

[What is an API? - YouTube](#)



API

- Application Programming Interface

An API is an intermediary between a primary service and a secondary service.



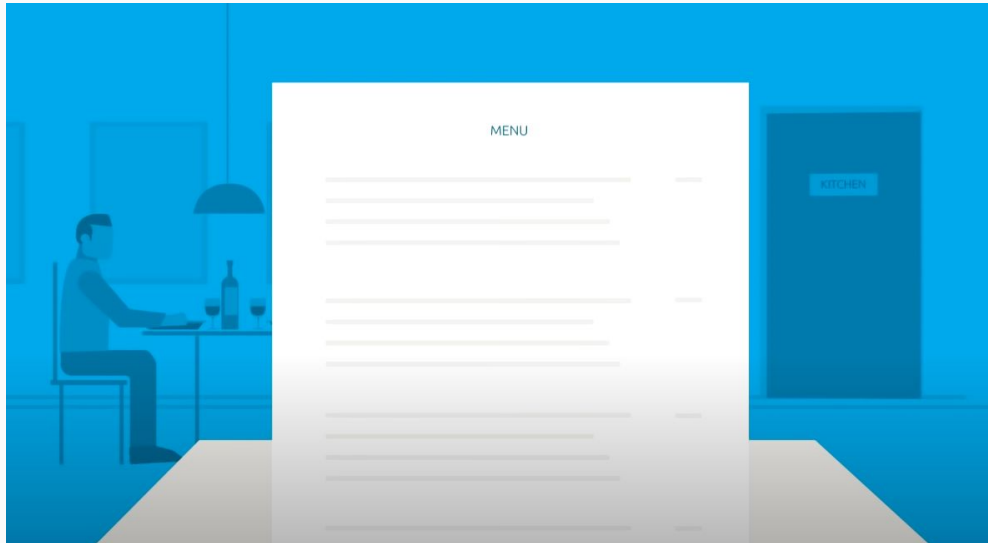
API

- Way for programs (or software) to speak to each other



API

- You'll send a request and get back information



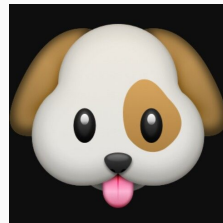
API

- It saves you from having to write code when you can use someone else's information



For example, let's say you wanted to create an app that shows how to get to the nearest dog park.

We don't have to create our own google maps. We can use their existing API.



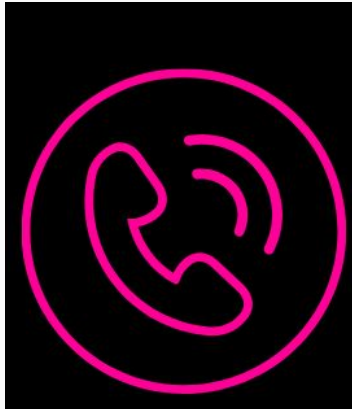
Call an API using:

- console application
- browser
- desktop application
- mobile application
- etc

Client

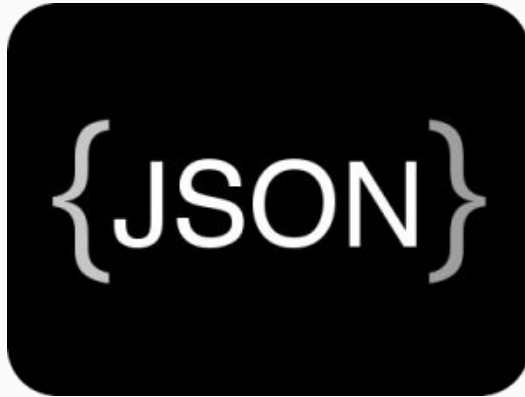
client - application or software that consumes or interacts with an API.

The client sends requests to the API and receives responses in return.



JSON

- Javascript Object Notation



JSON

We will call web APIs and receive a **response containing JSON**.



