PHP হচ্ছে একটি সার্ভার সাইড স্ক্রিপ্টিং ভাষা, যা তৈরী করেন রাসমাস লারডোর্ফ ১৯৯৪ সালে। যা বিশেষ ভাবে যেকোনো ডাইনামিক এবং ইন্টারেক্টিভ ওয়েব ডেভেলপমেন্ট অথবা ওয়েব ভিত্তিক সফটওয়্যার এবং ওয়েব ভিত্তিক এপ্লিকেশন বানানোর জন্য ব্যবহৃত হয়। বর্তমানে এটিকে জেনারেল পারপাস প্রোগ্রামিং ল্যাঙ্গুয়েজ ও বলা হয়। অর্থাৎ, অন্যান্য প্রোগ্রামিং ভাষা দিয়ে যা যা করা যায়, PHP দিয়েও মোটামুটি তার সবই করা যায়। এছাড়া PHP ওপেনসোর্স হওয়ায় ওয়েব প্রোগ্রামারদের কাছে অধিক জনপ্রিয় । তবে মূলত, ওয়েব ডেভেলপমেন্টের, ওয়েব ভিত্তিক সফটওয়্যার এবং ওয়েব ভিত্তিক এপ্লিকেশন বানানোর কাজেই এটি সর্বাধিক ব্যবহৃত হয়। এছাড়া পি এইচপি ওয়েবে তথ্য সংরক্ষণের ক্ষেত্রে সর্বোচ্চ নিরাপত্তা প্রদান করে। পি এইচ পি কে এইচ টি এম এল (HTML) এর সাথেও ব্যবহার করা যায়। ওয়ার্ডপ্রেস, জুমলা সহ বেশ কিছু এডভান্সড ওপেনসোর্স কনটেন্ট ম্যানেজমেন্ট সিস্টেম (CMS) পি এইচ পি দ্বারা তৈরি। পি এইচ পি ব্যবহার করে কাস্টম কনটেন্ট ম্যানেজমেন্ট সিস্টেম তৈরি করা যায়। এছাড়া ই-কমার্স, ম্যানেজমেন্ট, অনলাইন ব্যাংকিং সফটওয়্যার ইত্যাদি তৈরিতেও পি এইচ পি ব্যাপকভাবে ব্যবহৃত হচ্ছে। পি এইচ পি খুবই জনপ্রিয় একটি ভাষা, w3techs এর তথ্য অনুযায়ী পৃথিবীতে যত ওয়েব সাইট আছে তার শতকরা ৮২ ভাগই সার্ভার সাইড স্ক্রিপ্টিংয়ে PHP ব্যবহার করে। PHP ফ্রেমওয়ার্ক এ কাজ করতে হলে PHP Object Oriented Programming সম্পর্কে খুব ভালো ধারণা থাকা প্রয়োজন।

PHP Laravel Framework

বর্তমানে PHP Frameworkগুলোর মধ্যে সবচেয়ে শক্তিশালী এবং জনপ্রিয় ফ্রেমওয়ার্ক হচ্ছে Laravel। এটি একটি Opensource Secure PHP MVCWeb Framework যা সেইসব ডেভেলপারদের জন্য তৈরী করা হয়েছে যাদের একটি সহজ এবং মার্জিত টুলকিট প্রয়োজন যা পূর্ণ-বৈশিষ্ট্যযুক্ত ওয়েব অ্যাপ্লিকেশনগুলি তৈরি করতে পারে। বিভিন্ন অসামান্য বৈশিষ্ট্যগুলির জন্য এটি বিশ্বের বিভিন্ন পিএইচপি ওয়েব প্রোগ্রামারের মধ্যে খুব দ্রুত জনপ্রিয় হয়ে ওঠে । ২০১১ সালে Taylor Otwel প্রথম লারাভেল ডেভেলপ করেন। বর্তমানে laravel এর 5.5 ভার্সন চলছে।

PHP Laravel Framework শেখা শুরু করার আগে, আপনাকে অবশ্যই নিম্নের বিষয় গুলো জানতে হবে:

- PHP
- Object Oriented Concepts এবং
- MVC structure সম্পর্কে পরিষ্কার ধারণা।
- আর Laravel Framework এর Dependency Manage করার জন্য Composer অবশ্যই ইন্সটল করা থাকতে হবে।

Laravel File এবং Folder বিন্যাস

ডিফল্ট Laravel ফাইল এবং ফোল্ডার স্ট্রাকচার টি মূলত ছোট বড় সব ধরণের application development এর জন্য সাজানো হয়েছে, তবে আপনি চাইলে আপনার নিজের ইচ্ছে মতো করে ফাইল এবং ফোল্ডার structure সাজাতে পারেন। এর জন্য লারাভেল আপনাকে কোনো রেস্ট্রিকশন দিবেনা যতক্ষণ কম্পোজার class গুলো autoload করতে পারে।

- **app:**

  এটি মূলত আপনার জন্য, অর্থাৎ এই ডিরেক্টরিতে আপনার এপ্লিকেশন এর Core কোড গুলো থাকবে , যেগুলো আপনিই বানাবেন। যেমনঃ Controller, Middleware সহ অন্যান্য প্রয়োজনীয় class গুলো।

- **bootstrap:**

  এটির মধ্যে **app.php** ফাইলটি থাকে, যা দিয়ে Laravel Framework টি কে bootstraps করে। তা ছাড়া এই ফোল্ডার এর মধ্যে একটি cache folder ও থাকে, যার মধ্যে performance optimization এর জন্য Framework Generated file এবং folder গুলো থাকে , যেমন routes এবং services cache files .

