API

What all challenge are included under API testing?

API Documentation

Access to DB

Authorization overhead

List out few Authentication Techniques used in API’s?

Session/cookie Based Authentication.

Basic Authentication

Digest Authentication

OAuth

What exactly needs to verify in API testing?

We will verify the accuracy of data

Will see the HTTP status code

We will see the response time

Error codes in case API returns any errors

Authorization would be check

\*\*\* What is the API challans?

-Parameter selection

-Parameter combination

-Call sequencing

\*\*\* Do know what are the principal of API testing?

-Setup

-Execution

-Verification

-Reporting

-Clean up

\*S\*1 What is API ?

Application programing interface (API) that communicates two or multiple applications/program/servers and transfer data/ information.

\*\*\*2 API Communication happens in which level?

1. Presentation level = GUI (web System/leir)/ Graphics / Application >>> Selenium
2. Business level / logic leir = req level/business need >>>API
3. Database level = Store the data (Oracle)

\*\*\* Do you use any third party application or which third party connect your company? J(MVR >>motor vehicle report)

\*\*\* 3 You have Selenium Automation; then why do you need API testing?

1. Communicate between two or multiple program/application/server .
2. Not working in GUI level. (glue : Performance will be faster)
3. API works in business level (requirement & functions level )
4. Clint application can’t handle by selenium, but API can handle client application, API directly communicate with any application server or Database
5. API testing is platform independent and it uses the XML and JSON language

\*\*\* 4 Advantage of API Automation over selenium Automation

|  |  |
| --- | --- |
| API Automation | Selenium Automation |
| Client application / Web application/server/database | Only Web based application (HTML) |
| Faster (No GUI) (why faster-don’t work GUI level) | Slower than API |

\*\*\*5 What kind of application do you handle with API testing ?

1. My current project multiple applications and I handle application based on technology

* Web based application (HTML & Java) >>Selenium+java+testing+cucumber
* Web based application (Angular JS) >> Protector
* Client application >>UTP/QTP(tool),Test complete (tool),API
* No application or no GUI (directly need to talk with server) >> API
* Complex setup with multiple servers & router >> API

\*\*\*6 API types or classification

SOAP message is an ordinary XML document that contains the elements as a SOAP message.

Elements of SOAP message structure= Envelope, Header and Body.

Envelope= envelop defines the start and end of the message.

Header= (Authentication and Parameters) header is an optional element which contains information about the message being sent.

Body= Body contains XML Data message.

### How to find a broken link in your web application?

1st Find out all link by xpath of herf or Linktext

Then check the status code 200 or 400

If it is 200 then link is good

If it is 400 then it is a bad link or broken link.

API has different type:

1. Webservice Testing (OLD)

* SOAP (Simple object access protocol, More secure and reliable (XML only)
* REST (Representational state transfer architecture , Less secure and reliable (XML & JSON)

1. API (NEW)

* SOAP >> OLD
* REST >>NEW

a. Free API

b. Paid API

c. Private API

d. Public API (open for all)

\*\*\* 7 Tell me about API communication process or how API interact or how works internally?

1. API do bidirectional communication
2. Request(you) --------------------- Server / Application (one direction)
3. Response(you) ------------------- Server / Application (another direction)
4. This communication follow some rules (Known as protocol)

* HTTP >> most popular protocol in the market
* SMTP
* FTP

\*\*\* What is the API challans?

-Parameter selection

-Parameter combination

-Call sequencing

\*\*\* Do know what are the principal of API testing?

-Setup

-Execution

-Verification

-Reporting

-Clean up

\*\*\*8 What is HTTP? Difference between HTTP vs HTTPs?

* HTTP (Hypertext transfer protocol) is an API protocol

|  |  |
| --- | --- |
| HTTP | HTTPs |
| Not secured | Secured |
| Can not handle sensitive information | Used during login & payment |
| Hypertext transfer protocol | Hypertext transfer protocol secure |

\*\*\* 9 Can you tell me HTTP methods name?