JSON

- **(JavaScript Object Notation)** is a lightweight data-interchange format.
- It is easy for humans to read and write.
- It is easy for machines to parse and generate.
- Efficient data transfer
- Platform independent - (can be consumed by different types of clients)



JSON

Two structures:

- 1) **Objects**-- curly braces { }
- 2) **Arrays**-- square brackets[]

JSON

Two structures:

- 1) JObject
- 2) JArray

JSON Object

```
{"quote": "Coding is a craft"}
```

JSON Object

Name

Value



The diagram shows a JSON object `{"quote": "Coding is a craft"}` with two annotations. A blue arrow points from the label "Name" to the key `"quote"`, which is underlined in blue. A red arrow points from the label "Value" to the value `"Coding is a craft"`, which is underlined in red.

```
{"quote": "Coding is a craft"}
```

JSON Object

```
{'quote': 'Coding is a craft'}
```


JSON Object

I would use the name to grab the value

```
{"quote": "Coding is a craft"}
```

JSON Array

```
["I enjoy government functions like I enjoy getting kicked in the nuggets with a steel toed boot."]
```

JSON Array

Array one one item (one string)

```
["I enjoy government functions like I enjoy getting kicked in the nuggets with a steel toed boot."]
```

JSON Array

We know it's an array because of the brackets

```
[ 'I enjoy government functions like I enjoy getting kicked in the nuggets with a steel toed boot.' ]
```

JSON Array

Guess who said it

```
["I enjoy government functions like I enjoy getting kicked in the nuggets with a steel toed boot."]
```

JSON Array

Ron Swanson

```
["I enjoy government functions like I enjoy getting kicked in the nuggets with a steel toed boot."]
```



JSON Array

Index 0



```
["I enjoy government functions like I enjoy getting kicked in the nuggets with a steel toed boot."]
```

JSON Object



```
// myObj is a JSON object
myObj = {
  "name": "John",
  "age": 30,
  "cars": [
    { "name": "Ford", "models": [ "Fiesta", "Focus", "Mustang" ] },
    { "name": "BMW", "models": [ "320", "X3", "X5" ] },
    { "name": "Fiat", "models": [ "500", "Panda" ] }
  ]
}
```


- Object has three values
- The value of “cars” is an array of three objects

```
// myObj is a JSON object
myObj = {
  "name": "John",
  "age": 30,
  "cars": [
    { "name": "Ford", "models": [ "Fiesta", "Focus", "Mustang" ] },
    { "name": "BMW", "models": [ "320", "X3", "X5" ] },
    { "name": "Fiat", "models": [ "500", "Panda" ] }
  ]
}
```

This array has three objects

```
// myObj is a JSON object
myObj = {
  "name": "John",
  "age": 30,
  "cars": [
    0 { "name": "Ford", "models": [ "Fiesta", "Focus", "Mustang" ] },
    1 { "name": "BMW", "models": [ "320", "X3", "X5" ] },
    2 { "name": "Fiat", "models": [ "500", "Panda" ] }
  ]
}
```

We'll try to grab this value

```
// myObj is a JSON object
myObj = {
  "name": "John",
  "age": 30,
  "cars": [
    { "name": "Ford", "models": [ "Fiesta", "Focus", "Mustang" ] },
    { "name": "BMW", "models": [ "320", "X3", "X5" ] },
    { "name": "Fiat", "models": [ "500", "Panda" ] }
  ]
}
```

Use indexing!

```
// myObj is a JSON object
myObj = {
  "name": "John",
  "age": 30,
  "cars": [
    { "name": "Ford", "models": [ "Fiesta", "Focus", "Mustang" ] },
    { "name": "BMW", "models": [ "320", "X3", "X5" ] },
    { "name": "Fiat", "models": [ "500", "Panda" ] }
  ]
}
```

`myObj["cars"][0]["name"] = "Ford"`

```
// myObj is a JSON object
myObj = {
  "name": "John",
  "age": 30,
  "cars": [
    { "name": "Ford", "models": [ "Fiesta", "Focus", "Mustang" ] },
    { "name": "BMW", "models": [ "320", "X3", "X5" ] },
    { "name": "Fiat", "models": [ "500", "Panda" ] }
  ]
}
```

If I wanted to grab “Fiesta”

```
// myObj is a JSON object
myObj = {
  "name": "John",
  "age": 30,
  "cars": [
    { "name": "Ford", "models": [ "Fiesta", "Focus", "Mustang" ] },
    { "name": "BMW", "models": [ "320", "X3", "X5" ] },
    { "name": "Fiat", "models": [ "500", "Panda" ] }
  ]
}
```