- **config**

  এই Folder এর মধ্যে আপনার application এর সব configuration ফাইল গুলো থাকবে। যেমন ডাটাবেস, মেইল, সেশন ইত্যাদি।

- **database**

  এখানে আপনার application এর সব database migration এবং seeds থাকবে । এমনকি SQLite database ফাইলটাও এখানে থাকে। database migration এবং seeds কি? পরে বিস্তারিত আলোচনা করব।

- **public**

  Public Directory তে আপনার index.php file থাকবে, যেটি আপনার application এর সব ধরণের request এর entry point হিসেবে ব্যবহৃত হবে আর সব ধরনের asset যেমন CSS, JS, fonts, image ইত্যাদি থাকবে।

- **resources**

  এই folder এ আপনার ফ্রন্ট এন্ড ফাইল view থাকবে । তা ছাড়া যত রকমের uncompiled যেমন LESS, SASS, JavaScript এবং Language ফাইলগুলাও এখানে থাকবে।

- **routes**

  এই folder এ আপনার application এর সব route definitions গুলা থাকবে। যেমনঃ web.php, api.php, channels.php, এবং console.php. আমরা route নিয়ে পরে বিস্তারিত আলোচনা করব।

- **storage**

  এই folder এ আপনার application এর সব compiled Blade templates, file based sessions, file caches, এবং framework generated অন্যান্য file গুলো থাকবে। এই folder টি তিনটি folder যথাক্রমে app, framework, এবং logs folder এ বিভক্ত। app folder আপনার application generated ফাইল গুলো থাকবে, framework folder আপনার framework generated ফাইল গুলো থাকবে এবং logs folder আপনার application এর লগ ফাইল গুলো থাকবে।

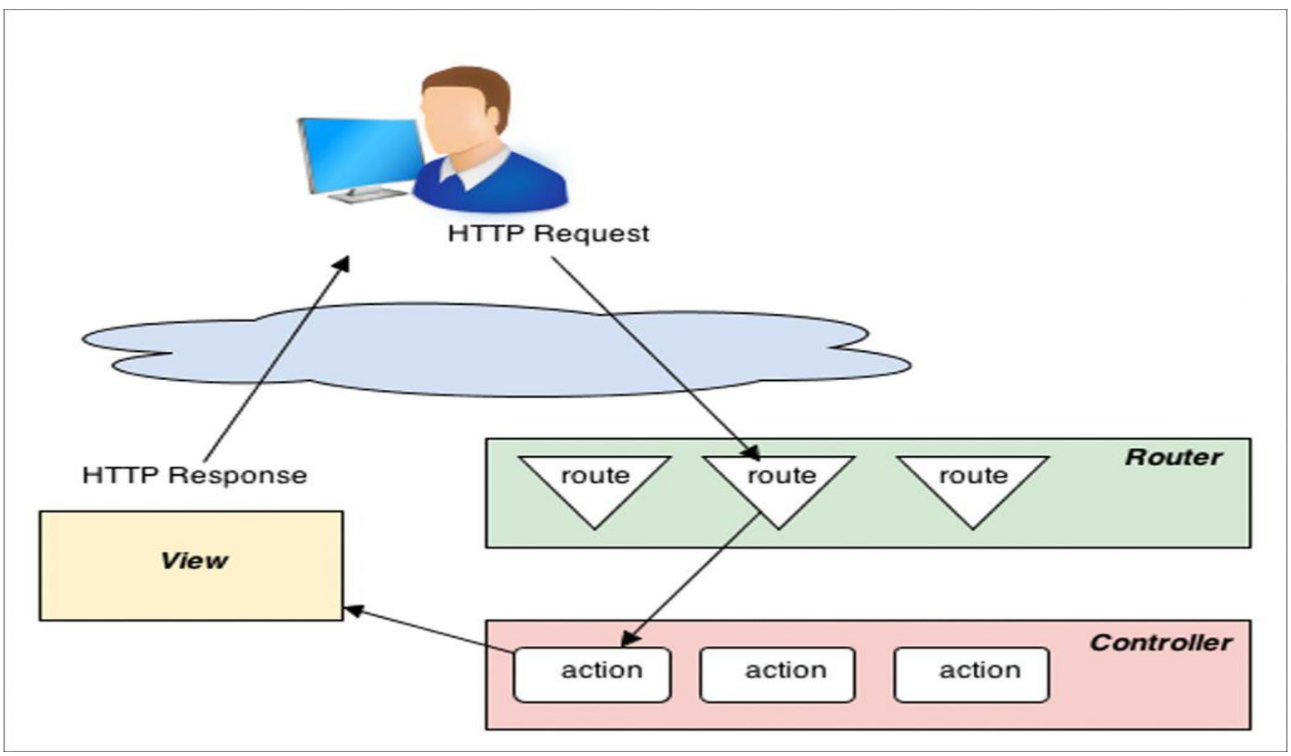
- **tests**

  এই folder এ আপনার application এর test ফাইল গুলা থাকে , যেমনঃ PHPUnit ফাইল।

- **vendor**

  এই folder এ আপনার সব Composer dependency ফাইলগুলা থাকে।

Laravel Framework এ routes কি?



