SFTP_ROOT'),
247    ],
248
249    'rackspace' => [
250        'driver' => 'rackspace',
251        'username' => env('RACKSPACE_USERNAME'),
252        'key' => env('RACKSPACE_API_KEY'),
253        'container' => env('RACKSPACE_CONTAINER'),
254    ],
255
256    'dropbox' => [
257        'driver' => 'dropbox',
258        'token' => env('DROPBOX_TOKEN'),
259    ],
260
261    'cloudinary' => [
262        'driver' => 'cloudinary',
263        'cloud_name' => env('CLOUDINARY_CLOUD_NAME'),
264        'api_key' => env('CLOUDINARY_API_KEY'),
265        'api_secret' => env('CLOUDINARY_API_SECRET'),
266    ],
267
268    'v3' => [
269        'driver' => 'v3',
270        'key' => env('V3_KEY'),
271        'secret' => env('V3_SECRET'),
272        'region' => env('V3_REGION'),
273        'bucket' => env('V3_BUCKET'),
274    ],
275
276    'gridFS' => [
277        'driver' => 'gridFS',
278        'connection' => env('MONGO_CONNECTION'),
279        'bucket' => env('MONGO_BUCKET'),
280    ],
281
282    'mogodb' => [
283        'driver' => 'mogodb',
284        'connection' => env('MONGO_CONNECTION'),
285        'bucket' => env('MONGO_BUCKET'),
286    ],
287
288    's3' => [
289        'driver' => 'aws',
290        'key' => env('AWS_ACCESS_KEY_ID'),
291        'secret' => env('AWS_SECRET_ACCESS_KEY'),
292        'region' => env('AWS_DEFAULT_REGION'),
293        'bucket' => env('AWS_BUCKET'),
294    ],
295
296    'sftp' => [
297        'driver' => 'sftp',
298        'host' => env('SFTP_HOST'),
299        'username' => env('SFTP_USERNAME'),
300        'password' => env('SFTP_PASSWORD'),
301        'root' => env('SFTP_ROOT'),
302    ],
303
304    'rackspace' => [
305        'driver' => 'rackspace',
306        'username' => env('RACKSPACE_USERNAME'),
307        'key' => env('RACKSPACE_API_KEY'),
308        'container' => env('RACKSPACE_CONTAINER'),
309    ],
310
311    'dropbox' => [
312        'driver' => 'dropbox',
313        'token' => env('DROPBOX_TOKEN'),
314    ],
315
316    'cloudinary' => [
317        'driver' => 'cloudinary',
318        'cloud_name' => env('CLOUDINARY_CLOUD_NAME'),
319        'api_key' => env('CLOUDINARY_API_KEY'),
320        'api_secret' => env('CLOUDINARY_API_SECRET'),
321    ],
322
323    'v3' => [
324        'driver' => 'v3',
325        'key' => env('V3_KEY'),
326        'secret' => env('V3_SECRET'),
327        'region' => env('V3_REGION'),
328        'bucket' => env('V3_BUCKET'),
329    ],
330
331    'gridFS' => [
332        'driver' => 'gridFS',
333        'connection' => env('MONGO_CONNECTION'),
334        'bucket' => env('MONGO_BUCKET'),
335    ],
336
337    'mogodb' => [
338        'driver' => 'mogodb',
339        'connection' => env('MONGO_CONNECTION'),
340        'bucket' => env('MONGO_BUCKET'),
341    ],
342
343    's3' => [
344        'driver' => 'aws',
345        'key' => env('AWS_ACCESS_KEY_ID'),
346        'secret' => env('AWS_SECRET_ACCESS_KEY'),
347        'region' => env('AWS_DEFAULT_REGION'),
348        'bucket' => env('AWS_BUCKET'),
349    ],
350
351    'sftp' => [
352        'driver' => 'sftp',
353        'host' => env('SFTP_HOST'),
354        'username' => env('SFTP_USERNAME'),
355        'password' => env('SFTP_PASSWORD'),
356        'root' => env('SFTP_ROOT'),
357    ],
358
359    'rackspace' => [
360        'driver' => 'rackspace',
361        'username' => env('RACKSPACE_USERNAME'),
362        'key' => env('RACKSPACE_API_KEY'),
363        'container' => env('RACKSPACE_CONTAINER'),
364    ],
365
366    'dropbox' => [
367        'driver' => 'dropbox',
368        'token' => env('DROPBOX_TOKEN'),
369    ],
370
371    'cloudinary' => [
372        'driver' => 'cloudinary',
373        'cloud_name' => env('CLOUDINARY_CLOUD_NAME'),
374        'api_key' => env('CLOUDINARY_API_KEY'),
375        'api_secret' => env('CLOUDINARY_API_SECRET'),
376    ],
377
378    'v3' => [
379        'driver' => 'v3',
380        'key' => env('V3_KEY'),
381        'secret' => env('V3_SECRET'),
382        'region' => env('V3_REGION'),
383        'bucket' => env('V3_BUCKET'),
384    ],
385
386    'gridFS' => [
387        'driver' => 'gridFS',
388        'connection' => env('MONGO_CONNECTION'),
389        'bucket' => env('MONGO_BUCKET'),
390    ],
391
392    'mogodb' => [
393        'driver' => 'mogodb',
394        'connection' => env('MONGO_CONNECTION'),
395        'bucket' => env('MONGO_BUCKET'),
396    ],
397
398    's3' => [
399        'driver' => 'aws',
400        'key' => env('AWS_ACCESS_KEY_ID'),
401        'secret' => env('AWS_SECRET_ACCESS_KEY'),
402        'region' => env('AWS_DEFAULT_REGION'),
403        'bucket' => env('AWS_BUCKET'),
404    ],
405
406    'sftp' => [
407        'driver' => 'sftp',
408        'host' => env('SFTP_HOST'),
409        'username' => env('SFTP_USERNAME'),
410        'password' => env('SFTP_PASSWORD'),
411        'root' => env('SFTP_ROOT'),
412    ],
413
414    'rackspace' => [
415        'driver' => 'rackspace',
416        'username' => env('RACKSPACE_USERNAME'),
417        'key' => env('RACKSPACE_API_KEY'),
418        'container' => env('RACKSPACE_CONTAINER'),
419    ],
420
421    'dropbox' => [
422        'driver' => 'dropbox',
423        'token' => env('DROPBOX_TOKEN'),
