

SITE KNOWLEDGE E-LEARNING SYMFONY

27/06/2025

APPLICATION KNOWLEDGE E-LEARNING

CREATION DU PROJET SYMFONY

L'installation de Symfony :

`symfony new nom_du_projet --webapp`

The screenshot displays a website for "KNOWLEDGE E-LEARNING". The top navigation bar includes links for Accueil, Boutique, Leçons, Panier, Se déconnecter, and Administration. The main content area features a section titled "Produits" (Products) with a grid of course offerings. Each product card includes a thumbnail image of a woman in a kitchen, a category label (Cuisine or Informatique), a course title, a brief description, a price, and a "Ajouter au panier" button.

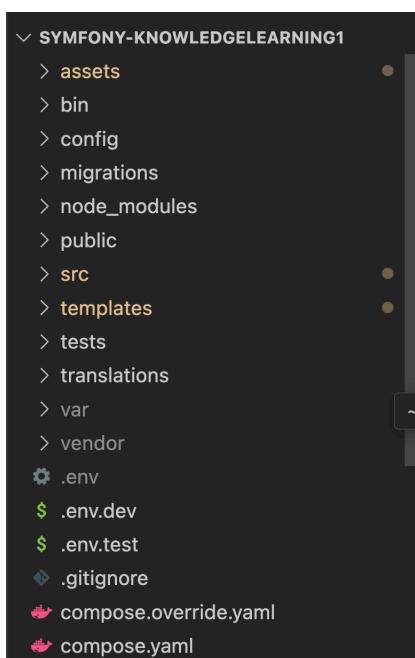
Category	Course Title	Description	Price	Action
Cuisine	Coursus d'initiation à l'art du dressage culinaire	Leçon n°2 : Harmoniser un repas à quatre plats Circa hos dies Lollianus primae lanuginis...	26 €	Ajouter au panier
Cuisine	Coursus d'initiation à l'art du dressage culinaire	Leçon n°1 : Mettre en œuvre le style dans l'assiette Circa hos dies Lollianus primae lanuginis...	26 €	Ajouter au panier
Cuisine	Cursus d'initiation à la cuisine	Leçon n°2 : Les saveurs Circa hos dies Lollianus primae lanuginis...	23 €	Ajouter au panier
Cuisine	Cursus d'initiation à la cuisine	Leçon n°1 : Les modes de cuisson Circa hos dies Lollianus primae lanuginis...	23 €	Ajouter au panier
Informatique	Coursus d'initiation au développement web	127.0.0.1:8000/home/page/product/12/show		
Informatique	Coursus d'initiation au développement web			

La structure de base du projet :

- `src/` : code source de l'application
- `templates/` : vues (Twig)
- `config/` : fichiers de configuration (routes, services...)
- `public/` : point d'entrée (index.php)

Configuration :

- `.env` pour les variables d'environnement (ex: base de données)
- `config/packages/` pour les bundles
- `config/routes.yaml` ou annotations pour les routes



Création de routes, contrôleurs et vues :

php bin/console make:controller NomController

Création des vues dans templates / avec twig

Création d'une entité :

php bin/console make:entity

Création et exécution des migrations :

php bin/console make:migration

php bin/console doctrine:migrations:migrate

Création des formulaires :