সাধারণত ইংরেজি শব্দ route যাকে অনেক সময় আমরা বাংলায় বলি রুট যার অর্থ দাঁড়ায় গন্তব্যস্থলে পৌঁছানোর রাস্তা। আর Laravel Application এ route হচ্ছে URL থেকে request গ্রহণ করে এবং application কে resource এর জন্য নির্দেশনা প্রদান করে। আরো সহজ ভাবে বলা যায় route হল আপনার অ্যাপ্লিকেশনের একটি request এর বিপরীতে কোন URL টি hit হবে? বা কোথায় থেকে কি response করবে তা নির্ধারণের একটি উপায়। Laravel 5.5 এ সব route গুলো routes ফোল্ডারে তৈরী করা থাকে। এর মধ্যে web application এর route সমূহ routes/web.php তে লিখা হয়। এবং API এর জন্য route সমূহ routes/api.php তে লিখা হয়। Laravel Framework এ route এর সবচেয়ে বড় সুবিধা হচ্ছে আপনি এক যায়গা থেকেই সমস্ত route কে নিয়ন্ত্রণ করতে পারবেন অর্থাৎ পরবর্তিতে route সম্পর্কিত যেকোনো ধরনের পরিবর্তন এখান থেকেই করতে পারবেন।

Laravel Route এর প্রাথমিক ধারণা

Laravel Route লিখার আগে আপনাকে PHP Anonymous Function, Closure এবং PHP Class এর Static Method সম্পর্কে ধারণা থাকা দরকার, আমরা ধরে নিলাম আপনি এই গুলো জানেন। তো আসা যাক কিভাবে route দিয়ে URL Request গুলো Manage করতে পারি। ধরুন আমাদের Web Application এ তিনটি পেজ আছে , সেগুলো যথাক্রমে Home, About এবং Contact. এখন Web Application ব্যবহারকারী এই তিনটি পেজ এর মধ্যে যাকে request করবে শুধু সংশ্লিষ্ট পেজটি response করবে। অনেকটা নিচের URL এর মত:

for Home page:

http://localhost:8000/

এখন এই URL Request এর জন্য আপনি Laravel এর routes/web.php ফাইল এ ঠিক নিচের মত করে লিখতে পারেন।

```
Route::get('/', function (){
   echo "<h2>This is Home Page</h2>;
});
```

এখন আপনি আপনার ব্রাউজার এর Address bar এ http://localhost:8000 লিখে hit করুন

নোট: অবশ্যই ” php artisan serve ” command দিয়ে server run রাখতে ভুলবেননা।

ব্যাখ্যা: উপরোক্ত উদাহরণে আমরা Route class এর get() মেথডকে call করেছি। Route Class এর get() মেথডটি দুইটি Argument গ্রহণ করে। প্রথম Argument হিসেবে URL Path অর্থাৎ যেই URL এ hit করবে সেটি, এবং দ্বিতীয় Argument হিসেবে একটি Anonymous Function সাথে closure থাকতে পারে। যেমনঃ উপরোক্ত উদাহরণে আমাদের Home URL এর জন্য প্রথম আর্গুমেন্ট হিসেবে শুধু forward slash (/) ব্যবহার করি , যার দ্বারা Laravel route রুট ডোমেইনকে নির্দেশ করে। একই ভাবে About URL এর জন্য forward slash সাথে about (/about) এবং Contact URL এর জন্য forward slash সাথে contact(/contact) লিখি। আর তিনটি

URL এর জন্যই দ্বিতীয় Argument হিসেবে একটি Anonymous Function ব্যবহার করি যার দ্বারা action কি হবে তা নির্ধারণ করি।

Laravel Route এবং URL Parameter নিয়ে কাজ

অবশ্যই, কখনও কখনও আপনাকে আপনার route মধ্যে URI এর segment গুলিকে receive বা capture করতে হবে। এটি করার জন্য Laravel Route এ দুটি উপায় আছে যার মাধ্যমে আপনি URL- এর মাধ্যমে পাঠানো Parameter গুলিকে Capture করতে পারেন।