424    ],
425
426    'cloudinary' => [
427        'driver' => 'cloudinary',
428        'cloud_name' => env('CLOUDINARY_CLOUD_NAME'),
429        'api_key' => env('CLOUDINARY_API_KEY'),
430        'api_secret' => env('CLOUDINARY_API_SECRET'),
431    ],
432
433    'v3' => [
434        'driver' => 'v3',
435        'key' => env('V3_KEY'),
436        'secret' => env('V3_SECRET'),
437        'region' => env('V3_REGION'),
438        'bucket' => env('V3_BUCKET'),
439    ],
440
441    'gridFS' => [
442        'driver' => 'gridFS',
443        'connection' => env('MONGO_CONNECTION'),
444        'bucket' => env('MONGO_BUCKET'),
445    ],
446
447    'mogodb' => [
448        'driver' => 'mogodb',
449        'connection' => env('MONGO_CONNECTION'),
450        'bucket' => env('MONGO_BUCKET'),
451    ],
452
453    's3' => [
454        'driver' => 'aws',
455        'key' => env('AWS_ACCESS_KEY_ID'),
456        'secret' => env('AWS_SECRET_ACCESS_KEY'),
457        'region' => env('AWS_DEFAULT_REGION'),
458        'bucket' => env('AWS_BUCKET'),
459    ],
460
461    'sftp' => [
462        'driver' => 'sftp',
463        'host' => env('SFTP_HOST'),
464        'username' => env('SFTP_USERNAME'),
465        'password' => env('SFTP_PASSWORD'),
466        'root' => env('SFTP_ROOT'),
467    ],
468
469    'rackspace' => [
470        'driver' => 'rackspace',
471        'username' => env('RACKSPACE_USERNAME'),
472        'key' => env('RACKSPACE_API_KEY'),
473        'container' => env('RACKSPACE_CONTAINER'),
474    ],
475
476    'dropbox' => [
477        'driver' => 'dropbox',
478        'token' => env('DROPBOX_TOKEN'),
479    ],
480
481    'cloudinary' => [
482        'driver' => 'cloudinary',
483        'cloud_name' => env('CLOUDINARY_CLOUD_NAME'),
484        'api_key' => env('CLOUDINARY_API_KEY'),
485        'api_secret' => env('CLOUDINARY_API_SECRET'),
486    ],
487
488    'v3' => [
489        'driver' => 'v3',
490        'key' => env('V3_KEY'),
491        'secret' => env('V3_SECRET'),
492        'region' => env('V3_REGION'),
493        'bucket' => env('V3_BUCKET'),
494    ],
495
496    'gridFS' => [
497        'driver' => 'gridFS',
498        'connection' => env('MONGO_CONNECTION'),
499        'bucket' => env('MONGO_BUCKET'),
500    ],
501
502    'mogodb' => [
503        'driver' => 'mogodb',
504        'connection' => env('MONGO_CONNECTION'),
505        'bucket' => env('MONGO_BUCKET'),
506    ],
507
508    's3' => [
509        'driver' => 'aws',
510        'key' => env('AWS_ACCESS_KEY_ID'),
511        'secret' => env('AWS_SECRET_ACCESS_KEY'),
512        'region' => env('AWS_DEFAULT_REGION'),
513        'bucket' => env('AWS_BUCKET'),
514    ],
515
516    'sftp' => [
517        'driver' => 'sftp',
518        'host' => env('SFTP_HOST'),
519        'username' => env('SFTP_USERNAME'),
520        'password' => env('SFTP_PASSWORD'),
521        'root' => env('SFTP_ROOT'),
522    ],
523
524    'rackspace' => [
525        'driver' => 'rackspace',
526        'username' => env('RACKSPACE_USERNAME'),
527        'key' => env('RACKSPACE_API_KEY'),
528        'container' => env('RACKSPACE_CONTAINER'),
529    ],
530
531    'dropbox' => [
532        'driver' => 'dropbox',
533        'token' => env('DROPBOX_TOKEN'),
534    ],
535
536    'cloudinary' => [
537        'driver' => 'cloudinary',
538        'cloud_name' => env('CLOUDINARY_CLOUD_NAME'),
539        'api_key' => env('CLOUDINARY_API_KEY'),
540        'api_secret' => env('CLOUDINARY_API_SECRET'),
541    ],
542
543    'v3' => [
544        'driver' => 'v3',
545        'key' => env('V3_KEY'),
546        'secret' => env('V3_SECRET'),
547        'region' => env('V3_REGION'),
548        'bucket' => env('V3_BUCKET'),
549    ],
550
551    'gridFS' => [
552        'driver' => 'gridFS',
553        'connection' => env('MONGO_CONNECTION'),
554        'bucket' => env('MONGO_BUCKET'),
555    ],
556
557    'mogodb' => [
558        'driver' => 'mogodb',
559        'connection' => env('MONGO_CONNECTION'),
560        'bucket' => env('MONGO_BUCKET'),
561    ],
562
563    's3' => [
564        'driver' => 'aws',
565        'key' => env('AWS_ACCESS_KEY_ID'),
566        'secret' => env('AWS_SECRET_ACCESS_KEY'),
567        'region' => env('AWS_DEFAULT_REGION'),
568        'bucket' => env('AWS_BUCKET'),
569    ],
570
571    'sftp' => [
572        'driver' => 'sftp',
573        'host' => env('SFTP_HOST'),
574        'username' => env('SFTP_USERNAME'),
575        'password' => env('SFTP_PASSWORD'),
576        'root' => env('SFTP_ROOT'),
577    ],
578
579    'rackspace' => [
580        'driver' => 'rackspace',
581        'username' => env('RACKSPACE_USERNAME'),
582        'key' => env('RACKSPACE_API_KEY'),
583        'container' => env('RACKSPACE_CONTAINER'),
584    ],
585
586    'dropbox' => [
587        'driver' => 'dropbox',
588        'token' => env('DROPBOX_TOKEN'),
589    ],
590
591    'cloudinary' => [
592        'driver' => 'cloudinary',
593        'cloud_name' => env('CLOUDINARY_CLOUD_NAME'),
594        'api_key' => env('CLOUDINARY_API_KEY'),
595        'api_secret' => env('CLOUDINARY_API_SECRET'),
596    ],
597
598    'v3' => [
599        'driver' => 'v3',
600        'key' => env('V3_KEY'),
