



SETUP & CONSOLE

```
composer create-project symfony/skeleton
myapp
composer require symfony/webapp-pack
php bin/console server:start
php bin/console make:controller User
php bin/console
doctrine:migrations:migrate
php bin/console debug:router
```

ROUTING

```
// PHP Attribute style:
#[Route('/users', name: 'user_list',
methods: ['GET'])]
public function index(): Response { ...
}
// YAML (config/routes.yaml):
user_list:
path: /users
controller:
App\Controller\UserController::index
```

CONTROLLERS

```
class UserController extends
AbstractController {
#[Route('/users', methods:['GET'])]
public function index(UserRepository
$repo): Response {
return $this->json($repo->findAll());
}
#[Route('/users/{id}', methods:['GET'])]
public function show(User $user):
Response {
return $this->json($user); //
ParamConverter
}
}
```

TWIG TEMPLATES

```
return
$this->render('user/index.html.twig',
['users'=>$users]);
{{ user.name }} {# comment #}
{% for u in users %}{{ u.name }}{%
endfor %}
{% if user.admin %}Admin{% endif %}
{% extends 'base.html.twig' %}
{% block body %}...{% endblock %}
{{ path('user_list') }}
```

DOCTRINE ORM

```
#[ORM\Entity] #[ORM\Table(name:'users')]
class User {
#[ORM\Id] #[ORM\GeneratedValue]
#[ORM\Column] private ?int $id = null;
#[ORM\Column(length: 180, unique: true)]
private string $email;
}
$em->persist($user); $em->flush();
$repo->find(1) /
$repo->findBy(['active'=>true])
```

SERVICES & DI

```
// Auto-wiring: constructor injection
works automatically
public function __construct(
private readonly UserRepository $repo,
private readonly MailerInterface $mailer
) {}
// config/services.yaml:
services: App\: resource: '../src/'
# Everything in src/ is auto-registered
as a service
```

FORMS

```
$form =
$this->createForm(UserType::class,
$user);
$form->handleRequest($request);
if ($form->isSubmitted() &&
$form->isValid()) {
$em->persist($user); $em->flush();
return
$this->redirectToRoute('user_list');
}
return
$this->render('user/edit.html.twig',
['form'=>$form]);
```

SECURITY

```
// security.yaml:
firewalls:
main:
stateless: true
jwt: ~ # LexikJWTAuthBundle
access_control:
- { path: ^/api/admin, roles: ROLE_ADMIN
}
// In controller:
$this->denyAccessUnlessGranted('ROLE_ADMIN');
```

CONSOLE COMMANDS

```
#[AsCommand(name: 'app:send-emails')]
class SendEmailsCommand extends Command
{
protected function
execute(InputInterface $in,
OutputInterface $out): int {
$out->writeln('Sending...');
return Command::SUCCESS;
}
}
php bin/console app:send-emails
```

COMMON COMMANDS

```
php bin/console debug:container // list
services
php bin/console debug:router // list
routes
php bin/console cache:clear
php bin/console
make:entity/controller/form
php bin/console doctrine:database:create
php bin/console doctrine:migrations:diff
// generate migration
```

COMMON GOTCHAS

Services are private by default
inject, don't fetch from container
Auto-wiring uses type-hints: same
type = one service conflict
Doctrine: flush() is needed to
persist to DB
ParamConverter: {id} param
auto-loads entity by ID
Twig escaping: use |raw only for
trusted HTML