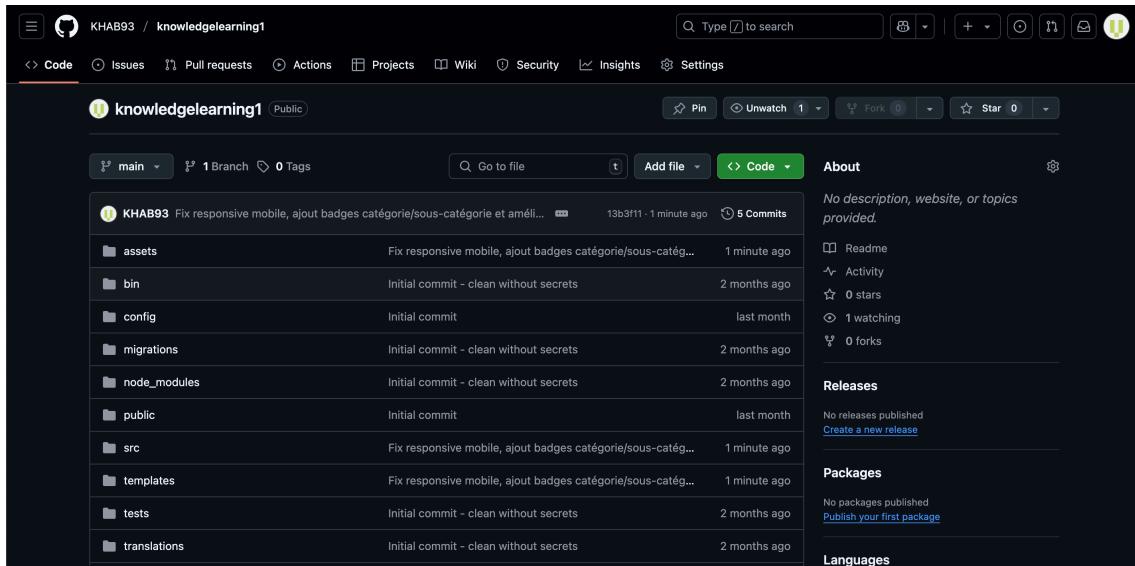
php bin/console make:form

CREATION D'UN DEPOT GIT HUB

Création d'un «New Repository »

Exécution des commandes suivantes :

```
git init  
git add .  
git commit -m "Initial commit"  
git branch -M main  
git remote add origin https://github.com/ton-pseudo/nom-du-depot.git  
git push -u origin main
```



MISE EN PLACE DE LA BASE DE DONNEES MYSQL ADAPTEE

Modification du fichier .env :

DATABASE_URL="mysql://utilisateur:motdepasse@127.0.0.1:3306/nom_de_la_base?serverVersion=8.0"

Remplacement utilisateur, mot de passe et le nom de la base de données.

Création de la base de données :

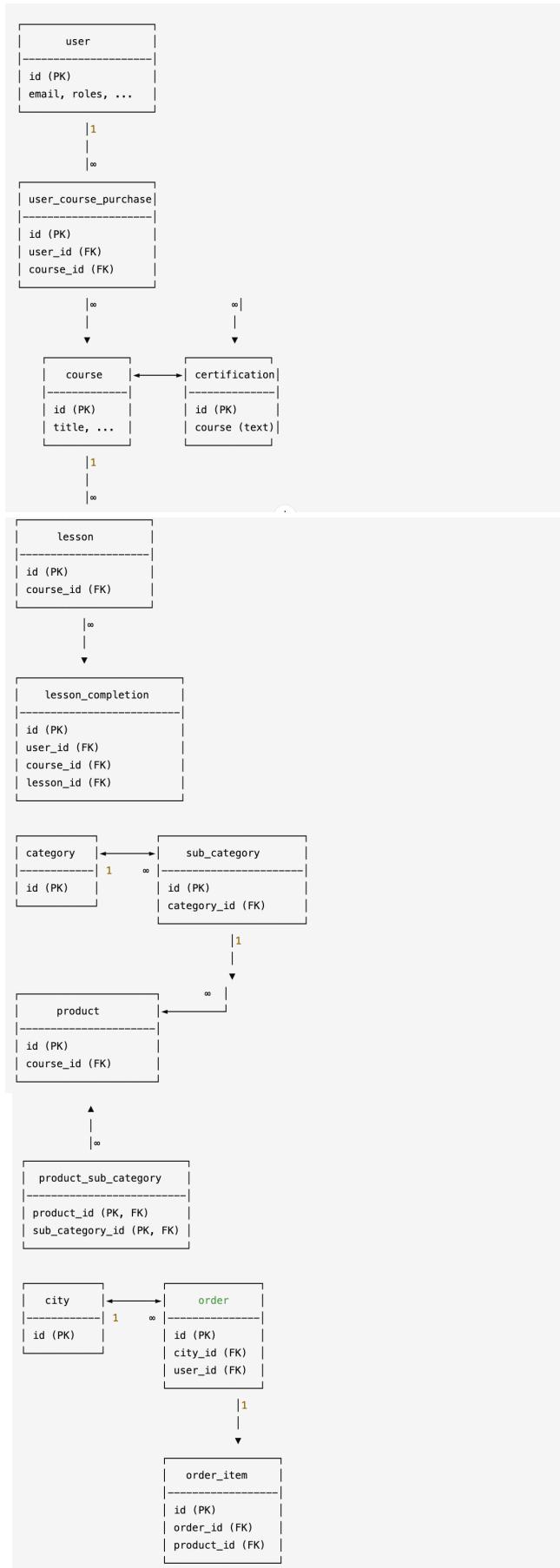
php bin/console doctrine:database:create