`myObj["cars"][0]["models"][0] = "Fiesta"`

```
// myObj is a JSON object
myObj = {
  "name": "John",
  "age": 30,
  "cars": [
    { "name": "Ford", "models": [ "Fiesta", "Focus", "Mustang" ] },
    { "name": "BMW", "models": [ "320", "X3", "X5" ] },
    { "name": "Fiat", "models": [ "500", "Panda" ] }
  ]
}
```

JSON Data Types

[JSON Data Types \(w3schools.com\)](https://www.w3schools.com/js/json_data_types.asp)

Valid Data Types

In JSON, values must be one of the following data types:

- a string
- a number
- an object (JSON object)
- an array
- a boolean
- *null*

Valid Data Types

In JSON, values must be one of the following data types:

- a string
- a number
- an object (JSON object)
- an array
- a boolean
- *null*

```
// A JSON object
{
  "firstName": "John",
  "lastName": "Smith",
  "isAlive": true,
  "age": 27,
  "address": {
    "streetAddress": "21 2nd Street",
    "city": "New York",
    "state": "NY",
    "postalCode": "10021-3100"
  },
  "phoneNumbers": [
    {
      "type": "home",
      "number": "212 555-1234"
    },
    {
      "type": "office",
      "number": "646 555-4567"
    },
    {
      "type": "mobile",
      "number": "123 456-7890"
    }
  ],
  "children": [],
  "spouse": null
}
```

Newtonsoft.Json

- Need to install this NUGET package!

Newtonsoft.Json

- popular library for JSON processing
- Has these classes **JObject** and **JArray**
- JObject and JArray represent different types of JSON structures.

When you hear API think REST



REST API

API's Utilize REST

REST: Representational State Transfer

[What is a REST API? | IBM](#)

REST



Architectural Style

A REST API follows a [client-server model](#), where the client (like a web application) interacts with the server (the API) over HTTP (Hypertext Transfer Protocol)

API's Utilize REST

- REST is a language-independent architecture that provides interoperability between computer systems on the Internet.
- It utilizes a stateless protocol aiming for fast performance and reliability.



Stateless

- Session state is entirely on the client.
- Each request from the client to the server is self-contained and doesn't rely on the server's previous state.
- The client keeps hold of the information and sends a pre-packaged request/series of packets to the server.
- **Nothing is stored on the server.**

REST Verbs



HTTP (Hypertext Transfer Protocol) is the foundation of any data exchange on the Web.

HTTP requests are used to perform actions on the web server. The main HTTP methods used by RESTful APIs are:

It translates like this:

Create	Read	Update	Delete
POST	GET	PUT	DELETE

REST Verbs



POST - Create a new resource on the server.

GET - Retrieve data from the server (ex. fetching a resource).

PUT - Update a resource on the server.

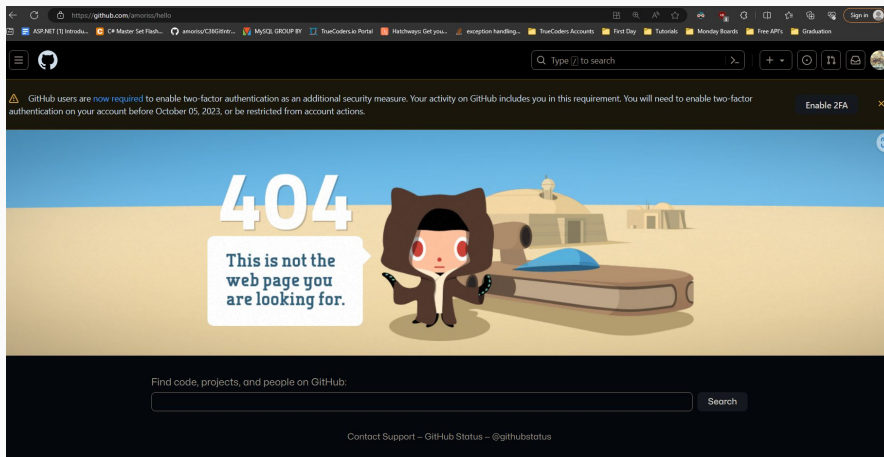
DELETE - Remove a resource from the server.

It translates like this:

Create	Read	Update	Delete
POST	GET	PUT	DELETE

HTTP Status Codes:

- HTTP Status Codes are three-digit numbers returned by a server in response to a client's request, such as a web browser's request for a web page.
- These status codes are part of the HTTP (Hypertext Transfer Protocol) response and are intended to inform the client about the outcome of its request.



