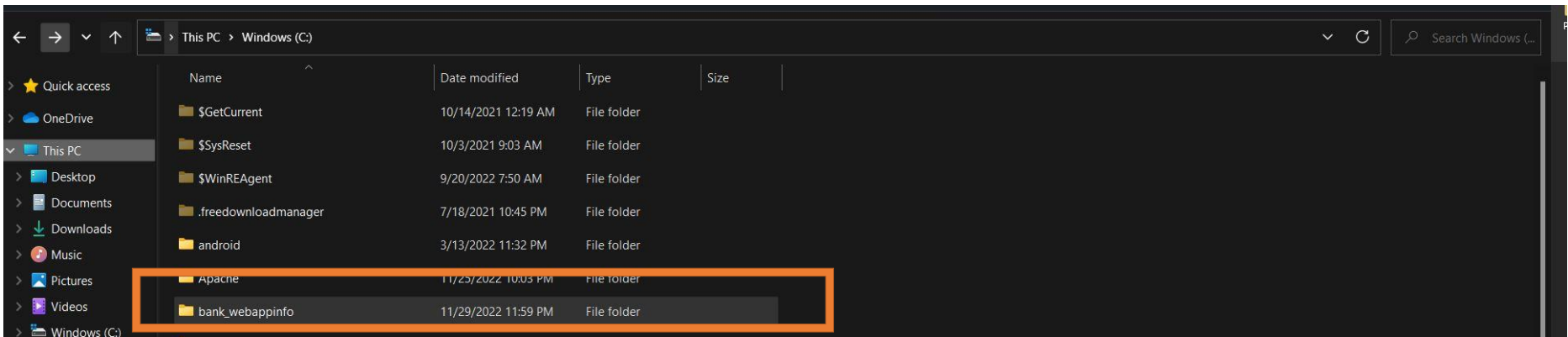


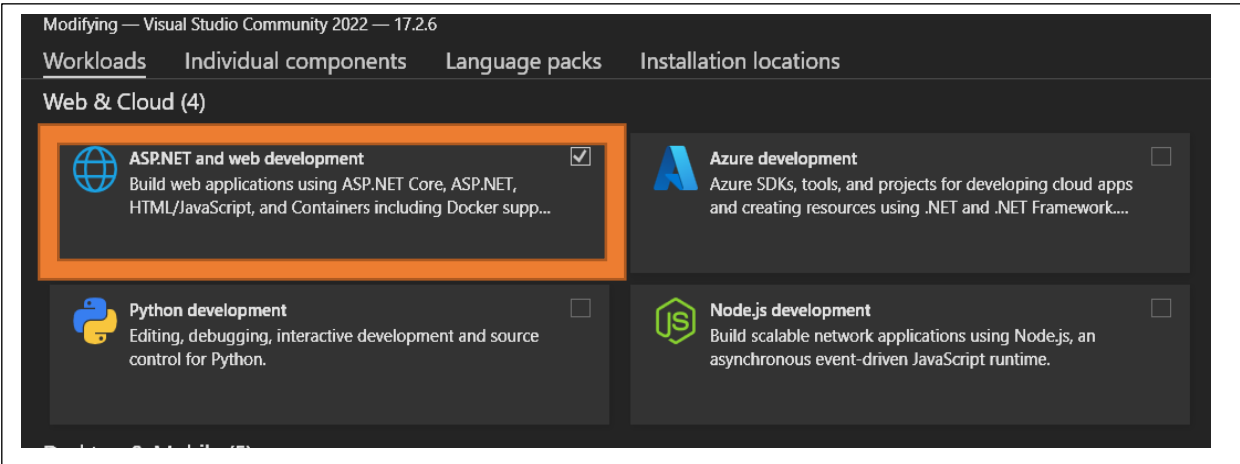
01.

I use C# to built Saop base web service and web application

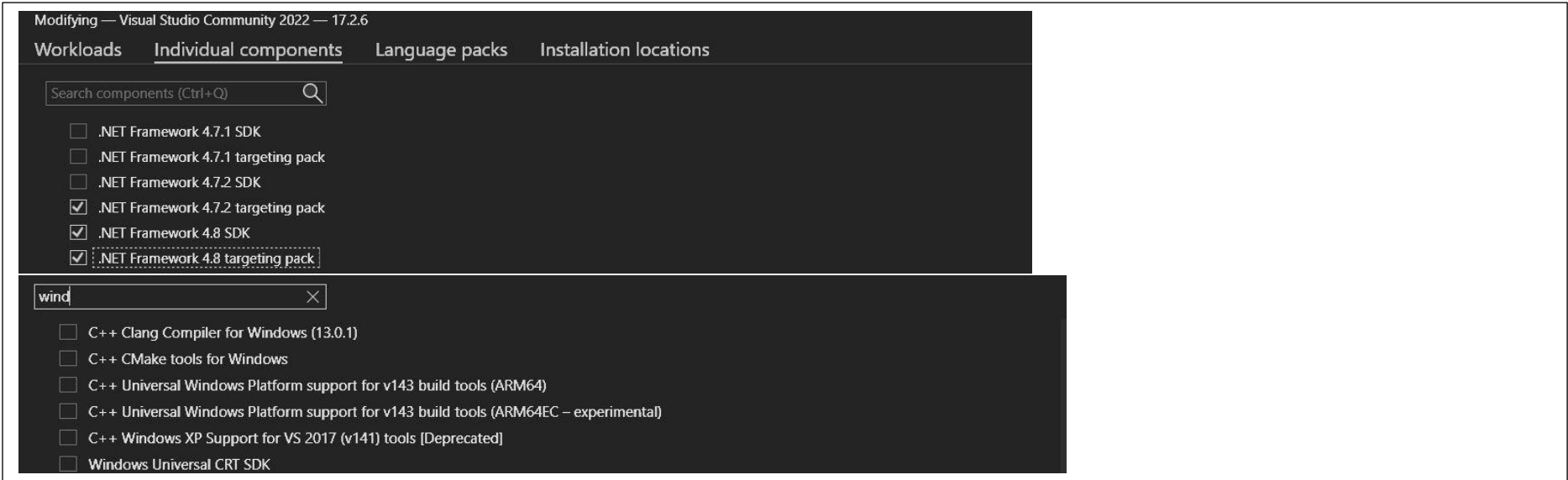
- Extract the zip file
- After unzip the zip file ,there is a folder (name -> bank_webappinfo). Put this folder c partition



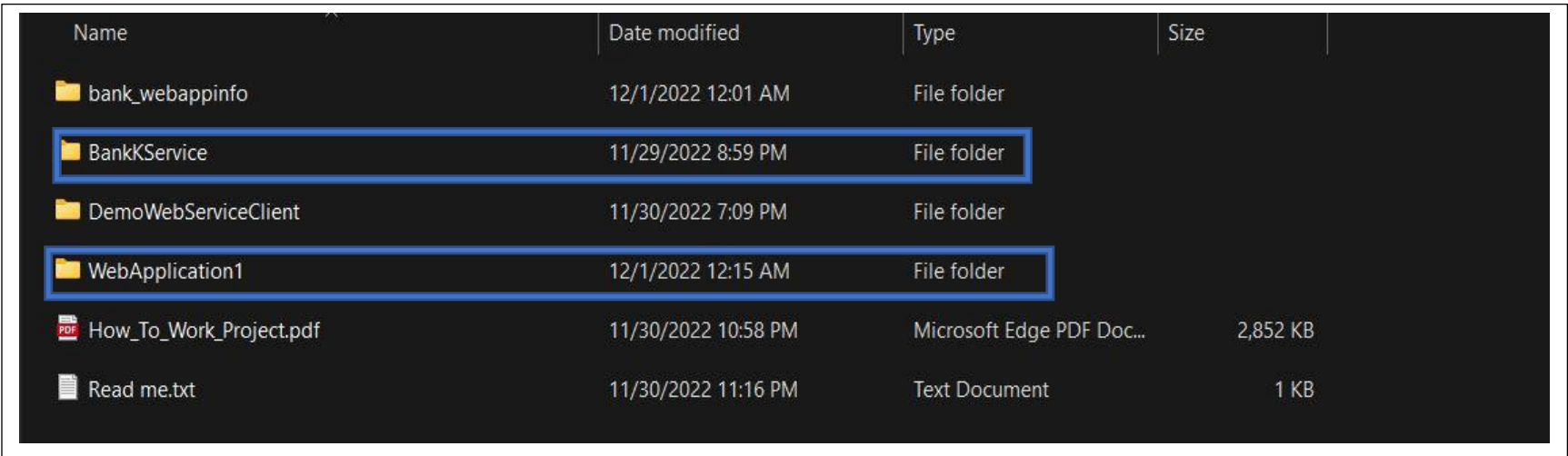
- Requirement of run this project
 1. Need to install visual studio
 2. Need to install ASP.NET and web development web framework