Générer les migrations :

php bin/console make:migration

php bin/console doctrine:migrations:migrate

Action	Lignes	Type	Interclassement	Taille	Perte
add_product_history	32,0	InnoDB	utf8mb4_unicode_ci	32,0 KiB	-
category	32,0	InnoDB	utf8mb4_unicode_ci	32,0 KiB	-
city	16,0	InnoDB	utf8mb4_unicode_ci	16,0 KiB	-
course	16,0	InnoDB	utf8mb4_unicode_ci	16,0 KiB	-
course_completion	48,0	InnoDB	utf8mb4_unicode_ci	48,0 KiB	-
lesson	32,0	InnoDB	utf8mb4_unicode_ci	32,0 KiB	-
lesson_completion	64,0	InnoDB	utf8mb4_unicode_ci	64,0 KiB	-
messenger_messages	64,0	InnoDB	utf8mb4_unicode_ci	64,0 KiB	-
order	48,0	InnoDB	utf8mb4_unicode_ci	48,0 KiB	-
order_item	48,0	InnoDB	utf8mb4_unicode_ci	48,0 KiB	-
product	48,0	InnoDB	utf8mb4_unicode_ci	48,0 KiB	-
product_sub_category	48,0	InnoDB	utf8mb4_unicode_ci	48,0 KiB	-
sub_category	32,0	InnoDB	utf8mb4_unicode_ci	32,0 KiB	-
user	32,0	InnoDB	utf8mb4_unicode_ci	32,0 KiB	-
user_course_purchase	48,0	InnoDB	utf8mb4_unicode_ci	48,0 KiB	-
Somm	624,0	InnoDB	utf8mb4_general_ci	624,0 KiB	0 o



Diagrammes de bases de données

Les cardinalités :

TABLE	RELATION	CARDINALITE
user → user_course_purchase	1 user achète plusieurs cours	1 → ∞
user → lesson	1 cours contient plusieurs leçons	1 → ∞
course → product	1 cours = 1 produit	1 → 1
product → order_item	1 produit peut être dans plusieurs commandes	1 → ∞
order → order_item	1 commande contient plusieurs produits	1 → ∞
sub_category → product (via product_sub_category)	N ↔ N	
category → sub_category	1 → ∞	
user → order	1 → ∞	

SYSTEME D'AUTHENTIFICATION

Installer le composant de sécurité :

composer require symfony/security-bundle

Création de l'entité USER

php bin/console make:user

Création de la table en base de données

php bin/console make:migration

php bin/console doctrine:migrations:migrate

Création du système de login

php bin/console make:auth

Configuration de security.yaml

```
access_control:
    - { path: ^/admin, roles: ROLE_ADMIN }
    - { path: ^/editor, roles: ROLE_EDITOR }
```

```
src > Security > SecurityAuthenticator.php
1  <?php
2
3  namespace App\Security;
4
5  use Symfony\Component\HttpFoundation\RedirectResponse;
6  use Symfony\Component\HttpFoundation\Request;
7  use Symfony\Component\HttpFoundation\Response;
8  use Symfony\Component\Routing\Generator\UrlGeneratorInterface;
9  use Symfony\Component\Security\Core\Authentication\Token\TokenInterface;
10 use Symfony\Component\Security\Http\Authenticator\AbstractLoginFormAuthenticator;
11 use Symfony\Component\Security\Http\Authenticator\Passport\Badge\CsrfTokenBadge;
12 use Symfony\Component\Security\Http\Authenticator\Passport\Badge\RememberMeBadge;
13 use Symfony\Component\Security\Http\Authenticator\Passport\Badge\UserBadge;
14 use Symfony\Component\Security\Http\Authenticator\Passport\Credentials\PasswordCredentials;
15 use Symfony\Component\Security\Http\Authenticator\Passport\Passport;
16 use Symfony\Component\Security\Http\SecurityRequestAttributes;
17 use Symfony\Component\Security\Http\Util\TargetPathTrait;
18
19 class SecurityAuthenticator extends AbstractLoginFormAuthenticator
20 {
21     use TargetPathTrait;
22
23     public const LOGIN_ROUTE = 'app_login';
24 }
```

```
src > Security > SecurityAuthenticator.php
19  class SecurityAuthenticator extends AbstractLoginFormAuthenticator
20 {
21     use TargetPathTrait;
22
23     public const LOGIN_ROUTE = 'app_login';
24
25     public function __construct(private UrlGeneratorInterface $urlGenerator)
26     {
27     }
28
29     public function authenticate(Request $request): Passport
30     {
31         $email = $request->getPayload()->getString('email');
32
33         $request->getSession()->set(SecurityRequestAttributes::LAST_USERNAME, $email);
34
35         return new Passport(
36             new UserBadge($email),
37             new PasswordCredentials($request->getPayload()->getString('password')),
38             [
39                 new CsrfTokenBadge('authenticate', $request->getPayload()->getString('_csrf_token')),
40                 new RememberMeBadge(),
41             ],
42         );
43     }
44 }
```

```
src > Security > SecurityAuthenticator.php
20 {
30     {
42     );
43 }
44
45     public function onAuthenticationSuccess(Request $request, TokenInterface $token, string $firewallName)
46     {
47         if ($targetPath = $this->getTargetPath($request->getSession(), $firewallName)) {
48             return new RedirectResponse($targetPath);
49         }
50
51         return new RedirectResponse($this->urlGenerator->generate('app_products'));
52         // For example:
53         // return new RedirectResponse($this->urlGenerator->generate('some_route'));
54         throw new \Exception('TODO: provide a valid redirect inside '.__FILE__);
55
56     }
57 }
58
59     protected function getLoginUrl(Request $request): string
60     {
61         return $this->urlGenerator->generate(self::LOGIN_ROUTE);
62     }
63 }
```