HTTP Status Codes:

- **1xx – Informational**
- **2xx – Success**
 - This class of status code indicates that the client's request was successfully received, understood, and accepted.
- **3xx – Redirection**
 - The client must take additional action to complete the request.
- **4xx – Client Error**
 - (Like 404 Page Not Found)
 - The **4xx** class of status code is intended for cases in which the client seems to have erred. Except when responding to a HEAD request, the server should include an entity containing an explanation of the error situation, and whether it is a temporary or permanent condition. These status codes are applicable to any request method. User agents should display any included entity to the user.
- **5xx – Server Error**
 - The server failed to fulfill an apparently valid request

Text to an Object

- Will receive JSON text back → need to parse it into an OBJECT
- C# is an object-oriented language
- We need to convert data into objects
- We will be parsing to do this!

Things we need to get started:

1. URL endpoint
2. Newtonsoft.Json Nuget Package (allows us to parse)
3. HTTP Client (to send requests)

Recommended: Download Postman to your desktop

[Download Postman | Get Started for Free](#)



Postman acts as a Client to send requests



Postman allows us to read our data in a “prettier” format




```

{"id":495,"brand":"maybelline","name":"Maybelline Face Studio Master Hi-Light Light Booster Bronzer","price":"14.99","price_sign":null,"currency":null,"image_link":"https://d3t32hsnjx07q6.cloudfront.net/i/991799d3e70b8856686979f8ff6dcfe0_ra,w158,h184_pa,w158,h184.png","product_link":"https://well.ca/products/maybelline-face-studio-master-88837.html","website_link":"https://well.ca","description":"Maybelline Face Studio Master Hi-Light Light Boosting bronzer formula has an expert \nbalance of shade + shimmer illuminator for natural glow. Skin goes \nsoft-lit with zero glitz.\n\n\t\t\tFor Best Results: Brush over all shades in palette and gently sweep over \ncheekbones, brow bones, and temples, or anywhere light naturally touches\nthe face.\n\n\t\t\t\t\t\n\n\t","rating":5.0,"category":null,"product_type":"bronzer","tag_list":[],"created_at":"2016-10-01T18:36:15.012Z","updated_at":"2017-12-23T21:08:50.624Z","product_api_url":"https://makeup-api.herokuapp.com/api/v1/products/495.json","api_featured_image":"/s3.amazonaws.com/donovanbailey/products/api_featured_images/000/000/495/original/open-uri20171223-4-9hrt04?1514063330","product_colors":[{"hex_value":"#CF9978","colour_name":"Medium Bronze"}]},{"id":477,"brand":"maybelline","name":"Maybelline Facestudio Master Contour Kit","price":"15.99","price_sign":null,"currency":null,"image_link":"https://d3t32hsnjx07q6.cloudfront.net/i/4f731de249cbd4cb819ea7f5f4cfb5c3_ra,w158,h184_pa,w158,h184.png","product_link":"https://well.ca/products/maybelline-facestudio-master-contour-120303.html?cat=328","website_link":"https://well.ca","description":"Maybelline Facestudio Master Contour Kit is the ultimate on the go all-in-one palette, with contouring brush included. Define and highlight in a New York minute with this effortless 3-step face contouring kit. This easy-to-use 3-step face contouring kit features a bronzer, blush and highlighter.","rating":null,"category":null,"product_type":"bronzer","tag_list":[],"created_at":"2016-10-01T18:35:40.504Z","updated_at":"2017-12-23T21:08:48.157Z","product_api_url":"https://makeup-api.herokuapp.com/api/v1/products/477.json","api_featured_image":"/s3.amazonaws.com/donovanbailey/products/api_featured_images/000/000/477/original/open-uri20171223-4-1m8bc4v?1514063328","product_colors":[{"hex_value":"#9B7163","colour_name":null},{"hex_value":"#D78A89","colour_name":null},{"hex_value":"#DBB7A9","colour_name":null}],"id":468,"brand":"maybelline","name":"Maybelline Face Studio Master Hi-Light Light Booster Blush","price":"14.99","price_sign":null,"currency":null,"image_link":"https://d3t32hsnjx07q6.cloudfront.net/i/4621032a92cb428ad640c105b944b39c_ra,w158,h184_pa,w158,h184.png","product_link":"https://well.ca/products/maybelline-face-studio-master-88831.html","website_link":"https://well.ca","description":"Maybelline Face Studio Master Hi-Light Light Boosting blush formula has an expert \nbalance of shade + shimmer illuminator for natural glow. Skin goes \nsoft-lit with zero glitz.\n\n\t\t\tFor Best Results: Brush over all shades in palette and gently sweep over \ncheekbones, brow bones, and temples, or anywhere light naturally touches\nthe face.\n\n\t\t\t\t\t\n\n\t","rating":null,"category":"powder","product_type":"blush","tag_list":[],"created_at":"2016-10-01T18:35:27.706Z","updated_at":"2017-12-23T21:08:50.624Z"}

