


FACULTÉ DES SCIENCES DE SFAX 	Département d'informatique et des Communications Année Universitaire : 2023-2024
	TP2 : Création de Classes et Gestion de Schémas XML (DTD et XSD) Matière : Architecture SOA et services web Filière : LSI3 Enseignantes : Amel ZAYANI, Amira ECHTIOUI, Roua TORJMEN

Objectifs opérationnels

L'objectif principal de ce TP est de vous permettre de développer une solide compréhension et des compétences pratiques dans la création de classes Java, la correspondance avec des schémas XML (XSD et DTD) pour valider la structure d'un fichier XML, ainsi que dans l'exploration de données XML à l'aide de XPath.

Type de l'activité Individuel

Livrables document Word + Lien du projet sur la plateforme GitHub

Ressources

Consultez le cours Architecture SOA et services web : séance 1.

Préparation de l'espace de travail

Vous êtes invité(e) à :

1. Dans votre espace de travail, créez un répertoire sous le nom **vousreNom_LSI3_SOA-SW_TP2**
2. Créer un document Word qui porte votre **VousreNom_TP2_SOA_SW.docx** dans lequel vous allez insérer les imprimés écrans de chaque étape demandée.

Activité 2-1: Transformation de la classe BanqueService en Java et DTD et XSD

Objectif : Dans cette activité, l'objectif est de transformer la classe BanqueService en une classe Java et en un schéma XSD (XML Schema Definition Language) pour faciliter la conversion de montants entre différentes devises. Cette transformation permettra une meilleure intégration dans une architecture orientée service (SOA).

Travail demandé :

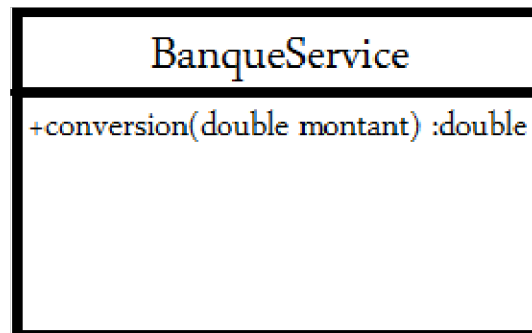
Vous allez reprendre l'exemple vu dans le cours du développement d'application répartie ainsi que dans ce présent cours Architecture SOA qui consiste à faire la conversion d'un montant d'une devise à une autre. Cette conversion est une méthode définie dans la classe **BanqueService**.

Vous êtes demandés à effectuer la transformation de l'entité nommée **BanqueService** en classe ainsi qu'en XSD

Pour cette raison, il faut écrire les deux codes correspondants.

Rq. L'utilité de cette transformation sera vue lorsque vous aller avancer dans le cours.

- 1- Lancez Eclipse et créez un nouveau projet Java dédié à cette activité.
- 2- À l'intérieur du projet, établissez un package nommé **"service"** pour organiser votre code de manière structurée.
- 3- À l'intérieur du package **"service"**, créez une nouvelle classe Java nommée **"BanqueService"** qui contient une méthode publique `double conversion(double montant)` pour effectuer la conversion de devises.



4. Créez également une classe principale (par exemple, "Main") destinée à tester la fonctionnalité de la classe **"BanqueService"**.
5. À présent, créez un package nommé **"xsd"** à la racine du projet pour stocker vos fichiers de définition de structure.
6. À l'intérieur du package "xsd", créez un fichier DTD nommé **"ServiceBanque.dtd"**. Ce fichier DTD sera utilisé pour définir la structure des données de la classe **"BanqueService"**.
7. En outre, dans le même package **"xsd"**, créez un fichier XSD nommé **"ServiceBanque.xsd"**. Ce fichier XSD permettra également de définir plus précisément la structure des données de la classe **"BanqueService"**.

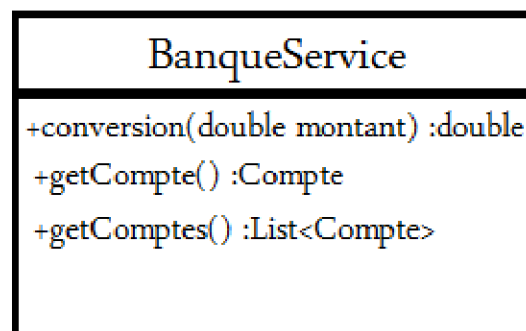
Activité 2-2: Amélioration de la classe BanqueService et de son DTD et son XSD

Objectif : Dans cette activité, l'objectif est d'améliorer la classe **BanqueService** en ajoutant des fonctionnalités et d'améliorer les fichiers DTD et XSD en tenant compte des besoins de l'application.

Travail demandé :

Vous êtes demandés d'améliorer la classe BanqueService ainsi que son XSD :

1. Dans la classe "**BanqueService**", ajoutez les méthodes `Compte getCompte()` et `List<Compte> getComptes()`. Assurez-vous également de créer une nouvelle classe Java nommée "**Compte**" contenant les variables suivantes : **code**, **solde**, **dateCreation**, ainsi qu'un constructeur prenant des paramètres, ainsi que des getters et setters pour ces variables.



2. Dans les fichiers "**ServiceBanque.dtd**" et "**ServiceBanque.xsd**", effectuez les mises à jour nécessaires pour inclure les nouvelles fonctionnalités ou éléments supplémentaires requis par l'application. Assurez-vous que le schéma XSD reflète avec précision les modifications apportées à la classe "**BanqueService**".
3. Mettez à jour la classe principale (par exemple, "Main") afin de tester les nouvelles fonctionnalités ajoutées à la classe "**BanqueService**".
4. Générez un fichier **XML** en utilisant le fichier **XSD** "**ServiceBanque.xsd**" commemodèle.
5. Utilisez des requêtes **XPath** pour interroger le fichier **XML** et extraire les données nécessaires.