Création d'un utilisateur

Pour un profil administrateur : je lui donne le rôle ADMIN
Pour un profil utilisateur : je lui donne le rôle USER

```
["ROLE_USER", "ROLE_EDITOR", "ROLE_ADMIN"]
```

Insertion dans registration form et registration controller :

- **@Assert\NotBlank** : Vérifie que le mot de passe n'est pas vide.
- **@Assert\Length** : Spécifie la longueur minimale (et potentiellement maximale) du mot de passe.
- **@Assert\Regex** : Utilise une expression régulière pour imposer des règles de format (ici, au moins une lettre et un chiffre)

Ajout dans entity user.php :

```
/**  
 * @var string The hashed password  
 */  
/**  
 * @ORM\Column(type="string", length=255)  
 * @Assert\NotBlank(message="Le mot de passe ne peut pas être vide")  
 * @Assert\Length(  
 *     min=8,  
 *     minMessage="Le mot de passe doit comporter au moins {{ limit }} caractères"  
 * )  
 * @Assert\Regex(  
 *     pattern="/^(?=.*[A-Za-z])(?=.*\d)[A-Za-z\d@$!%*?&]{8,}$/",  
 *     message="Le mot de passe doit contenir au moins une lettre et un chiffre"  
 * )  
 */
```

PAGE D'ACCUEIL

Création du contrôleur :

```
php bin/console make:controller HomeController
```

Modification de la route :

```
#Route('/', name: 'app_home_page', methods: ['GET'])
public function index(Repository $productRepository): Response
{
    $products = $productRepository->findBy([
        'id' => [1, 6, 9, 13]
    ]);
    return $this->render('homePage/index.html.twig', [
        'products' => $products,
    ]);
}
```

Personnalisation du template :

```
3  {% block title %} accueil {% endblock %}
4
5  {% block body %}
6  <div class="container mt-4">
7      <div class="row row-cols-1 row-cols-md-4 g-4">
8          {% for product in products %}
9              <div class="col">
10                  <div class="card h-100">
11                      
16
17
18                  <div class="card-body d-flex flex-column">
19                      <a
20                          href="{{ path('app_home_page_product_show',{'id':product.id}) }}"
21                          class="text-decoration-none text-dark"
22                      >
23                          <h5 class="card-title">{{ product.name }}</h5>
24                      </a>
25                      <p class="card-text">{{ product.description|slice(0,60) ~ '...' }}</p>
26                      <p class="card-text fw-bold">{{ product.price }} €</p>
27
28
29                      <a
30                          href="{{ path('app_cart_new',{'id':product.id}) }}"
31                          class="btn btn-success mt-auto"
32                      >
33                          Ajouter au panier
34                      </a>
```

