

TP1

Composer

Composer est un outil de gestion de dépendances fait en PHP. Il vous permet de déclarer les librairies dont votre projet a besoin dans le fichier **composer.json** et Composer gèrera l'installation et la mise à jour pour vous. Par défaut, Composer n'installe pas les dépendances en global mais dans le dossier **vendor** à la racine de votre projet.

Pour pouvoir initier un projet sous Symfony, utilisez l'image PHP dans docker, avec composer installé pour vous.

Symfony 6

Pour créer un projet [Symfony Website Skeleton](#) via Composer :

```
composer create-project symfony/skeleton:"6.2.*@dev" my_project_directory
```

La **version « Website Skeleton »** comprend le nécessaire pour le développement d'applications web.

Symfony Flex

Symfony Flex est la nouvelle façon d'installer et de gérer les applications Symfony. En effet, ce n'est pas une nouvelle version de Symfony, mais un outil (ou une surcouche) qui remplace et améliore le programme d'installation de Symfony. Symfony Flex peut marcher sur un projet Symfony 3.3 mais devient obligatoire depuis les projets Symfony 4.

En d'autres termes, Symfony Flex est un plugin Composer qui modifie le comportement des commandes require, update et remove.

Lors de l'installation ou de la suppression de dépendances dans une application compatible Flex, Symfony peut effectuer des tâches avant et après l'exécution des tâches Composer.

Lorsque Symfony Flex est installé dans l'application et que vous exécutez composer require, l'application envoie une requête au serveur Symfony Flex avant d'essayer d'installer le paquet avec Composer :

- S'il n'y a aucune information sur ce paquet, le serveur Flex ne renvoie rien et l'installation du paquet suit la procédure habituelle basée sur Composer;
- S'il existe des informations spéciales sur ce package, Flex le renvoie dans un fichier appelé "recette" et l'application l'utilise pour décider du package à installer et des tâches automatisées à exécuter après l'installation.

Symfony Maker

Symfony Maker a été développé dans le but d'aider les développeurs à créer des commandes vides, des contrôleurs, des classes de formulaire, des tests etc afin que les développeurs puissent oublier d'écrire du code standard. Ce bundle est une alternative à SensioGeneratorBundle pour les applications Symfony modernes et nécessite l'utilisation de Symfony 3.4 minimum.

Une simple commande magique permet de l'installer :

```
composer require --dev symfony/maker-bundle
```

Quelques exemples de l'utilisation du MakerBundle :

<i>make:command</i>	Créer une nouvelle classe de commande
<i>make:controller</i>	Créer un nouveau contrôleur
<i>make:entity</i>	Créer une nouvelle entité doctrine

PHP-CS-FIXER

Les normes sont utiles afin que tout le monde puisse facilement lire du code source avec une certaine uniformité. En PHP la norme est le PSR, nous allons donc voir comment utiliser **php-cs-fixer** pour pouvoir rendre votre code conforme (PSR-1 et PSR-2).

<https://github.com/FriendsOfPHP/PHP-CS-Fixer>

Let's go !

Maintenant que l'environnement est installé, nous allons pouvoir démarrer le projet.

Avant de démarrer le projet qui sera votre fil rouge sur l'ensemble des TPs, il vous faut découvrir les concepts principaux.

- ☐ Afin d'utiliser le principe de routing et votre premier contrôleur, afficher un Hello World sur la route **/helloworld**
- ☐ Améliorez l'action de votre contrôleur en passant en paramètre un nom afin d'afficher un *Hello {name}* mis en page via un template Twig.
- ☐ Créez une commande qui écrit sur la sortie standard une liste de personnes stockée dans un fichier yaml.

Les Petites annonces – Partie 1

L'objectif est de concevoir un site de petites annonces : Les Petites Annonces.

Pour cela, vous aurez besoin d'une Entité : Annonce.

Une annonce est composée de :

- Titre
- Contenu
- Un prix
- Un code postal
- Date de création
- Photos (optionnel)

☐ Utilisez le **DoctrineFixturesBundle** pour charger des annonces dans votre base (vous pouvez vous aider de la librairie **Faker**).

☐ Créez une page qui liste l'ensemble des annonces que vous avez en base de données.

☐ Améliorez votre page en prenant en compte le **responsive design** (Bootstrap).