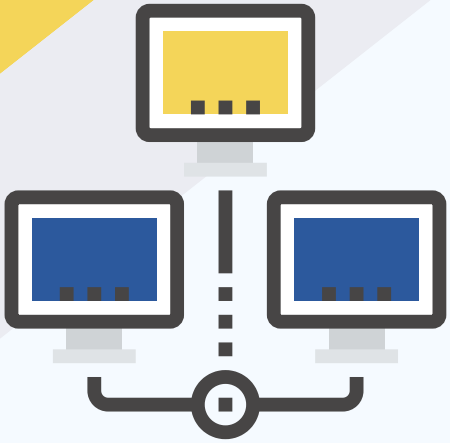
2023



TP 2 REPORT

YOUNESS ZYADI MIAAD

@WEBSERVICE "BANQUEWS"

IMPLÉMENTATION DU WEB SERVICE JAXWS

BanqueService.java - ws_miaad - Visual Studio Code

src > main > java > ws > BanqueService.java > {} ws

```java
package ws;

import jakarta.jws.WebMethod;
import jakarta.jws.WebParam;
import jakarta.jws.WebService;

import java.util.Date;
import java.util.List;
// POJO Plain Simple Java Object : Simple Java Class
@WebService(serviceName = "BanqueWS")
public class BanqueService {
    @WebMethod(operationName = "Convert")
    public double conversion(@WebParam(name = "montant") double mt){
        return mt*11.10;
    }
    @WebMethod
    public Compte getCompte(@WebParam(name = "code")int code){
        return new Compte(code,Math.random()*9888,new Date());
    }
    @WebMethod
    public List<Compte> listComptes(){
        return List.of(
                new Compte(1,Math.random()*9888,new Date()),
                new Compte(2,Math.random()*9888,new Date()),
                new Compte(3,Math.random()*9888,new Date())
        );
    }
}
```

WEB-SERVICE

Compte.java - ws_miaad - Visual Studio Code

src > main > java > ws > Compte.java > Compte > datecreation

```java
package ws;

import jakarta.xml.bind.annotation.XmlAccessType;
import jakarta.xml.bind.annotation.XmlAccessorType;
import jakarta.xml.bind.annotation.XmlRootElement;
import jakarta.xml.bind.annotation.XmlTransient;

import java.util.Date;
@XmlRootElement(name = "compte")
@XmlAccessorType(XmlAccessType.FIELD)
public class Compte {
    private int code;
    private double solde;
    @XmlTransient
    private Date datecreation;//ignore datecreation

    public Compte(int code, double solde, Date datecreation) {
        this.code = code;
        this.solde = solde;
        this.datecreation = datecreation;
    }

    public Compte() {
    }
```

COMPTE CLASS

DEPLOY WS ON SIMPLE SERVER JAX WS :

Go Run Terminal Help ServerJWS.java - ws_miaad - Visual Studio Code

Compte.java 1 | ServerJWS.java 1 ×

src > main > java > ServerJWS.java > ...

```java
import jakarta.xml.ws.Endpoint;
import ws.BanqueService;

public class ServerJWS {
    public static void main(String[] args) {
        Endpoint.publish("http://0.0.0.0:9191/", new BanqueService());
        System.out.println("Web Service Deployed on : http://0.0.0.0:9191");
    }
}
```

WSDL :

localhost:9191/BanqueWS?wsdl

localhost:9191/BanqueWS?wsdl

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 is XML-WS Runtime 4.0.0 git-revision#129f787. --
<!-- Generated by XML-WS Runtime (https://github.com/eclipse-ee4j/metro-jax-ws). Runtime's version is XML-WS Runtime 4.0.0 git-revision#129f787. --
<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:wsam="http://www.w3.org/2007/05/addressing/metadata" xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/" xmlns:tns="http://ws/" xmlns:xsd="http:
 name="BanqueWS">
  <types>
    <xsd:schema>
      <xsd:import namespace="http://ws/" schemaLocation="http://localhost:9191/?xsd=1"/>
    </xsd:schema>
  </types>
  <message name="Convert">
    <part name="parameters" element="tns:Convert"/>
  </message>
  <message name="ConvertResponse">
    <part name="parameters" element="tns:ConvertResponse"/>
  </message>
  <message name="getCompte">
    <part name="parameters" element="tns:getCompte"/>
  </message>
  <message name="getCompteResponse">
    <part name="parameters" element="tns:getCompteResponse"/>
  </message>
  <message name="listComptes">
    <part name="parameters" element="tns:listComptes"/>
  </message>
  <message name="listComptesResponse">
    <part name="parameters" element="tns:listComptesResponse"/>
  </message>
  <portType name="BanqueService">
    <operation name="Convert">
      <input wsam:Action="http://ws/BanqueService/ConvertRequest" message="tns:Convert"/>
      <output wsam:Action="http://ws/BanqueService/ConvertResponse" message="tns:ConvertResponse"/>
    </operation>
    <operation name="getCompte">
      <input wsam:Action="http://ws/BanqueService/getCompteRequest" message="tns:getCompte"/>
      <output wsam:Action="http://ws/BanqueService/getCompteResponse" message="tns:getCompteResponse"/>
    </operation>
    <operation name="listComptes">
      <input wsam:Action="http://ws/BanqueService/listComptesRequest" message="tns:listComptes"/>
      <output wsam:Action="http://ws/BanqueService/listComptesResponse" message="tns:listComptesResponse"/>
    </operation>
  </portType>
  <binding name="BanqueServicePortBinding" type="tns:BanqueService">
```

TEST WEB SERVICE USING SOAPUI

TEST CONVERT METHOD

SOAP REQUEST :

SoapUI 5.7.0

File Project Suite Case Step Tools Desktop Help

Empty SOAP REST Import Save All Forum Trial Preferences Proxy Endpoint Explorer

Navigator

Projects
- Test1
  - BanqueServicePortBinding
    - Convert
      - Request 1
    - getCompte
      - Request 1
    - listComptes
      - Request 1

SoapUI Start Page

Request 1

http://localhost:9191/

XML Raw