FORMULAIRE DE CONNEXION

php bin/console make:auth

```
templates > security > login.html.twig
  {% extends 'base.html.twig' %}

  {% block title %}Log in!{% endblock %}

  {% block body %}
    <div class="container security-form">
      <form method="post" class=login-form>
        {% if error %}
          <div class="alert alert-danger">{{ error.messageKey|trans(error.messageData, 'security') }}</div>
        {% endif %}
        {% if app.user %}
          <div class="mb-3">
            | You are logged in as {{ app.user.userIdentifier }}, <a href="{{ path('app_logout') }}>Logout</a>
          </div>
        {% endif %}

        <h1 class="h3 mb-3 font-weight-normal">Se connecter</h1>
        <label for="inputEmail">Email</label>
        <input type="email" value="{{ last_username }}" name="email" id="inputEmail" class="form-control" autocomplete="email" required="required" />
        <label for="inputPassword">Password</label>
        <input type="password" name="password" id="inputPassword" class="form-control" autocomplete="current-password" required="required" />
        <input type="hidden" name="_csrf_token" value="{{ csrf_token('authenticate') }}>

        <div class="checkbox mb-3">
          <label>
            | <input type="checkbox" name="_remember_me"> Remember me
          </label>
        </div>
    </form>
  </div>
```

FORMULAIRE D'INSCRIPTION

php bin/console make:controller RegistrationController

Création d'un formulaire d'inscription

php bin/console make:form RegistrationFormType

```
templates > registration > register.html.twig
  1  {% extends 'base.html.twig' %}

  2
  3  {% block title %}Register{% endblock %}

  4
  5  {% block body %}
  6    {% for flash_error in app.flashes('verify_email_error') %}
  7      <div class="alert alert-danger" role="alert">{{ flash_error }}</div>
  8    {% endfor %}
  9
 10  <div class="container security-form">
 11    <div class="login-form">
 12      <h1>Register</h1>
 13
 14      {{ form_errors(registrationForm) }}
 15
 16      {{ form_start(registrationForm, { method: 'POST' }) }}
 17
 18      {{ form_row(registrationForm.email, {'attr': {'class': 'form-control', 'placeholder': 'Email'}}) }}
 19
 20      {{ form_widget(registrationForm.plainPassword,{ 'attr': {'class': 'form-control', 'placeholder': 'Mot de passe'}}) }}
 21
 22      <button type="submit" class="btn btn-primary mt-3">S'inscrire</button>
 23
 24    {{ form_end(registrationForm) }}
 25
 26  </div>
 27
 28  {% endblock %}
```

PAGE PRODUIT

Création de l'entité produit :

```
php bin/console make:entity Product
```

Création du controller

```
php bin/console make:controller ProductController
```

```
1  {% extends 'base.html.twig' %}  
2  
3  {% block title %}Boutique{% endblock %}  
4  
5  {% block body %}  
6      <div class="container mt-4">  
7          <h1 class="mb-4">Produits</h1>  
8          <div class="row">  
9              {% for product in products %}  
10                 <div class="col-6 col-md-3 mb-4">  
11                     <div class="card h-100">  
12                           
15                         <div class="card-body d-flex flex-column">  
16  
17                             {%# Affichage des catégories et sous-catégories #}  
18                             {% if product.subCategories is not empty %}  
19                                 <p class="mb-2">  
20                                     {% for subCategory in product.subCategories %}  
21                                         <span class="badge bg-info text-dark">  
22                                         {{ subCategory.category.name }}  
23                                         </span>  
24                                         <span class="badge bg-secondary">  
25                                         {{ subCategory.name }}  
26                                     </span>  
27                                 </p>  
28                             {% endif %}  
29                         </div>  
30                     </div>  
31                 </div>  
32             </div>  
33         </div>  
34     </div>  
35 </div>
```

PAGE PANIER

Création d'un fichier service Panier :

Création d'un controller cart :

```
php bin/console make:controller CartController
```