601        'secret' => env('V3_SECRET'),
602        'region' => env('V3_REGION'),
603        'bucket' => env('V3_BUCKET'),
604    ],
605
606    'gridFS' => [
607        'driver' => 'gridFS',
608        'connection' => env('MONGO_CONNECTION'),
609        'bucket' => env('MONGO_BUCKET'),
610    ],
611
612    'mogodb' => [
613        'driver' => 'mogodb',
614        'connection' => env('MONGO_CONNECTION'),
615        'bucket' => env('MONGO_BUCKET'),
616    ],
617
618    's3' => [
619        'driver' => 'aws',
620        'key' => env('AWS_ACCESS_KEY_ID'),
621        'secret' => env('AWS_SECRET_ACCESS_KEY'),
622        'region' => env('AWS_DEFAULT_REGION'),
623        'bucket' => env('AWS_BUCKET'),
624    ],
625
626    'sftp' => [
627        'driver' => 'sftp',
628        'host' => env('SFTP_HOST'),
629        'username' => env('SFTP_USERNAME'),
630        'password' => env('SFTP_PASSWORD'),
631        'root' => env('SFTP_ROOT'),
632    ],
633
634    'rackspace' => [
635        'driver' => 'rackspace',
636        'username' => env('RACKSPACE_USERNAME'),
637        'key' => env('RACKSPACE_API_KEY'),
638        'container' => env('RACKSPACE_CONTAINER'),
639    ],
640
641    'dropbox' => [
642        'driver' => 'dropbox',
643        'token' => env('DROPBOX_TOKEN'),
644    ],
645
646    'cloudinary' => [
647        'driver' => 'cloudinary',
648        'cloud_name' => env('CLOUDINARY_CLOUD_NAME'),
649        'api_key' => env('CLOUDINARY_API_KEY'),
650        'api_secret' => env('CLOUDINARY_API_SECRET'),
651    ],
652
653    'v3' => [
654        'driver' => 'v3',
655        'key' => env('V3_KEY'),
656        'secret' => env('V3_SECRET'),
657        'region' => env('V3_REGION'),
658        'bucket' => env('V3_BUCKET'),
659    ],
660
661    'gridFS' => [
662        'driver' => 'gridFS',
663        'connection' => env('MONGO_CONNECTION'),
664        'bucket' => env('MONGO_BUCKET'),
665    ],
666
667    'mogodb' => [
668        'driver' => 'mogodb',
669        'connection' => env('MONGO_CONNECTION'),
670        'bucket' => env('MONGO_BUCKET'),
671    ],
672
673    's3' => [
674        'driver' => 'aws',
675        'key' => env('AWS_ACCESS_KEY_ID'),
676        'secret' => env('AWS_SECRET_ACCESS_KEY'),
677        'region' => env('AWS_DEFAULT_REGION'),
678        'bucket' => env('AWS_BUCKET'),
679    ],
680
681    'sftp' => [
682        'driver' => 'sftp',
683        'host' => env('SFTP_HOST'),
684        'username' => env('SFTP_USERNAME'),
685        'password' => env('SFTP_PASSWORD'),
686        'root' => env('SFTP_ROOT'),
687    ],
688
689    'rackspace' => [
690        'driver' => 'rackspace',
691        'username' => env('RACKSPACE_USERNAME'),
692        'key' => env('RACKSPACE_API_KEY'),
693        'container' => env('RACKSPACE_CONTAINER'),
694    ],
695
696    'dropbox' => [
697        'driver' => 'dropbox',
698        'token' => env('DROPBOX_TOKEN'),
699    ],
699
700    'cloudinary' => [
701        'driver' => 'cloudinary',
702        'cloud_name' => env('CLOUDINARY_CLOUD_NAME'),
703        'api_key' => env('CLOUDINARY_API_KEY'),
704        'api_secret' => env('CLOUDINARY_API_SECRET'),
705    ],
706
707    'v3' => [
708        'driver' => 'v3',
709        'key' => env('V3_KEY'),
710        'secret' => env('V3_SECRET'),
711        'region' => env('V3_REGION'),
712        'bucket' => env('V3_BUCKET'),
713    ],
714
715    'gridFS' => [
716        'driver' => 'gridFS',
717        'connection' => env('MONGO_CONNECTION'),
718        'bucket' => env('MONGO_BUCKET'),
719    ],
720
721    'mogodb' => [
722        'driver' => 'mogodb',
723        'connection' => env('MONGO_CONNECTION'),
724        'bucket' => env('MONGO_BUCKET'),
725    ],
726
727    's3' => [
728        'driver' => 'aws',
729        'key' => env('AWS_ACCESS_KEY_ID'),
730        'secret' => env('AWS_SECRET_ACCESS_KEY'),
731        'region' => env('AWS_DEFAULT_REGION'),
732        'bucket' => env('AWS_BUCKET'),
733    ],
734
735    'sftp' => [
736        'driver' => 'sftp',
737        'host' => env('SFTP_HOST'),
738        'username' => env('SFTP_USERNAME'),
739        'password' => env('SFTP_PASSWORD'),
740        'root' => env('SFTP_ROOT'),
741    ],
742
743    'rackspace' => [
744        'driver' => 'rackspace',
745        'username' => env('RACKSPACE_USERNAME'),
746        'key' => env('RACKSPACE_API_KEY'),
747        'container' => env('RACKSPACE_CONTAINER'),
748    ],
749
750    'dropbox' => [
751        'driver' => 'dropbox',
752        'token' => env('DROPBOX_TOKEN'),
753    ],
754
755    'cloudinary' => [
756        'driver' => 'cloudinary',
757        'cloud_name' => env('CLOUDINARY_CLOUD_NAME'),
758        'api_key' => env('CLOUDINARY_API_KEY'),
759        'api_secret' => env('CLOUDINARY_API_SECRET'),
760    ],
761
762    'v3' => [
763        'driver' => 'v3',
764        'key' => env('V3_KEY'),
765        'secret' => env('V3_SECRET'),
766        'region' => env('V3_REGION'),
767        'bucket' => env('V3_BUCKET'),
768    ],
769
770    'gridFS' => [
771        'driver' => 'gridFS',
772        'connection' => env('MONGO_CONNECTION'),
773        'bucket' => env('MONGO_BUCKET'),
774    ],
775
776    'mogodb' => [
777        'driver' => 'mogodb',