```
<soapenv:Envelope xmlns:soapenv="http://schemas.x
    <soapenv:Header/>
    <soapenv:Body>
        <ws:Convert>
            <montant>100</montant>
        </ws:Convert>
    </soapenv:Body>
</soapenv:Envelope>
```

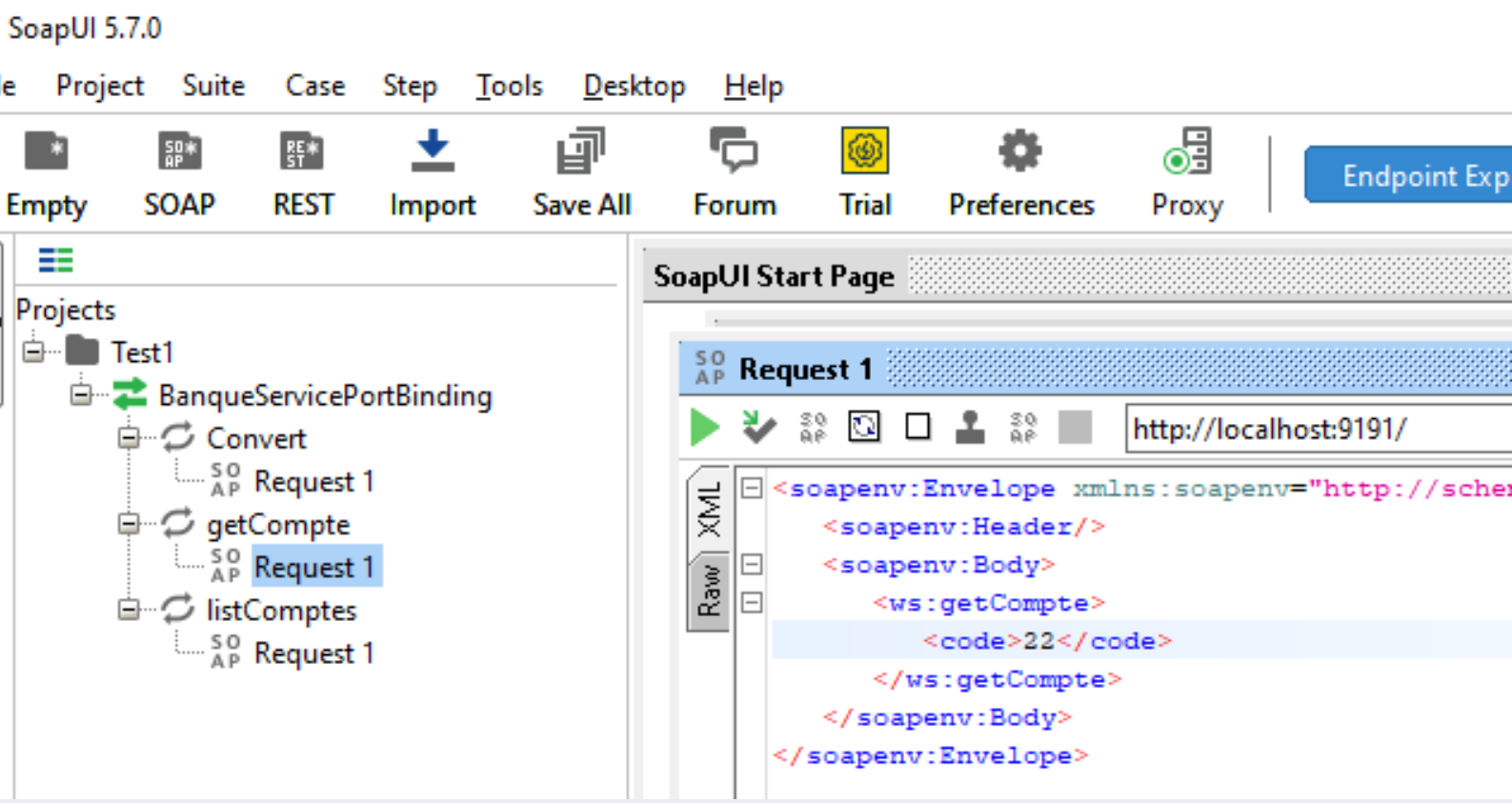
SOAP RESPONSE :

XML Raw

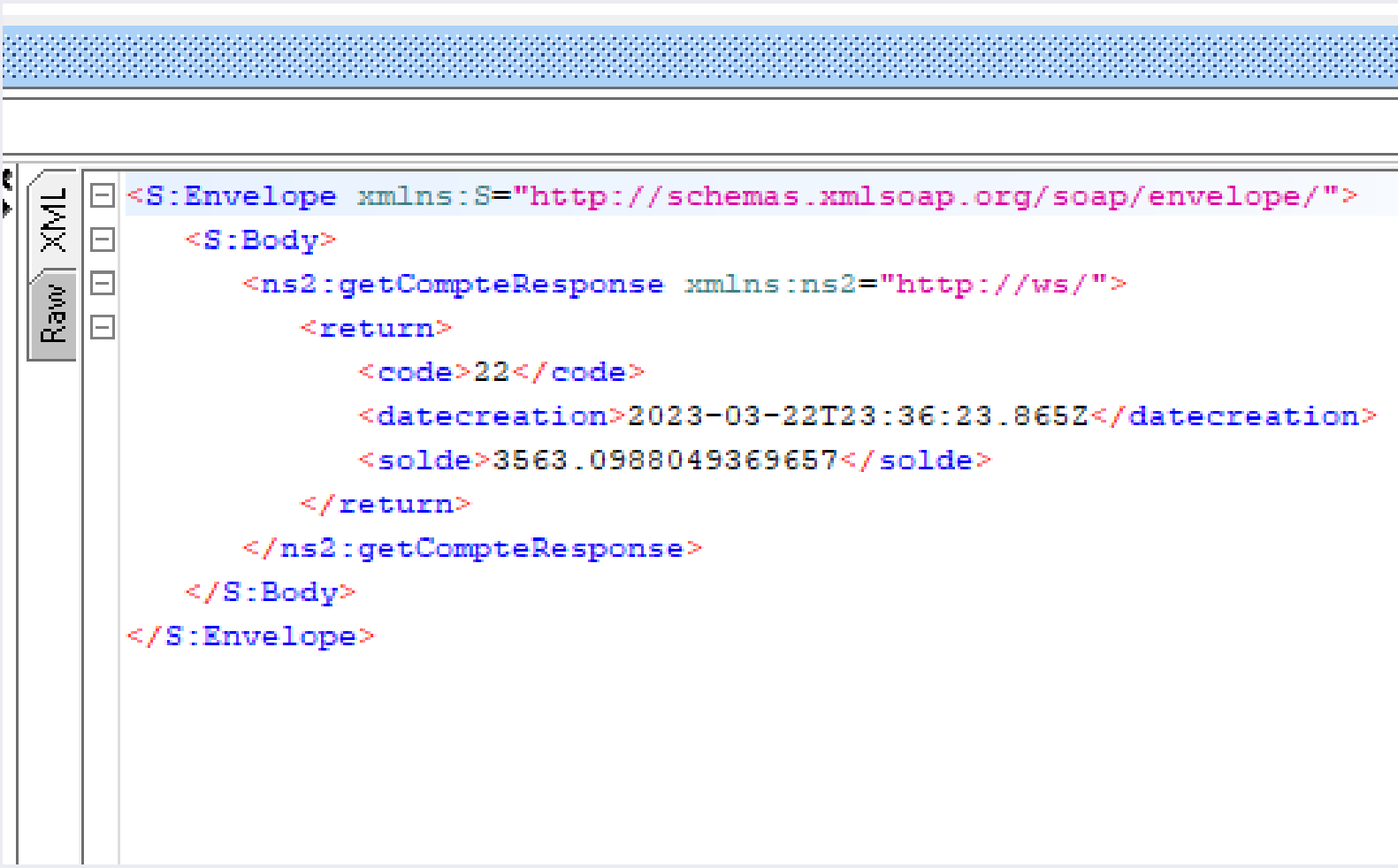
```
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/enve
    <S:Body>
        <ns2:ConvertResponse xmlns:ns2="http://ws/">
            <return>1110.0</return>
        </ns2:ConvertResponse>
    </S:Body>
