文件夹 app/Http/Middleware php artisan make:middleware public function handle(\$request, Closure \$next) // Perform action public function handle(\$request, Closure \$next) return \$next(\$request); if (\$request->age <= 200) {</pre> return redirect('home'); public function handle(\$request, Closure \$next) return \$next(\$request); \$response = \$next(\$request); // Perform action return \$response; Route::get('admin/ profile', function () { })->middleware('auth'); Route::get('/', function () { 先要分配middleware 在app/Http/ Kernel.php 中 })->middleware('first', 'second'); 先要在 \$routeMiddleware 中配置 使用Raw_Middleware use App\Http\Middleware\CheckAge; Route::get('admin/profile', function () { })->middleware(CheckAge::class); Route::get('/', function () { })->middleware('web'); 在 app/Http/Kernel.php 中注册group Route::group(['middleware' => Middleware groups ['web']], function () { }); public function handle(\$request, Closure \$next, \$role) if (! \$request->user()->hasRole(\$role)) { // Redirect... Middleware parameters return \$next(\$request); Route::put('post/{id}', function (\$id) { 多个参数使用commas(逗号) 连接 })->middleware('role:editor'); For example, the "session" middleware included with Laravel writes the session data to storage after the response has been sent to the browser. Terminable Middleware If you define a terminate public function handle(\$request, Closure \$next) method on your middleware,

Middleware

it will automatically be called after the response is sent to the browser

```
return $next($request);
public function terminate($request, $response)
   // Store the session data...
```