1. GET() >> get data from server/Application
2. POST() >> send data to server /Application
3. PUT() >> update (whole or all) information /data to server/Application
4. PATCH() >> update(specific data) information /data to server/Application
5. DELETE() >> data information /data from server /Application

\*\*\*10 Difference between HTTP Get() vs Post()

|  |  |
| --- | --- |
| GET() | POST() |
| Get data from server/Application | Send data to server/Application |
| Get request sent via URL | Post requests send via HTTP message body |
| Get request remain in browser history and it can be cashed or bookmarked | Not possible |
| Length restriction: maximum URL length is 2048 characters | No restriction |
| Never be used when dealing with sensitive data | Yes |

\*\*\* 11 Difference between HTTP Post() vs Put()

|  |  |
| --- | --- |
| Post() | Put() |
| Send data to server /Application | Update data |
| Calling a POST request repeatedly have side effects of creating the same resource or data multiple times | Calling the same PUT request multiple times will always produce the same result |

\*\*\*12 Difference between HTTP Put() vs Patch()

|  |  |
| --- | --- |
| Put() | Patch() |
| To update data | To update data |
| Can update all data inside server /Application | Can update specific data inside server /Application |

\*\*\* 13 Difference between URL vs URI

|  |  |
| --- | --- |
| URL | URI |
| Uniform Resource Locator | Uniform Resource identify |
| Its gives server location | It provides server identity |
| Its like address | Its like ID,SSN |

\*\*\* 14 Difference between internet vs Intranets

|  |  |
| --- | --- |
| Internet | Intranets |
| Open to connect anywhere in the world | Only inside the office can not go outside office |
| No restriction | Restricted |

\*\*\* 15 Tell me about HTTP status code

We need to understand some code to understand API testing status (pass/fail)

* 1XX >> informational
* 2XX >>Success/OK(Test pass)

|  |  |
| --- | --- |
| 200 code | 201 (sometimes 202 or 200) |
| Get(), Put(), Delete() | Post() |

* 3xx >> redirection
* 4XX >> Client Error/application Error (Application down) >> Test Fail
* 5XX >> Server error (Server down) >> Test Fail

\*\*\* 16 Important status code with meaning for interview

|  |  |  |  |
| --- | --- | --- | --- |
| 4XX status(Client) | meaning | 5XX status(Server) | meaning |
| 400 | Bad request | 500 | Internal server error |
| 404 | Not found | 502 | Bad Gateway |
| 401 | Unauthorized | 503 | Service unavailable |
| 403 | Forbidden | 504 | Gateway Timeout |
| 406 | Not acceptable |  |  |

\*\*\* 17. API Testing process

1. How to test

* Manual >> by using postman tool
* Automation >> by using Rest Assured

1. What to Test ?

* HTTP methods (like get,post,put,delete)
* Check HTTP status code to understand test status passed or failed
* Testing data is null or empty
* Validate data by testing assert & verify (soft assert)

1. Testing types handle by API

It can handle any kind of testing except GUI testing

* Smoke
* Sanity
* Functional
* Regression
* Performance
* ETOE Testing
* GUI testing >>No
* Unit testing
* Integration testing

1. API Documentation

* Need all manual & automation usual docs like test strategy, test plan, RTM
* Also, must need one more document know as API document

1. What is API document?

- API document >> Give idea to new hire employee to understand

1. General information about the API

2. Methods \*\*\* Who wrote the API document . Ans: Dev and me

3. Setup

4. Testing process

5. How to handle data

- Created by Dev or QA

- API document template

1. In my current project: my Dev created API document in JIRA confluence

2. Also, I know Swagger document template is very popular in the market

f. API Manual testing process

- Need to understand requirements – acceptance criteria

- Write test cases in excel or JIRA xray

- Tool for manual testing >>Postman

- Application or server need to ready and stable before testing

- Check API Authentication for login

- Test HTTP methods and status code

|  |  |  |
| --- | --- | --- |
| HTTP method | Passed | Failed |
| Get | 2XX (200) | 4XX/5XX |
| Post | 2XX(201) | 4XX/5XX |
| Put | 2XX(200) | 4XX/5XX |
| Delete | 2XX(200) | 4XX/5XX |

