

Installation et configuration du framework Symfony

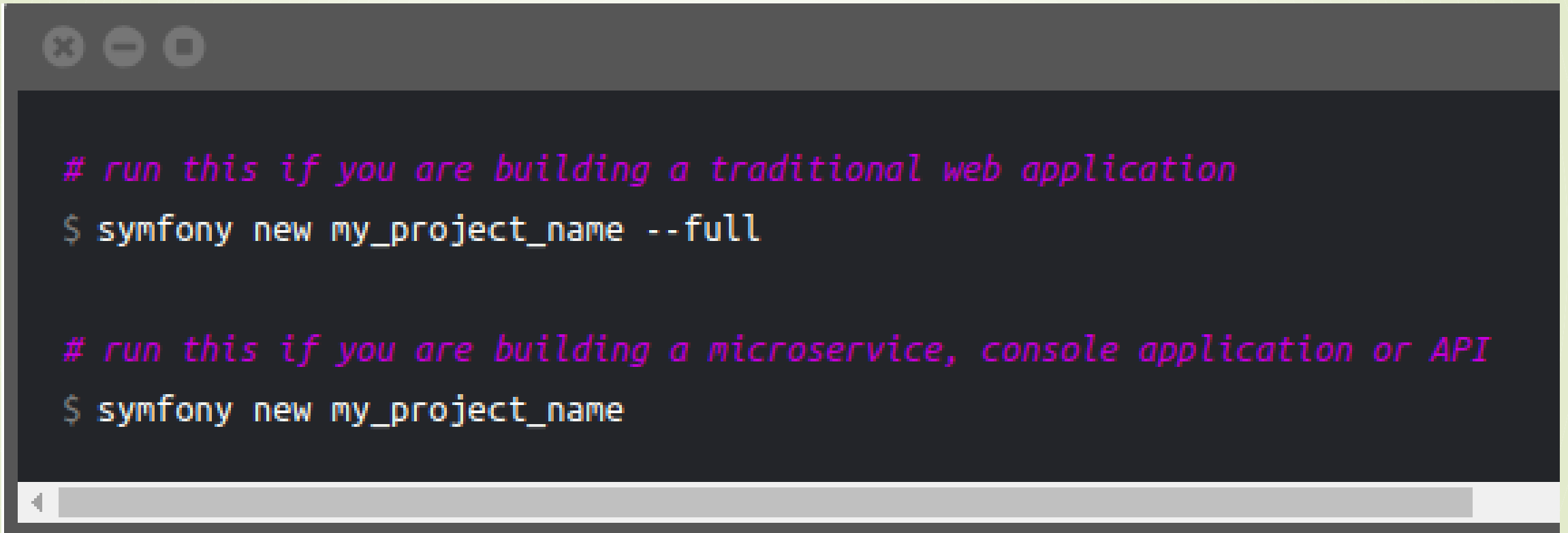
Lotfi KHEDIRI



Exigences techniques

- ❑ Installez **PHP 7.2.5** ou supérieur (en cas problème installer l'extension php-xml)
- ❑ Installez **Composer** : gestionnaire de dépendances php, utilisé pour installer les packages PHP;
- ❑ Installez **l'installateur Symfony** : un binaire appelé "symfony" qui fournit tous les outils dont vous avez besoin pour développer votre application localement.
- ❑ Pour créer une application symfony on 2 alternatives :
 - ❑ En utilisant **l'installateur** (commande) "symfony" ou
 - ❑ En utilisant **composer**
- ❑

Créer une nouvelle application avec l'installateur

A terminal window with a dark background and a light gray title bar. The title bar contains three window control icons (close, minimize, maximize). The terminal displays two code blocks, each preceded by a comment in pink. The first block shows the command to create a full Symfony application. The second block shows the command to create a microservice, console application, or API.

```
# run this if you are building a traditional web application  
$ symfony new my_project_name --full  
  
# run this if you are building a microservice, console application or API  
$ symfony new my_project_name
```

--full : option qui installe tous les packages dont vous avez généralement besoin pour créer des applications Web



Créer la nouvelle application Symfony à l'aide de Composer:

```
# run this if you are building a traditional web application
```

```
$ composer create-project symfony/website-skeleton my_project_name
```

```
# run this if you are building a microservice, console application or API
```

```
$ composer create-project symfony/skeleton my_project_name
```

Exécuter des applications Symfony

- ❑ En mode **production** : on utilise un serveur Web : Nginx ou Apache
- ❑ En mode **développement** : on utilise le *serveur Web local* fourni par Symfony.

- ❑

```
$ cd dossierProjet  
$ symfony serve
```

- ❑

- ❑

- ❑

- ❑ Ouvrez votre navigateur et accédez à <http://localhost:8000/>

- ❑

Configurer un projet Symfony existant

```
# clone the project to download its contents
$ cd projects/
$ git clone ...

# make Composer install the project's dependencies into vendor/
$ cd my-project/
$ composer install
```

personnaliser votre fichier .env et effectuer quelques autres tâches spécifiques au projet

```
$ php bin/console about
```