</S:Envelope>
```

TEST GETCOMPTE METHOD

SOAP REQUEST :

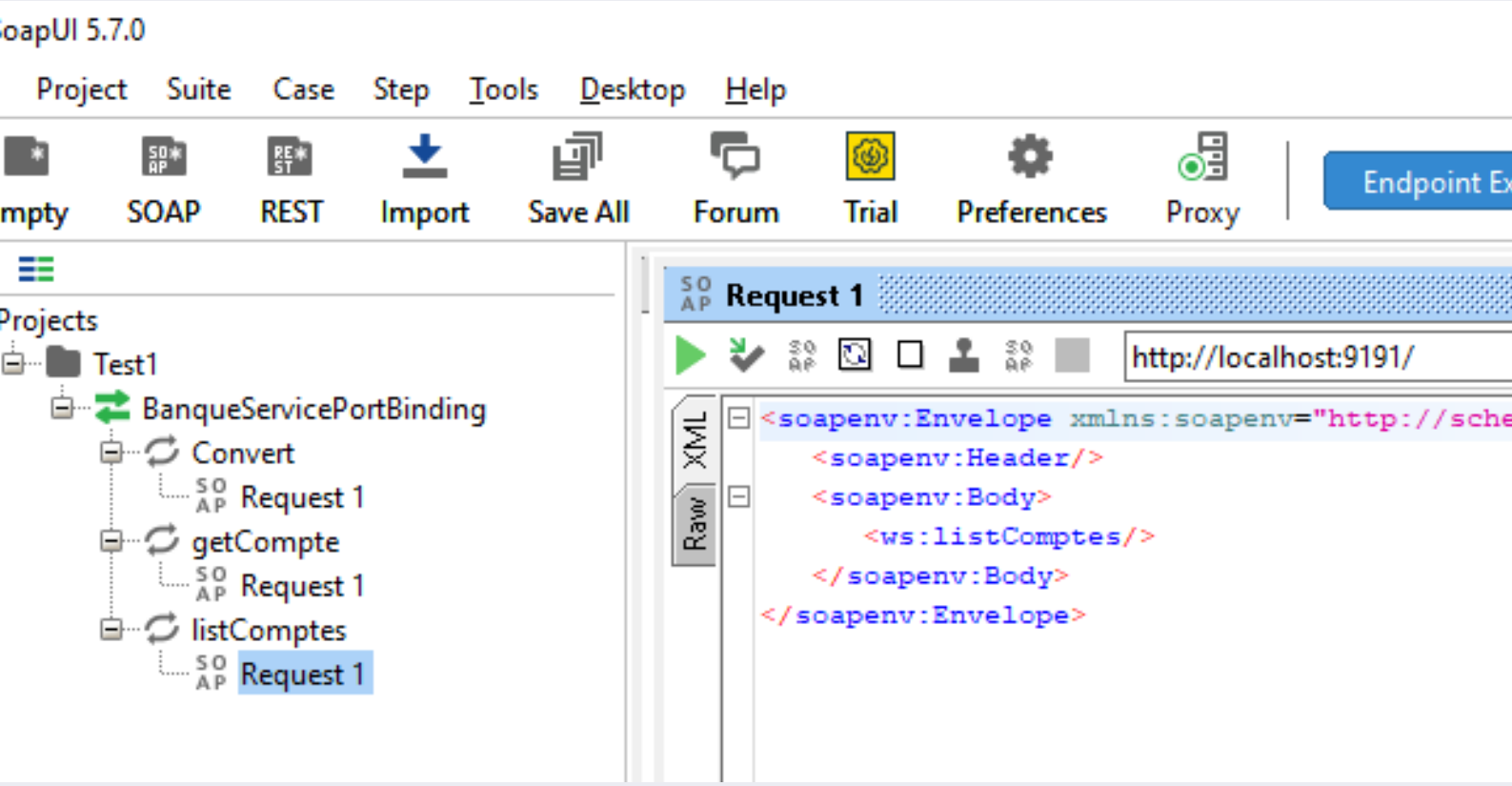


SOAP RESPONSE :



TEST LISTCOMPTES METHOD

SOAP REQUEST :



SOAP RESPONSE :

CLIENT SOAP JAVA

CODE :

File Edit Selection View Go Run Terminal Help — Client.java - Client_WS - Visual Studio Code

EXPLORER

OPEN EDITORS
- Client.java src\m...

CLIENT_WS
- .idea
- src
  - main
    - java
      - proxy
      - Client.java
    - resources
  - test
- target
- pom.xml

Client.java

src > main > java > Client.java > Client > main(String[])