1. How to validate Manual API testing in postman

* Postman has options to write small code in javaScript language to validate testing or data
* Where to write: HTTP method >> test >> use test templates to write test cases

1. What are parts of HTTP request or response ?

* Header >> all information like data type (JSON/XML)
* Body >> It contains actual message or data

1. What kind of data need to handle in manual or Automation API testing ?

* SOAP >> XML (extensible markup language)
* REST >> JSON (JavaScript object notation)

\*\*\*18. What is JSON ?

* JSON >> JavaScript object notation
* It’s a programing language
* Used in middle end or webserver
* Difference between HTML vs JSON?

|  |  |
| --- | --- |
| HTML | JSON/XML |
| Display data | Transfer data |
| Front end | Middle end language |
| <> | XML <>, JSON {} |

## KEY DIFFERENCES:

* Web Server is s responsible for accepting HTTP requests from clients and serving back that HTTP responses whereas Application server exposes business logic to the clients, which generates dynamic content.
* Web servers are used for producing produce static or dynamic, hypertext documents and Application servers use for text document generation for the computation on provided data.
* Web server consumes fewer resources like CPU memory compared with the application server while the application server utilizes more resources.
* The web server supports HTTP/s Protocol but the application server supports HTTP/s and RPC/RMI protocols.
* Web server provides an environment to run a web application but the application server gives an environment to run the web with enterprise applications.

## Features of Web Server:

Here are important features of a web server:

* Handles HTTP Protocol (static contents)
* No Server-side Programming.
* Support web-Based Applications (JSP, Servlets, PHP, HTML, etc.)
* Not support Database Connection Pooling.
* Not provide EJB support.

## Features of Application Server:

Here are important features of a web server:

* Serves dynamic business logic.
* It helps you to manage backend logic like calculations, database, processing, etc.
* It helps you to deploy applications, dependency injection, security, etc. database pooling, and EJB.
* The superior server of Web Server.

Difference Between Application Server & Web Server

Here are some important differences between Web Server and Application Server:

|  |  |
| --- | --- |
| **Web Server** | **Application Server** |
| Web Server is a computer program or a computer that runs the application. | An application server is a sever type that helps you to host applications. |
| It is responsible for accepting HTTP requests from clients and serving back that HTTP responses. | It exposes business logic to the clients, which generates dynamic content. |
| Subset of the application server. | Superset of a web server. |
| Web servers are used for producing produce static or dynamic, hypertext documents. | Application servers use for text document generation for the computation on provided data. |
| Web servers need a web browser to display the result of the HTTP request. This is commonly known as an HTTP response. | The client-side application is need to continuous data exchange between the application server and client application. |
| It provides an environment for server-side programs to execute and produce HTTP response in results. | The application server offers an ambiance to expose the functionality of the software installed at the server-side to the clients. |
| Web servers are accessed using HTTP request and HTTP protocols. | Application servers are accessed using APIs. |
| The result is a hypertext document storing the information which is displayed to the user on a web browser. | The result is XML, JSON, and HTML, etc. files that contain required data and can serve a special purpose depending upon the user's needs. |
| The web server delivers static content. | The application server delivers dynamic content. |
| Reduces longer running processes that are resource-intensive. | Reduces web traffic, which is not resource-intensive. |
| Web server consumes fewer resources like CPU memory compared with the application server. | The application server utilizes more resources. |
| The web server supports HTTP Protocol. | The application server supports HTTP and RPC/RMI protocols. |
| Web server enables to serve web-based applications. | The application server helps you to serve web-based applications as well as enterprise-based applications. |
| Web servers support scripting languages like PHP, Perl, JSP, ASP, etc. | Assist Application server services such as transaction support, connection pooling, object pooling. |
| The web server does not support multithreading. | Application server assists multithreading and distributed transactions. |
| Web servers are primarily designed to serve HTTP content. | Application server can also serve HTTP content. However, it is also assists protocol like RMI/RPC. |
| It provides an environment to run a web application. | The application server gives an environment to run the web with enterprise applications. |

