

SYSTÈMES DISTRIBUÉS

MASTER INTELLIGENCE
ARTIFICIELLE ET ANALYSE
DES DONNÉES

RÉALISÉE PAR :

ANOADA NOHAYLA

ENCADRÉ PAR :

M. MOHAMED YOUSSEFI



ACTIVITÉ PRATIQUE N°5

WEB SERVICES SOAP WSDL

1. CRÉER UN WEB SERVICE QUI PERMET

1.1 CLASSE COMPTE

représente un compte bancaire avec des attributs pour le code du compte, le solde et la date de création.

```
public class Compte
{
    3 usages
    private int code;
    3 usages
    private double solde;
    3 usages
    private Date dateCreation;
    no usages
    public Compte() {}
    4 usages
    public Compte(int code, double solde, Date dateCreation)
    {
        this.code = code;
        this.solde = solde;
        this.dateCreation = dateCreation;
    }
    no usages
    public int getCode() { return code;}
    no usages
    public void setCode(int code) { this.code = code; }
    no usages
    public double getSolde()
    { return solde; }
    no usages
    public void setSolde(double solde)
    { this.solde = solde; }
    no usages
    public Date getDateCreation()
    { return dateCreation; }
    no usages
    public void setDateCreation(Date dateCreation)
    { this.dateCreation = dateCreation;}
}
```

1.2 CLASSE BANQUESERVICE

une classe de service web qui expose trois opérations: conversion de monnaie, obtention d'un compte par code et liste des comptes.

```
@WebService(serviceName = "BanqueWS")
public class BanqueService
{
    no usages
    @WebMethod(operationName = "conversionEuroToDH")
    public double conversion(@WebParam(name = "montant") double mt)
    { return mt*11.3; }
    no usages
    @WebMethod
    public Compte getCompte(@WebParam(name = "code") int code)
    {
        return new Compte(code, solde: Math.random()*80000, new Date());
    }
    no usages
    @WebMethod
    public List<Compte> listComptes()
    {
        return List.of(
            new Compte( code: 1, solde: Math.random()*80000, new Date()),
            new Compte( code: 2, solde: Math.random()*80000, new Date()),
            new Compte( code: 3, solde: Math.random()*80000, new Date())
        );
    }
}
```

Méthodes:

- **conversion:** Convertit un montant en euros (EUR) en dirhams marocains (MAD) en utilisant un taux fixe de 11.3.
- **getCompte:** Retourne un objet Compte avec un code spécifique. Le solde est généré aléatoirement.
- **listComptes:** Retourne une liste de trois comptes avec des soldes générés aléatoirement.

1.3 CLASSE SERVERJWS

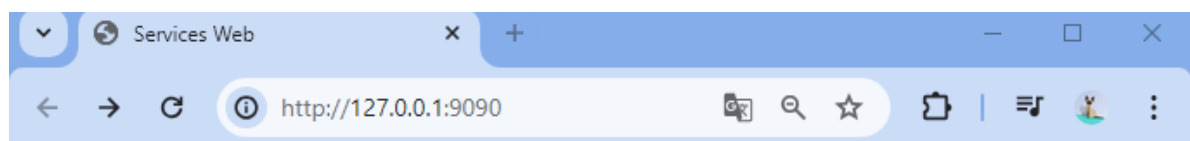
La classe ServerJWS contient le point d'entrée de l'application qui publie le service web.

```
public class ServerJWS
{
    public static void main(String[] args)
    {
        String url = "http://127.0.0.1:9090/";
        Endpoint.publish(url, new BanqueService());
        System.out.println("Web service deployed on " + url);
    }
}
```

Méthode main:

- URL du Service: Définit l'URL où le service web sera déployé (<http://127.0.0.1:9090/>).
- Publication du Service: Utilise `Endpoint.publish` pour publier le service web à l'URL spécifiée.
- Message de Confirmation: Affiche un message dans la console pour confirmer que le service web est déployé.