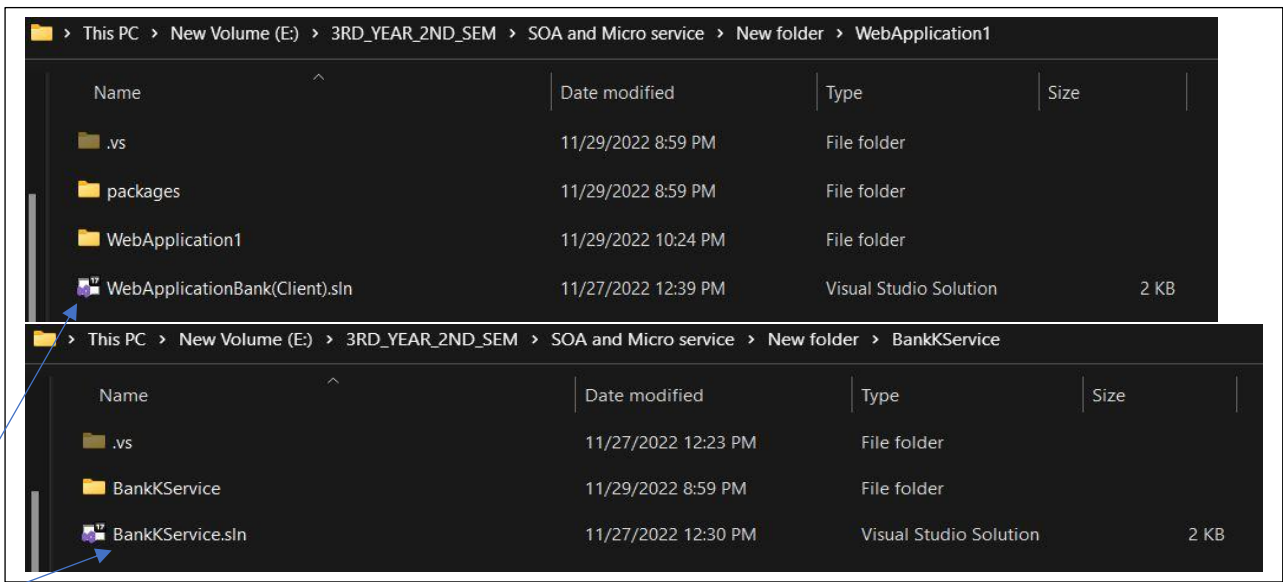
3. Need to install following individual component
 - .NET Framework 4.8 targeting pack
 - .NET Framework 4.7.2 targeting pack
 - Windows Communication Foundation



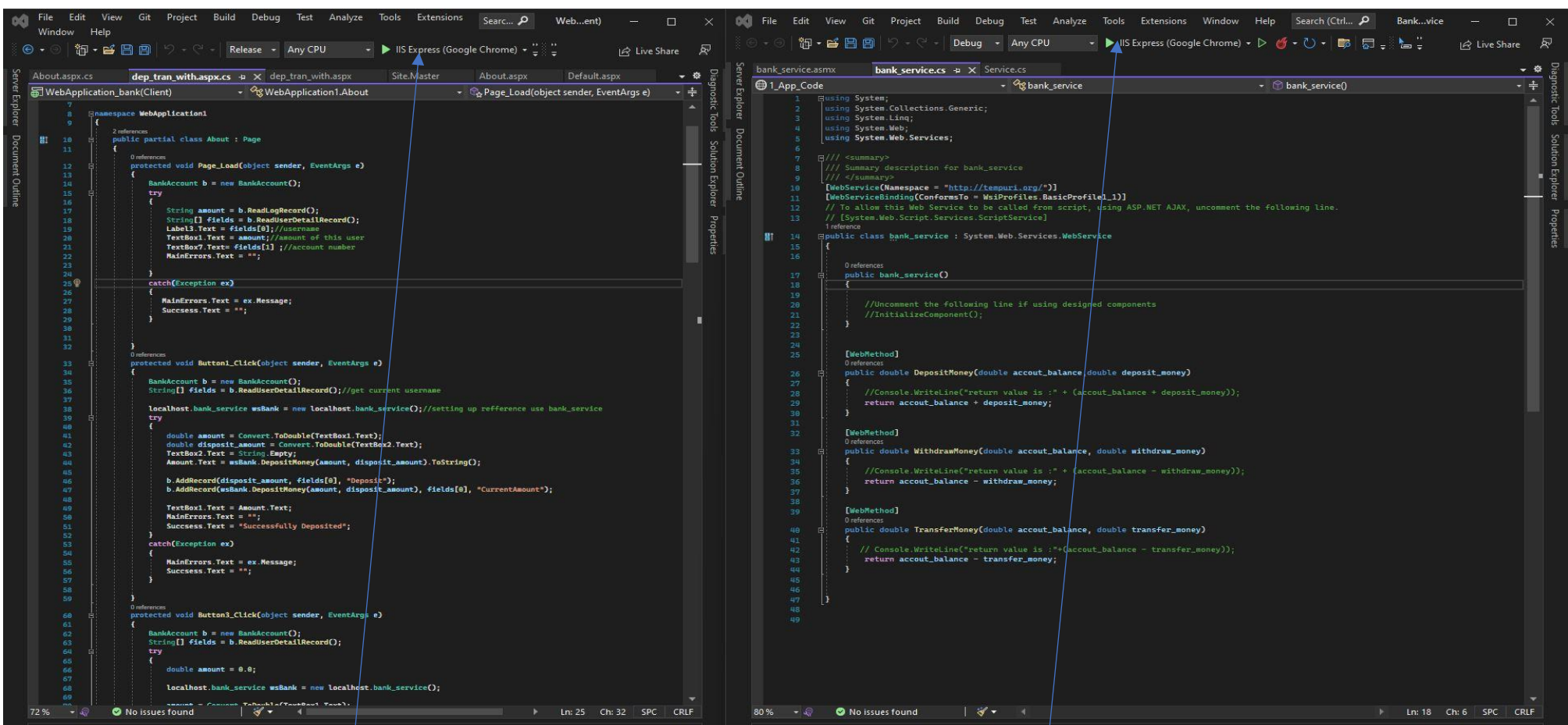
- Extract folder there are two folders as a following name



- Open folders click .sin file then projects will open inside visual studio



Click these file individually then open 2 visual studio windows



First run Bank web Client (Web application)

First run Bank web Service

- First run bank_service then service will open default web browser

bank_service

The following operations are supported. For a formal definition, please review the [Service Description](#).

- [DepositMoney](#)
- [TransferMoney](#)
- [WithdrawMoney](#)

This web service is using <http://tempuri.org/> as its default namespace.

Recommendation: Change the default namespace before the XML Web service is made public.

Each XML Web service needs a unique namespace in order for client applications to distinguish it from other services on the Web. <http://tempuri.org/> is available for XML Web services that are under development, but published XML Web services should use a more permanent namespace.

Your XML Web service should be identified by a namespace that you control. For example, you can use your company's Internet domain name as part of the namespace. Although many XML Web service namespaces look like URLs, they need not point to actual resources on the Web. (XML Web service namespaces are URIs.)

For XML Web services created using ASP.NET, the default namespace can be changed using the `WebService` attribute's `Namespace` property. The `WebService` attribute is an attribute applied to the class that contains the XML Web service methods. Below is a code example that sets the namespace to `"http://microsoft.com/webservices/"`:

```
C#
[WebService(Namespace="http://microsoft.com/webservices/")]
public class MyWebService {
    // implementation
}

Visual Basic
<WebService Namespace="http://microsoft.com/webservices/"> Public Class MyWebService
    ' implementation
End Class

C++
[WebService(Namespace="http://microsoft.com/webservices/")]
public ref class MyWebService {
    // implementation
};
```

For more details on XML namespaces, see the W3C recommendation on [Namespaces in XML](#).

For more details on WSDL, see the [WSDL Specification](#).

For more details on URIs, see [RFC 2396](#).

Click this you can go to deposit money service page

Click this you can go to withdraw money service page

Click this you can go to WSDL page

bank_service

Click [here](#) for a complete list of operations.

DepositMoney

Test

To test the operation using the HTTP POST protocol, click the 'Invoke' button.

