***Compte-rendu TD3 Web Dynamique Théorique***

* 1. ***Persistance des données***

Question 1 : Que signifie JPA ? Quel est son intérêt ?

JPA veut dire : **JAVA PERSISTENCE API** qui est une collection de classes et de méthodes pour stocker de manière persistante les grandes quantités de données fournies par Oracle Corporation

* 1. ***Présentation des modèles (dans app /models)***

***Question 2 : Quels sont les deux modèles.***

Nous avons le modèle **Utilisateur** et **Product :** ****

***Question 3 : A quoi sert le ”@Id” dans la classe Product ?***

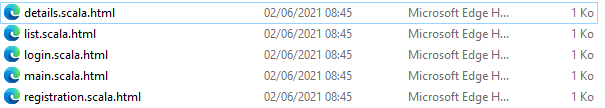
Le « @ID » est une annotation qui marque un champ en tant que champ de clé primaire, lorsqu’un champ de clé primaire est défini, la valeur de la clé primaire est automatiquement injectée dans ce champ par ObjectDB

***Question 4 : A quoi correspond le dossier validators ?***

Le dossier «**validato**r » contient des fichiers EAN :  et ces fichiers sont chargé de vérifier les codes barres, on peut le voir ci-contre , on sait qu’un code barre est composé de 13 chiffres qui sont compris entre 0-9

***1.3 Présentation de la vue (dans app/views)***

***Question 5 : Listez les 5 vues***

On a ces 5 vues : 

***Question 6 : Quelle est l’utilité de details.scala ?***

Il permet de faire appel à tous les fichiers dans différents dossiers et d’en faire une description

***Question 7 : Que fait listing 1 dans list.scala ?***

Le fichier **listing.scala.html** permet de structurer les données avec un code barre contenue dans le fichier

***Question 8 : Que fait listing 2 dans list.scala ?***

Il permet de supprimer un produit en un seul clic

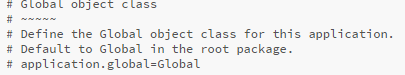
***Question 9 : Quelle est la difference entre login.scala et registration.scala ?***

La différence entre **login.scala** et **registration.scala** est le fait qu’il y a un fichier qui permet de s’inscrire et l’autre de se connecter (en mettant leur nom, @mail et mot de passe)

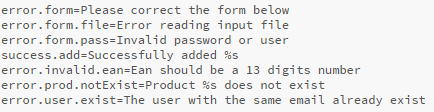
* 1. ***Présentation des contrôleurs***

***Question 10 : Quelles sont les méthodes de la classe Application dans le dossier conf en mentionnant leur utilité ?***

* **Application.conf**: définit les paramètres de base



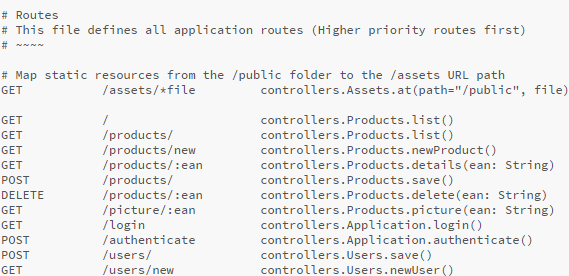
* **Messages**: nous montre les erreurs rencontrées



* **Messages.fr :** nous montre une erreur si on entre un code barre de façon erronée



* **Routes :** nous montre les routes entre l’URL et ce qui est censé être affiché



***Question 11 : Dans Products.java, quel type de structure de données contient les produits ? Quels sont ses avantages et inconvénients par rapport au tableau ? par rapport au vecteur ?***

***Question 12 : Que fait listing 3 dans users.java ?***

Il nous dit que la création d’un email a été un succès et qu’il sera ajouté à la base de données

***Question 13 : Détaillez un schéma UML avec les classes de contrôleur et les modèles.***

Controleur

Modèle

Route

Vue

***1.5 Analyse des routes dans dossier conf***

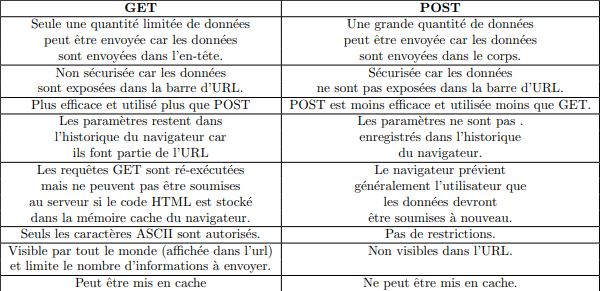
***Question 13 : Quelle est Utilité de /:ean dans listing 4 ?***

**Route.Products.list**

Il permet de donner une route pour **:/ean** pour qu’il puisse vérifier un produit via son code-barre

***Question 14 : Quelles sont les differences entre une route avec GET et celle***

***avec POST ?***



***Question 15 : Que fait cette route dans listing 5 ?***

Cette route permet de supprimer un produit si son code-barre n’est pas valide

***Question 16 : Quelle est l'utilité du fichier "messages" dans conf ?***

Il regroupe tous les messages d’erreurs rencontrées

***Question 17 : A quoi sert le dossier dao dans application ?***

DAO en JAVA est une section qui décrit le modèle pour spécifier l’accessibilité des données à partir de la base de données

***Question 18 : Schématiser les différentes relations entre le contrôleur, la vue et l’utilisateur en utilisant les routes et les actions definies dans ce programme.***

User

**Route.Product.list**

**Route.Application.login**

**Route.Application.login**

Product

Application

Secured