778        'connection' => env('MONGO_CONNECTION'),
779        'bucket' => env('MONGO_BUCKET'),
780    ],
781
782    's3' => [
783        'driver' => 'aws',
784        'key' => env('AWS_ACCESS_KEY_ID'),
785        'secret' => env('AWS_SECRET_ACCESS_KEY'),
786        'region' => env('AWS_DEFAULT_REGION'),
787        'bucket' => env('AWS_BUCKET'),
788    ],
789
790    'sftp' => [
791        'driver' => 'sftp',
792        'host' => env('SFTP_HOST'),
793        'username' => env('SFTP_USERNAME'),
794        'password' => env('SFTP_PASSWORD'),
795        'root' => env('SFTP_ROOT'),
796    ],
797
798    'rackspace' => [
799        'driver' => 'rackspace',
800        'username' => env('RACKSPACE_USERNAME'),
801        'key' => env('RACKSPACE_API_KEY'),
802        'container' => env('RACKSPACE_CONTAINER'),
803    ],
804
805    'dropbox' => [
806        'driver' => 'dropbox',
807        'token' => env('DROPBOX_TOKEN'),
808    ],
809
810    'cloudinary' => [
811        'driver' => 'cloudinary',
812        'cloud_name' => env('CLOUDINARY_CLOUD_NAME'),
813        'api_key' => env('CLOUDINARY_API_KEY'),
814        'api_secret' => env('CLOUDINARY_API_SECRET'),
815    ],
816
817    'v3' => [
818        'driver' => 'v3',
819        'key' => env('V3_KEY'),
820        'secret' => env('V3_SECRET'),
821        'region' => env('V3_REGION'),
822        'bucket' => env('V3_BUCKET'),
823    ],
824
825    'gridFS' => [
826        'driver' => 'gridFS',
827        'connection' => env('MONGO_CONNECTION'),
828        'bucket' => env('MONGO_BUCKET'),
829    ],
830
831    'mogodb' => [
832        'driver' => 'mogodb',
833        'connection' => env('MONGO_CONNECTION'),
834        'bucket' => env('MONGO_BUCKET'),
835    ],
836
837    's3' => [
838        'driver' => 'aws',
839        'key' => env('AWS_ACCESS_KEY_ID'),
840        'secret' => env('AWS_SECRET_ACCESS_KEY'),
841        'region' => env('AWS_DEFAULT_REGION'),
842        'bucket' => env('AWS_BUCKET'),
843    ],
844
845    'sftp' => [
846        'driver' => 'sftp',
847        'host' => env('SFTP_HOST'),
848        'username' => env('SFTP_USERNAME'),
849        'password' => env('SFTP_PASSWORD'),
850        'root' => env('SFTP_ROOT'),
851    ],
852
853    'rackspace' => [
854        'driver' => 'rackspace',
855        'username' => env('RACKSPACE_USERNAME'),
856        'key' => env('RACKSPACE_API_KEY'),
857        'container' => env('RACKSPACE_CONTAINER'),
858    ],
859
860    'dropbox' => [
861        'driver' => 'dropbox',
862        'token' => env('DROPBOX_TOKEN'),
863    ],
864
865    'cloudinary' => [
866        'driver' => 'cloudinary',
867        'cloud_name' => env('CLOUDINARY_CLOUD_NAME'),
868        'api_key' => env('CLOUDINARY_API_KEY'),
869        'api_secret' => env('CLOUDINARY_API_SECRET'),
870    ],
871
872    'v3' => [
873        'driver' => 'v3',
874        'key' => env('V3_KEY'),
875        'secret' => env('V3_SECRET'),
876        'region' => env('V3_REGION'),
877        'bucket' => env('V3_BUCKET'),
878    ],
879
880    'gridFS' => [
881        'driver' => 'gridFS',
882        'connection' => env('MONGO_CONNECTION'),
883        'bucket' => env('MONGO_BUCKET'),
884    ],
885
886    'mogodb' => [
887        'driver' => 'mogodb',
888        'connection' => env('MONGO_CONNECTION'),
889        'bucket' => env('MONGO_BUCKET'),
890    ],
891
892    's3' => [
893        'driver' => 'aws',
894        'key' => env('AWS_ACCESS_KEY_ID'),
895        'secret' => env('AWS_SECRET_ACCESS_KEY'),
896        'region' => env('AWS_DEFAULT_REGION'),
897        'bucket' => env('AWS_BUCKET'),
898    ],
899
900    'sftp' => [
901        'driver' => 'sftp',
902        'host' => env('SFTP_HOST'),
903        'username' => env('SFTP_USERNAME'),
904        'password' => env('SFTP_PASSWORD'),
905        'root' => env('SFTP_ROOT'),
906    ],
907
908    'rackspace' => [
909        'driver' => 'rackspace',
910        'username' => env('RACKSPACE_USERNAME'),
911        'key' => env('RACKSPACE_API_KEY'),
912        'container' => env('RACKSPACE_CONTAINER'),
913    ],
914
915    'dropbox' => [
916        'driver' => 'dropbox',
917        'token' => env('DROPBOX_TOKEN'),
918    ],
919
920    'cloudinary' => [
921        'driver' => 'cloudinary',
922        'cloud_name' => env('CLOUDINARY_CLOUD_NAME'),
923        'api_key' => env('CLOUDINARY_API_KEY'),
924        'api_secret' => env('CLOUDINARY_API_SECRET'),
925    ],
926
927    'v3' => [
928        'driver' => 'v3',
929        'key' => env('V3_KEY'),
930        'secret' => env('V3_SECRET'),
931        'region' => env('V3_REGION'),
932        'bucket' => env('V3_BUCKET'),
933    ],
934
935    'gridFS' => [
936        'driver' => 'gridFS',
937        'connection' => env('MONGO_CONNECTION'),
938        'bucket' => env('MONGO_BUCKET'),
939    ],
940
941    'mogodb' => [
942        'driver' => 'mogodb',
943        'connection' => env('MONGO_CONNECTION'),
944        'bucket' => env('MONGO_BUCKET'),
945    ],
946

```

# File Storage

## Create Symbolic Link

The public disk is intended for files that are going to be publicly accessible.

By default, the public disk uses the local driver and stores these files in `storage/app/public`.