Parameter	Value
accout_balance:	<input type="text"/>
deposit_money:	<input type="text"/>

Invoke

SOAP 1.1

The following is a sample SOAP 1.1 request and response. The placeholders shown need to be replaced with actual values.

```
POST /bank_service.asmx HTTP/1.1
Host: localhost
Content-Type: text/xml; charset=utf-8
Content-Length: length
SOAPAction: "http://tempuri.org/DepositMoney"

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <DepositMoney xmlns="http://tempuri.org/">
      <accout_balance>double</accout_balance>
      <deposit_money>double</deposit_money>
    </DepositMoney>
  </soap:Body>
</soap:Envelope>

HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <DepositMoneyResponse xmlns="http://tempuri.org/">
      <DepositMoneyResult>double</DepositMoneyResult>
    </DepositMoneyResponse>
  </soap:Body>
</soap:Envelope>
```

SOAP 1.2

The following is a sample SOAP 1.2 request and response. The placeholders shown need to be replaced with actual values.

```
POST /bank_service.asmx HTTP/1.1
Host: localhost
Content-Type: application/soap+xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
<soap12:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap12="http://www.w3.org/2003/05/soap-envelope">
  <soap12:Body>
    <DepositMoney xmlns="http://tempuri.org/">
      <accout_balance>double</accout_balance>
      <deposit_money>double</deposit_money>
    </DepositMoney>
  </soap12:Body>
</soap12:Envelope>

HTTP/1.1 200 OK
Content-Type: application/soap+xml; charset=utf-8
Content-Length: length
```

You can check the web methods using this section

bank_service

Click [here](#) for a complete list of operations.

WithdrawMoney

Test

To test the operation using the HTTP POST protocol, click the 'Invoke' button.

Parameter	Value
accout_balance:	<input type="text"/>
withdraw_money:	<input type="text"/>

Invoke

SOAP 1.1

The following is a sample SOAP 1.1 request and response. The placeholders shown need to be replaced with actual values.

```
POST /bank_service.asmx HTTP/1.1
Host: localhost
Content-Type: text/xml; charset=utf-8
Content-Length: length
SOAPAction: "http://tempuri.org/WithdrawMoney"

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <WithdrawMoney xmlns="http://tempuri.org/">
      <accout_balance>double</accout_balance>
      <withdraw_money>double</withdraw_money>
    </WithdrawMoney>
  </soap:Body>
</soap:Envelope>

HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <WithdrawMoneyResponse xmlns="http://tempuri.org/">
      <WithdrawMoneyResult>double</WithdrawMoneyResult>
    </WithdrawMoneyResponse>
  </soap:Body>
</soap:Envelope>
```

SOAP 1.2

The following is a sample SOAP 1.2 request and response. The placeholders shown need to be replaced with actual values.

```
POST /bank_service.asmx HTTP/1.1
Host: localhost
Content-Type: application/soap+xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
<soap12:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap12="http://www.w3.org/2003/05/soap-envelope">
  <soap12:Body>
    <WithdrawMoney xmlns="http://tempuri.org/">
      <accout_balance>double</accout_balance>
      <withdraw_money>double</withdraw_money>
    </WithdrawMoney>
  </soap12:Body>
</soap12:Envelope>

HTTP/1.1 200 OK
Content-Type: application/soap+xml; charset=utf-8
Content-Length: length
```

← → ↻ ⌂ ⓘ http://localhost:20328/bank_service.asmx?op=DepositMoney

New folder

New folder

Gmail

YouTube

Pause

Maps

Torrent Search Engi...

bank_service

Click [here](#) for a complete list of operations.

DepositMoney

Test

To test the operation using the HTTP POST protocol, click the 'Invoke' button.

Parameter	Value
accout_balance:	10000
deposit_money:	2000

Invoke

SOAP 1.1

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

```
<double xmlns="http://tempuri.org/">12000</double>
```

← → ↻ ⌂ ⓘ http://localhost:20328/bank_service.asmx?op=WithdrawMoney

New folder

New folder

Gmail

YouTube

Pause

Maps

Torrent Search Engi...

bank_service

Click [here](#) for a complete list of operations.

WithdrawMoney

Test

To test the operation using the HTTP POST protocol, click the 'Invoke' button.

Parameter	Value
accout_balance:	10000
withdraw_money:	2000

Invoke

SOAP 1.1

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

```
<double xmlns="http://tempuri.org/">8000</double>
```