```

<https://makeup-api.herokuapp.com/api/v1/products.json?brand=maybelline>



POSTMAN

```
1  {
2    {
3      "id": 495,
4      "brand": "maybelline",
5      "name": "Maybelline Face Studio Master Hi-Light Light Booster Bronzer",
6      "price": "14.99",
7      "price_sign": null,
8      "currency": null,
9      "image_link": "https://d3t32hsnjxo7q6.cloudfront.net/i/991799d3e70b8856686979f8ff6dcfe0_ra,w158,h184_pa,w158,h184.png",
10     "product_link": "https://well.ca/products/maybelline-face-studio-master_88837.html",
11     "website_link": "https://well.ca",
12     "description": "Maybelline Face Studio Master Hi-Light Light Boosting bronzer formula has an expert \nbalance of shade + shimmer illuminator for natural glow. Skin goes \nsoft-lit with
13     "rating": 5.0,
14     "category": null,
15     "product_type": "bronzer",
16     "tag_list": [],
17     "created_at": "2016-10-01T18:36:15.012Z",
18     "updated_at": "2017-12-23T21:08:50.624Z",
19     "product_api_url": "https://makeup-api.herokuapp.com/api/v1/products/495.json",
20     "api_featured_image": "https://s3.amazonaws.com/donovanbailey/products/api_featured_images/000/000/495/original/open-uri120171223-4-9hrto4?1514063330",
21     "product_colors": []
22   },
23   {
24     "id": 488,
25     "brand": "maybelline",
26     "name": "Maybelline Fit Me Bronzer",
27     "price": "10.29",
28     "price_sign": null,
29     "currency": null,
30     "image_link": "https://d3t32hsnjxo7q6.cloudfront.net/i/d4f7d82b4858c622bb3c1cef07b9d850_ra,w158,h184_pa,w158,h184.png",
31     "product_link": "https://well.ca/products/maybelline-fit-me-bronzer_31265.html",
32     "website_link": "https://well.ca",
33     "description": "Why You'll Love It\n\nLightweight pigments blend easily and wear evenly\n\nProvides a natural, fade-proof bronzed color that leaves skin the way it was meant to be...fresh
34     "rating": 4.5,
35     "category": null,
36     "product_type": "bronzer",
37     "tag_list": [],
38     "created_at": "2016-10-01T18:36:05.584Z",
39     "updated_at": "2017-12-23T21:08:49.985Z",
```

(Reminder) Things we need to get started:

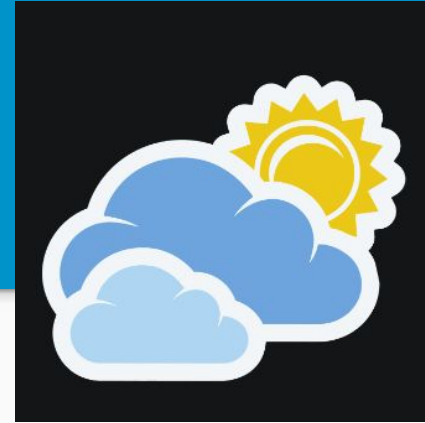
1. URL endpoint
2. Newtonsoft.Json Nuget Package (allows us to parse)
3. HTTP Client (to send requests)

Recommended: Download Postman to your desktop

[Download Postman | Get Started for Free](#)



Notes for Exercise 2



1. MUST create openweathermap account
2. and activate key with link sent to email!

URL EndPoint to find weather by city in Fahrenheit:

```
https://api.openweathermap.org/data/2.5/weather?q={city name}&appid={API key}&units=imperial
```