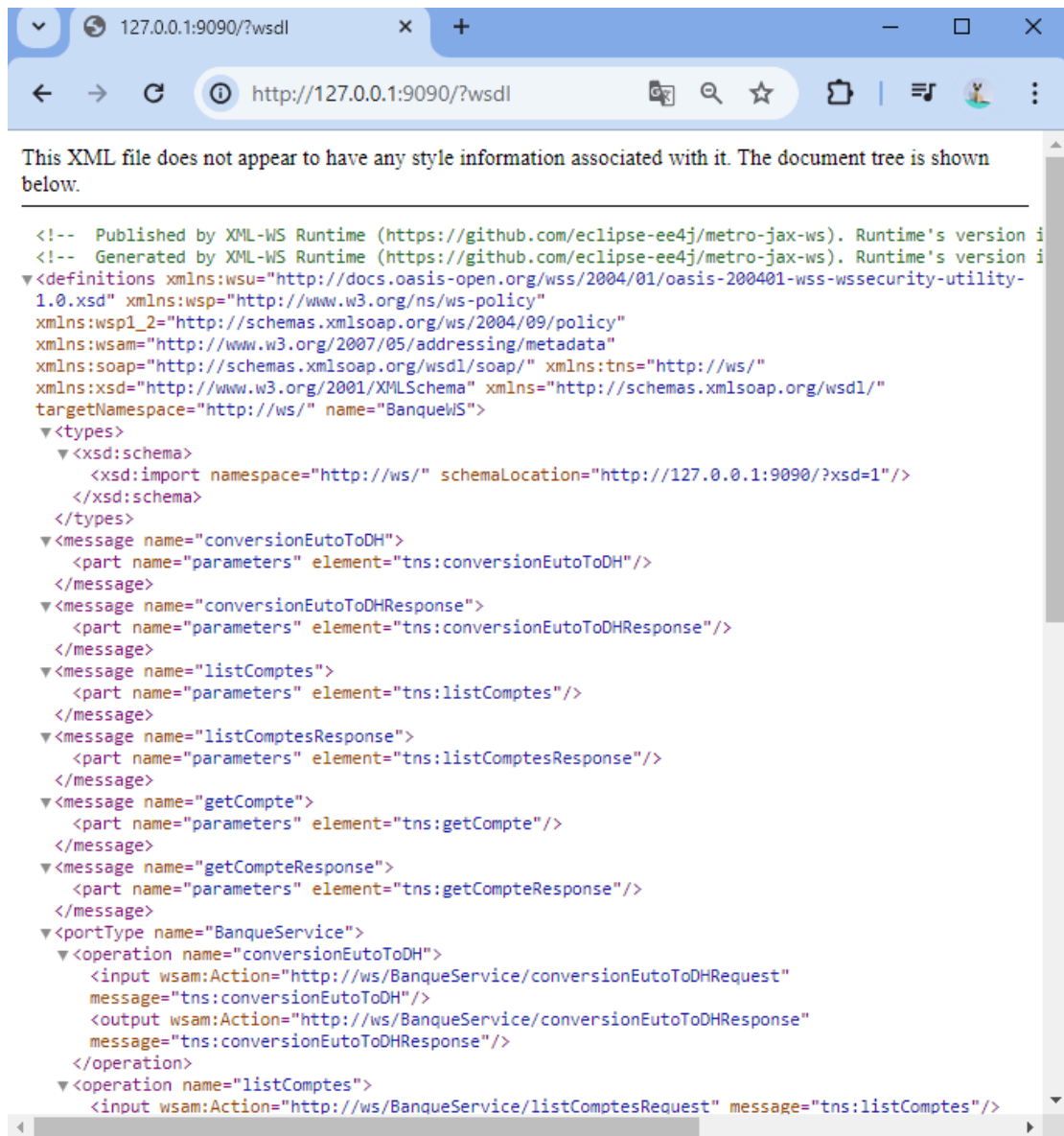
2. CONSULTER ET ANALYSER LE WSDL AVEC UN BROWSER HTTP



Services Web

Adresse	Informations
Nom de service : {http://ws/BanqueWS	Adresse : http://127.0.0.1:9090/
Nom de port : {http://ws/BanqueServicePort	WSDL: http://127.0.0.1:9090/?wsdl
	Classe d'implémentation : ws.BanqueService

cette page fournit les informations essentielles nécessaires pour interagir avec le service web SOAP publié.