WSDL

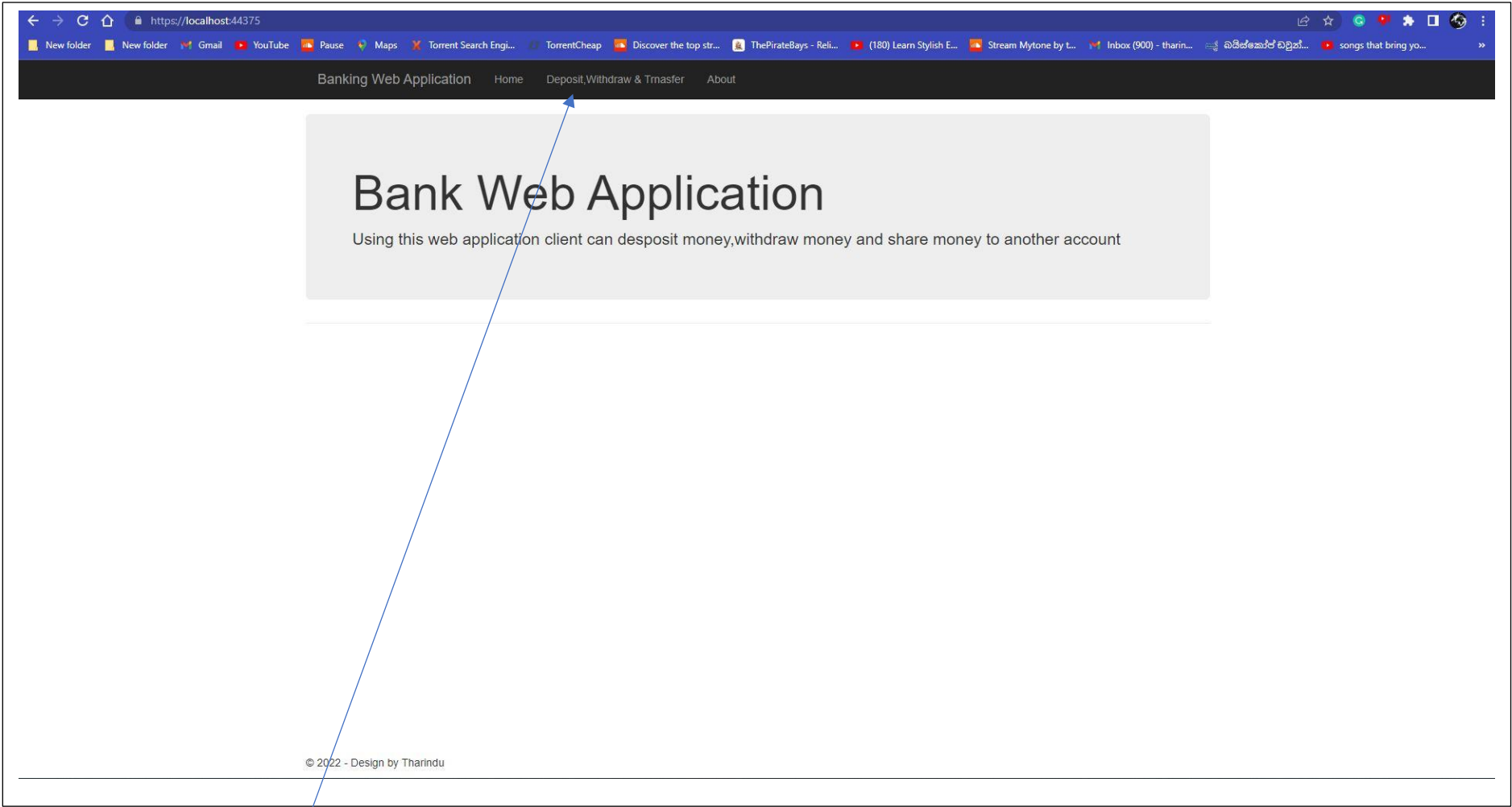
This XML file does not appear to have any style information associated with it. The document tree is shown below.

```

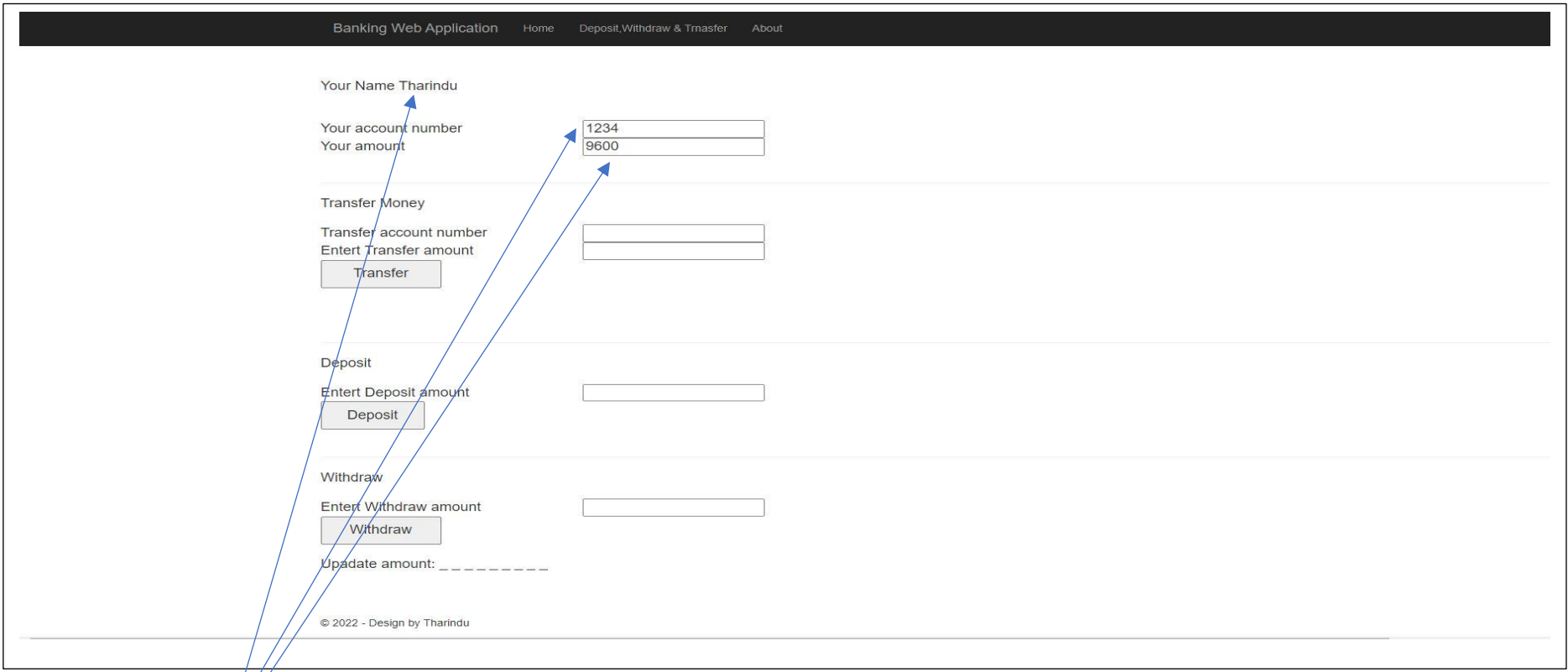
wsdl:definitions xmlns:s="http://www.w3.org/2001/XMLSchema" xmlns:soap12="http://schemas.xmlsoap.org/wsdl/soap12/" xmlns:tns="http://tempuri.org/" xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/" xmlns:tm="http://microsoft.com/wsdl/mime/textMatching/" xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/" targetNamespace="http://tempuri.org/">
  <wsdl:types>
    <xs:schema elementFormDefault="qualified" targetNamespace="http://tempuri.org/">
      <xs:element name="DepositMoney">
        <xs:complexType>
          <xs:sequence>
            <xs:element minOccurs="1" maxOccurs="1" name="accout_balance" type="s:double"/>
            <xs:element minOccurs="1" maxOccurs="1" name="deposit_money" type="s:double"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:element name="DepositMoneyResponse">
        <xs:complexType>
          <xs:sequence>
            <xs:element minOccurs="1" maxOccurs="1" name="DepositMoneyResult" type="s:double"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:element name="WithdrawMoney">
        <xs:complexType>
          <xs:sequence>
            <xs:element minOccurs="1" maxOccurs="1" name="accout_balance" type="s:double"/>
            <xs:element minOccurs="1" maxOccurs="1" name="withdraw_money" type="s:double"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:element name="WithdrawMoneyResponse">
        <xs:complexType>
          <xs:sequence>
            <xs:element minOccurs="1" maxOccurs="1" name="WithdrawMoneyResult" type="s:double"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:element name="TransferMoney">
        <xs:complexType>
          <xs:sequence>
            <xs:element minOccurs="1" maxOccurs="1" name="accout_balance" type="s:double"/>
            <xs:element minOccurs="1" maxOccurs="1" name="transfer_money" type="s:double"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:element name="TransferMoneyResponse">
        <xs:complexType>
          <xs:sequence>
            <xs:element minOccurs="1" maxOccurs="1" name="TransferMoneyResult" type="s:double"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
    </xs:schema>
  </wsdl:types>
  <wsdl:message name="DepositMoneySoapIn">
    <wsdl:part name="parameters" element="tns:DepositMoney"/>
  </wsdl:message>
  <wsdl:message name="DepositMoneySoapOut">
    <wsdl:part name="parameters" element="tns:DepositMoneyResponse"/>
  </wsdl:message>
  <wsdl:output>
    <soap:body use="literal"/>
  </wsdl:output>
  <wsdl:operation name="WithdrawMoney">
    <soap:operation soapAction="http://tempuri.org/WithdrawMoney" style="document"/>
    <wsdl:input>
      <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <soap:body use="literal"/>
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="TransferMoney">
    <soap:operation soapAction="http://tempuri.org/TransferMoney" style="document"/>
    <wsdl:input>
      <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <soap:body use="literal"/>
    </wsdl:output>
  </wsdl:operation>
  <wsdl:binding name="bank_serviceSoap12" type="tns:bank_serviceSoap">
    <soap12:binding transport="http://schemas.xmlsoap.org/soap/http"/>
    <wsdl:operation name="DepositMoney">
      <soap12:operation soapAction="http://tempuri.org/DepositMoney" style="document"/>
      <wsdl:input>
        <soap12:body use="literal"/>
      </wsdl:input>
      <wsdl:output>
        <soap12:body use="literal"/>
      </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="WithdrawMoney">
      <soap12:operation soapAction="http://tempuri.org/WithdrawMoney" style="document"/>
      <wsdl:input>
        <soap12:body use="literal"/>
      </wsdl:input>
      <wsdl:output>
        <soap12:body use="literal"/>
      </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="TransferMoney">
      <soap12:operation soapAction="http://tempuri.org/TransferMoney" style="document"/>
      <wsdl:input>
        <soap12:body use="literal"/>
      </wsdl:input>
      <wsdl:output>
        <soap12:body use="literal"/>
      </wsdl:output>
    </wsdl:operation>
  </wsdl:binding>
  <wsdl:service name="bank_service">
    <wsdl:port name="bank_serviceSoap" binding="tns:bank_serviceSoap">
      <soap:address location="http://localhost:20328/bank_service.asmx"/>
    </wsdl:port>
    <wsdl:port name="bank_serviceSoap12" binding="tns:bank_serviceSoap12">
      <soap12:address location="http://localhost:20328/bank_service.asmx"/>
    </wsdl:port>
  </wsdl:service>
</wsdl:definitions>
  <wsdl:part name="parameters" element="tns:TransferMoney"/>
</wsdl:message>
  <wsdl:message name="TransferMoneySoapOut">
    <wsdl:part name="parameters" element="tns:TransferMoneyResponse"/>
  </wsdl:message>
  <wsdl:portType name="bank_serviceSoap">
    <wsdl:operation name="DepositMoney">
      <wsdl:input message="tns:DepositMoneySoapIn"/>
      <wsdl:output message="tns:DepositMoneySoapOut"/>
    </wsdl:operation>
    <wsdl:operation name="WithdrawMoney">
      <wsdl:input message="tns:WithdrawMoneySoapIn"/>
      <wsdl:output message="tns:WithdrawMoneySoapOut"/>
    </wsdl:operation>
    <wsdl:operation name="TransferMoney">
      <wsdl:input message="tns:TransferMoneySoapIn"/>
      <wsdl:output message="tns:TransferMoneySoapOut"/>
    </wsdl:operation>
  </wsdl:portType>
  <wsdl:binding name="bank_serviceSoap" type="tns:bank_serviceSoap">
    <soap:binding transport="http://schemas.xmlsoap.org/soap/http"/>
    <wsdl:operation name="DepositMoney">
      <soap:operation soapAction="http://tempuri.org/DepositMoney" style="document"/>
      <wsdl:input>
        <soap:body use="literal"/>
      </wsdl:input>
      <wsdl:output>
        <soap:body use="literal"/>
      </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="WithdrawMoney">
      <soap:operation soapAction="http://tempuri.org/WithdrawMoney" style="document"/>
      <wsdl:input>
        <soap:body use="literal"/>
      </wsdl:input>
      <wsdl:output>
        <soap:body use="literal"/>
      </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="TransferMoney">
      <soap:operation soapAction="http://tempuri.org/TransferMoney" style="document"/>
      <wsdl:input>
        <soap:body use="literal"/>
      </wsdl:input>
      <wsdl:output>
        <soap:body use="literal"/>
      </wsdl:output>
    </wsdl:operation>
  </wsdl:binding>
  <wsdl:binding name="bank_serviceSoap12" type="tns:bank_serviceSoap">
    <soap12:binding transport="http://schemas.xmlsoap.org/soap/http"/>
    <wsdl:operation name="DepositMoney">
      <soap12:operation soapAction="http://tempuri.org/DepositMoney" style="document"/>
      <wsdl:input>
        <soap12:body use="literal"/>
      </wsdl:input>
      <wsdl:output>
        <soap12:body use="literal"/>
      </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="WithdrawMoney">