1. Required Parameters
2. Optional Parameters

Required Parameters

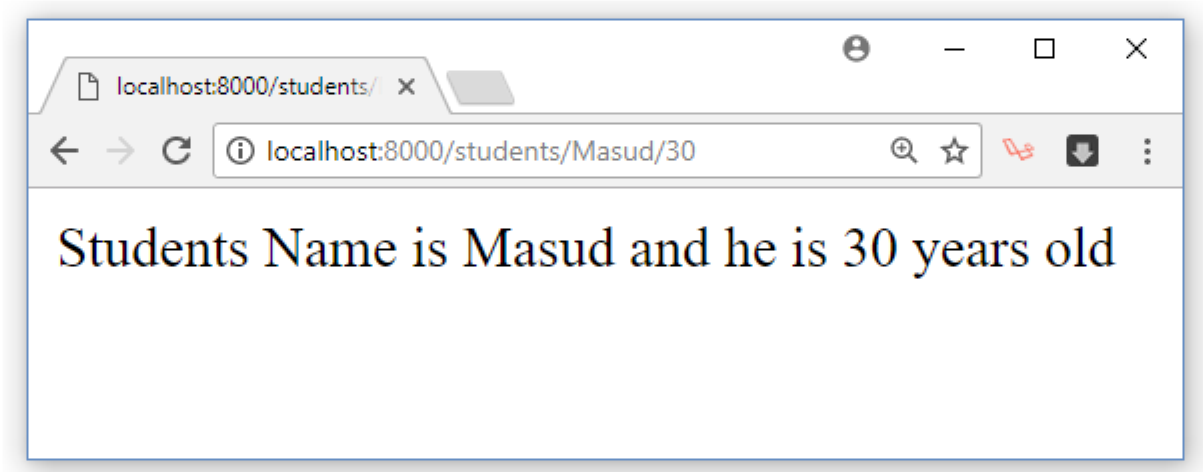
অর্থাৎ আপনার URL এ অবশ্যই Parameter থাকতে হবে। ধরুন আপনাকে URL থেকে ইউজারদের নাম capture করতে হতে পারে। আপনাকে route এ Parameter define করে তা করতে হবে। একটি উদাহরণ দিয়ে যাক।

```
// get the parameter of name
Route::get('students/{name}', function($name) {
    echo 'Students Name is ' . $name;
});
```

এখন আপনার ব্রাউজারে, আপনি নিচের মতো করে student name এ access করতে পারেন।

http://localhost:8000/students/Masud

http://localhost:8000/students/Masud/30

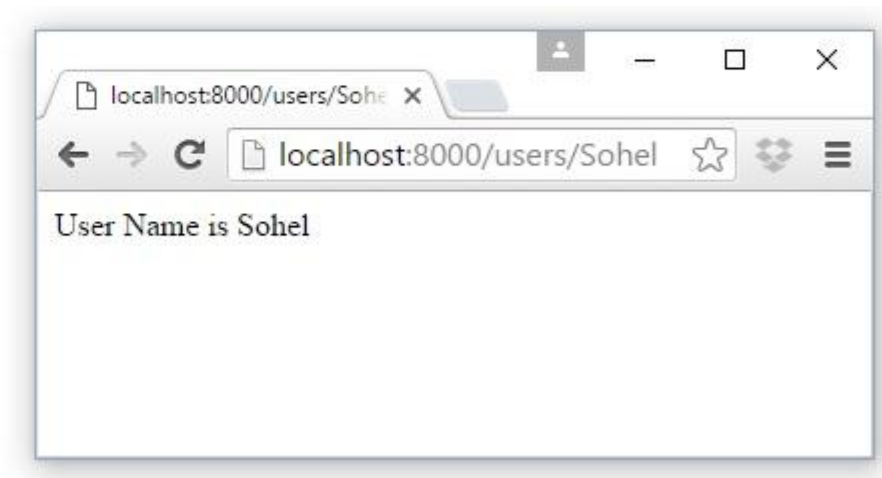


Optional Parameters

আপনি যদি আপনার URL এ এমন কিছু Parameter পাঠাতে চান , যা থাকতেও পারে আবার নাও থাকতে পারে। এই সব ক্ষেত্রে আপনাকে Laravel এর Optional Parameter System ব্যবহার করতে হবে। তখন URL এ এই প্যারামিটার গুলির থাকার আবশ্যকীয় নয়। আর এই Parameter গুলির নামের পরে “?” চিহ্ন দিতে হয়। নিম্নের কোডটি লক্ষ্য করুন :

```
Route::get('users/{name?}', function($name="Masud Alam") {
    echo 'User Name is ' . $name;
});
```

এখন আপনার URL এ যেই Parameter পাঠাবেন সেটা দেখাবে, অন্যথা default Parameter Masud Alam দেখাবে। নিম্নের ছবিতে লক্ষ্য করুন :

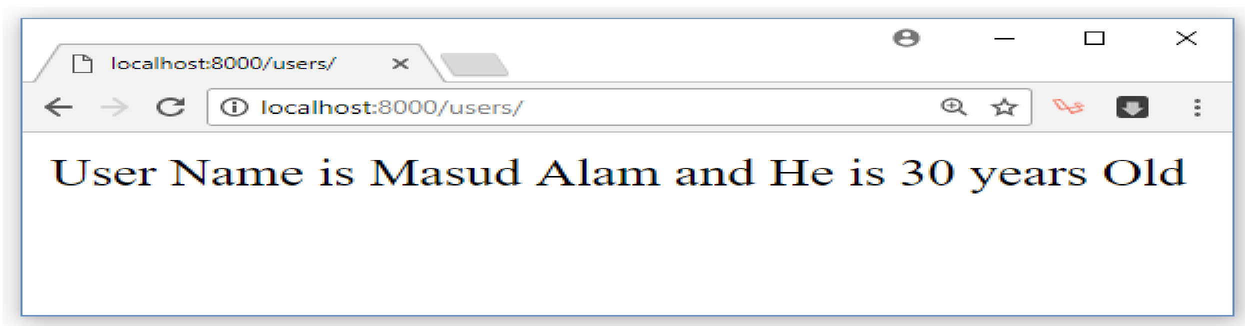


একইভাবে আপনি একাধিক Optional Parameter পাঠাতে পারেন, নিচের উদাহরণ লক্ষ্য করুন :