**1. What is API Testing?**

[API testing](https://www.katalon.com/resources-center/tutorials/introduction-api-testing/) is a kind of software testing which determines if the developed APIs meet expectations regarding the functionality, reliability, performance, and security of the application.

2. What are the advantages of API Testing?

In an API interview, they are likely to ask about the advantages of API testing. So be prepared with the significant ones such as:

* Test for Core Functionality:API testing provides access to the application without a user interface. The core and code-level of functionalities of the application will be tested and evaluated early before the GUI tests. This will help detect the minor issues which can become bigger during the GUI testing.
* ***Time Effective:*** API testing usually is less time consuming than functional GUI testing. The web elements in GUI testing must be polled, which makes the testing process slower. Particularly, API test automation requires less code so it can provide better and faster test coverage compared to GUI test automation. These will result in the cost saving for the testing project.
* Language-Independent: In API testing, data is exchanged using XML or JSON. These transfer modes are completely language-independent, allowing users to select any code language when adopting automation testing services for the project.
* ***Easy Integration with GUI:*** API tests enable highly integrable tests, which is particularly useful if you want to perform functional GUI tests after API testing. For instance, simple integration would allow new user accounts to be created within the application before a GUI test started.

### 3) What are the types of API testing?

API testing involves the following types of testing:

* Unit Testing
* Functional Testing
* Load Testing
* Runtime/Error Detection
* Security Testing
* UI Testing
* Interoperability and WS compliance Testing
* Penetration Testing
* Fuzz Testing

4) What are the protocols used in API Testing?

Protocols used in API testing are:

* HTTP
* REST
* SOAP
* JMS
* UDDI

5) What are the tools used for API Testing?

Tools used for API testing are:

* Parasoft SOAtest
* PostMan
* AlertSite API monitoring

6) What is API test environment?

For API the test environment is a quite complex method where the configuration of server and database is done as per the requirement of the software application. API testing does not involve graphical user interface (GUI).

API is checked for its proper functioning after installation.

7) What is API framework?

API framework is described by the config. File which consist of the list of all APIs that are required to be activated and are activated for any particular program run. This is essential as every test run does not require all APIs.

8) What are the limits of API usage?

Many APIs have certain limit set up by the provider. Hence, try to estimate our usage and understand how that will impact the overall cost of the offering.

**9) What are the common tests performed on API’s?**

The common tests performed on API’s

* Verification of the API whether it is updating any data structure
* Verify if the API does not return anything
* Based on input conditions, returned values from the API’s are checked
* Verification of the API whether it triggers some other event or calls another API

**10) Mention the key difference between UI level testing and API testing?**

UI ( User Interface) refers to testing graphical interface such as how user interacts with the applications, testing application elements like fonts, images, layouts etc. UI testing basically focuses on look and feel of an application.

While, API enables communication between two separate software systems. A software system implementing an API contains functions or sub-routines that can be executed by another software system

**11) Explain what is SOAP?**

SOAP-stands for Simple Object Access Protocol, and it is an XML based protocol for exchanging information between computers.

**12) Explain what is REST API?**

It is a set of functions to which the developers perform requests and receive responses. In REST API interaction is made via HTTP protocol

REST – stands for Representational State Transfer, it is quickly becoming defacto standard for API creation.