```java
import proxy.BanqueService;
import proxy.BanqueWS;
import proxy.Compte;

import java.util.List;

public class Client {
    Run | Debug
    public static void main(String[] args) {
        BanqueService stub = new BanqueWS().getBanqueServicePort();
        System.out.println(stub.convert(montant:1999));
        Compte cp = stub.getCompte(code:22);
        System.out.println("The Code of your account is :" + cp.getCode());
        System.out.println("The Solde of your account is :" + cp.getSolde());
        System.out.println("Liste des comptes");
        List<Compte> cptes = stub.listComptes();
        for (Compte c : cptes) {
            System.out.println(c.getCode() + "----" + c.getSolde());
        }
    }
}
```

RESULT :

OUTPUT DEBUG CONSOLE TERMINAL — Run: Client

```
PS C:\Users\youne\Desktop\Master\S2\System Distributed\TPs\Client_WS>  & 'C:\Users\youne\.vscode\extensions\redhat.java-1.16.0-win32-x64\jre\17.0.6-win32-x86_64\bin\java.exe' '@C:\Users\youne\AppData\Local\Temp\cp_3kmz4eobapund2ns7zjos0t6j.argfile' 'Client'
22188.899999999998
The Code of your account is :22
The Solde of your account is :6575.1042366736365
Liste des comptes
1----9352.732185679513
2----2666.565711554437
3----2667.923758285237
PS C:\Users\youne\Desktop\Master\S2\System Distributed\TPs\Client_WS>
```

Ln 17, Col 41 Spaces: 4 UTF-8 CRLF {} Java Go Live Prettier

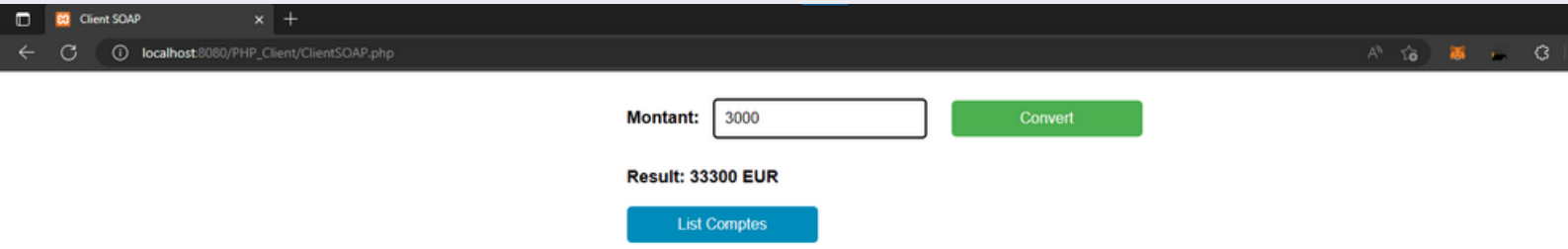
Java

CLIENT SOAP PHP

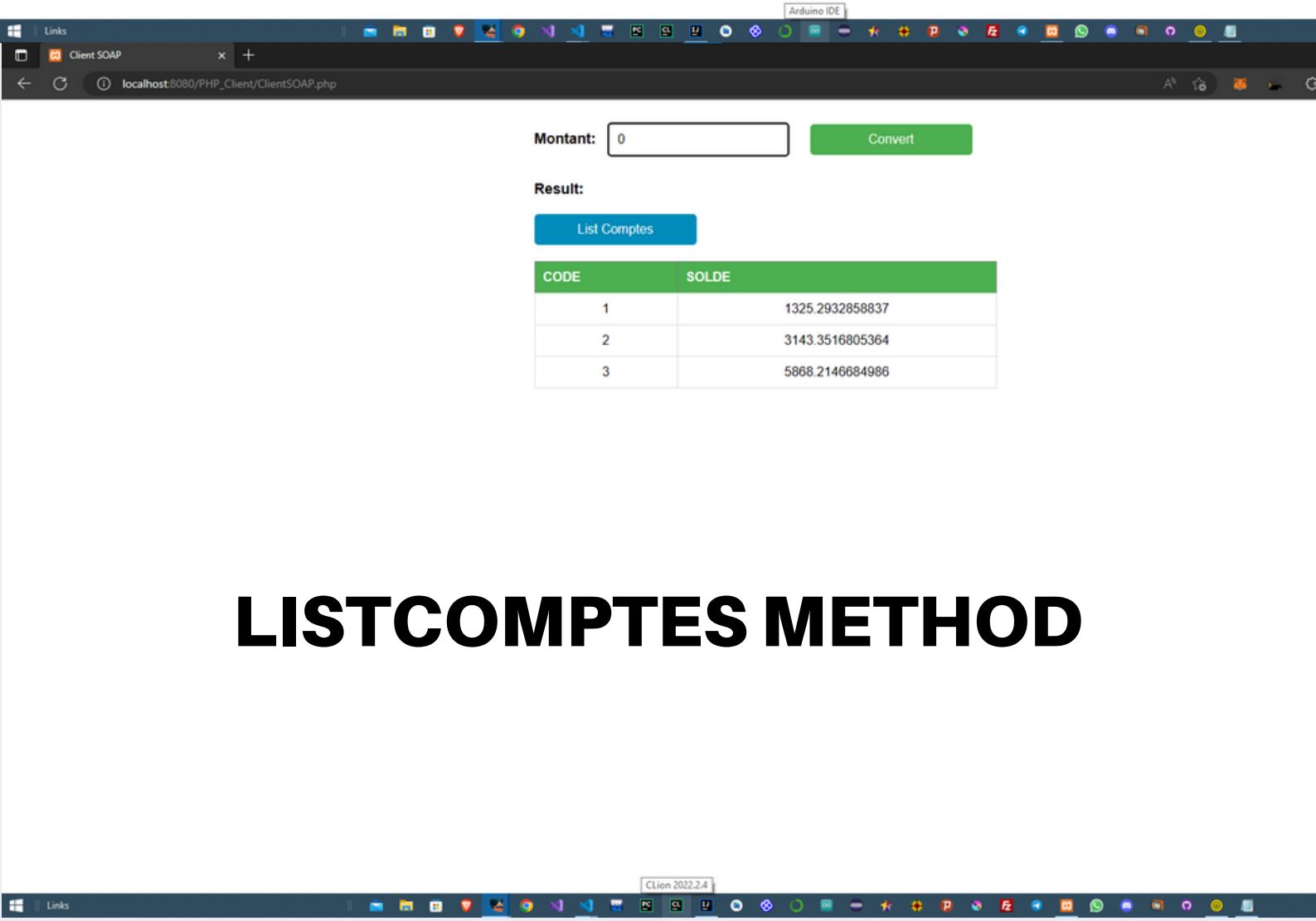
CODE :