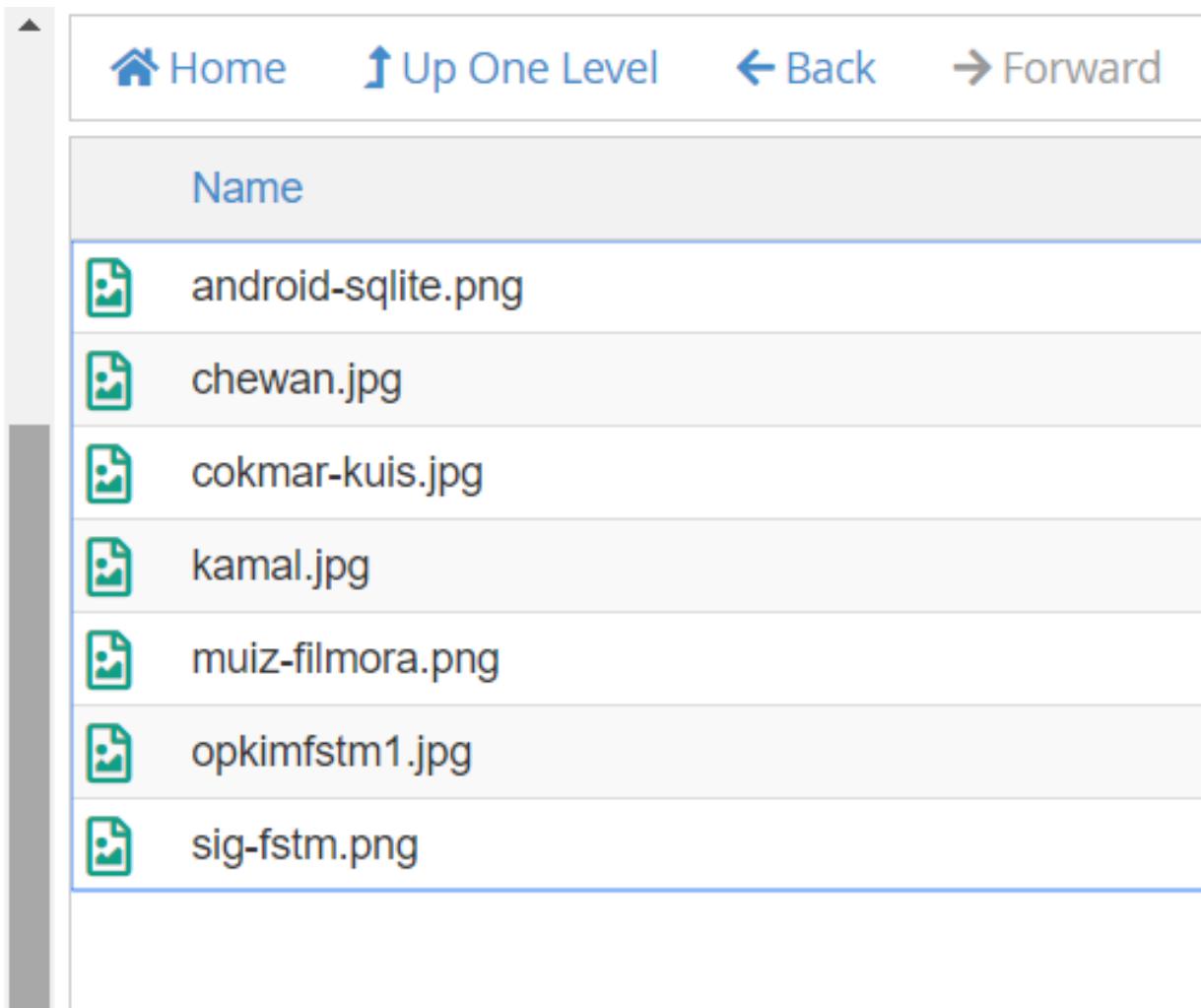
To make them accessible from the web, you should create a symbolic link from `public/storage` to `storage/app/public`.

To create the symbolic link, you may use the `storage:link` Artisan command:

```
php artisan storage:link
```

# Virtual link of public file storage

```
logs  
+ mail  
+ public_ftp  
- public_html  
  - larafstm  
    + css  
    fonts  
    + images  
    + img  
    + js  
    + sadmin  
    + storage  
    + vendors  
+ ionicpwa  
+ js
```



# Physical link of public file storage

The screenshot shows a file explorer interface. On the left, the directory structure of a Laravel application is displayed:

- (/home/---) (highlighted with a red box)
  - cache
  - + etc
  - larafstm
    - + app
    - + bootstrap
      - config
    - + database
    - + public
    - + resources
      - routes
    - storage
      - app
        - public
          - uploads
    - + framework
    - + logs
    - + tests
    - logs
  - + mail
  - + public\_ftp
  - public\_html
  - larafstm
    - + css
    - fonts
    - + images
    - + img

	android-sqlite.png
	chewan.jpg
	cokmar-kuis.jpg
	kamal.jpg
	kursus-it-opkim.fstm.png
	muiz-filmora.png
	opkimfstm1.jpg
	sig-fstm.png

## The actual physical location of the file storage

To make them accessible from the web, you should create a symbolic link from **public/storage** to **storage/app/public**.