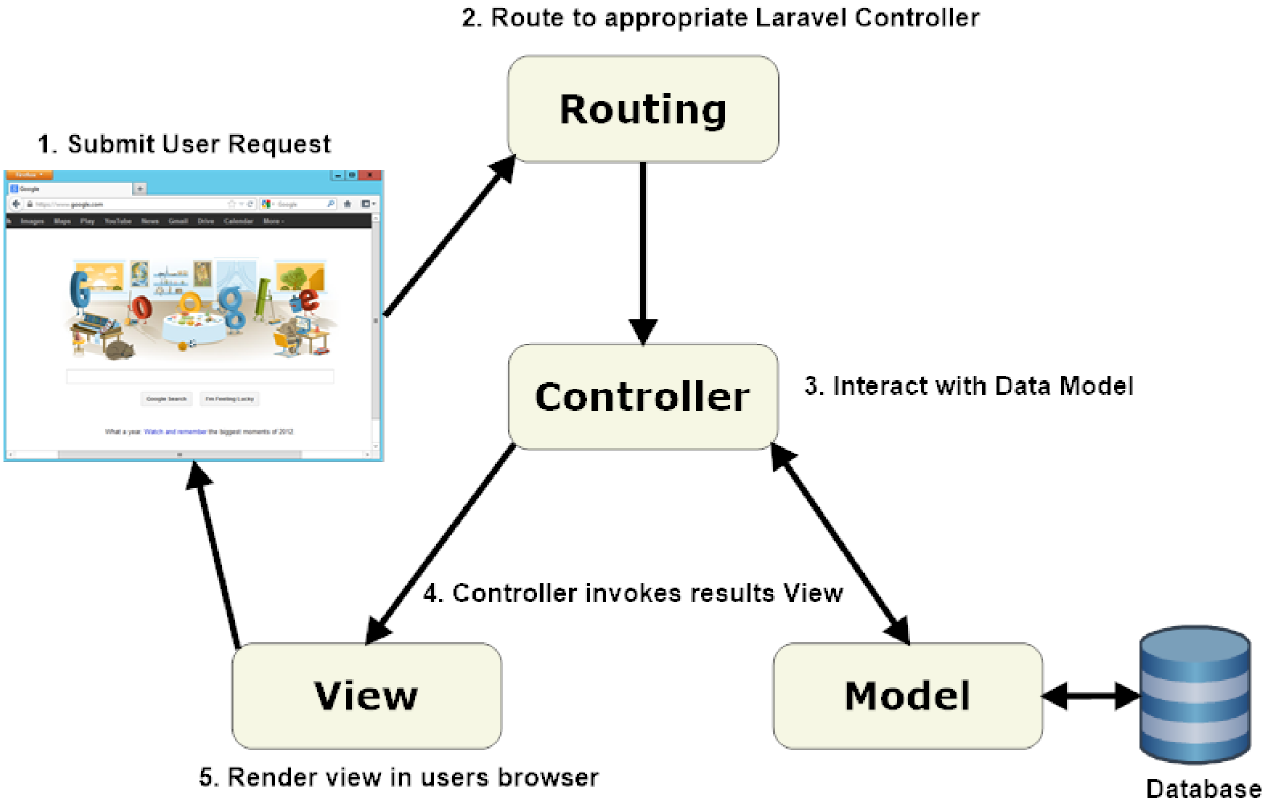
```
Route::get('users/{name?}/{age?}', function($name="Masud Alam",$age=30) {
    echo "User Name is $name and He is $age years Old";
});
```

এখন আপনার ব্রাউজারে, আপনি নিচের মতো করে user name এবং তার age এ access করতে পারেন।

http://localhost:8000/users

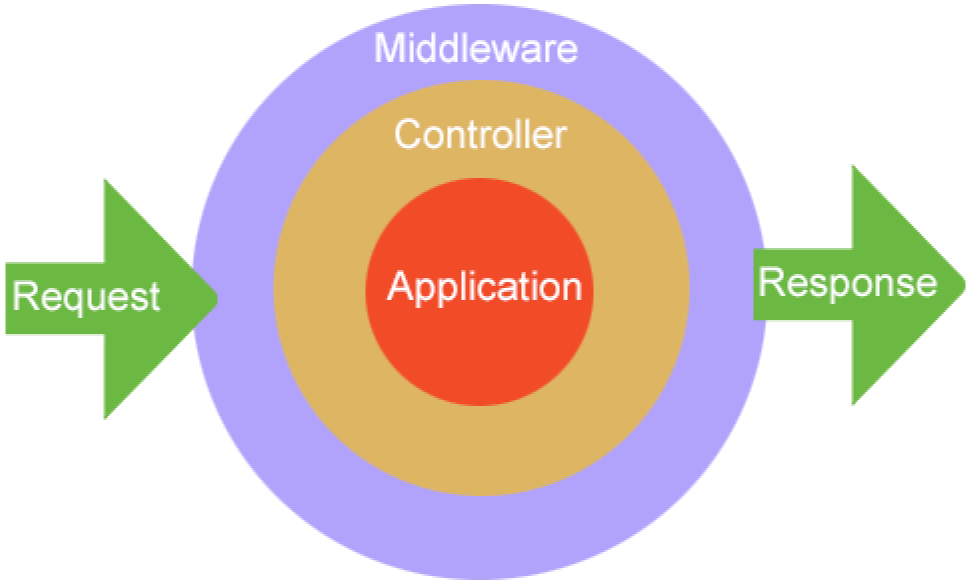


Larvel Framework এ View কি?