```php
<?php
$mt = 0;
if (isset($_POST['action'])) {
    $action = $_POST['action'];
    if ($action == "Convert") {
        $mt = $_POST['montant'];
        $client = new SoapClient("http://localhost:9191/BanqueWS?wsdl");
        $param = new stdClass();
        $param->montant = $mt;
        $rep = $client->__soapCall("Convert", array($param));
        $res = $rep->return;
    } elseif ($action == "List Comptes") {
        $client = new SoapClient("http://localhost:9191/BanqueWS?wsdl");
        $res2 = $client->__soapCall("listComptes", array());
    }
}
?>
<!DOCTYPE html>
<html>

<head>
    <title>Client SOAP</title>
    <link rel="stylesheet" href="style.css">
</head>

<body>
    <div class="container">
        <form method="post" action="ClientSOAP.php">
            <h3>Montant:
                <input type="text" name="montant" value="<?php echo ($mt) ?>">
                <input type="submit" value="Convert" name="action">
            </h3>
        </form>
        <div class="result">
            <h3>Result:
                <?php
                if (isset($res)) {
                    echo ($res) . " EUR";
                }
                ?>
            </h3>
        </div>
        <form method="post" action="ClientSOAP.php">
            <input type="submit" value="List Comptes" name="action">
        </form>
        <?php if (isset($res2)) { ?>
            <table border="1" width="100%">
                <tr>
                    <th>CODE</th>
                    <th>SOLDE</th>
                </tr>
                <?php foreach ($res2->return as $cp) { ?>
                    <tr>
                        <td><?php echo ($cp->code) ?></td>
                        <td><?php echo ($cp->solde) ?></td>
                    </tr>
                <?php } ?>
            </table>
        <?php } ?>
    </div>
</body>