This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
<!-- Published by XML-WS Runtime (https://github.com/eclipse-ee4j/metro-jax-ws). Runtime's version i
<!-- Generated by XML-WS Runtime (https://github.com/eclipse-ee4j/metro-jax-ws). Runtime's version i
<definitions xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-
1.0.xsd" xmlns:wsp="http://www.w3.org/ns/ws-policy"
xmlns:wsp1_2="http://schemas.xmlsoap.org/ws/2004/09/policy"
xmlns:wsam="http://www.w3.org/2007/05/addressing/metadata"
xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/" xmlns:tns="http://ws/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns="http://schemas.xmlsoap.org/wsdl/"
targetNamespace="http://ws/" name="BanqueWS">
  <types>
    <xsd:schema>
      <xsd:import namespace="http://ws/" schemaLocation="http://127.0.0.1:9090/?xsd=1"/>
    </xsd:schema>
  </types>
  <message name="conversionEutoToDH">
    <part name="parameters" element="tns:conversionEutoToDH"/>
  </message>
  <message name="conversionEutoToDHResponse">
    <part name="parameters" element="tns:conversionEutoToDHResponse"/>
  </message>
  <message name="listComptes">
    <part name="parameters" element="tns:listComptes"/>
  </message>
  <message name="listComptesResponse">
    <part name="parameters" element="tns:listComptesResponse"/>
  </message>
  <message name="getCompte">
    <part name="parameters" element="tns:getCompte"/>
  </message>
  <message name="getCompteResponse">
    <part name="parameters" element="tns:getCompteResponse"/>
  </message>
  <portType name="BanqueService">
    <operation name="conversionEutoToDH">
      <input wsam:Action="http://ws/BanqueService/conversionEutoToDHRequest"
        message="tns:conversionEutoToDH"/>
      <output wsam:Action="http://ws/BanqueService/conversionEutoToDHResponse"
        message="tns:conversionEutoToDHResponse"/>
    </operation>
    <operation name="listComptes">
      <input wsam:Action="http://ws/BanqueService/listComptesRequest" message="tns:listComptes"/>
```


Le fichier WSDL décrit trois opérations pour le service BanqueService :


- **conversionEutoToDH**: Convertit des euros en dirhams marocains.
- **listComptes**: Liste tous les comptes disponibles.
- **getCompte**: Récupère les détails d'un compte spécifique.

Chaque opération a des messages d'entrée et de sortie, et utilise SOAP sur HTTP pour la communication. Le fichier WSDL fournit les détails nécessaires pour permettre aux clients de générer du code pour interagir avec le service web.

3. TESTER LES OPÉRATIONS AVEC L'OUTIL SOAPUI

3.1 CRÉATION DE PROJET SOAP DANS SOAPUI

 New SOAP Project ✕

New SOAP Project 

Creates a WSDL/SOAP based Project in this workspace


Project Name:

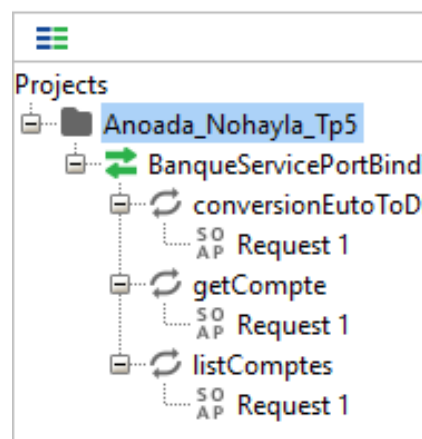
Initial WSDL:

Create Requests: ☒ Create sample requests for all operations?

Create TestSuite: ☐ Creates a TestSuite for the imported WSDL

Relative Paths: ☐ Stores all file paths in project relatively to project file (requires save)





3.1 TESTE DES REQUÊTE

CONVERSIONEUTOTODH

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Header/>
  <soapenv:Body>
    <ws:conversionEutoToDH>
      <montant>50</montant>
    </ws:conversionEutoToDH>
  </soapenv:Body>
</soapenv:Envelope>
```