# create.blade.php - Display validation errors

```
create.blade.php x TrainingController.php x filesystems.php x Training.php x show.blade.php x register.blade.php x login.blade.php x sadmin2.blade.php x .env
7 <div class="card">
8     <div class="card-header"><h3>Insert New Record</h3></div>
9
10    <div class="card-body">
11        <!-- show validation error messages -->
12        @if ($errors->any())
13            <div class="alert alert-danger">
14                <ul>
15                    @foreach ($errors->all() as $error)
16                        <li>{{ $error }}</li>
17                    @endforeach
18                </ul>
19            </div>
20        @endif
21
```

# Github.com/khirulnizam/lara7

Slides &  
codes

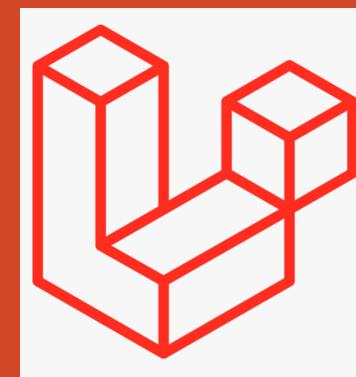
T12 Tutorial  
clone/download  
github repo

The screenshot shows a GitHub repository page for the user 'khirulnizam' with the repository name 'lara7'. The page displays the following statistics: 1 commit, 1 branch, 0 packages, 0 releases, and 1 contributor. The 'Branch: master' dropdown is selected. A 