```


- After web application you can above interface



Click this you can go to withdraw , deposit and money transfer



These details get from computer local storage txt file

Page load

```
protected void Page_Load(object sender, EventArgs e)
{
    BankAccount b = new BankAccount();
    try
    {
        String amount = b.ReadLogRecord();//read the current user log file and get his current amount
        String[] fields = b.ReadUserDetailRecord();//read the user deatails file and get the first user [we assume first user is current user and 2nd user is another user]
        Label3.Text = fields[0];//username
        TextBox1.Text = amount;//amount of this user
        TextBox7.Text= fields[1] ;//account number
        MainErrors.Text = "";//remove error msges
    }
    catch(Exception ex)
    {
        MainErrors.Text = ex.Message;//handle error
        Success.Text = "";
    }
}
```

Deposit money

Banking Web Application

Home

Deposit,Withdraw & Tmaste

About

Your Name Tharindu

Your account number1234

Your amount10000

Transfer Money

Transfer account number

Entert Transfer amount

Transfer

Deposit

Entert Deposit amount200

Deposit

Withdraw

Entert Withdraw amount

Withdraw

Upadate amount: _ _ _ _ _

© 2022 - Design by Tharindu

Banking Web Application

Home

Deposit,Withdraw & Tmaste

About

Successfully Deposited

Your Name Tharindu

Your account number1234

Your amount10200

Transfer Money

Transfer account number

Entert Transfer amount

Transfer

Deposit

Entert Deposit amount

Deposit

Withdraw

Entert Withdraw amount

Withdraw

Update amount: 10200

© 2022 - Design by Tharindu

Deposit web method

```
[WebMethod]
0 references
public double DepositMoney(double accout_balance,double deposit_money)
{
    //Console.WriteLine("return value is :" + (accout_balance + deposit_money));
    return accout_balance + deposit_money;
}
```

Log file(Cllient)

200,Tharindu,Deposit
10200,Tharindu,CurrentAmount

User Details File

Tharindu,1234
Sumith,1235

There are two users in this userdetails txt file. For Demonstrate purpose money transfer another account

Deposit Related Client Side Codes

```
protected void Button1_Click(object sender, EventArgs e)
{
    BankAccount b = new BankAccount();//create reference user's bank account
    String[] fields = b.ReadUserDetailRecord();//get current user username

    localhost.bank_service wsBank = new localhost.bank_service();//setting up reference use bank_service
    try
    {
        double amount = Convert.ToDouble(TextBox1.Text);//get the Textbox value and convert to double
        double disposit_amount = Convert.ToDouble(TextBox2.Text);//get the user entered value and convert to double
        TextBox2.Text = String.Empty;//clear the text filed
        Amount.Text = wsBank.DepositMoney(amount, disposit_amount).ToString();//pass the value web method and response conver to string and assign to Amount text filed

        b.AddRecord(disposit_amount, fields[0], "Deposit");//add record in user log file
        b.AddRecord(wsBank.DepositMoney(amount, disposit_amount), fields[0], "CurrentAmount");//add record in user log file

        TextBox1.Text = Amount.Text;
        MainErrors.Text = "";
        Success.Text = "Successfully Deposited";//show the msg
    }
    catch(Exception ex)
    {
        MainErrors.Text = ex.Message;//handle errors
        Success.Text = "";
    }
}
```

Add record method

```
class BankAccount
{
    //client bank account
    6 references
    public void AddRecord(double amount,String name,String typeOfWork)//first user
    {
        using (System.IO.StreamWriter file = new System.IO.StreamWriter("C://bank_webappinfo//log.txt", true))//path of log file
        {
            file.WriteLine(amount+","+name+","+typeOfWork);//write the file using this format
        }
    }
}
```


Withdraw Money

Banking Web ApplicationHomeDeposit,Withdraw & TmansferAbout

Your Name Tharindu

Your account number1234

Your amount8200

Transfer Money

Transfer account number

Entert Transfer amount

Transfer

Deposit

Entert Deposit amount

Deposit

Withdraw

Entert Withdraw amount2000

Withdraw

Update amount: _____

© 2022 - Design by Tharindu

Banking Web ApplicationHomeDeposit,Withdraw & TmansferAbout

Successfully Withdrawed
Your Name Tharindu

Your account number1234

Your amount6200

Transfer Money

Transfer account number

Entert Transfer amount

Transfer

Deposit

Entert Deposit amount

Deposit

Withdraw

Entert Withdraw amount

Withdraw

Uupdate amount: 6200

© 2022 - Design by Tharindu

Withdraw Web Method

```
[WebMethod]
0 references
public double WithdrawMoney(double accout_balance, double withdraw_money)
{
    //Console.WriteLine("return value is :" + (accout_balance - withdraw_money));
    return accout_balance - withdraw_money;
}
```

Withdraw Related Client Side Codes

```
protected void Button3_Click(object sender, EventArgs e)
{
    BankAccount b = new BankAccount();//create reference user's bank account
    String[] fields = b.ReadUserDetailRecord();//get current user username
    try
    {
        double amount = 0.0;

        localhost.bank_service wsBank = new localhost.bank_service();

        amount = Convert.ToDouble(TextBox1.Text);//get the Textbox value and convert to double
        double withdraw_amount = Convert.ToDouble(TextBox5.Text);//get the user entered value and convert to double

        TextBox5.Text = String.Empty;//clear the text filed

        Amount.Text = wsBank.WithdrawMoney(amount, withdraw_amount).ToString();//pass the value web method and response conver to string and assign to Amount text filed

        b.AddRecord(withdraw_amount, fields[0], "Withdraw");//add record in user log file
        b.AddRecord(wsBank.WithdrawMoney(amount, withdraw_amount), fields[0], "CurrentAmount");//add record in user log file

        TextBox1.Text = Amount.Text;
        MainErrors.Text = "";
        Success.Text = "Successfully Withdrawed";//show the msg
    }
    catch (Exception ex)
    {
        MainErrors.Text=ex.Message;//handle errors
        Success.Text = "";//clear successtext because we need to only show error
    }
}
```

Log file(Cllient)

2000,Tharindu,Withdraw
6200,Tharindu,CurrentAmount

Transfer Method

Your Name Tharindu

Your account number1234

Your amount9000

Transfer Money

Transfer account number1235

Entert Transfer amount200

Transfer

Deposit

Entert Deposit amount

Deposit

Withdraw

Entert Withdraw amount

Withdraw

Update amount: _____

© 2022 - Design by Tharindu

Successfully Transfered

Your Name Tharindu

Your account number1234

Your amount8800

Transfer Money

Transfer account number

Entert Transfer amount

Transfer

Deposit

Entert Deposit amount

Deposit

Withdraw

Entert Withdraw amount

Withdraw

Update amount: 8800

© 2022 - Design by Tharindu

Log file(Client)

200,Tharindu,Transfer
8800,Tharindu,CurrentAmount

Transfer Account owner’s log file(another user)

3224,Sumith,CurrentAmount
200,Sumith,Trasfer
3424,Sumith,CurrentAmount

Transfer method develop by using withdraw and deposit bank services

0 references

```
protected void Button4_Click(object sender, EventArgs e)
{
    BankAccount b = new BankAccount();//create reference user's bank account
    String[] fields = b.ReadUserDetailRecord();//get current user username

    double amount = Convert.ToDouble(Balance.Text);//get the current user account balance
    double transfer_amount = Convert.ToDouble(TextBox6.Text);//get the user entered transfer amount and conver to double
    String transfer_account = TextBox8.Text;//get the transfer account number [2nd user of userdetail txt]

    TextBox6.Text = String.Empty;//clear the text box
    TextBox8.Text = String.Empty;//clear the text box

    Amount.Text = Convert.ToString(b.MoneyTransfer(transfer_account, transfer_amount));//call the client method to transfer the money other user and the money transfer return double value
    //conver to string if any error return 1 otherwise return current user upadeted balance after moeny transfer
    if(Amount.Text == "1")//check the transfer account exist or not [1 = account not found]
    {
        Error1.Text = "Enter valid Account Number";//show the error in user interface
        Success.Text = "";//clear other messages
    }
    else
    {
        b.AddRecord(transfer_amount, fields[0], "Transfer");//if transfer account exist then add record in current user's log file
        b.AddRecord(Convert.ToDouble(Amount.Text), fields[0], "CurrentAmount");//if transfer account exist then add record in current user's log file
        Balance.Text = Amount.Text;
        Error1.Text = "";
        Success.Text = "Successfully Transfered";
    }
}
```