**13) Difference API and Unit Testing?**

|  |  |
| --- | --- |
| API testing | UNIT testing |
| * API is owned by QA team | * Unit testing is owned by development team |
| * API is mostly black box testing | * Unit testing is white box testing |
| * Full functionality of the system is considered in API testing as it will be used by the end-user (external developers who will use your API ) | * Unit testing is done to verify whether each unit in isolation performs as expected or not |
| * API test are often run after the build is ready and authors do not have access to the source code | * For each of their module the developers are expected to build unit tests for each of their code modules and have to ensure that each module pass unit test before the code is included in a build |

**14) How to test API’s ?**

To test the API’s you should follow the following steps

* Select the suite in which you want to add the API test case
* Choose test development mode
* Develop test cases for the desired API methods
* Configure application control parameters
* Configure test conditions
* Configure method validation
* Execute API test
* View test reports
* Filter API test cases
* Sequence API test cases

**15) Mention what the main areas to be taken in consideration while writing API document ?**

The key area to be considered when writing API documents are

* Source of the content
* Document plan or sketch
* Delivery layout
* Information required for each function in the document
* Automatic document creation programs

**16) In API document explain how to document each function? What are the tools used for documentation?**

* **Description:** Small description about what a function does
* **Syntax:** Syntax about the parameter of the code, the sequence in which they occur, required and optional elements etc.
* **Parameters:** Functions parameters
* **Error Messages:** Syntax of error messages
* **Example Code:** Small snippet of code
* **Related Links:** Related functions

Popular tools used for API documentations are JavaDoc (for Java code ) Doxygen (for .Net code)

**17) Explain API framework?**

API framework is self-explanatory. Values for test run and for holding the configurable parts, config file is used.  Automated test cases must represent in “ parse-table” format within config file.  When testing API, it is not necessary to test each API so the config file have some section whose all API are activated for that specific run.

**18) How does the API Builder work?**

API Builder is a PLSQL program consists of four SQL files

* For setting API parameters and starting the process one file is responsible
* Two files are created for temporary tables and Master package to create the outputted code
* Fourth file creates “spooled” output of the code into a file called “output\_script\_.sql”

**19) Explain what is TestApi ?**

TestApi is a library of utility and test APIs that enables testers and developers to create testing tools and automated tests for .NET and Win32 application.  It provides a set of common test building blocks, types, data-structure and algorithms.

**20) What is Input injection and what are different ways of doing it ?**

Input Injection:  It is the act of simulating user input, in several ways you can simulate user input.

* Direct Method Invocation
* Invocation using an accessibility interface
* Simulation using low-level input
* Simulation using a device driver
* Simulation using a robot

**21) What are the main challenges of API testing?**

The main challenges in API testing is

* Parameter Selection
* Parameter Combination
* Call sequencing

**22) What is API testing with runscope ?**

Runscope is a web application that provides backend services and easy to use interface for testing APIs.

**23) Explain what are the principles of API test design?**

The principle for API test design are

* **Setup :** Create objects, start services, initialize data etc
* **Execution:** Steps to exercise API or scenario, also logging
* **Verification:** Oracles to evaluate execution outcome
* **Reporting:** Pass, failed or blocked
* **Clean up:** Pre-test state

**24) What are the types of Bugs will API testing finds?**

The types of Bugs, API will find

* Missing or duplicate functionality
* Fails to handle error conditions gracefully
* Stress
* Reliability
* Security
* Unused flags
* Not implemented errors
* Inconsistent error handling
* Performance
* Multi-threading issues
* Improper errors

**25) What are the tools used for API test automation?**

While testing Unit and API testing,  both target source code, if an API method is using code  based on .NET then the tool which is supporting should have .NET

Automation tools for API testing can be used are

* NUnit for .NET
* JUnit for Java
* HP UFT
* Soap UI

**26) Mention the steps for testing API ?**

API testing steps

* Select the test case that has to be fulfilled
* For API call develop a test case
* To meet the test case configure the API parameters
* Determine how will you validate a successful test
* Using programming language like PHP or .NET execute the API call
* Allow the API call to return the data to validate

**27) What are the common protocols that are testing in API tesing ?**

* HTTP
* JMS
* REST
* SOAP
* UDDI

### 28) What is API framework?

A framework or software framework is a platform for developing software applications. API framework is a foundation on which software developer can build applications for a specific platform.