'New pull request' button is visible. On the right, there are 'Find file' and 'Clone or download' buttons, with 'Clone or download' being green. A 'Clone with HTTPS' section contains the URL <https://github.com/khirulnizam/lara7.git>. Below this, there are two large blue buttons: 'Open in Desktop' and 'Download ZIP'. The commit history lists seven commits, each with a timestamp of '16 minutes ago':

File/Folder	Commit Message	Timestamp
app	Initial commit	16 minutes ago
bootstrap	Initial commit	16 minutes ago
config	Initial commit	16 minutes ago
database	Initial commit	16 minutes ago
public	Initial commit	16 minutes ago
resources	Initial commit	16 minutes ago
routes	Initial commit	16 minutes ago

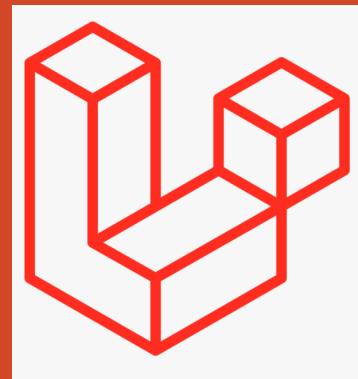
# Laravel t21

## Table relationship (Jadual dengan perhubungan 1-Many)



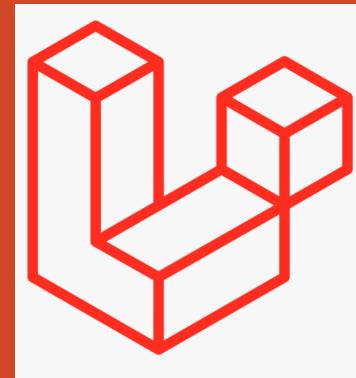
# Laravel t22

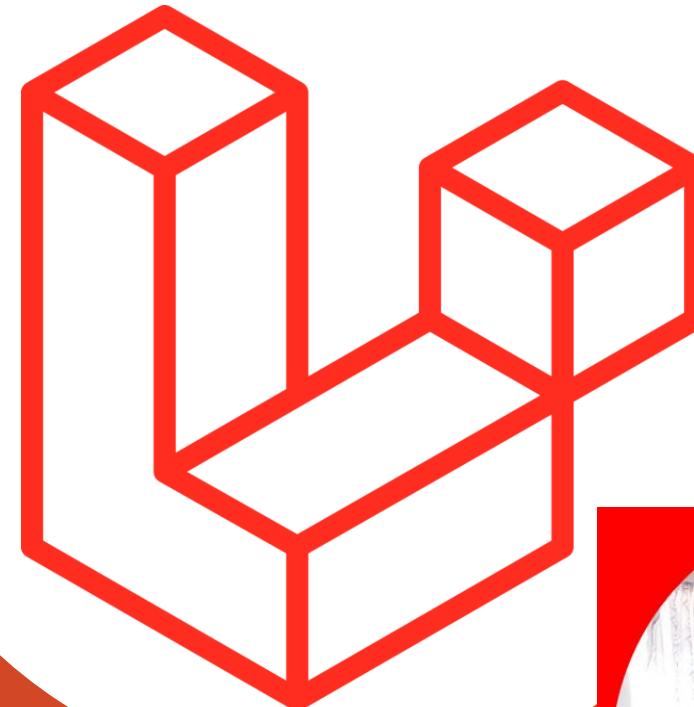
Insert record with drop  
down data option  
**(Masukkan data baharu  
dengan jadual berhubungan)**



# Laravel t23

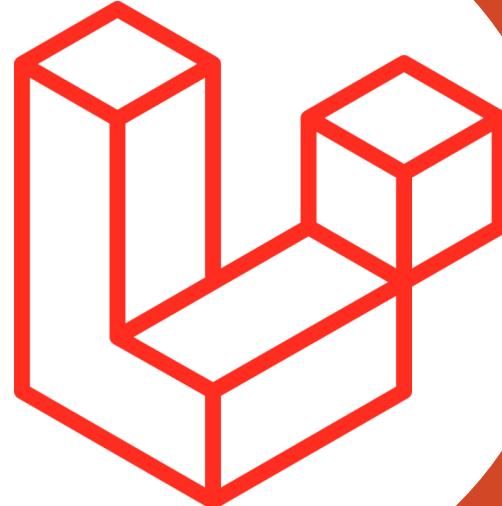