Transfer Method Client Side(using withdraw and deposit web service)

```
1 reference
public double MoneyTransfer(String trasfer_account_number,double tranfer_amount )
{
    Boolean check = false;//for flag
    BankAccount b = new BankAccount();//first user
    BankAccount b2 = new BankAccount();//2nd user
    double user_current_amount = Convert.ToDouble( b.ReadLogRecord());//get user current amount
    localhost.bank_service wsBank = new localhost.bank_service();

    //checking usersDetailslist if there match acc number with trasfer acc number
    String[] lines = System.IO.File.ReadAllLines("userDetails.txt");//get all line of userDetails.txt, there are 2 users in this file

    for(int i=0; i < lines.Length; i++)//check line by line
    {
        String[] fields = lines[i].Split(',');//get one line and split data
        if(trasfer_account_number == fields[1])//check the transfer account is exisit or not
        {
            user_current_amount = wsBank.WithdrawMoney(user_current_amount, tranfer_amount);//if 2nd user exist then withdraw money from this user
            double amount = Convert.ToDouble(b2.ReadLogRecord2());//this will return relavant user current balance ,but i am not use database therefore only one user log file is exist, therefore i create
            amount= wsBank.DepositMoney(amount, tranfer_amount);//deposit that transfer ammount 2nd user

            b2.AddRecord2(tranfer_amount, fields[0], "Trasfer");//add record 2nd user log file log2.txt his trasfer amount
            b2.AddRecord2(amount, fields[0], "CurrentAmount");//add record 2nd user log file log2.txt his current amount
            check = true;
        }
    }
    if (check == false)
    {
        return 1;
        // 1 ->account not fount
    }

    return user_current_amount;
}
```

```
1 reference
public void AddRecord2(double amount, String name, String typeOfWork)//2nd user
{
    using (System.IO.StreamWriter file = new System.IO.StreamWriter("C://bank_webappinfo//log2.txt", true))//path of log file
    {
        file.WriteLine(amount + "," + name + "," + typeOfWork);
    }
}
```

```
1 reference
public String ReadLogRecord2()// this method for another one user because i am not using database for this
{
    String[] lines = System.IO.File.ReadAllLines("C://bank_webappinfo//log2.txt");
    int len = lines.Length;
    String[] fields = lines[len - 1].Split(',');
    return fields[0];//return user current balance
}
```

Error handling

If user enter not valid input

Input string was not in a correct format.

Your Name Tharindu

Your account number1234

Your amount9000

Transfer Money

Transfer account number

Entert Transfer amount

Transfer

Deposit

Entert Deposit amount

Deposit

Withdraw

Entert Withdraw amount

Withdraw

Update amount: _____

© 2022 - Design by Tharindu

If user not valid account number for money transfer

Your Name Tharindu

Your account number1234

Your amount9000

Transfer Money

Transfer account number

Entert Transfer amount

Transfer

Enter valid Account Number

Deposit

Entert Deposit amount

Deposit

Withdraw

Entert Withdraw amount

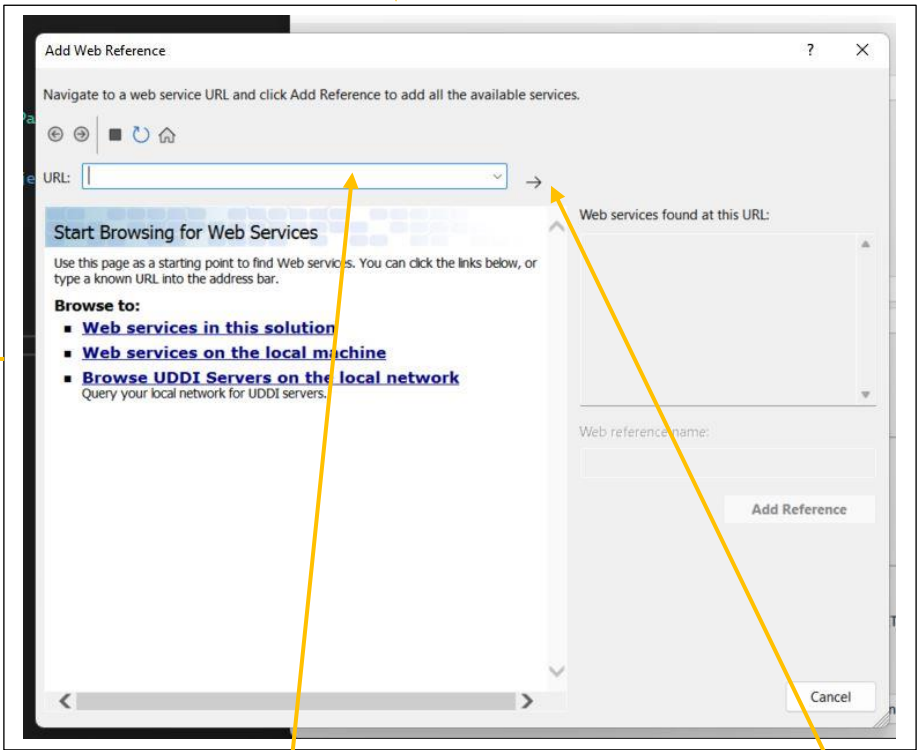
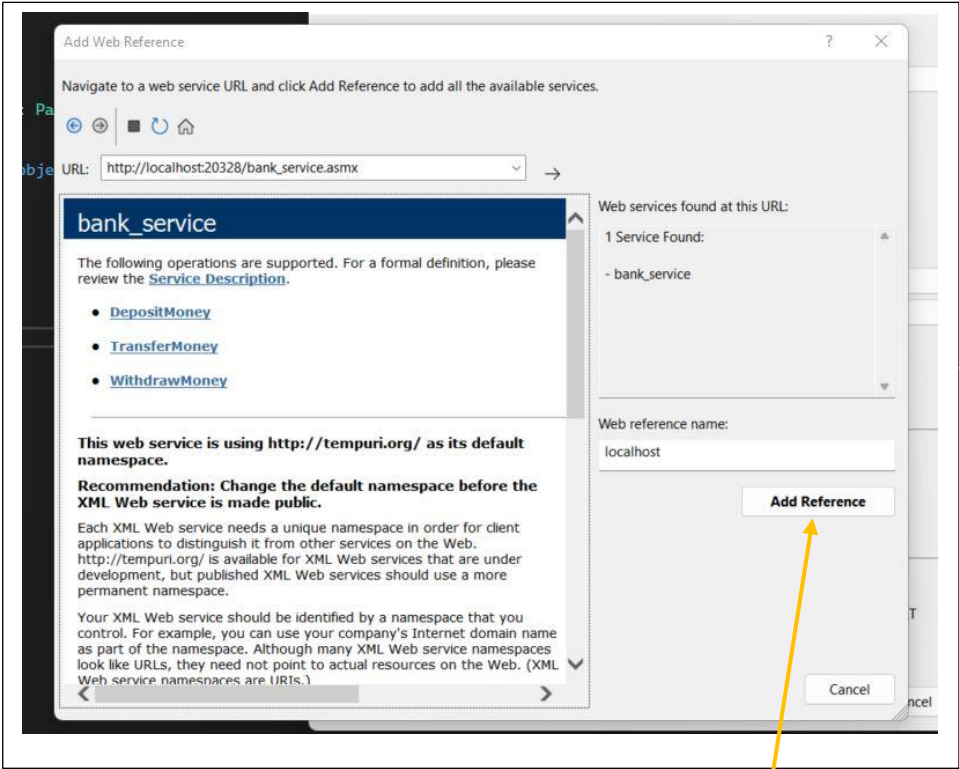
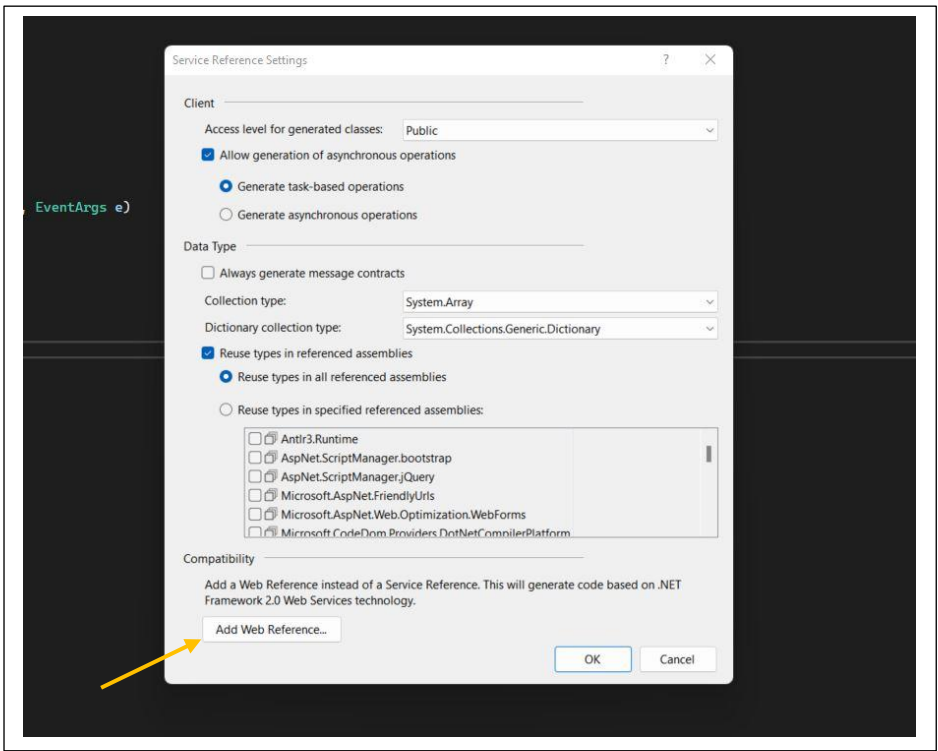
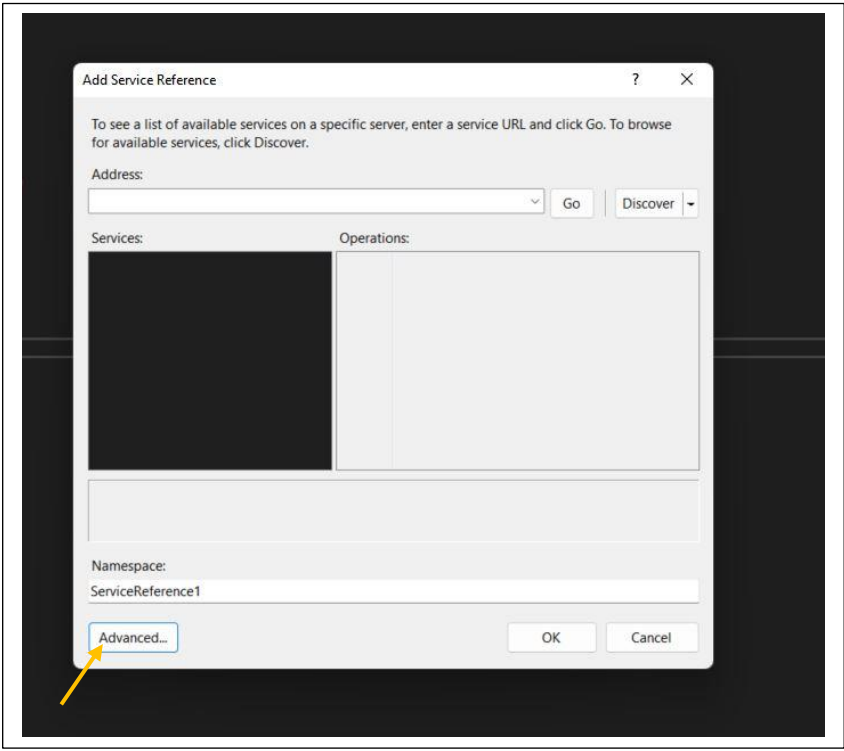
Withdraw

