ExoSalariés @ Symfony

# Installation et configuration du projet

## Création de l’arborescence Symfony

C:\Laragon\www\Jean-Michel\Symfony  
> **symfony new ExoSalaries –full**  
\* Creating a new Symfony project with Composer  
 (running C:\ProgramData\ComposerSetup\bin\composer.phar create-project symfony/website-skeleton F:\Laragon\www\Jean-Michel\Symfony\ExoSalaries)  
  
\* Setting up the project under Git version control  
 (running git init F:\Laragon\www\Jean-Michel\Symfony\ExoSalaries)  
  
                                                                                                
 [OK] Your project is now ready in F:\Laragon\www\Jean-Michel\Symfony\ExoSalaries               
                                                                                                
  
  
C:\Laragon\www\Jean-Michel\Symfony  
> **cd ExoSalaries**C:\Laragon\www\Jean-Michel\Symfony\ExoSalaries (master)  
>

## Mise à jour de l’environnement pour lien avec la base de données (fichier “.env”)

# In all environments, the following files are loaded if they exist,  
# the latter taking precedence over the former:  
#  
# \* .env contains default values for the environment variables needed by the app  
# \* .env.local uncommitted file with local overrides  
# \* .env.$APP\_ENV committed environment-specific defaults  
# \* .env.$APP\_ENV.local uncommitted environment-specific overrides  
#  
# Real environment variables win over .env files.  
#  
# DO NOT DEFINE PRODUCTION SECRETS IN THIS FILE NOR IN ANY OTHER COMMITTED FILES.  
#  
# Run "composer dump-env prod" to compile .env files for production use (requires symfony/flex >=1.2).  
# https://symfony.com/doc/current/best\_practices.html#use-environment-variables-for-infrastructure-configuration  
  
###> symfony/framework-bundle ###  
APP\_ENV=dev  
APP\_SECRET=0844b1e281a81a0ca8fe0c3b926912aa  
#TRUSTED\_PROXIES=127.0.0.0/8,10.0.0.0/8,172.16.0.0/12,192.168.0.0/16  
#TRUSTED\_HOSTS='^(localhost|example\.com)$'  
###< symfony/framework-bundle ###  
  
###> doctrine/doctrine-bundle ###  
# Format described at https://www.doctrine-project.org/projects/doctrine-dbal/en/latest/reference/configuration.html#connecting-using-a-url  
# For an SQLite database, use: "sqlite:///%kernel.project\_dir%/var/data.db"  
# For a PostgreSQL database, use: "postgresql://db\_user:db\_password@127.0.0.1:5432/db\_name?serverVersion=11&charset=utf8"  
# IMPORTANT: You MUST configure your server version, either here or in config/packages/doctrine.yaml  
DATABASE\_URL=**mysql://root:@127.0.0.1:3306/exoSalaries**?serverVersion=5.7  
###< doctrine/doctrine-bundle ###  
  
###> symfony/mailer ###  
# MAILER\_DSN=smtp://localhost  
###< symfony/mailer ###

## Création de la base de données

C:\Laragon\www\Jean-Michel\Symfony\ExoSalaries (master)  
> **php bin/**console **doctrine:database:create**  
Created database `exoSalaries` for connection named default

# Création des entités, des contrôleurs et des repositories associés

## Création des entites

### Entite Salarie

C:\Laragon\www\Jean-Michel\Symfony\ExoSalaries (master)  
> **php bin/console make:entity**  
  
 Class name of the entity to create or update (e.g. OrangeElephant):  
 > Salarie  
Salarie  
  
 created: src/Entity/Salarie.php  
 created: src/Repository/SalarieRepository.php  
  
 Entity generated! Now let's add some fields!  
 You can always add more fields later manually or by re-running this command.  
  
 New property name (press <return> to stop adding fields):  
 > nom  
  
 Field type (enter ? to see all types) [string]:  
 >  
  
  
 Field length [255]:  
 > 50  
  
 Can this field be null in the database (nullable) (yes/no) [no]:  
 >  
  
 updated: src/Entity/Salarie.php  
  
Add another property? Enter the property name (or press <return> to stop adding fields):  
 > prenom  
  
 Field type (enter ? to see all types) [string]:  
 >  
  
  
 Field length [255]:  
 > 20  
  
 Can this field be null in the database (nullable) (yes/no) [no]:  
 >  
  
 updated: src/Entity/Salarie.php  
  
 Add another property? Enter the property name (or press <return> to stop adding fields):  
 > dateNaissance  
  
 Field type (enter ? to see all types) [string]:  
 > date  
date  
  
 Can this field be null in the database (nullable) (yes/no) [no]:  
 >  
  
 updated: src/Entity/Salarie.php  
  
 Add another property? Enter the property name (or press <return> to stop adding fields):  
 > adresse  
  