Création d'un template

Ajouter un bouton « Ajouter au panier » sur la page produit

[Ajouter au panier]({{ path('cart_add', { id: product.id }) }})

```

templates > cart > index.html.twig
1  {% extends 'base.html.twig' %}
2
3  {% block title %}Mon Panier{% endblock %}
4
5  {% block body %}
6      <div class="container">
7          <div class="row">
8              <div class="col-8">
9                  <h1>Votre Panier</h1>
10
11                  <table class="table">
12                      <thead>
13                          <tr>
14                              <th>Produit</th>
15                              <th>Quantité</th>
16                              <th>Prix Unitaire</th>
17                              <th>Total</th>
18                              <th>Action</th>
19                          </tr>
20                      </thead>
21                      <tbody>
22                          {% if items is empty %}
23                              <tr>
24                                  <td colspan="5" class="text-center">Votre panier est vide</td>
25                          
```

```

templates > cart > index.html.twig
5  {% block body %}
6      <div class="container">
7          <div class="row">
8              <div class="col-8">
9                  <table class="table">
10
11                      </thead>
12                      <tbody>
13                          {% if items is empty %}
14                              <tr>
15                                  <td colspan="5" class="text-center">Votre panier est vide</td>
16                              </tr>
17                          {% else %}
18                              {% for item in items %}
19                                  <tr>
20                                      <td>{{ item.product.name }}</td>
21                                      <td>{{ item.quantity }}</td>
22                                      <td>{{ item.product.price }} €</td>
23                                      <td>{{ item.product.price * item.quantity }} €</td>
24                                      <td>
25                                          <a href="{{ path('app_cart_product_remove', {'id':item.product.id}) }}>Supprimer</a>
26                                      </td>
27                                  </tr>
28                              {% endfor %}
29                          {% endif %}
30
31
32
33
34
35
36
37
38 
```

```
templates > cart > index.html.twig
 5   {% block body %}
 6     <div class="container">
 7       <div class="row">
 8         <div class="col-8">
 9           <table class="table">
10             <tbody>
11               <tr>
12                 <td>
13                   ...
14                 </td>
15               </tr>
16             {% endfor %}
17             ...
18           {% endif %}
19         </tbody>
20       </table>
21 
22       <div class="total-price">
23         <p>Total: {{ total }} €</p>
24       </div>
25     </div>
26 
27     <div class="col-4">
28       <!-- Bouton ou lien vers la validation de la commande -->
29 
30         <a href="{{ path('app_order') }}" class="btn btn-success">Valider la commande</a>
31 
32     </div>
33 
34   </div>
```



Accueil Boutique Leçons Panier Se déconnecter Administration ▾

Votre Panier

Valider la commande

Produit	Quantité	Prix Unitaire	Total	Action
Leçon n°2 : Les saveurs	1	23 €	23 €	Supprimer le panier

Total: 23 €



Accueil Boutique Leçons Panier Se déconnecter Administration

Paiement

First name

Last name

Phone

Adresse

Ville

Paris

[continuer](#)

Montant à payer:

23 Eur

Veuillez renseigner ce champ.

Knowledgelearning

Adresse : Piccadilly Circus, London W1J 0DA, Royaume-Uni

Email : stubborn@blabla.com

"Don't compromise on your look"

© 2025 Knowledgelearning. Tous droits réservés.

PAGE BACK-OFFICE

Création du rôle admin :

php bin/console make:user

Attribuer le rôle ADMIN :

["ROLE_ADMIN"]

Ajout de l'accès dans security.yaml

- { path: ^/admin, roles: ROLE_ADMIN }

Produits					
Id	Name	Description	Price	actions	
1	Leçon n°1 : Découverte de l'instrument	Circa hos dies Lollianus primae lanuginis adulescens, Lampad ...	26	voir	modifier
2	Leçon n°2 : Les accords et les gammes	Circa hos dies Lollianus primae lanuginis adulescens, Lampad ...	26	voir	modifier
3	Leçon n°1 : Découverte de l'instrument 1	Circa hos dies Lollianus primae lanuginis adulescens, Lampad ...	26	voir	modifier
4	Leçon n°2 : Les accords et les gammes 2	Circa hos dies Lollianus primae lanuginis adulescens, Lampad ...	26	voir	modifier
5	Leçon n°1 : Les langages Html et CSS	Circa hos dies Lollianus primae lanuginis adulescens, Lampad ...	32	voir	modifier
6	Leçon n°2 : Dynamiser votre site avec Javascript	Circa hos dies Lollianus primae lanuginis adulescens, Lampad ...	32	voir	modifier
7	Leçon n°1 : Les outils du jardinier	Circa hos dies Lollianus primae lanuginis adulescens, Lampad ...	16	voir	modifier
8	Leçon n°2 : Jardiner avec la lune	Circa hos dies Lollianus primae lanuginis adulescens, Lampad ...	14	voir	modifier
9	Leçon n°1 : Les modes de cuisson	Circa hos dies Lollianus primae lanuginis adulescens, Lampad ...	23	voir	modifier
10	Leçon n°2 : Les saveurs	Circa hos dies Lollianus primae lanuginis adulescens, Lampad ...	23	voir	modifier
11	Leçon n°1 : Mettre en œuvre le style dans l'assiette	Circa hos dies Lollianus primae lanuginis adulescens, Lampad ...	26	voir	modifier
12	Leçon n°2 : Harmoniser un repas à quatre plats	Circa hos dies Lollianus primae lanuginis adulescens, Lampad ...	26	voir	modifier

