

Fiche de synthèse : Les routes Laravel

1. Route basique (GET)

```
Route::get('/accueil', function () {  
    return view('accueil');  
});
```

2. Route avec contrôleur

```
Route::get('/users', [UserController::class, 'index']);
```

3. Route avec paramètre

```
Route::get('/user/{id}', function ($id) {  
    return 'Utilisateur n°'.$id;  
});
```

4. Route avec paramètre optionnel

```
Route::get('/user/{name?}', function ($name = 'Invité') {  
    return 'Bonjour '.$name;  
});
```

5. Route avec expression régulière

```
Route::get('/user/{id}', function ($id) {  
    return $id;  
})->where('id', '[0-9]+');
```

6. Nommer une route

```
Route::get('/profil', [UserController::class, 'show'])->name('profil');  
// Utilisation : route('profil')
```

7. Route POST

```
Route::post('/contact', [ContactController::class, 'store']);
```

8. Route PUT/PATCH

```
Route::put('/user/{id}', [UserController::class, 'update']);  
Route::patch('/user/{id}', [UserController::class, 'update']);
```

9. Route DELETE

```
Route::delete('/user/{id}', [UserController::class, 'destroy']);
```

10. Routes avec middleware

```
Route::middleware(['auth'])->group(function () {  
    Route::get('/dashboard', function () {  
        return view('dashboard');  
    });  
});
```

Fiche de synthèse : Les routes Laravel

11. Groupes avec préfixe

```
Route::prefix('admin')->group(function () {  
    Route::get('/users', [AdminController::class, 'users']);  
});
```

12. Routes resources (CRUD)

```
Route::resource('posts', PostController::class);  
// Génère toutes les routes pour index, create, store, show, edit, update, destroy
```

13. Routes API (stateless)

```
Route::middleware('api')->prefix('api')->group(function () {  
    Route::get('/users', [ApiUserController::class, 'index']);  
});
```

14. Routes nommées avec paramètres

```
// Exemple : route('posts.show', ['post' => 5])  
Route::get('/posts/{post}', [PostController::class, 'show']->name('posts.show'));
```

15. Routes de fallback

```
Route::fallback(function () {  
    return response()->view('404', [], 404);  
});
```