Update amount: 9000

© 2022 - Design by Tharindu

Additional detail

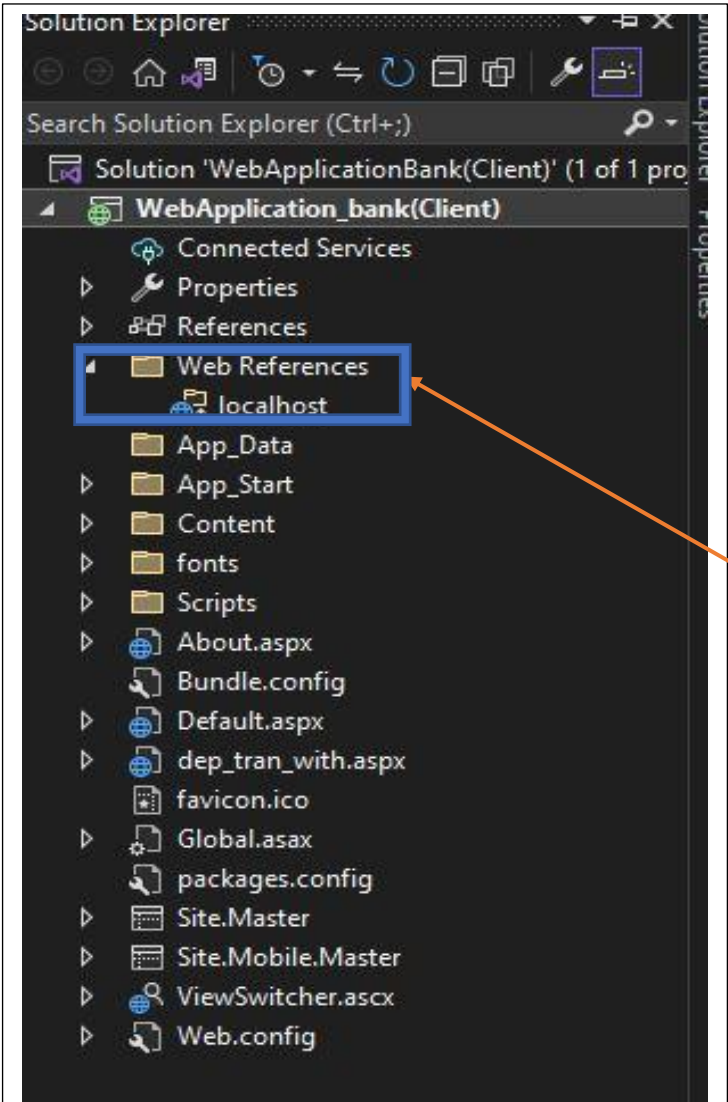
After create Bank web service we need to add reference in bank web app(client side)