[Create new](#)

```
templates > category > update.html.twig
1   {% extends 'base.html.twig' %}
2
3   {% block body%}
4
5       <div class="container">
6
7           <div class="form">
8               <h2>Modifier une categorie</h2>
9               {{ form_start(form) }}
10              {{ form_row(form.name,['attr':{'class': 'form form-control'}]) }}
11
12              <input type='submit' value="Modifier" class="btn-primary">
13              {{ form_end(form) }}
14
15         </div>
16     </div>
17
18   {%endblock%}
```

```
templates > category > new.html.twig
1   {% extends 'base.html.twig' %}
2
3   {% block body%}
4
5       <div class="container">
6
7           <div class="form">
8               <h2>Ajouter une categorie</h2>
9               {{ form_start(form) }}
10              {{ form_row(form.name,['attr':{'class': 'form form-control'}]) }}
11              <input type='submit' value="sauvegarder" class="btn-primary">
12              {{ form_end(form) }}
13
14         </div>
15     </div>
16
17   {%endblock%}
```

```

templates > category > index.html.twig
1  [% extends 'base.html.twig' %]
2
3  {% block title %}categories{% endblock %}
4
5  {% block body %}
6
7  <div class="container"></h1> <br><br>
8  <h1>Catégories</h1> <br><br>
9
10  {% include 'layouts/_flash_message.html.twig' %}
11  <table class="table">
12      <tr>
13          <th>id</th>
14          <th>nom de la categorie</th>
15          <th>actions</th>
16      </tr>
17      {% for category in categories %}
18          <tr>
19              <td>{{ category.id }}</td>
20              <td>{{ category.name }}</td>
21              <td>
22                  <a href="{{ path('app_category_update', {'id':category.id}) }}">Modifier</a>
23                  <a href="{{ path('app_category_delete',{'id':category.id}) }}" class="text-danger">Supp
24                  </td>
25          </tr>

```

```

templates > category > index.html.twig
7  <div class="container"></h1> <br><br>
11  <table class="table">
17      {% for category in categories %}
18          <tr>
19              <td>{{ category.id }}</td>
20              <td>{{ category.name }}</td>
21              <td>
22                  <a href="{{ path('app_category_update', {'id':category.id}) }}">Modifier</a>
23                  <a href="{{ path('app_category_delete',{'id':category.id}) }}" class="text-danger">Supp
24                  </td>
25          </tr>
26          {%else%}
27              <p>Aucune categorie à afficher pour l'instant</p>
28      {% endfor %}
29
30  </table>
31  <br>
32  <a class="btn btn-outline-primary" href="{{ path('app_category_new') }}>Ajouter une categorie</a>
33 </div>
34  {% endblock %}
35
36
37

```



STRIPE

Installer le package stripe

composer require stripe/stripe-php

Ajout de la clé secrète dans le projet .env

Création du contrôleur StripeController.php

php bin/console make:controller StripeController

Création d'un template

```
src > Service >StripePayment.php
1  <?php
2
3  namespace App\Service;
4
5  use Stripe\Stripe;
6  use Stripe\Checkout\Session;
7
8
9  class StripePayment
10 {
11     private $redirectUrl;
12
13     public function __construct(){
14         Stripe::setApiKey($_SERVER['STRIPE_SECRET']);
15         Stripe::setApiVersion('2024-06-20');
16     }
17
18     public function startPayment($cart,$shippingCost){
19
20         $cartProducts = $cart['items'];
21
22         $products = [
23             [
24                 'qte'=> 1,
25                 'price' => $shippingCost.
```

```

src > Service > 💎 StripePayment.php
10  {
18      public function startPayment($cart,$shippingCost){
22          $products = [
23              [
24                  'qte'=> 1,
25                  'price' => $shippingCost,
26                  'name' => "frais de livraison"
27              ]
28          ];
29
30         foreach ($cartProducts as $value){
31             $productItem = [];
32             $productItem['name']= $value['product']->getName();
33             $productItem['price']= $value['product']->getPrice();
34             $productItem['qte']= $value['quantity'];
35             $products[] = $productItem;
36         }
37
38         $session = Session::create([
39             'line_items'=>[
40                 array_map(fn(array $product)=>[
41                     'quantity'=> $product['qte'],
42                     'price_data' => [
43                         'currency'=> 'EUR',
44                         'product_data'=> [

