

Atelier 2

Matière : Spring Boot

Classe : GLSI 2

Enseignant : Mohamed Anouar DAHDEH

Création d'une application de boutique en ligne (onlinestore) avec Spring MVC et Thymeleaf

Objectifs

- Acquérir une expérience pratique dans le développement d'une application web à l'aide de Spring MVC et de Thymeleaf.
- Comprendre la syntaxe de base et les fonctionnalités de Thymeleaf pour le rendu de contenu dynamique.
- Découvrir comment configurer des classes de contrôleur et gérer les requêtes HTTP dans un projet Spring MVC.
- S'entraîner à travailler avec des données statiques et à les afficher dans une application Web.

Introduction :

Dans cet atelier, vous allez créer une application de boutique en ligne pour smartphone à l'aide de Spring MVC et de Thymeleaf. L'application permettra aux utilisateurs de parcourir et de consulter une liste statique de smartphones. L'objectif de cet atelier est d'acquérir une expérience pratique avec Spring MVC et Thymeleaf, ainsi que de comprendre les concepts de base de la création d'une application Web.

Travail à faire :

Votre tâche consiste à développer une application de boutique en ligne simple en réalisant les étapes suivantes :

- Créez un nouveau projet Spring MVC nommé « **onlinestore** » à l'aide de votre STS Vs Code.
- Structure du projet :
 - Créez les packages suivants dans le répertoire « src/main/java » :
 - com.example.onlinestore.web.controllers
 - com.example.onlinestore.web.models
- Définissez une classe de modèle « Product » pour représenter un produit avec les attributs suivants :
 - id (Long)
 - code (String)
 - name (String)
 - price (Double)
 - quantity (Integer)
 - image (String)

4. Implémentez la classe Controller :

- Créez une classe de contrôleur « **ProductController** » pour gérer les requêtes HTTP liées à la boutique en ligne.
- Définissez une liste statique de produits. Utilisez l'extrait de code suivant comme point de départ :

```
private static List<Product> products = new ArrayList<Product>();
private static Long idCount = 0L;
static {
    products.add(new Product(++idCount, "SS-S9", "Samsung Galaxy S9", 5000, 50, "samsung-s9.png"));
    products.add(new Product(++idCount, "NK-5P", "Nokia 5.1 Plus", 600, 60, null));
    products.add(new Product(++idCount, "IP-7", "iPhone 7", 6000, 30, "iphone-7.png"));
}
```

- Implémentez une seule action de contrôleur dans la classe « ProductController » qui répondra à deux requêtes : « / » et « /products ».
 - Pour la requête « / », l'action doit récupérer la liste des produits et les afficher à l'aide du modèle Thymeleaf.
 - Pour la requête « /products », l'action doit également récupérer la liste des produits et les afficher à l'aide du modèle Thymeleaf.
5. Créez le modèle Thymeleaf :
- Concevez un modèle de feuille de thymeleaf « **list.html** » pour afficher la liste des produits
 - Utilisez la syntaxe Thymeleaf pour afficher dynamiquement les informations sur le produit dans le modèle « list.html ».
6. Ajoutez du Style CSS:
- Créez un fichier « **style.css** » dans le répertoire approprié ou bien utiliser **Bootstrap** .
 - Appliquez les styles CSS aux modèles Thymeleaf pour créer une interface conviviale.
 - Ajouter une couleur d'arrière-plan
7. Testez l'application
- Exécutez l'application Spring Boot.
 - Accédez à l'application dans un navigateur Web et vérifiez que la liste des produits s'affiche correctement sur les points de terminaison racine (« / ») et « /products ».
 - Assurez-vous que le style du fichier « style.css » est appliqué à la page Web.

Instructions :

- Concentrez-vous sur la compréhension de la syntaxe et des fonctionnalités de base de Thymeleaf, telles que les expressions de variables, l'itération et le rendu conditionnel.
- Suivez le modèle architectural MVC pour séparer les problèmes et maintenir une structure de code propre.
- Utilisez les données statiques fournies pour la liste des produits.
- Si vous choisissez d'utiliser Lombok dans votre projet Spring Boot, assurez-vous d'inclure la dépendance de démarrage Lombok dans votre fichier pom.xml. En outre, vous pouvez installer le plug-in d'annotation Lombok pour Visual Studio Code afin d'améliorer votre expérience de développement avec les méthodes et les annotations générées par Lombok.

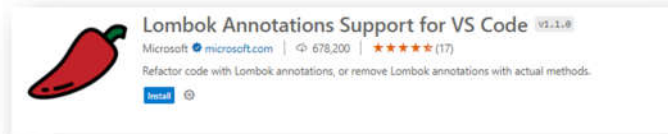


Figure 1: Extension Lombok pour VS Code

- Faites attention à l'expérience utilisateur et assurez-vous que l'application est intuitive et facile à utiliser.
- Assurez-vous d'avoir accès au dossier d'images contenant les images du produit. Contactez votre professeur pour obtenir le dossier d'images s'il ne vous a pas été fourni.
- Le modèle « list.html » doit être utilisé pour les points de terminaison racine (« / ») et « /products ».
- Affichez la liste des produits dans le modèle Thymeleaf en suivant la structure indiquée dans la figure fournie « list.html ».

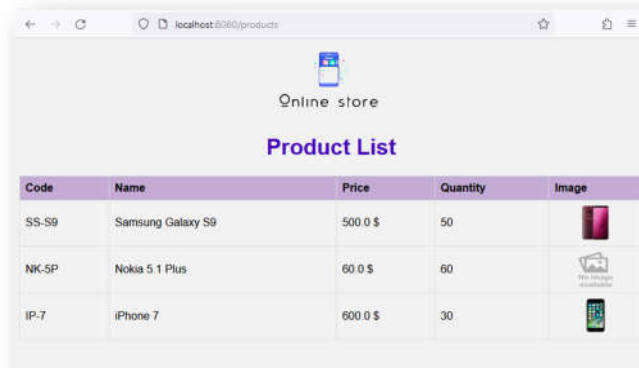


Figure 2: list.html

Indice :

- Définissez la couleur d'arrière-plan de la page Web sur #f2f2f2.
- Définissez la couleur d'arrière-plan de l'en-tête du tableau sur #c5acd5.
- Définissez la couleur de la bordure du tableau sur #ddd.

Bon Travail