Web Service plays an important role in the software industry, it is used to call other applications methodologies which are running on another platform

In this tutorial, we are going to see what is web service, why it is needed, how to use that, and the most important part how to invoke WSDL(Web Service Description Language) file in our application.

## **Web Service:**

Web Service is the specification, using this we can share information between the two interoperable (Implemented by 2 different technologies) application.

Suppose we have 2 applications one is Shopping and other is Bank, but both are implemented on different technologies, maybe Shopping application implement by Python and Bank application implemented using C and C++.
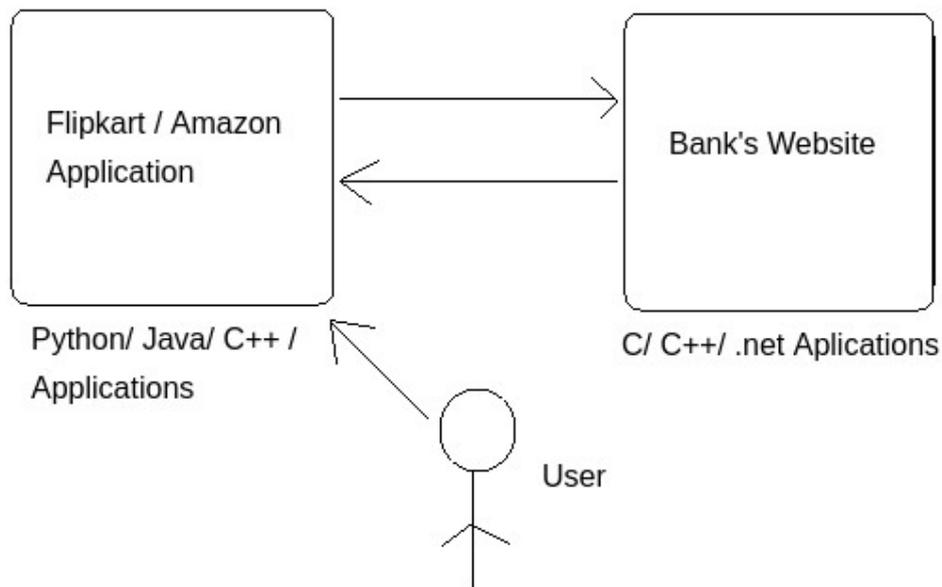
sufficient balance?

Here is the **Web Service** scenario comes, Flipkart or Amazon application communicates with the bank server with the provided data.

Now it seems web service is clear, but how it will communicate, most important what is the language used while communication because they are implemented on 2 different technologies. Before discussing this I Mentioned some old technologies which are used before the web service comes in reality.

## **Technologies Before Web Services:**

Before Web service comes in reality there were 6 scenarios presents are

 1) SP (Socket Programing)
 2) RMI (Remote Method invocation )
 3) EJB(Enterprise Java Beans)
 4) DCOM(Distributed Component Object Model)
 5) RPC(Remote Procedure Call)
 6) CORBA(Common Object Request Broker Architecture)

These concepts are used at the time of Java language booming in the Industry, among that SP, RMI, EJB, and RPC are only used when both the servers are implemented using the java languages only, those are not compatible to other languages. DCOM is implemented by Microsoft and supports only for the .Net platform.

CORBA is the first technology used to make communication between the 2 interoperability concepts (2 Servers can use any platform or language), for this the disadvantage is having more specification (Methods) given by Sun, Microsoft, C++. but those are not that much of the proper implementation, in some critical scenarios they were failing to share the information between 2 applications.

## How Web Service works:

Suppose we have a small application in java which is doing the addition something like below,

```java
public class AddTwoIntegers {

    public static void main(String[] args) {

        int A = self.A;
        int B = self.B;

        System.out.println("Enter two numbers: " + A + " " + B);
        int sum = A + B;

        System.out.println("The sum is: " + sum);
    }
}
```

To call this class in the same application we can simply create the object of the class and invoke the method, but I need to call from another application which is implemented by Python or C or C++, for that I need to create one formate to store the information of the current class with the unique language so that each application or language can simply execute them, and that universal language is XML, it can be understandable by any programming language.

We understood that we need to generate an XML file, the next point is what is the data that XML going to hold. In that XML we have to mention details of the class are Service class name, method names (in XML language we called operation name), parameter, data type, and at the last, we have to mention the endpoint (location of that class ). All combination of these data termes as **WSDL(Web Service Description Language**) file, Below is the simple format of the XML data or WSDL file.

```xml
<?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>
    <Add xmlns="http://tempuri.org/">
      <intA>int</intA>
      <intB>int</intB>
    </Add>
  </soap:Body>
</soap:Envelope>
```

<s:element name="Add">

```
<s:complexType>

<s:sequence>

<s:element minOccurs="1" maxOccurs="1" name="intA" type="s:int"/>

<s:element minOccurs="1" maxOccurs="1" name="intB" type="s:int"/>

</s:sequence>

</s:complexType>

</s:element>

<s:element name="AddResponse">

<s:complexType>

<s:sequence>

<s:element minOccurs="1" maxOccurs="1" name="AddResult" type="s:int"/>

</s:sequence>

</s:complexType>

</s:element>
```

You can find more detail by visting this link http://www.dneonline.com/calculator.asmx?op=Add  http://www.dneonline.com/calculator.asmx?WSDL

Just see above XML data it has add method in "<s:element name="Add">"  and parameter are A, B with the data type as int (integer). After successful response the result will comes in the variable name Addresult  with data type int .

FYI, We have some inbuilt tools to make the WSDL file for our class. We completed the server-side WSDL file but how this file can access the Client-side (Other applications want to use this method)

We have 2 procedures to share this WSDL file, By providing file directly to them or We need to deploy this on third party server and share the DNS or IP of that server,  same like the above add link. I suggest always 2nd procedure that is a simple and easy way, so we can update our WSDL file any time.

## **Implementation**

Once the client has a WSDL file, he can call that file and method name by using python's inbuilt library such as zeep pkg or a client from suds pkg as below.

So I imported the Client, you can use the zeep library also,

   **In [6]: from suds.client import Client**

after that call the WSDL file from the client, So it gathers all the methods present in the WSDL file.
   **In [7]: _client = Client('http://www.dneonline.com/calculator.asmx?WSDL')**

and you can check is the particular method is present in WSDL file using **getattr()** method, it will throw an error if the mentioned method is not present
   **In [8]: add = getattr(_client.service, 'Add')**

Now pass the parameter as describe datatypes to that method as below
  #For addition
   **In [9]: add(10,12)**
   **Out[9]: 22**

  #For Substraction

  **In [10]: substract = getattr(_client.service, 'Subtract')**
  **In [11]: substract(22, 10)**

  **Out[11]: 12**

Here, the mentioned  WSDL file is for the Java language and I did the addition using python

language. In the same way, our shopping application will communicate to the Bank server using the WSDL file and check the data via methods and send operation to deduct balance with acknowledgment, Once acknowledge is positive your shopping order will complete and money will detect from your bank.

That is all about Web Service in Python !!!
Happy Pythoning !!!