

## API :

**API is an abbreviation for Application Programming Interface** which is a collection of communication protocols and subroutines used by various programs to communicate between them. an API helps two programs or applications to communicate with each other by providing them with the necessary tools and functions. It takes the request from the user and sends it to the service provider and then again sends the result generated from the service provider to the desired user.

APIs are considered safe in terms of attacks as it includes authorization credentials and an API gateway to limit access so as to minimize security threats. To provide additional security layers to the data, HTTP headers, query string parameters, or cookies are used.

If we talk about the architectures, API's architectures are:

**Representational State Transfer (REST)** is an architectural style that defines a set of constraints to be used for creating web services. **REST API** is a way of accessing web services in a simple and flexible way without having any processing.

**Simple Object Access Protocol(SOAP)** is a network protocol for exchanging structured data between nodes

## Types of APIs:

1. WEB APIs
2. LOCAL APIs
3. PROGRAM APIs

**Make a GET Request and POST Request in JavaScript**

`fetch()` is a mechanism that lets you make simple AJAX (Asynchronous JavaScript and XML) calls with JavaScript.

When a response (data) is sent back from the API, the asynchronous tasks (fetch) resume. If it still sounds difficult, you can read my detailed introduction into [Asynchronous code](#).

GET request.

Simply call `fetch()` with the endpoint URL as the argument:

the fetch API returns a promise. Because of this, you need to nest a `then()` method to handle the resolution.

The data returned from the API is not usually in a useable form. So you'll need to convert the data to a form which your JavaScript can operate with. Thankfully, you can use the `json()` method to do just that:

- ... To make a simple GET request with fetch, you just need to pass in the URL endpoint as an argument.

For a post request, you'll need to pass an object of configuration options as a second argument. The optional object can take a lot of different parameters. In this case, include only the most necessary information.

Because you're sending a POST request, you'll need to declare that you're using the POST method.

You'll also need to pass some data to actually create the new blog post. Since you're sending JSON data, you'll need to set a header of *Content-Type* set to *application/json*. Finally, you'll need the body, which will be a single string of JSON data.

- ...To make a POST request, you'll need to pass along certain other parameters including a configuration object.