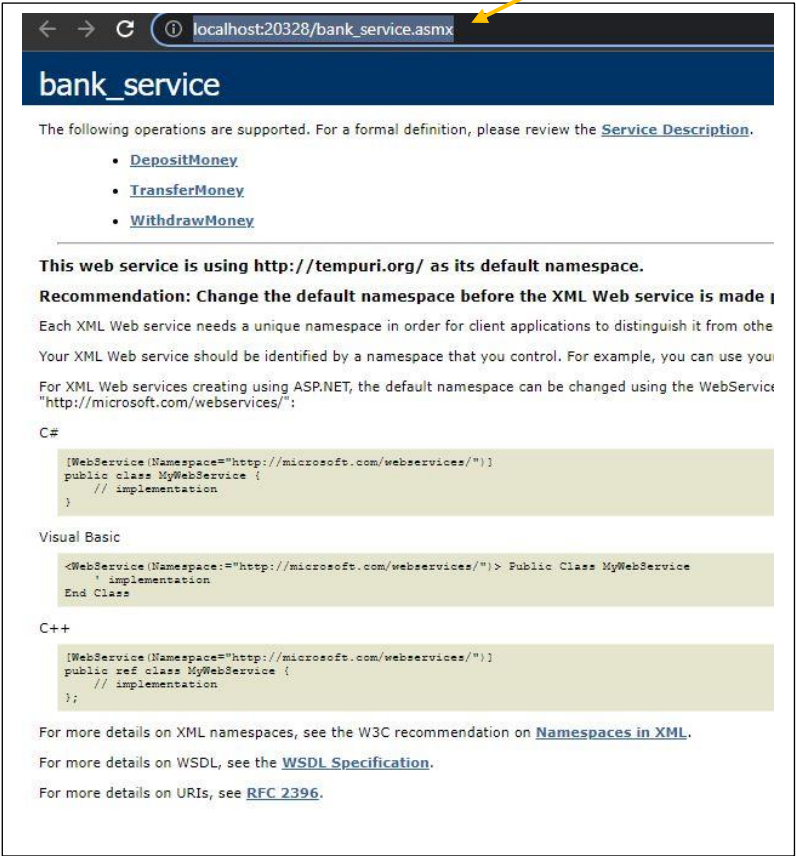
Click add reference

We need to pass bank_service url

This URL



After add reference create this folder



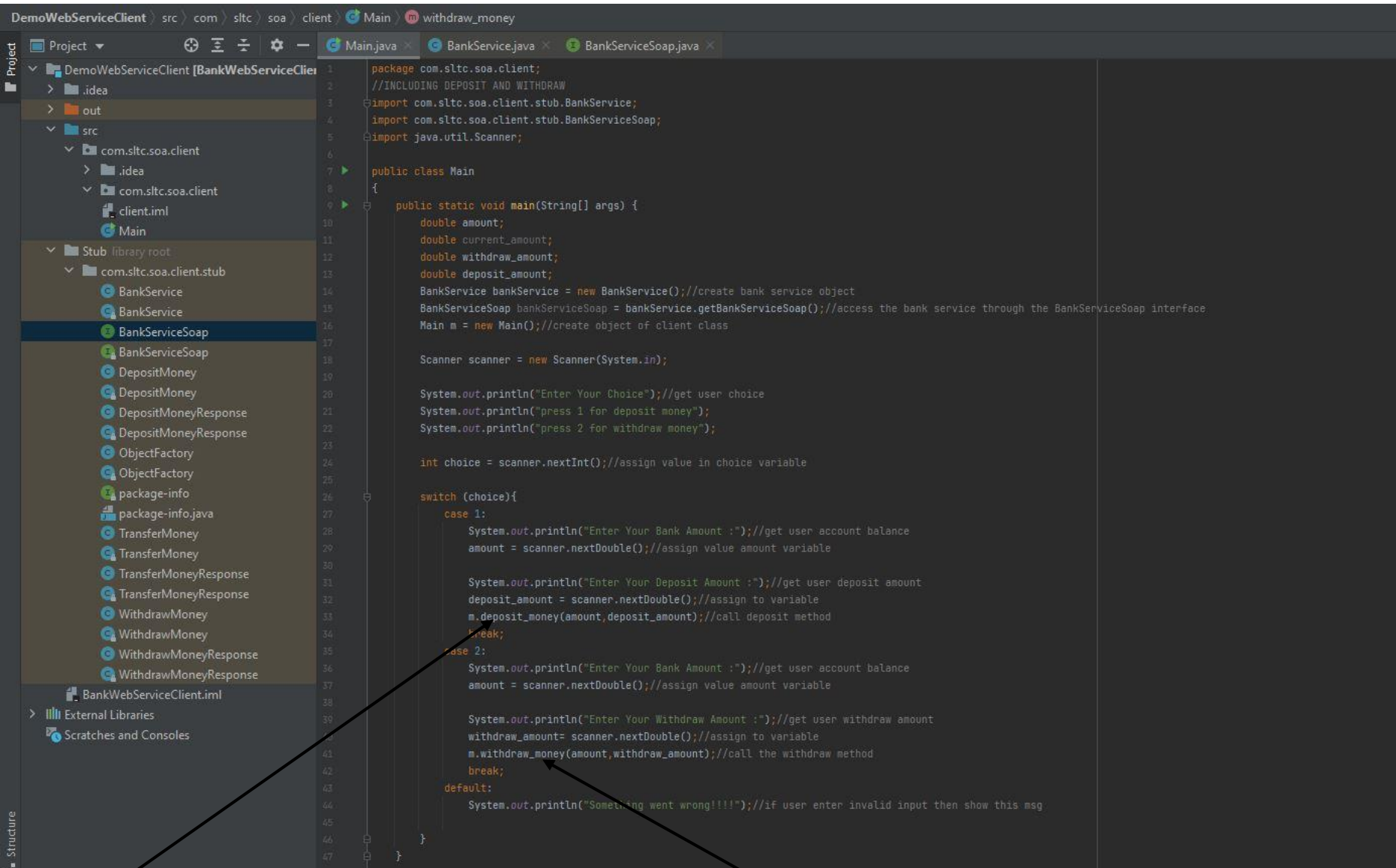
Afetr paste url click the arrow icon

Inside Unzip folder , there is folder name ‘DemoWebServiceClient’
This part is java client(Create java) using bank web service(Create C#)

Name	Date modified	Type	Size
bank_webappinfo	12/1/2022 12:01 AM	File folder	
BankKService	11/29/2022 8:59 PM	File folder	
DemoWebServiceClient	11/30/2022 7:09 PM	File folder	
WebApplication1	12/1/2022 12:15 AM	File folder	
How_To_Work_Project.pdf	11/30/2022 10:58 PM	Microsoft Edge PDF Doc...	2,852 KB
Read me.txt	11/30/2022 11:16 PM	Text Document	1 KB

Intelija idea ->go open project ->set project location-> click open

Main Method



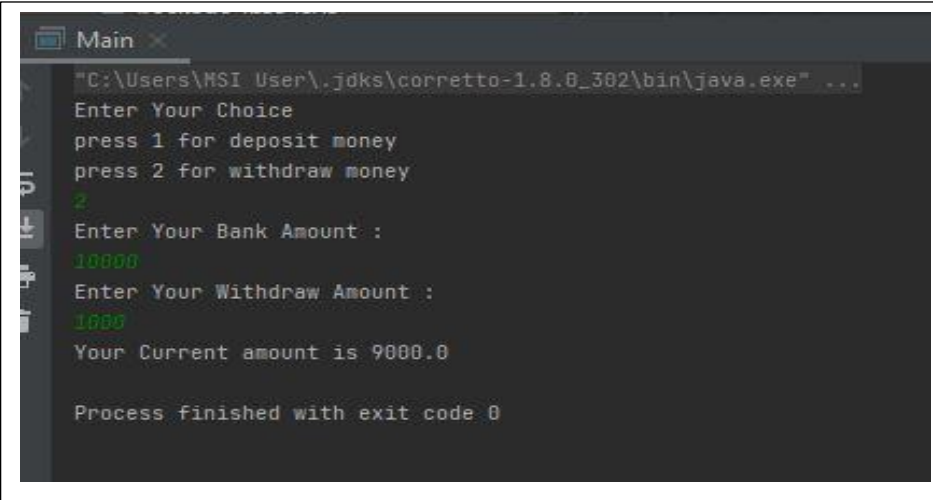
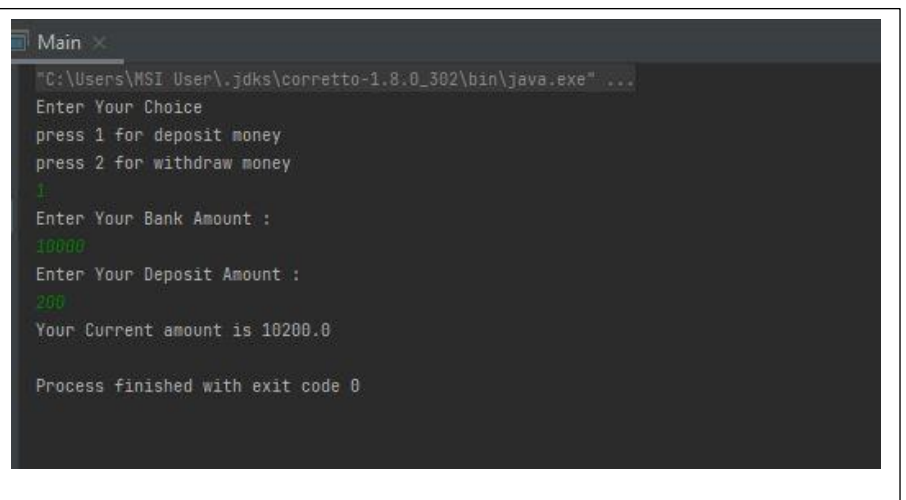
Deposit method

```
public void deposit_money(double amount,double d_amount){// deposit method
    double current_amount;
    BankService bankService = new BankService();
    BankServiceSoap bankServiceSoap = bankService.getBankServiceSoap();
    current_amount = bankServiceSoap.depositMoney(amount,d_amount);//call the deposit web method
    System.out.println("Your Current amount is "+current_amount);
}
```

Withdraw method

```
public void withdraw_money(double amount,double w_amount){//withdraw method
    double current_amount;
    BankService bankService = new BankService();
    BankServiceSoap bankServiceSoap = bankService.getBankServiceSoap();
    current_amount = bankServiceSoap.withdrawMoney(amount,w_amount);//call the withdraw web method
    System.out.println("Your Current amount is "+current_amount);
}
```

Run the programme



If user enter wrong input



```
C:\Users\MSI User\.jdk\corretto-1.8.0_302\bin>
Enter Your Choice
press 1 for deposit money
press 2 for withdraw money
345
Something went wrong!!!!

Process finished with exit code 0
|
```