**Example:** A framework can include predefined classes and functions that can be used to process input, manage hardware devices and interact with system software.

Framework is similar to an Application Programming Interface, technically framework includes API. Framework serves foundation for programming while API provides access to the elements supported by the framework. Framework also includes code libraries, compiler and other programs used in the software development process.

API framework is defined by configuration file which consists the list of all APIs that is required to be activated and activated for a particular program run.

### 29) What are the differences between API and Web Services?

|  |  |  |
| --- | --- | --- |
| **Sr. No.** | **API** | **Web Services** |
| **1.** | API may or may not need network for its operations. | Web Services always need network for its operation. |
| **2.** | API can be communicated through SOAP, REST, XML-RPC and CURL calls as well. API can also be exposed in number of ways like JAR, DLL, XML over HTTP, JSON over HTTP etc. | Web service can be communicated through SOAP, REST, AND RPC. |
| **3.** | API can perform all the operations which web service can't perform. | Web service can't perform all the operations like API. |
| **4.** | All APIs are not web service. | All web services are API |

30) What exactly needs to verify in API testing?

In API testing, we send a request to API with the known data and then analysis the response.

1. We will verify the accuracy of the data.
2. Will see the HTTP status code.
3. We will see the response time.
4. Error codes in case API returns any errors.
5. Authorization would be check.
6. Non-Functional testing such as performance testing, security testing.

### 31) What are the differences between SOAP and REST API?

|  |  |  |
| --- | --- | --- |
| **Sr. No.** | **SOAP API** | **REST API** |
| **1.** | **SOAP** stands as Simple Object Access Protocol. | **REST** stands as Representational State Transfer. |
| **2.** | **SOAP** is a protocol. | **REST** is an architectural pattern. |
| **3.** | **SOAP** can work with XML format. In SOAP all the data passed in XML format. | **REST** permit different data format such as Plain text, HTML, XML, JSON etc. But the most preferred format for transferring data is in JSON. |

32) What is a RESTFUL web services?

There are two kinds of web services

1. SOAP Web Services
2. RESTFUL Web Services

**1. SOAP (Simple Object Access Protocol) -** SOAP is a XML based method which is used in Web Services.

**2. RESTFUL Web Services -** To implement the concept of REST architecture HTTP method is used. RESTFUL Web Services defines URI (Uniform Resource Identifier), and also provides resource representation like JSON and a set of HTTP method.

33) What is Resource in REST?

REST architecture treats any content as resource, which can be text files, HTML pages, images, videos or dynamic business information. REST server gives the functionality to access the resources and modifies them. We can identify the each resources by URIs/ global IDs.

34) What is the way to represent the resource in REST?

REST uses different representation to define the resources like text, JSON and XML. The most popular representation of resources is JSON and XML.

35) What protocol is used by the RESTFUL Web Services?

RESTFUL Web Services uses the HTTP protocol. They use the HTTP protocol as a medium of communication between the client and the server.

36) What are the characteristics of REST?

Here, are the two characteristics of REST.

1. REST is stateless. With the use of the REST API the server has no status, we can restart the server between two calls, inspite of all the data is transferred to the server.
2. Web Services uses POST method to perform operations, while REST uses GET method to access the resources.

37) What is messaging in RESTFUL Web Services?

RESTFUL Web Services use the HTTP protocol as a communication tool between the client and the server. This is the technique when the client sends a message in the form of HTTP request the server send back the HTTP reply which is called Messaging. This message consists message data and Meta data i.e. information on the message itself.

38) What are the components of an HTTP request?