View নামটা দেখেই বুঝা যাচ্ছে Framework এর মধ্যে কোনো কিছু দেখানো বা show করা। হ্যাঁ আসলেই ঠিক, MVC Framework এর মধ্যে “V” অক্ষরটি View এর জন্য ব্যবহৃত হয়। Laravel অথবা যেকোনো MVC Framework এ View এর কাজ হচ্ছে route অথবা Controller এর মাধ্যমে কোনো কিছু ব্রাউজার এ show করা বা দেখানো। MVC ভিত্তিক যেকোনো software অথবা Web Application এর কোনো কিছু show বা দেখানোর কাজটি করা হয় View তে । User কোন কিছুর জন্য request করতে হলে সেটা View এর মাধ্যমেই করতে হয়, তারপর সেটা প্রথমে যায় Controller এ । Request এর Response যদি Database এর কোন Data প্রয়োজন হয় তবে Controller তা Model এর মাধ্যমে এনে user এর প্রয়োজন অনুযায়ী Process করে View দিয়েই ব্রাউজার দিয়ে show বা দেখতে হয়। এরপর User তার Request এর Response ও View এর মাধ্যমে দেখতে পায়। যদিও আপনি এই কাজটি Laravel এর view ছাড়া সরাসরি route অথবা Controller দিয়ে show বা দেখাতে পারেন। আলাদাভাবে View ব্যাবহার করা হচ্ছে MVC Platform এর একটি আদর্শ পদ্ধতি। আর এতে সুবিধা হচ্ছে আপনার software অথবা Web Application এর সব html code গুলো route অথবা controller এর মধ্যে না রেখে আলাদা view ফাইল এর মধ্যে রাখা। Laravel এ view ফাইল গুলা resources/views এই ডিরেক্টরিতে রাখতে হয়।

Laravel Framework এ Middleware কি?



Laravel Middleware

Laravel Framework এ Middleware হচ্ছে আপনার application এ বিভিন্ন সোর্স থেকে আসা HTTP request গুলোকে controller এ পাঠানোর আগে filtering করার একটা দুর্দান্ত Mechanism বা পদ্ধতি। ধরুন Laravel authentication Middleware এর কথাই ধরা যাক, যখন আপনার application এ কোনো user লগইন করতে চায়, সেক্ষেত্রে Laravel authentication Middleware কাজ হচ্ছে যেই user লগইন করতে চায় , তাকে authentication Middleware নিজে Middle Man হিসেবে verify করবে , যদি ইউজার authenticated না হয় , তাহলে Laravel authentication Middleware সেই user কে login screen এ পাঠাবে , আর যদি authenticated হন, তাহলে laravel authentication Middleware ইউজারকে পরবর্তী কাজ করার বা পরবর্তী ধাপে যাওয়ার অনুমতি প্রধান করবে। সহজ ভাবে বলা যায় , আপনার application এ যেকোনো request এর পর এবং response এর পূর্বে সব ধরণের verification এবং authentication করার Mechanism হচ্ছে Laravel Middleware. Laravel এ Middleware ফাইল গুলা app/Http/Middleware এই ডিরেক্টরিতে রাখতে হয়।

Laravel Framework এ কি কি ধরণের Middleware আছে?

Laravel এ authentication Middleware এর পাশাপাশি আরো অনেক গুলো Middleware বিভিন্ন কাজের জন্য লিখা হয়েছে, যেমন: সব গুলো রেসপন্স application ত্যাগ করার আগে তার সঠিক header গুলো সংযুক্ত করার জন্য তৈরী করা হয়েছে CORS Middleware . আর incoming request এর লগ রাখার জন্য logging Middleware তৈরী করা হয়েছে। এছাড়াও CSRF protection এর জন্য CSRF Middleware I এছাড়া আপনার নিজের প্রয়োজন মতো বিভিন্ন Custom Middleware খুব সহজে তৈরী করে নিতে পারেন।

Middleware ব্যবহার করে আপনার application এর কোন কোন কাজ গুলো করতে পারেন ?

Laravel Framework এ Middleware ব্যবহার করে আমরা বিভিন্ন ধরণের কাজ করতে পারি, তারমধ্যে উল্লেখযোগ্য হচ্ছে :

১. আপনার Application এর Request গুলোর Log রাখতে পারেন।
২. বিভিন্ন Condition এর উপর ভিত্তি করে ইউজারদেরকে ভিন্ন ভিন্ন page এ redirect করতে পারেন।
৩. আপনার Application এ Incoming Parameters গুলোকে altering এবং sanitizing করতে পারেন।
৪. Response গুলোকে Manipulate করতে পারেন।
৫. এবং আরো অনেক কিছু

Laravel Framework এ Blade Templating Engine কি?

