**API**

**(Application Programing Interface)**

API stands for Application Programming Interface. In the context of APIs, the word Application refers to any software with a distinct function. Interface can be thought of as a contract of service between two applications. This contract defines how the two communicate with each other using requests and responses.

It act as a bridge between different software and devices.

API architecture is usually explained in terms of client and server. The application sending the request is called the client, and the application sending the response is called the server.

There are four different ways that APIs can work depending on when and why they were created.

**SOAP APIs**

These APIs use Simple Object Access Protocol. Client and server exchange messages using XML. This is a less flexible API that was more popular in the past.

**RPC APIs**

These APIs are called Remote Procedure Calls. The client completes a function (or procedure) on the server, and the server sends the output back to the client.

**Websocket APIs**

[Websocket API](https://docs.aws.amazon.com/apigateway/latest/developerguide/apigateway-websocket-api-overview?pg=wianapi&cta=websocketapi) is another modern web API development that uses JSON objects to pass data. A WebSocket API supports two-way communication between client apps and the server. The server can send callback messages to connected clients, making it more efficient than REST API.

**REST APIs**

These are the most popular and flexible APIs found on the web today. The client sends requests to the server as data. The server uses this client input to start internal functions and returns output data back to the client.

## **What are REST APIs?**

REST stands for Representational State Transfer. REST defines a set of functions like GET, PUT, DELETE, etc. that clients can use to access server data. Clients and servers exchange data using HTTP.

## **What are the benefits of REST APIs?**

### 1. Integration

### 2. Innovation

### 3. Expansion

### 4. Ease of maintenance

## **What are the different types of APIs?**

APIs are classified both according to their architecture and scope of use. We have already explored the main types of API architectures so let’s take a look at the scope of use.

### Private APIs

These are internal to an enterprise and only used for connecting systems and data within the business.

### Public APIs

These are open to the public and may be used by anyone. There may or not be some authorization and cost associated with these types of APIs.

### Partner APIs

These are only accessible by authorized external developers to aid business-to-business partnerships.

### Composite APIs

These combine two or more different APIs to address complex system requirements or behaviors.

You would have to use HTTP method like GET,POST,PUT,DELETE,etc to make use of the endpoint

|  |  |  |
| --- | --- | --- |
| **No.** | **SOAP** | **REST** |
| 1) | SOAP is a **protocol**. | REST is an **architectural style**. |
| 2) | SOAP stands for **Simple Object Access Protocol**. | REST stands for **Representational State Transfer**. |
| 3) | SOAP **can't use REST** because it is a protocol. | REST **can use SOAP** web services because it is a concept and can use any protocol like HTTP, SOAP. |
| 4) | SOAP **uses services interfaces to expose the business logic**. | REST **uses URI to expose business logic**. |
| 5) | **JAX-WS** is the java API for SOAP web services. | **JAX-RS** is the java API for RESTful web services. |
| 6) | SOAP **defines standards**to be strictly followed. | REST does not define too much standards like SOAP. |
| 7) | SOAP **requires more bandwidth** and resource than REST. | REST **requires less bandwidth** and resource than SOAP. |
| 8) | SOAP **defines its own security**. | RESTful web services **inherits security measures** from the underlying transport. |
| 9) | SOAP **permits XML** data format only. | REST **permits different** data format such as Plain text, HTML, XML, JSON etc. |
| 10) | SOAP is **less preferred** than REST. | REST **more preferred** than SOAP. |

# 5 Essential HTTP Methods in RESTful API Development

**GET**

Use GET requests **to retrieve resource representation/information only** – and not modify it in any way. As GET requests do not change the resource’s state, these are said to be **safe methods**.

2. POST

The POST method sends data to create a ‘new record ‘on the server.

3. PUT

The PUT method sends data to update an ‘existing record ‘on the server.

4.PATCH

Like the PUT method, PATCH is also used to send data to update an ‘existing record’ on the server. But the important difference between PUT and PATCH is that PATCH only applies partial modifications to the record instead of replacing the whole record.

5.DELETE

The DELETE method is used to delete record(s) from the server.