</html>
```

RESULT :



CONVERT METHOD



LISTCOMPTES METHOD

CLIENT SOAP .NET

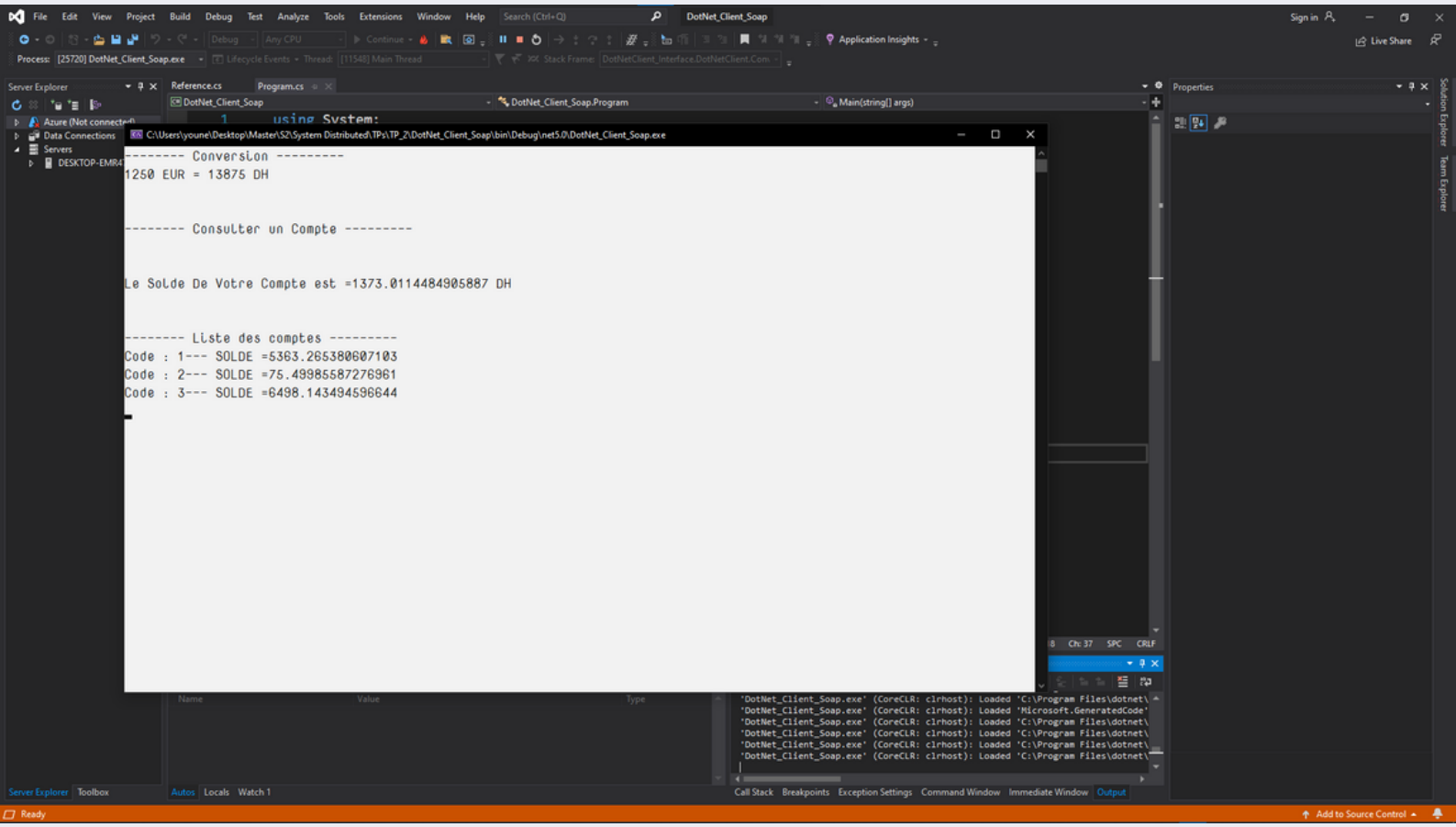
.NET CONSOLE CLIENT CODE :

```
using System;

namespace DotNet_Client_Soap
{
    class Program
    {
        static void Main(string[] args)
        {
            ServiceReference1.BanqueServiceClient stub =
            new ServiceReference1.BanqueServiceClient();
            Console.WriteLine("-------- Conversion ---------");
            Console.WriteLine("1250 EUR = "+stub.Convert(1250)+" DH");
            Console.WriteLine("\n");
            Console.WriteLine("-------- Consulter un Compte ---------");
            ServiceReference1.compte cp = stub.getCompte(2);
            Console.WriteLine("\n");
            Console.WriteLine("Le Solde De Votre Compte est =" + cp.solde+" DH");
            Console.WriteLine("\n");
            Console.WriteLine("-------- Liste des comptes ---------");
            ServiceReference1.compte[] cptes = stub.listComptes();
            for (int i = 0; i < cptes.Length; i++)
            {
                Console.WriteLine("Code : " +cptes[i].code + "--- SOLDE =" + cptes[i].solde);
            }
            Console.ReadLine();
        }
    }
}
```

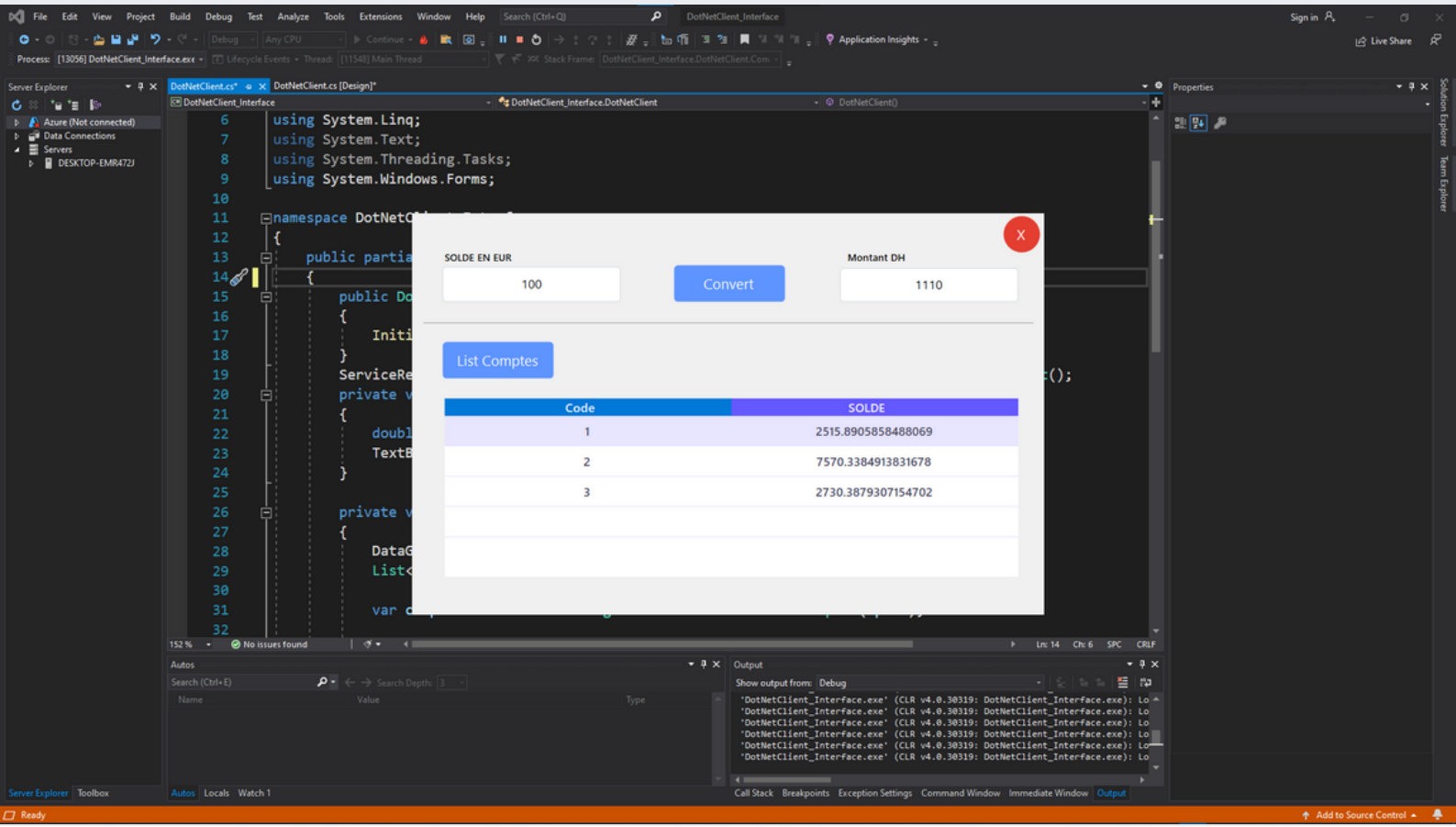
.NET INTERFACE CLIENT CODE:

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace DotNetClient_Interface
{
    public partial class DotNetClient : Form
    {
        public DotNetClient()
        {
            InitializeComponent();
        }
        ServiceReference1.BanqueServiceClient stub = new ServiceReference1.BanqueServiceClient();
        private void ConvertButton_Click(object sender, EventArgs e)
        {
            double Montant = double.Parse(TextBoxDH.Text);
            TextBoxEUR.Text = stub.Convert(Montant).ToString();
        }

        private void ButtonListComptes_Click(object sender, EventArgs e)
        {
            DataGridViewListComptes.AutoGenerateColumns = false;
            List<ServiceReference1.compte> cptes = stub.listComptes().ToList();

            var comptesList = new BindingList<ServiceReference1.compte>(cptes);

            DataGridViewListComptes.DataSource = comptesList;
        }

        private void ButtonClose_Click(object sender, EventArgs e)
        {
            this.Close();
        }
    }
}
```