## Upload record with file (Muat naik sebarang fail)



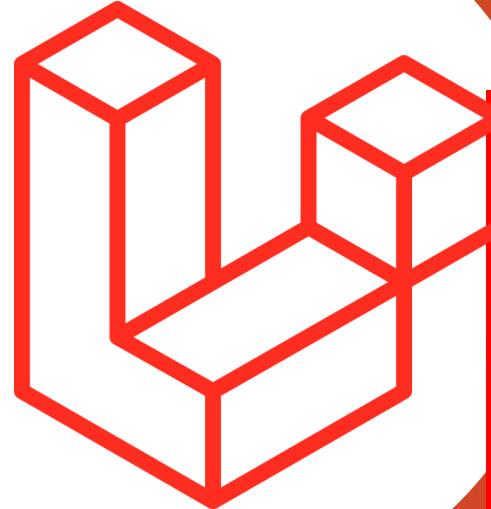


# Laravel



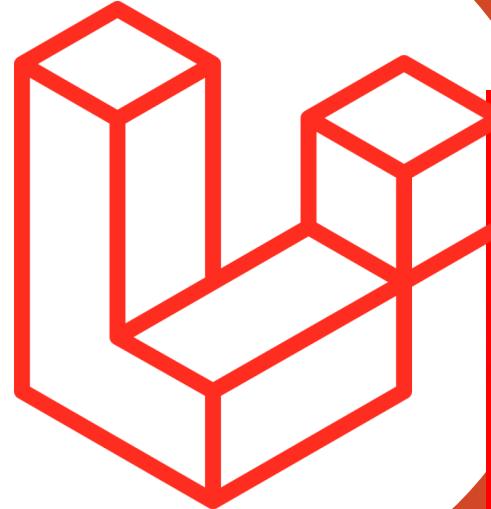


**TMMT6193 Advanced Web  
Week2 MVC, Laravel new  
project, views**



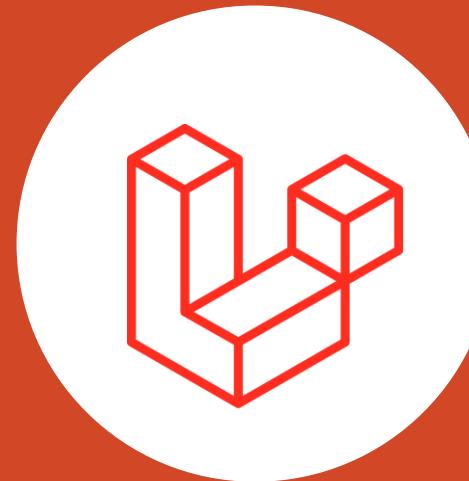
TMMT6193 Advanced Web

# Week3 CSS Framework using Bootstrap



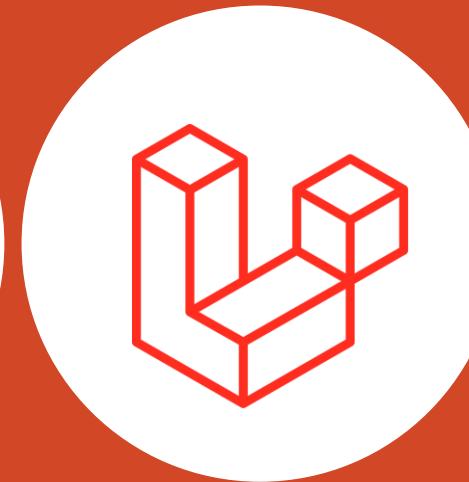
TMMT6193 Advanced Web Development

Week4 View,  
Controller, Routing



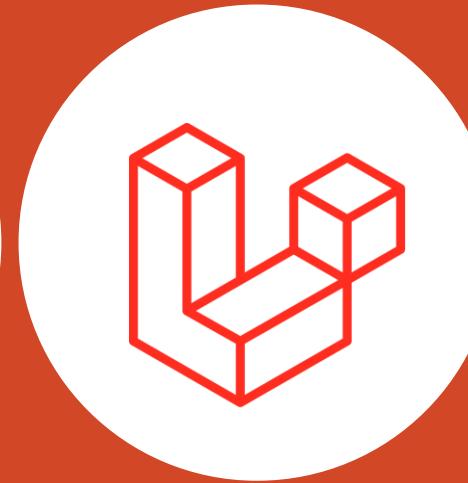
TMMT6193 Advanced Web Development

# Week6 Model & Insert Record to DB



TMMT6193 Advanced Web Development

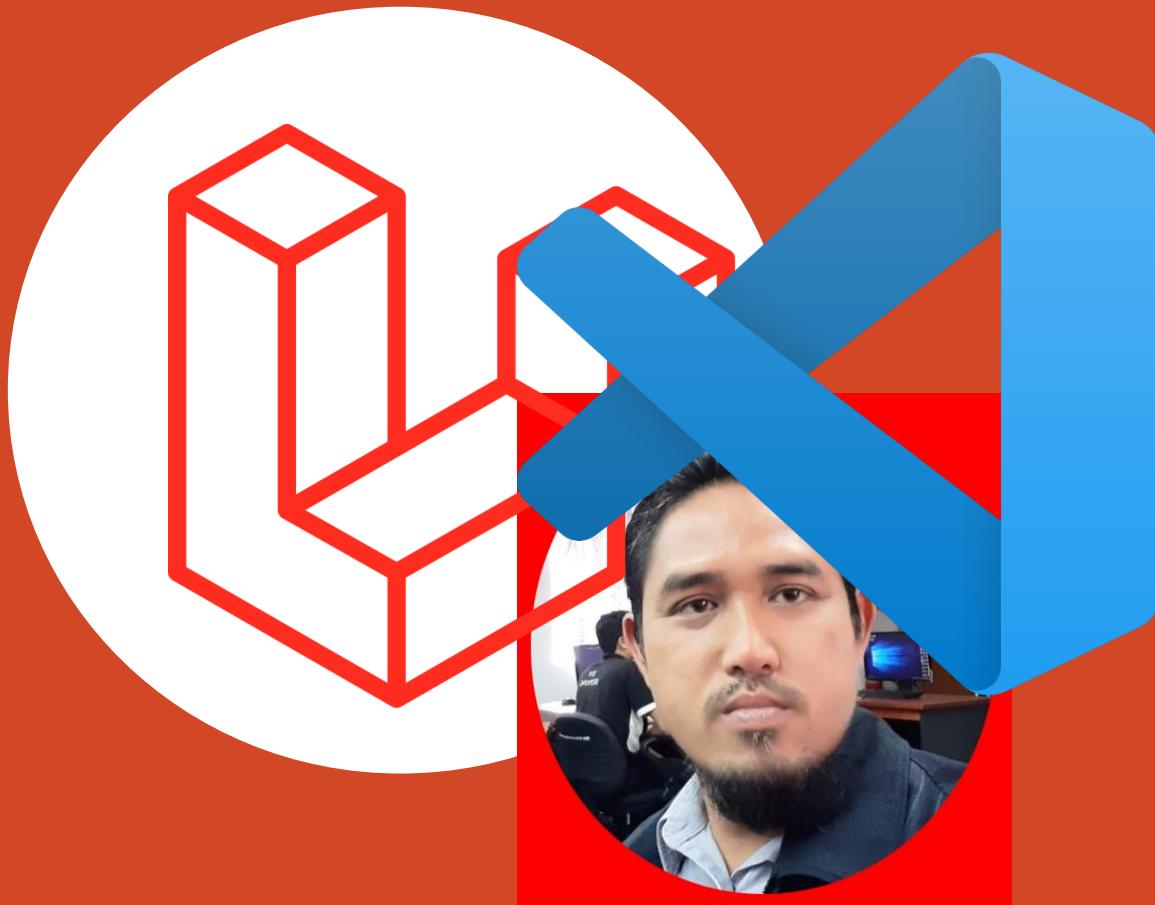
# Week7 Browse, Search Records



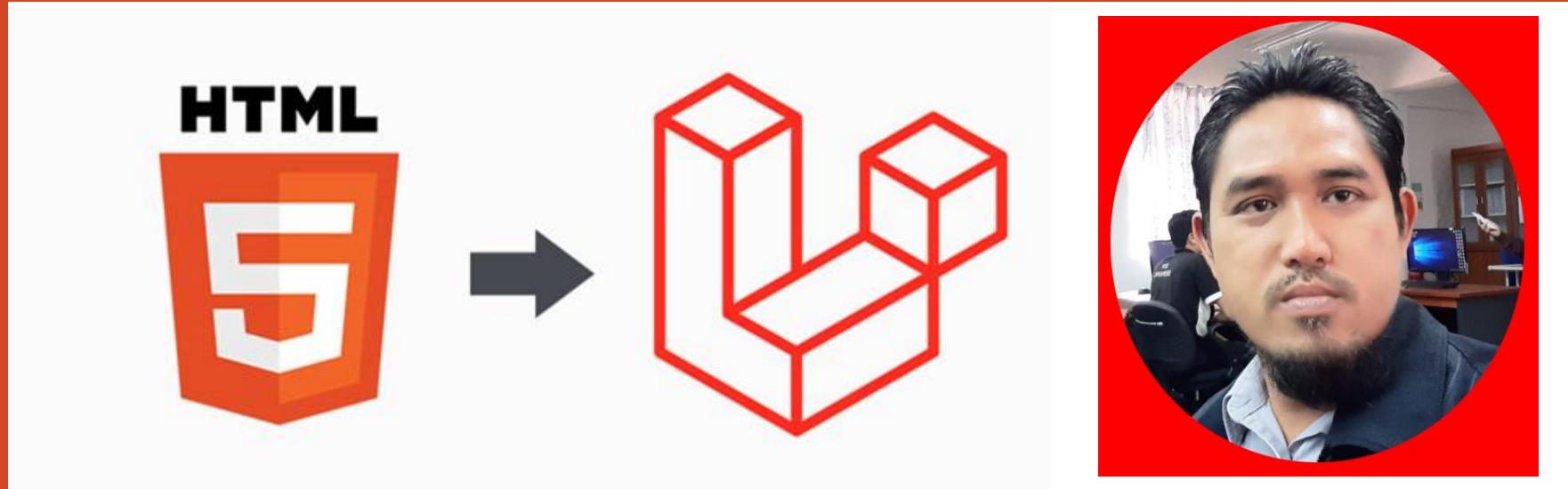
TMMT6193 Advanced Web Development

Week8

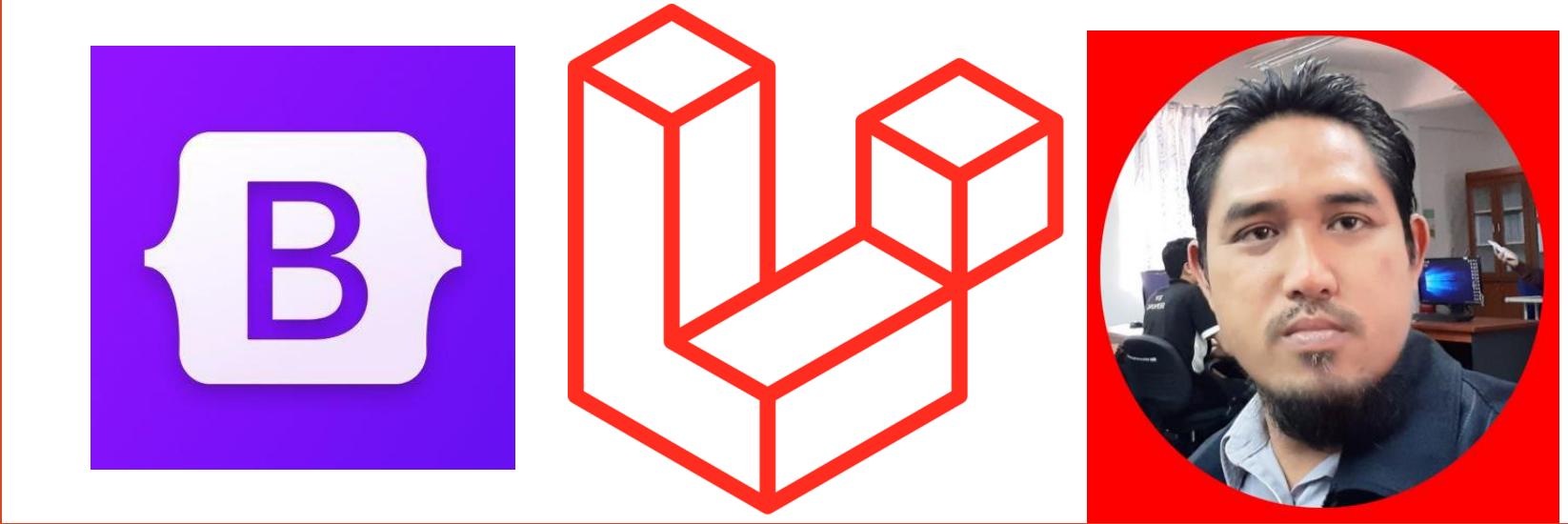
Update Records



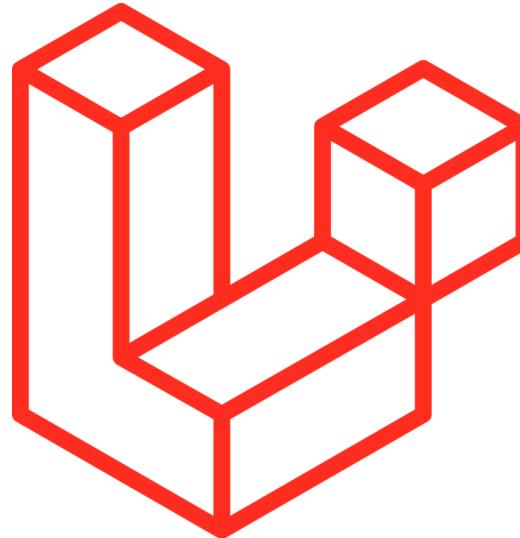
# vSCODE for Laravel



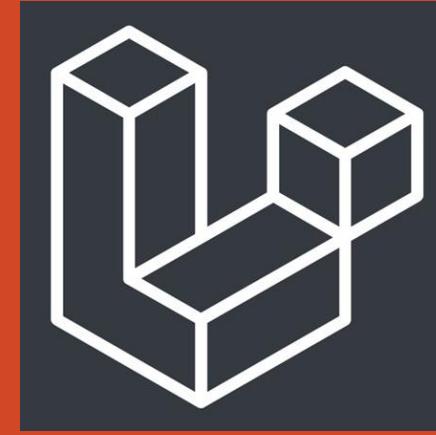
# Antaramuka HTML dalam Laravel View



# Bootstrap Template dalam Laravel View



Kemaskini rekod  
Update record



# W09 DELETE RECORD



# Login using Laravel/Breeze