Laravel Framework এ Plain PHP Code থেকে design part যাকে আমরা বলি Presentation Layer (অর্থাৎ HTML, CSS, JS) কে আলাদা করার জন্য Blade Template Engine ব্যবহার করা হয়। আর এটা ব্যাবহার করা খুবই সহজ। যদিও Blade আপনাকে আপনার view তে Plain PHP Code ব্যবহার করতে বাধা দেয় না। প্রকৃতপক্ষে, View তে ব্লেডের সব code গুলো Plain PHP কোডে Compiled হয় এবং cache করা হয়। এবং কোডে কোনো সংশোধন করা না হওয়া পর্যন্ত cache অবস্থায় থাকে , যার মানে Blade মূলত আপনার অ্যাপ্লিকেশনে কোনো বাড়তি বোঝা তৈরী করেনা। বরং Plain PHP Code গুলো থেকে HTML,CSS এবং JS কে আলাদা করায় আপনার এপ্লিকেশন এর Security, Performance, code reuse সহ আরো অনেক সুবিধা যোগ করে। Blade Template এর Syntax মূলত ASP.net Razor syntax সিনট্যাক্স দ্বারা অনুপ্রাণিত। Blade Template এ কোড লেখার জন্য আপনার View ফাইলের এক্সটেনশন হতে হবে .blade.php . Blade Template ফাইল গুলো ও **resources/views** ফোল্ডারে রাখতে হয়।

Blade template তৈরী করার নিয়ম

যেকোনো html Template কে Laravel এর Blade Template এ রূপান্তর করতে হলে প্রথমে আপনাকে আপনার template কে slicing করে header section নিয়ে header.blade.php, sidebar section নিয়ে sidebar.blade.php এবং footer section নিয়ে footer.blade.php ফাইল তৈরী করে নিতে হবে (যদি আপনার html template এ section গুলো থাকে)। আর মূল file কে master.blade.php নামে save করতে হবে। তারপর master.blade.php ফাইল এ Laravel Blade Command @include দিয়ে ফাইল গুলোকে include করে নিতে হবে। তারপর সেই Master Page কে Laravel Blade Command @extendsদিয়ে extend করে অন্য view Page গুলো বানাতে হবে। ধরুন আপনার View Page গুলোর নাম যথাক্রমে home,about এবং contact তাহলে Blade এর Format এ এগুলো হবে যথাক্রমে home.blade.php, about.blade.php এবং contact.blade.php.

প্রথমেই দেখে নেয়া যাক আমাদের Blade Template কেমন হবে :

এবার Blade Template এর File Structure দেখে নেয়া যাক :

File struture :-

```
-public
--assets
---css
---js
---fonts
---img
- resources
-- views
--- layout
---- master.blade.php
---- header.blade.php
---- footer.blade.php
---- sidebar.blade.php
--- pages
---- home.blade.php
---- about.blade.php
---- contact.blade.php
```

Blade Template Commands

এবার আমরা আমাদের Blade Template এ ব্যবহার করব এমন কিছু Blade Template Commands সম্পর্কে জানা যাক: –

@yield :- এটি যেকোনো value প্রদর্শন করতে ব্যবহৃত হয়।
@section :- এটি Template এর যেকোনো section define করতে ব্যবহৃত হয়।
@endsection :- এটি define করা section end করতে ব্যবহৃত হয়।
@include :- এক view file কে অন্য view file এ include করার জন্য ব্যবহৃত হয়।
@extends:- Master Layout কে Child Layout এ extends বা inherit করার জন্য ব্যবহৃত হয়।
{{ asset('your_asset_file_path') }} :- css file, js file এবং যেকোনো image link করার জন্য ব্যবহৃত হয়।
{{URL::to('/your_page_name')}} :-এটা মূলতঃ বিভিন্ন page এর সাথে লিংক করার জন্য ব্যবহৃত হয়

Laravel Framework এ Migration কি?

প্রযুক্তিগত অগ্রগতির সাথে সাথে , আমরা যারা আই.টি. পেশাদার, ওয়েব ডেভেলপার অথবা প্রোগ্রামার, আমরা অনেকেই 'Database Schema' নামটির সাথে পরিচিত এবং আমরা অনেকেই Database Schema কে create, modify এবং up-to-date রাখতে গিয়ে অনেক সমস্যার সম্মুখীন হই। আপনার Database কে up-to-data রাখা এবং সব জটিলতা সমাধান করার জন্য, Laravel Framework নিয়ে আসছে Migration সুবিধা। Laravel Framework এ Migration ফাইল গুলি /database/migrations ফোল্ডারে থাকে।

Laravel Framework এ, Migrations হচ্ছে আপনার Database Schema গুলোর যেকোনো গঠন পরিবর্তন করার জন্য একটি ফাইল-ভিত্তিক পদ্ধতি। আপনার Database এর Version Control System এর মতো। যা

