

MIDDLEWARE

REQUEST MIDDLEWARE

- Understanding over the middleware structure

Command : php artisan make:middleware MiddlewareName

What is middleware?

Ans: Middleware হচ্ছে request এবং response এর মাঝে যদি কোন কোড কাজ করে বা শর্ত থাকে ঐ কাজকে আমরা Middleware বলে থাকি।

Middleware ও এক ধরনের controller

request -> middleware -> controller -> response

kernal.php file 👍

kernal.php কোন কিছু (middleware) add করা মানে তা globally বা group আকারে defined করে দেওয়া

CheckRequestDetails.php

```
1  <?php
2  namespace App\Http\Middleware;
3  use Closure;
4  use Illuminate\Http\Request;
5  use Symfony\Component\HttpFoundation\Response;
6  class CheckRequestDetails
7  {
8      /**
9       * Handle an incoming request.
10      *
11      * @param  \Closure(\Illuminate\Http\Request): (\Symfony\Component\HttpFoundation\Response)  $next
12      */
13     public function handle(Request $request, Closure $next): Response
14     {
15         return $next($request);
16     }
17 }
18
```

REQUEST MIDDLEWARE

■ Check Request Header Inside Middleware

web.php

```
2 use App\Http\Controllers\DemoController;
3 use App\Http\Middleware\CheckRequestDetails;
4 use Illuminate\Support\Facades\Route;
5 Route::post('/DemoAction', [DemoController::class, 'DemoAction'])->middleware([CheckRequestDetails::class]);
```

CheckRequestDetails.php

```
13 public function handle(Request $request, Closure $next): Response
14 {
15     $key= $request->header('key');
16     if($key=="123"){
17         return $next($request);
18     }
19     else{
20         return response()->json("unauthorized",401);
21     }
22 }
```


REQUEST MIDDLEWARE

Redirect Request From Middleware

```
●●● CheckRequestDetails.php

13     public function handle(Request $request, Closure $next): Response
14     {
15         if($request->header('key')== "123"){
16             return $next($request);
17         }
18         else{
19             return redirect('/');
20         }
21     }
```

REQUEST MIDDLEWARE

Apply For Specific Route

```
Route::get('/DemoAction', [DemoController::class, 'DemoAction'])->middleware([CheckRequestDetails::class]);
```

Apply For Route Group

web.php

```
6 Route::middleware(['CheckRequestDetails'])->group(function () {
7     Route::get('/DemoAction1', [DemoController::class, 'DemoAction']);
8     Route::get('/DemoAction2', [DemoController::class, 'DemoAction']);
9     Route::get('/DemoAction3', [DemoController::class, 'DemoAction']);
10 });
11 Route::get('/DemoAction4', [DemoController::class, 'DemoAction']);
```

Kernel.php

```
55 protected $middlewareAliases = [
56     'CheckRequestDetails' => \App\Http\Middleware\CheckRequestDetails::class,
```


REQUEST MIDDLEWARE

■ Apply For Whole Application

```
Kernel.php

31     protected $middlewareGroups = [
32         'web' => [
33             \App\Http\Middleware\CheckRequestDetails::class,
34             \App\Http\Middleware\EncryptCookies::class,
35             \Illuminate\Cookie\Middleware\AddQueuedCookiesToResponse::class,
36             \Illuminate\Session\Middleware\StartSession::class,
37             \Illuminate\View\Middleware\ShareErrorsFromSession::class,
38             // \App\Http\Middleware\VerifyCsrfToken::class,
39             \Illuminate\Routing\Middleware\SubstituteBindings::class,
40         ],
41         'api' => [
42             \App\Http\Middleware\CheckRequestDetails::class,
43             // \Laravel\Sanctum\Http\Middleware\EnsureFrontendRequestsAreStateful::class,
44             \Illuminate\Routing\Middleware\ThrottleRequests::class.':api',
45             \Illuminate\Routing\Middleware\SubstituteBindings::class,
46         ],
47     ];
48
```

REQUEST MIDDLEWARE

- Manipulate ([ADD](#)) Request Details Inside Middleware

```
$request->headers->set('key3', 'variable3');
```

- Manipulate ([Remove](#)) Request Details Inside Middleware

```
$request->request->remove('name');
```

- Manipulate ([Replace](#)) Request Details Inside Middleware

```
$request->headers->replace(['name'=>'XXX', 'age'=>100]);
```


REQUEST RATE LIMITING

■ Inside Kernel file middleware group

[Request Rate limiting middleware \(throttle\)](#)

request rate limiting মানে রাউট point এর বা রাউট এর request গুলুর লিমিটেশন করে দেওয়া। যার ফলে ইউসারের অহেতুক request আটকানো যায়।

```
Kernel.php

31     protected $middlewareGroups = [
32         'web' => [
33             'throttle:60,1',
34             \App\Http\Middleware\EncryptCookies::class,
35             \Illuminate\Cookie\Middleware\AddQueuedCookiesToResponse::class,
36             \Illuminate\Session\Middleware\StartSession::class,
37             \Illuminate\View\Middleware\ShareErrorsFromSession::class,
38             // \App\Http\Middleware\VerifyCsrfToken::class,
39             \Illuminate\Routing\Middleware\SubstituteBindings::class,
40         ],
41         'api' => [
42             'throttle:60,1',
43             // \Laravel\Sanctum\Http\Middleware\EnsureFrontendRequestsAreStateful::class,
44             \Illuminate\Routing\Middleware\ThrottleRequests::class.':api',
45             \Illuminate\Routing\Middleware\SubstituteBindings::class,
46         ],
47     ];
```


REQUEST RATE LIMITING

■ For Specific Routing End Point

● ● ● web.php

```
1 <?php
2 use App\Http\Controllers\DemoController;
3 use Illuminate\Support\Facades\Route;
4 Route::post('/DemoAction', [DemoController::class, 'DemoAction'])->middleware('throttle:5,1');
5
```