Response

1. Basic Response

return response('Hello, World!');

1. JSON Response:

return response()->json(['message' => 'Success']);

1. Redirect Response:

Route::get('/dashboard', function () {

return redirect('home/dashboard');

});

4-[Redirecting With Flashed Session Data](https://laravel.com/docs/10.x/responses" \l "redirecting-with-flashed-session-data)

Route::post('/user/profile', function () {

return redirect('dashboard')->with('status', 'Profile updated!');

});

5-Redirect with Validation Errors:

return redirect('/form')->withErrors($validator);

6-[File Downloads](https://laravel.com/docs/10.x/responses" \l "file-downloads)

return response()->download($pathToFile);

file upload with laravel

1-[The Local Driver](https://laravel.com/docs/10.x/filesystem" \l "the-local-driver)

use Illuminate\Support\Facades\Storage;

Storage::disk('local')->put('example.txt', 'Contents');

2-[The Public Disk](https://laravel.com/docs/10.x/filesystem" \l "the-public-disk)

'links' => [

public\_path('storage') => storage\_path('app/public'),

public\_path('images') => storage\_path('app/images'),

],

For file path

use Illuminate\Support\Facades\Storage;

$path = Storage::path('file.jpg');

Before upload file we should validate and check for it

Eloquent: Relationships

1-One-to-One Relationship:

public function profile()

{

return $this->hasOne(Profile::class);

}

public function user()

{

return $this->belongsTo(User::class);

}

You can then retrieve the related record using `$user->profile` or `$profile->user`.

2-One-to-Many Relationship:

public function posts()

{

return $this->hasMany(Post::class);

}

public function user()

{

return $this->belongsTo(User::class);

}

3-Many-to-Many Relationship:

public function roles()

{

return $this->belongsToMany(Role::class);

}

public function users()

{

return $this->belongsToMany(User::class);

}