Raw XML

```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Body>
    <ns2:conversionEutoToDHResponse xmlns:ns2="http://ws/">
      <return>565.0</return>
    </ns2:conversionEutoToDHResponse>
  </S:Body>
</S:Envelope>
```

LISTCOMPTES

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Header/>
  <soapenv:Body>
    <ws:listComptes/>
  </soapenv:Body>
</soapenv:Envelope>

<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Body>
    <ns2:listComptesResponse xmlns:ns2="http://ws/">
      <return>
        <code>1</code>
        <dateCreation>2024-06-12T00:53:32.455+01:00</dateCreation>
        <solde>48429.42181609812</solde>
      </return>
      <return>
        <code>2</code>
        <dateCreation>2024-06-12T00:53:32.455+01:00</dateCreation>
        <solde>77470.70455923793</solde>
      </return>
      <return>
        <code>3</code>
        <dateCreation>2024-06-12T00:53:32.455+01:00</dateCreation>
        <solde>399.4162737669615</solde>
      </return>
    </ns2:listComptesResponse>
  </S:Body>
</S:Envelope>
```

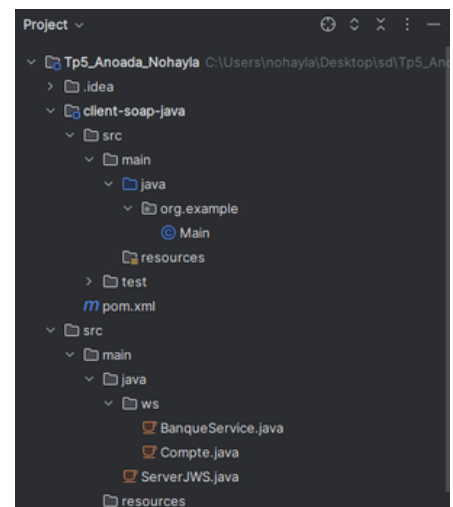
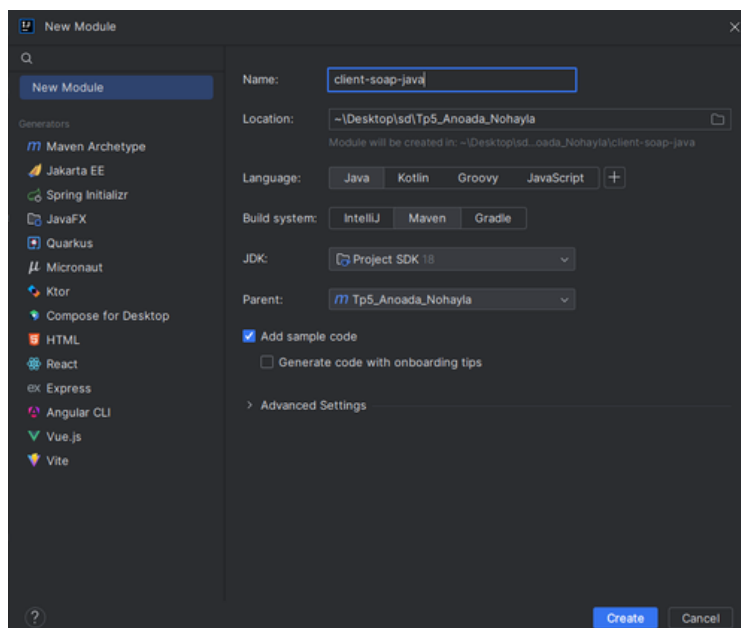
GETCOMPTE

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Header/>
  <soapenv:Body>
    <ws:getCompte>
      <code>3</code>
    </ws:getCompte>
  </soapenv:Body>
</soapenv:Envelope>

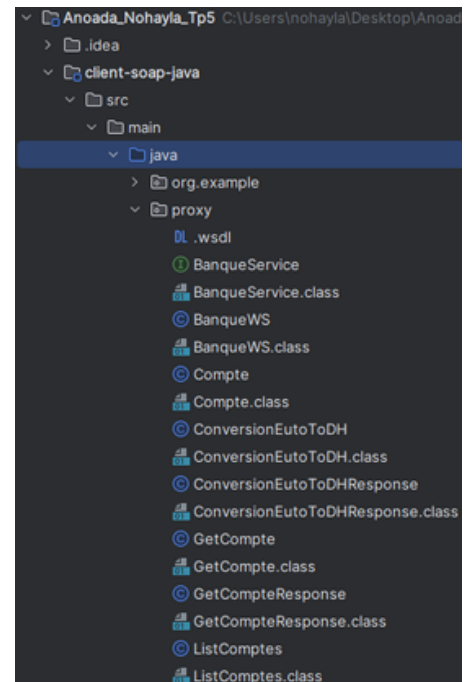
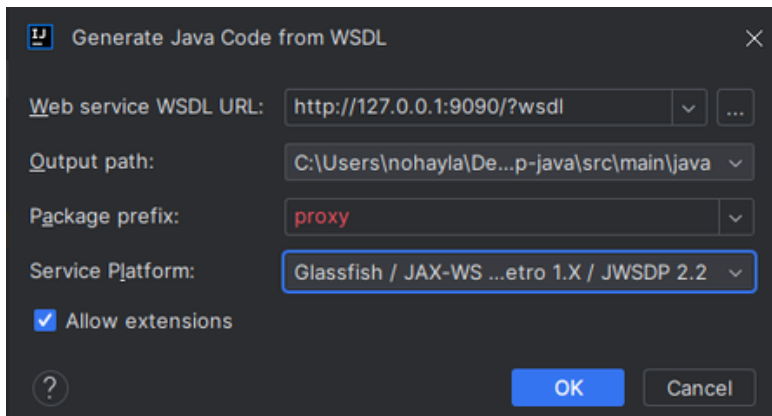
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Body>
    <ns2:getCompteResponse xmlns:ns2="http://ws/">
      <return>
        <code>3</code>
        <dateCreation>2024-06-12T00:52:18.472+01:00</dateCreation>
        <solde>77161.35975125541</solde>
      </return>
    </ns2:getCompteResponse>
  </S:Body>
</S:Envelope>
```