 Field type (enter ? to see all types) [string]:  
 >  
  
  
 Field length [255]:  
 > 50  
  
 Can this field be null in the database (nullable) (yes/no) [no]:  
 > yes  
  
 updated: src/Entity/Salarie.php  
  
 Add another property? Enter the property name (or press <return> to stop adding fields):  
 > cp  
  
 Field type (enter ? to see all types) [string]:  
 >  
  
  
 Field length [255]:  
 > 10  
  
 Can this field be null in the database (nullable) (yes/no) [no]:  
 > yes  
  
 updated: src/Entity/Salarie.php  
  
 Add another property? Enter the property name (or press <return> to stop adding fields):  
 > ville  
  
 Field type (enter ? to see all types) [string]:  
 >  
  
  
 Field length [255]:  
 > 30  
  
 Can this field be null in the database (nullable) (yes/no) [no]:  
 > yes  
  
 updated: src/Entity/Salarie.php  
  
 Add another property? Enter the property name (or press <return> to stop adding fields):  
 > dateEmbauche  
  
 Field type (enter ? to see all types) [string]:  
 > date  
date  
  
 Can this field be null in the database (nullable) (yes/no) [no]:  
 >  
  
 updated: src/Entity/Salarie.php  
  
 Add another property? Enter the property name (or press <return> to stop adding fields):  
 >  
  
  
              
  Success!    
              
  
 Next: When you're ready, create a migration with make:migration

### Entité Entreprise

C:\Laragon\www\Jean-Michel\Symfony\ExoSalaries (master)  
> **php bin/console make:entity**  
 Class name of the entity to create or update (e.g. GrumpyJellybean):  
 > Entreprise  
Entreprise  
  
 created: src/Entity/Entreprise.php  
 created: src/Repository/EntrepriseRepository.php  
  
 Entity generated! Now let's add some fields!  
 You can always add more fields later manually or by re-running this command.  
  
 New property name (press <return> to stop adding fields):  
 > raisonSociale  
  
 Field type (enter ? to see all types) [string]:  
 >  
  
  
 Field length [255]:  
 > 50  
  
 Can this field be null in the database (nullable) (yes/no) [no]:  
 >  
  
 updated: src/Entity/Entreprise.php  
  
 Add another property? Enter the property name (or press <return> to stop adding fields):  
 > siret  
  
 Field type (enter ? to see all types) [string]:  
 >  
  
  
 Field length [255]:  
 > 20  
  
 Can this field be null in the database (nullable) (yes/no) [no]:  
 >  
  
 updated: src/Entity/Entreprise.php  
  
 Add another property? Enter the property name (or press <return> to stop adding fields):  
 > adresse  
  
 Field type (enter ? to see all types) [string]:  
 >  
  
  
 Field length [255]:  
 > 50  
  
 Can this field be null in the database (nullable) (yes/no) [no]:  
 > yes  
  
 updated: src/Entity/Entreprise.php  
  
 Add another property? Enter the property name (or press <return> to stop adding fields):  
 > cp  
  
 Field type (enter ? to see all types) [string]:  
 >  
  
  
 Field length [255]:  
 > 10  
  
 Can this field be null in the database (nullable) (yes/no) [no]:  
 > yes  
  
 updated: src/Entity/Entreprise.php  
  
 Add another property? Enter the property name (or press <return> to stop adding fields):  
 > ville  
  
 Field type (enter ? to see all types) [string]:  
 >  
  
  
 Field length [255]:  
 > 30  
  
 Can this field be null in the database (nullable) (yes/no) [no]:  
 > yes  
  
 updated: src/Entity/Entreprise.php  
  
 Add another property? Enter the property name (or press <return> to stop adding fields):  
 > salaries  
  
 Field type (enter ? to see all types) [string]:  
 > ?  
?  
  
Main types  
 \* string  
 \* text  
 \* boolean  
 \* integer (or smallint, bigint)  
 \* float  
  
Relationships / Associations  
 \* relation (a wizard will help you build the relation)  
 \* ManyToOne  
 \* OneToMany  
 \* ManyToMany  
 \* OneToOne  
  
Array/Object Types  
 \* array (or simple\_array)  
 \* json  
 \* object  
 \* binary  
 \* blob  
  
Date/Time Types  
 \* datetime (or datetime\_immutable)  
 \* datetimetz (or datetimetz\_immutable)  
 \* date (or date\_immutable)  
 \* time (or time\_immutable)  
 \* dateinterval  
  
Other Types  
 \* decimal  
 \* guid  
 \* json\_array  
  
  
 Field type (enter ? to see all types) [string]:  
 > OneToMany  
OneToMany  
  
 What class should this entity be related to?:  
 > Salarie  
Salarie  
  
 A new property will also be added to the Salarie class so that you can access and set the related Entreprise object from it.  
  
 New field name inside Salarie [entreprise]:  
 >  
  
 Is the Salarie.entreprise property allowed to be null (nullable)? (yes/no) [yes]:  
 > no  
  
 Do you want to activate orphanRemoval on your relationship?  
 A Salarie is "orphaned" when it is removed from its related Entreprise.  
 e.g. $entreprise->removeSalarie($salarie)  
  
 NOTE: If a Salarie may \*change\* from one Entreprise to another, answer "no".  
  