.NET CONSOLE CLIENT RESULT :



.NET INTERFACE CLIENT RESULT:



DEPLOY WEB SERVICE ON SPRING BOOT

DEPLOY WEB SERVICE ON SPRING BOOT

WEB SERVICE CODE :

```java
package com.sid.wsbanque.web;

import com.sid.wsbanque.entities.Compte;
import com.sid.wsbanque.repositories.CompteRepository;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Component;

import javax.jws.WebMethod;
import javax.jws.WebParam;
import javax.jws.WebService;
import java.util.List;

@Component
@WebService(serviceName = "BanqueWS")
public class CompteSoapService {
    @Autowired
    private CompteRepository compteRepository;

    @WebMethod
    public List<Compte> compteList() {

        return compteRepository.findAll();
    }

    @WebMethod
    public Compte getcompte(@WebParam(name = "id") Long id) {

        return compteRepository.findById(id).get();
    }

    @WebMethod
    public float Convert(@WebParam(name = "montant") float balance) {
        float conversionRate = 10.815f;
        float convertedBalance = balance * conversionRate;
        convertedBalance = Math.round(convertedBalance * 100.0f) / 100.0f;
        return convertedBalance;
    }
}

```

CONTROLLER CODE :

```java
package com.sid.wsbanque;

import com.sid.wsbanque.entities.Compte;
import com.sid.wsbanque.web.CompteSoapService;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.ui.Model;
import org.springframework.web.bind.annotation.*;

import java.util.List;

@RestController
public class CompteController {
    @Autowired
    private CompteSoapService compteSoapService;

    @GetMapping("/list")
    public List<Compte> list(Model model) {
        List<Compte> comptes = compteSoapService.compteList();
        model.addAttribute("comptes", comptes);
        return comptes;
    }

    @GetMapping("/getone/{id}")
    public Compte view(@PathVariable("id") Long id, Model model) {
        Compte compte = compteSoapService.getcompte(id);
        model.addAttribute("compte", compte);
        return compte;
    }

    @PostMapping("/convert")
    public float convert(@RequestParam("montant") float balance, Model model) {
        float convertedBalance = compteSoapService.Convert(balance);
        model.addAttribute("montant", balance);
        model.addAttribute("result", convertedBalance);
        return convertedBalance;
    }
}
```