দিয়ে আপনার টিমকে অ্যাপ্লিকেশন গুলোর Database Schema কে সহজেই create, modify এবং share করার সুযোগ দিতে পারেন। এবং current schema কে up-to-date রাখতে পারেন। যেমন, আপনি আপনার Database এ table তৈরি এবং Drop করা, table এ নতুন column যোগ করা, আপডেট করা, মুছে ফেলা, table এ index যোগ করা সহ অন্যান্য কাজের জন্য Laravel Migration ব্যবহার করতে পারেন। পরবর্তীতে যেকোনো ভুলের জন্য অথবা যেকোনো পরিবর্তনের জন্যে আপনি খুব সহজে আগের অবস্থায় অথবা পিছনের যেকোনো ভার্সন এ ফিরে আসতে পারেন। কারণ প্রতিটি মাইগ্রেশন একটি টেক্সট ফাইলে সংরক্ষণ করা হয়, আপনি আপনার Project Repository তে রেখে তাদের পরিচালনা করতে পারেন। Laravel Framework এ Migrations সাধারণত Laravel এর schema builder সঙ্গে সংযুক্ত থাকে , যাতে সহজেই আপনার অ্যাপ্লিকেশন এর Database Schema তৈরী করতে পারেন।

ধরুন, আপনি এবং আপনার টীম git Version Control ব্যবহার করেন। হঠাৎ আপনার মনে হলো user টেবিল এ নতুন একটি ফিল্ড লাগবে যার নাম status এবং টাইপ হবে boolean । এখন আপনি Laravel Framework এর migration সুবিধা ব্যবহার করে টেবিলের পরিবর্তন টা করলেন ও git এ পাঠিয়ে দিলেন। এখন আপনার team এর অন্যান্য সদস্যরা চাইলে git merge করে Laravel migration ফাইলটি রান করে Database কে up-to-date করে নিতে পারেন। আগের মতো আর .sql ফাইল গুলো একজন আরেকজনের কাছ থেকে খুঁজে নিতে হবেনা ।

Migration নিয়ে কাজ করার পূর্ব প্রস্তুতি :

Laravel Framework এ Migration নিয়ে কাজ করতে হলে আপনাকে প্রথমে Database এর host এর সাথে connect এবং Database Select করতে হবে। আর এর জন্য আপনাকে আপনার project এর root ফোল্ডারের .env ফাইল এ Database এর host, username, password এবং Database name বলে দিতে হবে। নিচের ছবিটি লক্ষ্য করুন :

```
DB_CONNECTION=mysql
DB_HOST=127.0.0.1
DB_PORT=3306
DB_DATABASE=laravel
DB_USERNAME=root
DB_PASSWORD=''
```

Laravel Database Connection in .env file-

Note: .env file edit করার পর অবশ্যই php artisan সার্ভার টি রিস্টার্ট দিতে হবে। অথবা “php artisan config:clear” এই command টি run করতে হবে।

Laravel Framework এ Requests কি?

যদিও ওয়েব দুনিয়ায় Request বিষয়টি ব্যাপক ভাবে ব্যবহৃত হয়, এক কথায় বলা যায় , ক্লায়েন্ট বা ইউজার পিসি থেকে সার্ভার পিসিতে যখন কোনো কিছু চাওয়া হয় , তখন আমরা তাকে বলি Requests. Larvel Framework এ সবধরণের Request গুলো Illuminate\Http\Request Class এর মাধ্যমে হ্যান্ডলিং করা হয়।

Laravel Framework এ Database Query Builder কি?

Query Builder হচ্ছে Laravel এর একটি Facades Class যা আপনাকে Database query গুলো তৈরী এবং চালনা করার জন্য একটি সহজ এবং স্বচ্ছ ইন্টারফেস এর সুবিধা দেয়। এটি আপনার অ্যাপ্লিকেশনে Database সংশ্লিট সব ধরণের কাজের জন্য ব্যবহার করা যেতে পারে এবং Laravel এ support করে এমন সব Database System (যেমন: MySQL, PostgreSQL,SQLite,SQL Server) এর সাথে কাজ করে। query builder যেহেতু PDO parameter binding ব্যবহার করে তাই এটি ব্যবহারের ফলে আপনার application এ SQL injection attack হওয়া থেকে বিরত রাখে। এবং Binding Parameter হিসাবে পাস করা স্ট্রিং clean করার কোন প্রয়োজন হয়না।

facade কি?

Note: facade হচ্ছে class গুলোকে একটি static interface প্রধান করে বা Laravel Framework এ facade গুলো Service Container এর মধ্যে static froxy হিসেবে কাজ করে। অর্থাৎ class এর Method গুলোকে Traditional static Method গুলোর তুলনায় আরো মসৃণ, সংক্ষিপ্ত static code করার সুযোগ দেয় ।

Query Builder নিয়ে কাজ করার পূর্ব প্রস্তুতি :

Laravel Framework এ Database Query Builder নিয়ে কাজ করতে হলে আপনাকে প্রথমে Database এর host এর সাথে connect এবং Database Select করতে হবে। আর এর জন্য আপনাকে আপনার project এর root ফোল্ডারের .env ফাইল এ Database এর host, username, password এবং Database name বলে দিতে হবে।