4. CRÉER UN CLIENT SOAP JAVA

création de nouveau module dans un projet



Générer code java à partir du WSDL



Création de client SOAP pour le web service

```
public class Main
{
    public static void main(String[] args)
    {
        BanqueService proxy = new BanqueWS().getBanqueServicePort();
        System.out.println(proxy.conversionEutoToDH( montant: 90.0));
        System.out.println("-----");
        Compte compte = proxy.getCompte( code: 4);
        System.out.println(compte.getCode());
        System.out.println(compte.getSolde());
        System.out.println(compte.getDateCreation());
        System.out.println("-----");
        proxy.listComptes().forEach(cp -> {
            System.out.println("-----");
            System.out.println(cp.getCode());
            System.out.println(cp.getSolde());
            System.out.println(cp.getDateCreation());
        });
    }
}
```



```
"C:\Program Files\Java\jdk-18.0.1.1\bin\java.exe" ...  
1017.0000000000001  
-----  
4  
9964.6983081921  
2024-06-14T14:48:32.854+01:00  
-----  
-----  
1  
1430.943646646483  
2024-06-14T14:48:32.963+01:00  
-----  
2  
35213.60102234271  
2024-06-14T14:48:32.963+01:00  
-----  
3  
75385.61325992091  
2024-06-14T14:48:32.963+01:00  
  
Process finished with exit code 0
```