**Rest full services:**

* Rest services use HTTP.
* HTTP: includes following methods,

1. Get: it is used to retrieve some data.
2. Put: update or add data in to the application.
3. Head: it act’s same as get method but only give response to header part.
4. Post: is less secure.

* Rest stands for representational state transfer, it sends the data by XML or Json. If there is an interaction b/w front end & back end as UI then we use Json where as for .NET and other programming languages we use XML.

**WSDL:(web services description language)**

* It is an XML file.
* It can easily readable
* It is xml based interface that is used to describe the functionalities of the web services.

**EXAMPLE:**

<message name="getTermRequest">  
  <part name="term" type="xs:string"/>  
</message>  
  
<message name="getTermResponse">  
  <part name="value" type="xs:string"/>  
</message>  
  
<portType name="glossaryTerms">  
  <operation name="getTerm">  
    <input message="getTermRequest"/>  
    <output message="getTermResponse"/>  
  </operation>  
</portType>

**SOAP:**

* Stands for simple object access protocol.
* It is used to communicate over internet.
* Information/messages is exchange in XML format only.
* Soap messages consists of envelope which contain the information about the header and body.
* Header: it provides the information about messages only and it include authentication, complex types and routing information.
* Body: it contains the actual requested data meant to be send to the server.
* <?xml version="1.0"?>  
    
  <soap:Envelope  
  xmlns:soap="http://www.w3.org/2003/05/soap-envelope/"  
  soap:encodingStyle="http://www.w3.org/2003/05/soap-encoding">  
    
  <soap:Header>  
  ...  
  </soap:Header>  
    
  <soap:Body>  
  ...  
    <soap:Fault>  
    ...  
    </soap:Fault>  
  </soap:Body>  
    
  </soap:Envelope>
* SOAP defines three attributes in the default namespace. These attributes are: mustUnderstand, actor, and encodingStyle.
* The attributes defined in the SOAP Header define how a recipient should process the SOAP message.
* <?xml version="1.0"?>  
    
  <soap:Envelope  
  xmlns:soap="http://www.w3.org/2003/05/soap-envelope/"  
  soap:encodingStyle="http://www.w3.org/2003/05/soap-encoding">  
    
  <soap:Header>  
    <m:Trans xmlns:m="https://www.w3schools.com/transaction/"  
    soap:mustUnderstand="1">234  
    </m:Trans>  
  </soap:Header>  
  ...  
  ...  
  </soap:Envelope>
* **MustUnderstand:**
* The SOAP mustUnderstand attribute can be used to indicate whether a header entry is mandatory or optional for the recipient to process.
* If you add mustUnderstand="1" to a child element of the Header element it indicates that the receiver processing the Header must recognize the element. If the receiver does not recognize the element it will fail when processing the Header.

**Syntax:** soap:mustUnderstand="0|1"

* **Actor:** the SOAP actor attribute is used to address the Header element to a specific endpoint.
* Syntax: soap:actor="*URI*"
* **Encodingstyle:** The encodingStyle attribute is used to define the data types used in the document. This attribute may appear on any SOAP element, and it will apply to that element's contents and all child elements.
* Syntax: soap:encodingStyle="*URI*"

**HTTP PROTOCOLS:**

* HTTP communicates over TCP/IP. An HTTP client connects to an HTTP server using TCP. After establishing a connection, the client can send an HTTP request message to the server:
* POST /item HTTP/1.1  
  Host: 189.123.255.239  
  Content-Type: text/plain  
  Content-Length: 200
* The server then processes the request and sends an HTTP response back to the client. The response contains a status code that indicates the status of the request:
* 200 OK  
  Content-Type: text/plain  
  Content-Length: 200
* In the example above, the server returned a status code of 200. This is the standard success code for HTTP.
* If the server could not decode the request, it could have returned something like this:
* 400 Bad Request  
  Content-Length: 0

**SOAP BINDING:**

* SOAP bindings are mechanisms, which allow SOAP messages to be effectively exchanged using a transport protocol like HTTP & SMTP.
* HTTP is synchronous and widely used. A SOAP HTTP request specifies at least two HTTP headers: Content-Type and Content-Length.

**CONTENT TYPE:**

The Content-Type header for a SOAP request and response defines the MIME type for the message and the character encoding (optional) used for the XML body of the request or response.

**EXAMPLE:** POST /item HTTP/1.1  
Content-Type: application/soap+xml; charset=utf-8

**CONTENT LENGTH:**

The Content-Length header for a SOAP request and response specifies the number of bytes in the body of the request or response.

**EXAMPLE:** POST /item HTTP/1.1  
Content-Type: application/soap+xml; charset=utf-8  
Content-Length: 250