What is an API? (Beginner Friendly Explaination)

Imagine you're at a restaurant.

You don't go into the kitchen and start cooking, right? You look at a **MENU**.

The menu lists the dishes the restaurant can make and what you need to know about them (like ingredients and prices).

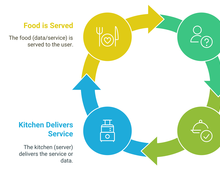
You tell the waiter (**who acts as a go-between**) what you want, and they relay your order to the kitchen.

Then, they bring you the cooked food.

In this analogy:

- **You** are the user or application.
- **The Menu** is the API (Application Programming Interface). It's a list of what the "kitchen" (the service) can do.
- **The Waiter** is the intermediary, handling your request and delivering the result.
- **The Kitchen** is the server or application providing the service.
- **The Food** is the data or service delivered.

API Interaction Cycle



Essentially, an API is a set of rules that allows different software applications to communicate with each other.

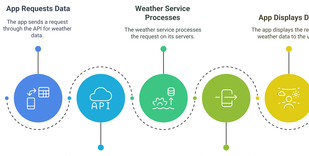
It defines how they can request and exchange information.

Another Real-Life Example: Checking the Weather

You open a weather app on your phone. You see the current temperature, forecast, and maybe even a radar map.

- Your weather app doesn't have its own weather station.
- Instead, it uses an API provided by a weather service (like AccuWeather or OpenWeatherMap).
- Your app sends a request through the API, asking for the weather data for your location.
- The weather service's servers send back the data, and your app displays it.

Weather App API Interaction



Going a Little Deeper: Requests and Responses

APIs work by sending "requests" and receiving "responses."

- **Request:** When you use an app that uses an API, the app sends a request to the server. This request asks for specific information or tells the server to perform a specific action.
- **Response:** The server receives the request, processes it, and sends back a response. This response usually contains the data you asked for or a confirmation that the action was performed.

Types of Data (Don't worry too much about these, but good to know):

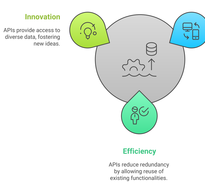
Often, the data sent back and forth is in a format called **JSON (JavaScript Object Notation)**. Think of JSON as a structured way of organising information, like a well-organised list. It's easy for computers (and humans) to read and understand.

```
{
  "model": "gpt-4o",
  "messages": [
    {
      "role": "user",
      "content": "Hello, how are you?"
    }
  ]
}
```

Why are APIs important?

- **They allow different applications to work together:** This makes it possible to create complex services by combining the capabilities of multiple applications.
- **They save time and effort:** Developers don't have to build everything from scratch. They can use existing APIs to access data and functionality.
- **They enable innovation:** APIs make it easier to create new applications and services by providing access to a wide range of data and functionality.

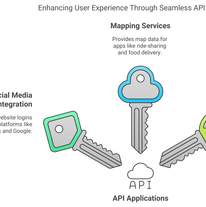
API Contributions to Software Development



Simple Everyday API uses:

- **Social Media Login:** When you log into a website using your Facebook or Google account, that website is using an API to communicate with Facebook or Google.
- **Maps in Apps:** When you see a map in a ride-sharing app or a food delivery app, that map is usually provided by an API from a mapping service like Google Maps.
- **Online Payments:** When you pay for something online, the payment gateway uses APIs to communicate with your bank and the merchant's bank.

Enhancing User Experience Through Seamless API Integration



In essence, APIs are the invisible bridges that connect the digital world, allowing applications to share information and work together seamlessly.

TYPES of API Requests

GET & POST

Imagine you're at a restaurant:

- **GET:** You're looking at the menu to see what dishes are available. You're simply *retrieving* information.
  - **Real-life example:** Checking the weather app for today's forecast. You're *getting* the current weather data.
- **POST:** You're placing an order. You're *sending* information to the kitchen (server) to create something new.
  - **Real-life example:** Placing a food order through a delivery app. You're *sending* your order details to the restaurant.

Choose the appropriate API request method for your task

In API terms:

- **GET:** Used to *retrieve* data from a server. Think of it as asking, "Give me the information about [something]."
- **POST:** Used to *send* data to a server to create, update, or delete something. Think of it as saying, "Here's the information, please [do something with it]."

Key differences:

- **Data visibility:** GET data is visible in the URL, while POST data is hidden.
- **Security:** POST is generally more secure for sensitive information.
- **Data size:** GET has limitations on data size due to URL length restrictions, while POST can handle larger amounts of data.