1. A web service is a collection of open protocols and standards used for exchanging data between applications or systems. Software applications written in various programming languages and running on various platforms can use web services to exchange data over computer networks like the Internet in a manner similar to inter-process communication on a single computer.
2. Simple object access protocol (SOAP) - it us based on XML message format.

(Representational state transfer) REST and XML- REST uses the simple HTTP request just like the regular browsers that would send to a web server. Here the data is transferred in form of XML document

REST and JSON- the data is transferred to JSON over XML

XML RPC has both a response and a request format

**Advantages of using web services.**

1. **Standardized protocol** – web services use a standardized protocol for the purposes of communication. All the four layers use well defined protocols in the web services protocol stack that is service transport, XML messaging, service description and service discovery layers.

**Low cost communication-** SOAP and HTTP protocol are used since they still exist and can be used to implement a web service

**Interoperability**- they allow numerous applications to share data and services among themselves and other applications can also use the web services.

**Exposing the existing function on the network-** allows a programmer to expose the functionality of the existing code over a network. Once the functionality has been exposed other applications can use it

**Disadvantage of using web services**

Web services use plain text protocols that use fairly verbose method to identify data indicating that web services requests are more that the request encoded with a binary protocol.

HTTP and HTTPS were not meant for long term session. A browser makes an HTTP connection, requests a Web page and maybe some images, and then disconnects.

HTTP and HTTPS are stateless- the interaction between the server and client is typically brief and when there is no data being exchanged, the server and client have no knowledge of each other.

1. Service oriented architecture- is a style of software design where services are provided to the other components by [application components](https://en.wikipedia.org/wiki/Application_components), through a [communication protocol](https://en.wikipedia.org/wiki/Communications_protocol) over a network.
2. Message Exchanging Formats means the way in which the client XML structure should look like so that the provider can understand the document. there are 4 Message Encoding Formats

* Remote Procedure Call-Encoded
* Remote Procedure Call-literal
* document-encoded (This combination is not recommended by WS-I BP 1.0 & 2.0)
* document-literal

while web service standards is the definition of a set of services supporting the description and discovery of businesses, organizations, and other web services providers; the web services they make available; and the technical interfaces which may be used to access those services.

var xmlHttp=new XMLHttpRequest();

var url="<https://jsonplaceholder.typicode.com/posts>";

xmlHttp.onreadystatechange=function(){

if(this.readyState==4 && this.status==200){

var myArr= JSON.parse(this.responseText);

alert("number of posts: "+myArr.length+" ");

} } xmlHttp.open("GET",url,true); xmlHttp.send();

$.get("[https://jsonplaceholder.typicode.com/posts",function](https://jsonplaceholder.typicode.com/posts%22,function) (data,status) {

if (status=="success") {

alert("the number of posts is "+data.length);

}else {

alert("the status is "+status);

}

}

);