**Laravel**

1.Project Creation Command : composer create-project --prefer-dist laravel/laravel blog

2.php artisan serve

3.php artisan make:Controller HomeController (for making new Controller)

4.Php artisan make:Model HomeModel (For making new Model)

5.php artisan make:Middleware HomeMiddleware (For making new Middleware)

6.php artisan make:Migration HomeMigration (For making new Migration)

7.php artisan cache:clear (for clearing application cache)|| cofig.clear,view:clear,migration:clear,route:clears

8.route/web.php contains the the definition of route files

9. Route::get('/', function () {

    return view('welcome');

}); //Returning a view

10.index.blade.php (we use .blade.php so we can use the blade template engine functionality)

11. <a href="{{url('/')}}">Home</a><br>

<a href="{{url('/about')}}">About</a><br>

(blade template engine feature)

12.For the Higher version of 8 we have to include Route file Web.php use App\Http\Controllers\DemoController; then use the syntax as Route::get('/',[DemoController::class, 'index']);

Route::get('/about', [DemoController::class, 'about']);

13.

Route::get('/{name}', function($name){

    return view('welcome',['name' => $name]);

}); //for passing name in url

<h1>This is Welcome Section</h1>

<h1> The name is {{$name}} </h1>

14. return redirect('about'); (for redirecting )

15.Components: For reusability alternative of php include and require (php artisan make:component Header)

16. <x-header /> This is like include in php where header is the file name.

**Blade Template**

1.<?php echo $somthing ?> -🡪 {{}}

2. @if(condition) @endif

Ex-a: @if($user=='shuvo')

<h1>This was {{$user}} </h1>

@endif

Ex-b: @for($i=0;$i<=10;$i++)

<h1>Hello </h1>

@endfor

return view('form',['user' => ['shuvo','shawon','shiful','sahabuddin']]);

3. @include('user') //include view inside view

4. var v =@json($user[0]); //for getting value from controller in javascript code.

[5.@csrf](mailto:5.@csrf) //for prevent cross site request

**HTML FORM**

1.function getData(Request $req)

    {

        return $req -> input();

    }

2. <form action="users" method="POST">

    @csrf

    <input type="text" name="username" placeholder="Enter Your username"/> <br><br>

    <input type="password" name="password" placeholder="Enter Your password"/> <br><br>

    <button type='submit'>Login</button>

</form>

3. Route::post('users',[FormController::class,'getData']);

Route::view('/login','users');

OUTPUT: {"\_token":"GKKsBfr0hrX3WuuiGEnJxQcoYZDhz9eAifzHdxMR","username":"wqrwq","password":"fwqwqf"}

**Middleware**

Three types of middleware : 1.global 2.group 3.route

We have to include newly created middleware in the kernel.php like \App\Http\Middleware\ageMiddleware::class,

1.  if($request -> age && $request->age <18)

        {

            return redirect('noaccess');

        }

2.

a.

For group middleware we have to include array in kernel.php like:

'protectedPage' => [

            \App\Http\Middleware\ageMiddleware::class,

        ],

b. Route::group(['middleware'=> ['protectedPage']], function(){

  Route::view('add','add');

}); //in web.php

**3.Route middleware: only for specific route**

a. 'protectedPage' => \App\Http\Middleware\ageMiddleware::class, //in kernel.php

b. Route::view('/add','add')->middleware('protectedPage'); //in Route

c. public function handle(Request $request, Closure $next)

    {

        if($request -> age && $request->age <18)

        {

            return redirect('noaccess');

        }

        return $next($request);

    }//in your created middleware