An HTTP request have five components. These are:

1. **Action showing HTTP method** like GET, PUT, POST, DELETE.
2. **Uniform Resource Identifier (URI):** URI is the identifier for the resource on the server.
3. **HTTP version:** Indicate the HTTP version like- HTTP V1.1.
4. **Request Header:** Request Header carries metadata for the HTTP request message. Metadata could be a client type, format supported by the client, format of a message body, cache setting etc.
5. **Request Body:** Resource body indicates message content or resource representation.

39) What is the HTTP protocol supported by REST?

**GET:** GET is used to request data from the specified resource.

GET request can be cached and bookmark. It remains in the browser history and has length restriction. When dealing with sensitive data GET requests should not be used.

**POST:** POST is used to send data to server for creation or updating the resources.

POST requests are never cached or bookmark.

**PUT:** PUT replaces the current representation of the target resource with the request payload.

**DELETE:** DELETE removes the specified resource.

**OPTIONS:** OPTION is used to describe the communication option for the target resources.

**HEAD:** HEAD asks for response which is identical to GET requests, but without the response body.

40) Can we use GET request instead of PUT to create a resource?

PUT or POST method is used create a resource. GET is only used to request the resources.

41) What is URI? What is the purpose of web-based service and what is it's format?

URI stands for Uniform Resource Identifier. It is a string of characters designed for unambiguous identification of resources and extensibility by the URI scheme. The purpose of URI is to locate the resource on the server hosting of the web service.

A URIs format is ***<protocol>://<service-name>/<Resource Type>/<ResourceID>***

42) What are SOAP Web Services?

**SOAP** (*Simple Object Access Protocol*) is defined as the XML based protocol. SOAP is also known for developing and designing web services and also enable the communication between the applications developed on different platform by using different programming languages on the internet. SOAP is platform and language independent.

43) When we can use SOAP API?

We can use SOAP API to perform the operation on records like create, retrieve, update or delete. We can use API to manage password, perform searches etc.

# What is web service ?

# A web service is an [application](https://techterms.com/definition/application) or [data](https://techterms.com/definition/data) source that is accessible via a standard web protocol ([HTTP](https://techterms.com/definition/http) or [HTTPS](https://techterms.com/definition/https)). Unlike [web applications](https://techterms.com/definition/web_application), web services are designed to communicate with other [programs](https://techterms.com/definition/program), rather than directly with users. While web services can provide data in a number of different formats, [XML](https://techterms.com/definition/xml) and [JSON](https://techterms.com/definition/json) are the most common. These standard text-based formats can be easily recognized and [parsed](https://techterms.com/definition/parse) by another program that receives the data. The most common web service protocol – [SOAP](https://techterms.com/definition/soap) (Simple Object Access Protocol) – simply adds a [header](https://techterms.com/definition/header) to each XML message before it is transferred over HTTP.

Business-oriented web services may use a standard called [UDDI](https://techterms.com/definition/uddi). This formats data in a specific type of XML known as the Web Services Description Language, or WSDL. While UDDI transmits [.WSDL](https://fileinfo.com/extension/wsdl) files instead of standard [.XML](https://fileinfo.com/extension/xml) files, it may still use the SOAP [protocol](https://techterms.com/definition/protocol) to transfer data.

Most web services provide an [API](https://techterms.com/definition/api), or a set of functions and commands, that can be used to access the data. For example, [Twitter](https://techterms.com/definition/twitter) provides an API that allows [developers](https://techterms.com/definition/developer) to access [tweets](https://techterms.com/definition/tweet) from the service and receive the data in JSON format. Yelp provides an API for programmers to access information about businesses, which can be displayed directly in an [app](https://techterms.com/definition/app) or [website](https://techterms.com/definition/website). Google Maps provides an API for receiving geographical data and directions from the Google Maps [database](https://techterms.com/definition/database).

**NOTE:** An *API* is a specific set of commands and guidelines used for accessing data, while a *web service* is an actual service provided by an Internet-based source.

Source : javaTpoint,99

1