affiche des informations sur le projet



Installation de packages

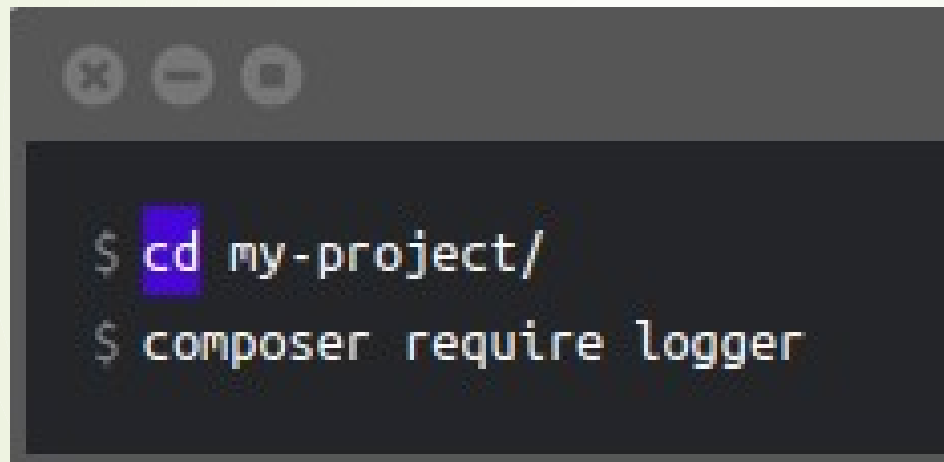
- ▮ Une pratique courante lors du développement d'applications Symfony consiste à installer des **packages** (Symfony les appelle des **bundles**) qui fournissent des fonctionnalités prêtes à l'emploi.
- ▮ Les packages nécessitent généralement une configuration
- ▮ Cette configuration peut être automatisée avec Symfony Flex
- ▮ **Symfony Flex** = plugin Composer qui est installé par défaut lors de la création d'une nouvelle application Symfony
- ▮ **Flex automatise les tâches les plus courantes des applications Symfony**
- ▮ Les packages / bundles Symfony définissent des "**recettes**" , qui sont un ensemble d'instructions automatisées pour installer et activer les packages dans les applications Symfony

▮

Installation de packages :

Exemple

cette commande installe et active tous les packages nécessaires pour utiliser le logger officiel Symfony

A terminal window with a dark background and light gray window controls at the top. It contains two lines of text: the first line is '\$ cd my-project/' with 'cd' highlighted in blue, and the second line is '\$ composer require logger'.

```
$ cd my-project/  
$ composer require logger
```




Symfony Packs

- ▮ Parfois, une seule fonctionnalité nécessite l'installation de **plusieurs packages** et bundles : Les Packs
- ▮ Sont des méta-packages Composer qui incluent plusieurs dépendances
- ▮ Par exemple, pour ajouter des fonctionnalités de débogage dans votre application,
\$ composer require --dev debug
- ▮ Ceci installe le **symfony/debug-pack**, qui à son tour installe
plusieurs paquets comme **symfony/debug** bundle, **symfony/monolog-bundle**, **symfony/var-dumper**, etc.

Versions Symfony LTS

- ❑ les versions de «Long Term Support» (ou **LTS** pour faire court) sont publiées tous les deux ans.
- ❑ Par défaut, la commande "symfony" crée de nouvelles applications Symfony qui utilise la dernière version **stable**.
- ❑ Si on souhaite utiliser une version **LTS**, ajoutez l'option: **--version**

```
# use the most recent LTS version
```

```
$ symfony new my_project_name --version=lts
```

```
# use the 'next' Symfony version to be released (still in development)
```

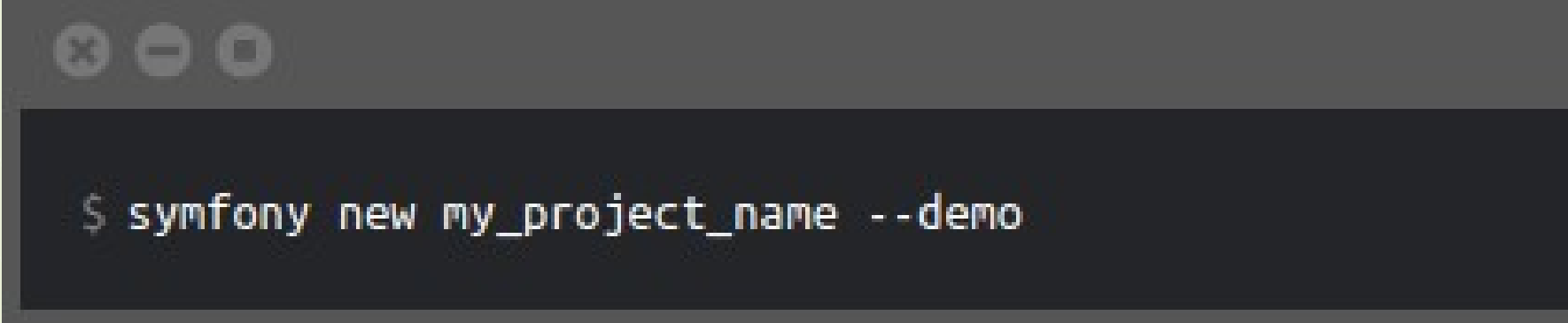
```
$ symfony new my_project_name --version=next
```

```
# you can also select an exact specific Symfony version
```

```
$ symfony new my_project_name --version=4.4
```

L'application Symfony Demo

- ▮ L'application de démonstration Symfony : une application entièrement fonctionnelle qui montre la manière recommandée de développer des applications Symfony.



```
$ symfony new my_project_name --demo
```