```

```

src > Service > 💎 StripePayment.php
10  {
18      public function startPayment($cart,$shippingCost){
38          $session = Session::create([
39              'line_items'=>[
40                  array_map(fn(array $product)=>[
45                      [
46                          [
47                              [
48                                  [
49                                      [
50                                          [
51                                              [
52                                                  [
53                                                      [
54                                                          [
55                                                              [
56                      ],
57                      [
58                          [
59                              [
60                                  [
61                                      [
62                                          [
63                                              [

```

The screenshot shows a payment summary on the left and a payment form on the right.

Summary:

- Payer Environnement de test STUBBORN
- 23,00 €
- frais de livraison: 0,00 €
- Leçon n°2 : Les saveurs: 23,00 €

Payment Options:

- Payer par
- Informations de livraison
- Courriel:
- Adresse de livraison
- Nom complet: France
- Adresse:
- Saisir l'adresse manuellement
- Moyen de paiement
- Carte (VISA, Mastercard, American Express)
- Bancontact
- Bnp EPS
- Klarna
- Enregistrer mes informations en toute sécurité

Le compte admin d'authentification est le suivant :

Compte administrateur :

Email : lisa@gmail.com

Mot de passe : lisa1993

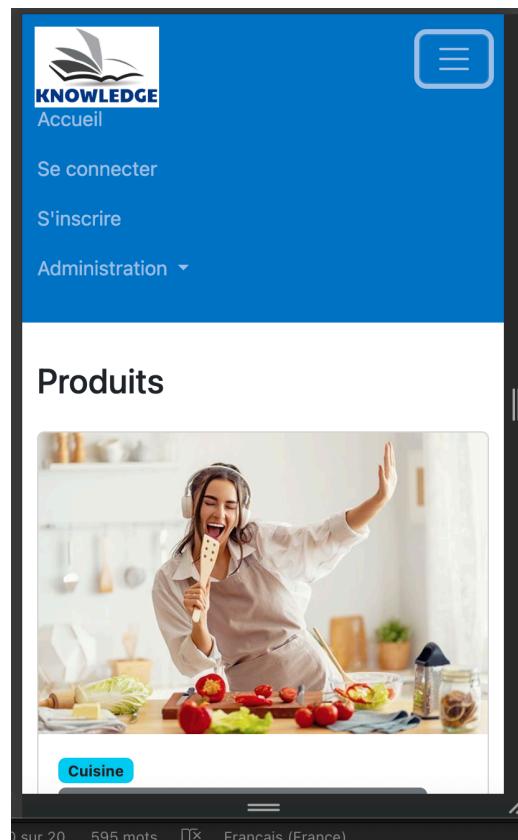
	id	email	roles (DC2Type:json)	password	is_active	activation_token
<input type="checkbox"/>	1	lisa@gmail.com	["ROLE_USER", "ROLE_EDITOR", "ROLE_ADMIN"]	\$2y\$13\$R1gTcl3JLvnByp4b2OpYoiFlfXUa/2x6GT4IREP1...	0	25c0a2ff11a929475bce4cf3b7bcd620fe14ea5293396

SECURITE

Symfony protège automatiquement les formulaires contre les attaques CSRF.

Le système est déjà intégré et activé par défaut.

REPOSITIVE



```
5      /* ===== Mobile seulement ===== */
6  ↘ @media (max-width: 767.98px) {
7      /* Styles pour smartphones */
8  ↘ .row > [class*="col-"] {
9          flex: 0 0 100% !important;
10         max-width: 100% !important;
11     }
12 }
13
14      /* ===== Desktop seulement ===== */
15 ↘ @media (min-width: 992px) {
16      /* Styles pour écrans d'ordinateurs */
17  ↘ .login-form {
18      width: 300px;
19  }
20  ↘ .security-form {
21      flex-direction: row;
22      text-align: left;
23  }
24 }
```