 Do you want to automatically delete orphaned App\Entity\Salarie objects (orphanRemoval)? (yes/no) [no]:  
 >  
  
 updated: src/Entity/Entreprise.php  
 updated: src/Entity/Salarie.php  
  
 Add another property? Enter the property name (or press <return> to stop adding fields):  
 >  
  
  
              
  Success!    
              
  
 Next: When you're ready, create a migration with make:migration

## Création des contrôleurs

### Controleur Salarie

C:\Laragon\www\Jean-Michel\Symfony\ExoSalaries (master)  
> **php bin/console make:controller**  
  
 Choose a name for your controller class (e.g. GentleJellybeanController):  
 > SalarieController  
  
 created: src/Controller/SalarieController.php  
 created: templates/salarie/index.html.twig  
  
              
  Success!    
              
  
 Next: Open your new controller class and add some pages!

### Controleur Entreprise

C:\Laragon\www\Jean-Michel\Symfony\ExoSalaries (master)  
> **php bin/console make:controller**  
  
 Choose a name for your controller class (e.g. DeliciousPizzaController):  
 > EntrepriseController  
  
 created: src/Controller/EntrepriseController.php  
 created: templates/entreprise/index.html.twig  
  
              
  Success!    
              
  
 Next: Open your new controller class and add some pages!

# Premier lancement de l’application

## Migration de la base de données (⬄ création des tables)

C:\Laragon\www\Jean-Michel\Symfony\ExoSalaries (master)  
> **php bin/console make:migration**  
  
              
  Success!    
              
  
 Next: Review the new migration "src/Migrations/Version20200206103356.php"  
 Then: Run the migration with php bin/console doctrine:migrations:migrate  
 See https://symfony.com/doc/current/bundles/DoctrineMigrationsBundle/index.html  
  
C:\Laragon\www\Jean-Michel\Symfony\ExoSalaries (master)  
> **php bin/console doctrine:migrations:migrate**  
                                                                
                    Application Migrations                      
                                                                
  
  
WARNING! You are about to execute a database migration that could result in schema changes and data loss. Are you sure you wish to continue? (y/n)y  
Migrating up to 20200206103356 from 0  
  
 ++ migrating 20200206103356  
  
 -> CREATE TABLE salarie (id INT AUTO\_INCREMENT NOT NULL, entreprise\_id INT NOT NULL, nom VARCHAR(50) NOT NULL, prenom VARCHAR(20) NOT NULL, date\_naissance DATE NOT NULL, adresse VARCHAR(50) DEFAULT NULL, cp VARCHAR(10) DEFAULT NULL, ville VARCHAR(30) DEFAULT NULL, date\_embauche DATE NOT NULL, INDEX IDX\_828E3A1AA4AEAFEA (entreprise\_id), PRIMARY KEY(id)) DEFAULT CHARACTER SET utf8mb4 COLLATE `utf8mb4\_unicode\_ci` ENGINE = InnoDB  
 -> CREATE TABLE entreprise (id INT AUTO\_INCREMENT NOT NULL, raison\_sociale VARCHAR(50) NOT NULL, siret VARCHAR(20) NOT NULL, adresse VARCHAR(50) DEFAULT NULL, cp VARCHAR(10) DEFAULT NULL, ville VARCHAR(30) DEFAULT NULL, PRIMARY KEY(id)) DEFAULT CHARACTER SET utf8mb4 COLLATE `utf8mb4\_unicode\_ci` ENGINE = InnoDB  
 -> ALTER TABLE salarie ADD CONSTRAINT FK\_828E3A1AA4AEAFEA FOREIGN KEY (entreprise\_id) REFERENCES entreprise (id)  
  
 ++ migrated (took 184ms, used 44M memory)  
  
 ------------------------  
  
 ++ finished in 189.1ms  
 ++ used 44M memory  
 ++ 1 migrations executed  
 ++ 3 sql queries

## Lancement du serveur Symfony

C:\Laragon\www\Jean-Michel\Symfony\ExoSalaries (master)  
> **symfony server:start**                                                                                                
 [WARNING] run "symfony.exe server:ca:install" first if you want to run the web server with     
 TLS support, or use "--no-tls" to avoid this warning                                           
                                                                                                
  
  
Feb 6 19:05:13 |DEBUG| PHP Reloading PHP versions  
Feb 6 19:05:13 |DEBUG| PHP Using PHP version 7.2.27 (from default version in $PATH)  
Feb 6 19:05:14 |INFO | PHP listening path="F:\\Laragon\\bin\\php\\php-7.2.27-Win32-VC15-x64\\php-cgi.exe" php="7.2.27" port=51529  
Feb 6 19:05:14 |DEBUG| PHP started  
                                                                                                
 [OK] Web server listening on http://127.0.0.1:8000 (PHP CGI 7.2.27)                            
                                                                                              