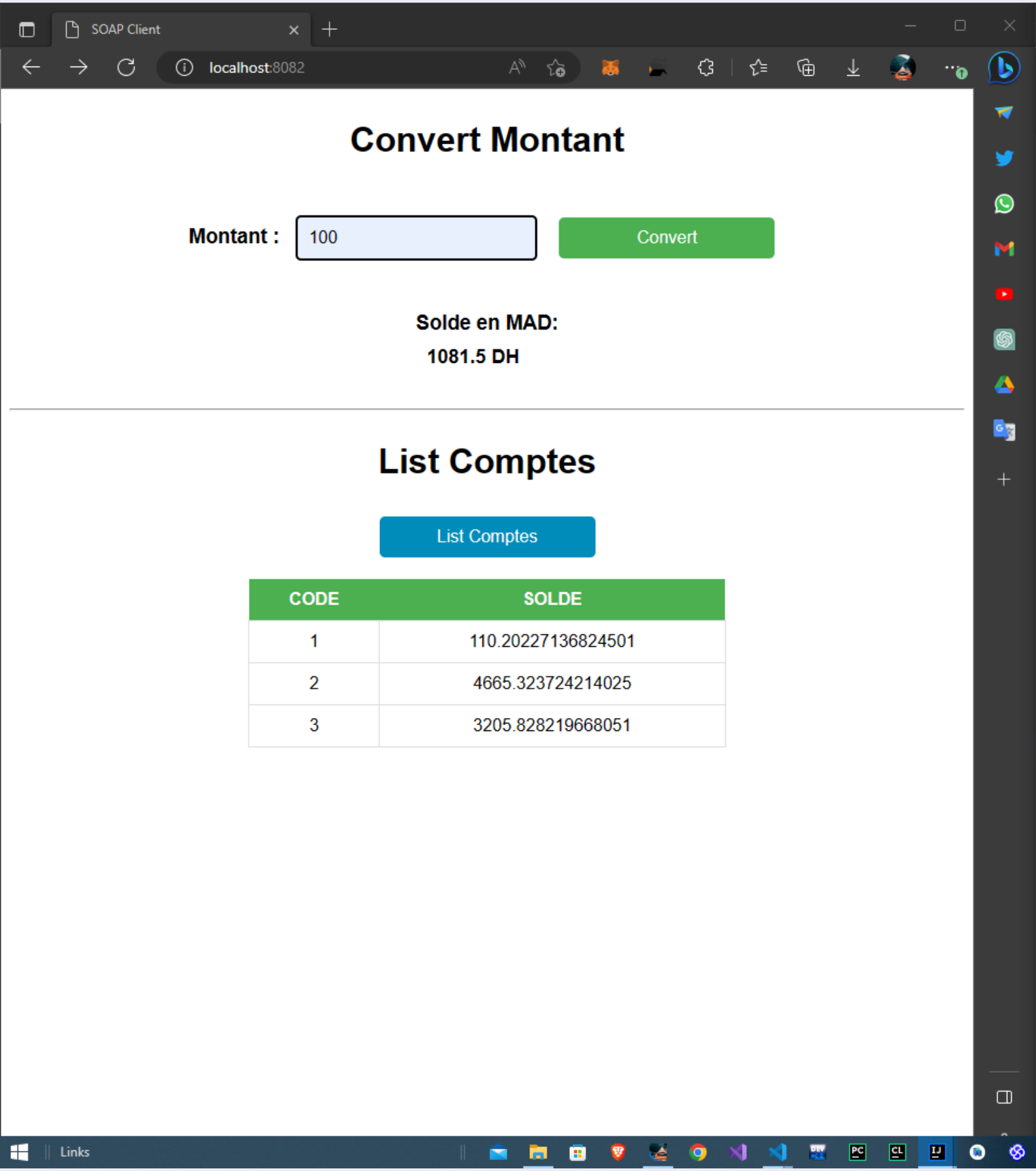
CONFIGURATION CODE :

```java
package com.sid.wsbanque;

import org.springframework.context.annotation.Bean;
import org.springframework.context.annotation.Configuration;
import org.springframework.remoting.jaxws.SimpleJaxWsServiceExporter;

@Configuration
public class Myconfig {
    @Bean
    SimpleJaxWsServiceExporter serviceExporter() {
        SimpleJaxWsServiceExporter serviceExporter = new SimpleJaxWsServiceExporter();
        serviceExporter.setBaseAddress("http://0.0.0.0:8888/");
        return serviceExporter;
    }

}

```

APPLICATION. PROPERTIES CODE :

```properties
spring.datasource.url=jdbc:h2:mem:banque
server.port=8082
```

WEB PAGE CODE

```html
<!DOCTYPE html>
<html xmlns:th="http://www.w3.org/1999/xhtml">
<head>
    <title>SOAP Client</title>
    <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
    <script type="text/javascript">
        $(document).ready(function () {
            $('#convert-form').submit(function (event) {
                event.preventDefault();
                var formData = $(this).serialize();
                $.ajax({
                    type: "POST",
                    url: "/convert",
                    data: formData,
                    success: function (result) {
                        $("#result").text(result + " DH");
                    },
                    error: function (e) {
                        console.log("ERROR : ", e);
                    }
                });
            });
        });
    </script>
    <script type="text/javascript">
        $(document).ready(function() {
            $('#list-comptes-btn').click(function(e) {
                e.preventDefault();
                $.get('/list', function(data) {
                    var comptes = data;
                    var tableBody = $('#comptes-table tbody');
                    tableBody.empty();
                    comptes.forEach(function(compte) {
                        var tr = $('<tr>');
                        tr.append($('<td>').text(compte.id));
                        tr.append($('<td>').text(compte.solde));
                        tableBody.append(tr);
                    });
                });
            });
        });
    </script>

</head>
<body>

<form id="convert-form">
    <h1>Convert Montant</h1>
    <h3>Montant : <input type="text" name="montant"><button type="submit">Convert</button></h3>
    <h3>Solde en MAD:<p id="result"></p></h3>
</form>
<hr>
<form method="get" action="/list">
    <h1>List Comptes</h1>
    <button type="submit" id="list-comptes-btn" >List Comptes</button>
<table id="comptes-table">
    <thead>
    <tr>
        <th>CODE</th>
        <th>SOLDE</th>
    </tr>
    </thead>
    <tbody >
    <tr th:each="compte : ${comptes}">
        <td th:text="${compte.id}"></td>
        <td th:text="${compte.solde}"></td>
    </tr>
    </tbody>
</table>
</form>
</body>
</html>
```

WEB PAGE RESULT :