**WEBSERVICES & SOAP**

**what is web service?**

**Web Services** are the means by which devices communicate over the World Wide Web. Whether you use a mobile application, search engine or an enterprise system, the user piece of the application resides on your device. The data, and potentially the business rules, live on some other server on the network. How your interface communicates with the server piece is the role of Web Services.

**what is XML and what will be format?**

XML, or Extensible Markup Language, is a markup language that you can use to create your own tags. It was created by the World Wide Web Consortium (W3C) to overcome the limitations of HTML, the Hypertext Markup Language that is the basis for all Web pages. Like HTML, XML is based on SGML -- Standard Generalized Markup Language. Although SGML has been used in the publishing industry for decades, its perceived complexity intimidated many people that otherwise might have used it (SGML also stands for "Sounds great, maybe later"). XML was designed with the Web in mind.

<address>

<name>

<title>Mrs.</title>

<first-name> Mary </first-name>

<last-name> McGoon </last-name>

</name>

<street> 1401 Main Street </street>

<city>Anytown</city>

<state>NC</state>

<postal-code> 34829 </postal-code>

</address>

**what is requestxml and responseXML?**

**Request XML is used to send request for data to web sever where as Response XML is used to send the requested data response.**

**Different http methods?**

**GET:** The GET method is used to retrieve information from the given server using a given URI. Requests using GET should only retrieve data and should have no other effect on the data.

**HEAD:** Same as GET, but transfers the status line and header section only.

**POST:** A POST request is used to send data to the server, for example, customer information, file upload, etc. using HTML forms.

**PUT:** Replaces all current representations of the target resource with the uploaded content.

**DELETE:** Removes all current representations of the target resource given by a URI.

**what is json and how is the format look like?**

JSON: **J**ava**S**cript **O**bject **N**otation. JSON is a syntax for storing and exchanging data. JSON is an easier-to-use alternative to XML.

{"employees":[  
    {"firstName":"John", "lastName":"Doe"},  
    {"firstName":"Anna", "lastName":"Smith"},  
    {"firstName":"Peter", "lastName":"Jones"}  
]}

how to read from json string (for validation purpose) and how to create json data for input?

The **JSON.parse()** method parses a JSON string, constructing the JavaScript value or object described by the string.

**Difference between SOAP and REST webservices?**

SOAP permits only XML data format only

REST permits different data formats such as plain text, HTML, XML, JSON etc.

Assertions in soapUI?

* Assertion means act of affirming or stating something. It can also be interpreted as check point or a validation point.
* Once a request is sent to a web server a response is received. We need to validate if the response contains the data that we expect. In order to validate the response, we need to use assertions.

Below are assertions available in soap UI :

* Property content
* Compliance status standard
* SLA
* JMS

**Xpath:**

Xpath can be used to navigate through elements and attributes in an XML document.

**what is / and // in xpath?**

/ - selects from the root node.

// - Selects nodes in the document from the current node that match the selection no matter where they are

**How to read attributes in xpath?**

1. **<book** id="7890"**>**
2. **<title>**A New Kind of Science**</title>**
3. **<edition>**2**</edition>**
4. **<author** id="1234"**>**Stephen Wolphram**</author>**
5. **<prices>**
6. **<price** currency="usd"**>**20.00**</price>**
7. **<price** currency="can"**>**25.00**</price>**
8. **</prices>**
9. **</book>**

Imagine that you wanted to obtain the ID for the book titled "A New Kind of Science".  Note that the book and author elements both come with an "id" attribute.  To get the ID for the book, you'd do this:

/books/book[title = "A New Kind of Science"]/@id

Which would return the attribute as follows:

id="7890"

**Different functions in xpath?**

Contains() – The contains function determines whether the first argument string contains the second argument string and returns boolean true or false.

floor() - The floor function evaluates a decimal number and returns the largest integer less than or equal to the decimal number.

Boolean() - The boolean function evaluates an expression and returns true or false.

Ceiling() - The ceiling function evaluates a decimal number and returns the smallest integer greater than or equal to the decimal number.

**How to read child nodes in xpath?**

child::\* selects all element children of the context node

child::text() selects all text node children of the context node

child::node() selects all the children of the context node, whatever their node type

**what is XSD (schema), wsdl, wadl and what is contains**

An XML Schema describes the structure of an XML document.

The XML Schema language is also referred to as XML Schema Definition (XSD).

WSDL:

An WSDL document describes a web service. It specifies the location of the service, and the methods of the service, using these major elements:

|  |  |
| --- | --- |
| <types> | Defines the (XML Schema) data types used by the web service |
| <message> | Defines the data elements for each operation |

**what is endpoint and operation names?**

WSDL is an XML format for describing network services as a set of **endpoints**operating on messages containing either document-oriented or procedure-oriented information. The **operations** and messages are described abstractly, and then bound to a concrete network protocol and message format to define an **endpoint**.

Groovy script:

It is a dynamic language with features similar to those of Python, Ruby, Perl, and Smalltalk. It can be used as a **scripting** language for the Java Platform, is dynamically compiled to Java Virtual Machine (JVM) bytecode, and interoperates with other Java code and libraries. **Groovy** uses a Java-like curly-bracket syntax.

Calling soap services directly in selenium with jax-rs and jax-ws frameworks

A Web Service is an application deployed on a machine (server). Web services accept incoming requests from client and send response to client in either plain text, JSON or XML format.

The beautiful thing about web services is they are language independent. So a web service written in Python can be requested from a client written in Java, C#, Ruby etc.

Web services are classified into two categories:

* SOAP – In SOAP-based web services both client and service transfer SOAP messages to communicate.
* REST – While in REST-style services both client and service usually transfer raw XML to communicate.

Java provides API for creating both SOAP and REST-style web services

JAX-WS – JAX-WS (Java API for XML Web Services) is a Java API for creating both SOAP and REST-style web services. There is a common misconception that JAX-WS is only for SOAP-based services, which is not true. Using JAX-WS you can create both SOAP and REST-style services

JAX-RS – JAX-RS (Java API for restful Web Services) is a Java API to write restful web services easily

Although JAX-WS can be used to write restful web services, it’s not the best way to create REST-style services. In a production mode you should probably use JAX-RS, Restlet framework etc

We have annotated the class with @webserviceprovider, which means exchanged messages will be XML document.

The annotation @servicemode with value MESSAGE indicates that the service wants access to the entire message (For example,  HTTP headers and body). Bindingtype annotation declares that this service deals with raw XML over HTTP instead of SOAP over HTTP.

In the invoke method, we first retrieve the messagecontext from webservicecontext. We then check the request method for the incoming request and if it is GET, we call it the doget method passing the messagecontext. Then we extract the value of the city parameter from the query string and call it the which returns an XML document.We store the returned XML document in a stringbuffer.

**23) Calling rest services using bdd/cucumber or in feature file**

Scenario: List fruit

Given the system knows about the following fruit:

| name | color |

| banana | yellow |

| strawberry | red |

When the client requests a list of fruit

Then the response is a list containing 2 fruits

And one fruit has the following attributes:

| attribute | type | value |

| name | String | banana |

| color | String | yellow |

And one fruit has the following attributes:

| attribute | type | value |

| name | String | strawberry |

| color | String | red |

In order to make them pass, we need to

* Create Simple Web service which returns JSON
* Organize our Project Code with Step Definitions & Support directory
* Implement Step Definitions to make them pass