API Response Formats

William Trang

Quick Recap

- What is an API?
- Different types of APIs
  - Public
  - Private
  - Paired
- Different API architectures
  - REST
  - SOAP

API Responses

Similar to HTTP requests, APIs typically work on the request/response format

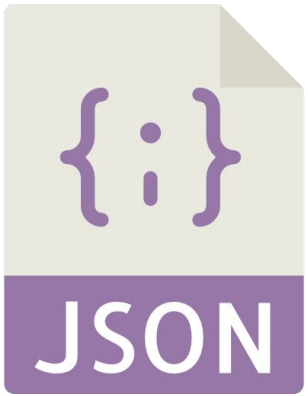
- Request: When you want to access data or services from an application, you send a request to its API
- Response: The API then sends back the requested data or performs the desired action, providing a response

API Responses

Last week, we briefly discussed what you can do with APIs. This time, we'll talk go in detail about API responses and common API response file formats

JSON

JSON stands for JavaScript Object Notation, and it is a text format that stores data as objects, consisting of key-value pairs, similar to dictionaries in Python. JSON is flexible, structured, and easy to parse and manipulate by machines, and is by far the most popular data transfer file format in modern day



```
{
    hey: "guy",
    anumber: 243,
  - anobject: {
        whoa: "nuts",
      - anarray: [
            1,
            2,
            "thr<h1>ee"
        ],
        more: "stuff"
    },
    awesome: true,
    bogus: false,
    meaning: null,
    japanese: "明日がある。",
    link: http://jsonview.com,
    notLink: "http://jsonview.com is great"
}
```

```
{
    "friends":
    [
        {
            "name": "John Ferreira",
            "age": 26,
            "city": "Porto",
            "profession": "Full Stack Web Developer",
            "hobbies": ["Fitness", "Games"]
        },
        {
            "name": "Leonardo Marinho",
            "age": 18,
            "city": "London",
            "profession": "Electric Engineer",
            "hobbies": ["Build legos", "Robots", "Swim"]
        },
        {
            "name": "Caroline Azevedo",
            "age": 34,
            "city": "Salvador",
            "profession": "Entrepreneur",
            "hobbies": ["Sing", "play guitar"]
        }
    ]
}
```

XML

XML stands for Extensible Markup Language, and it is a text format that stores data as elements, attributes, and text, enclosed by tags. XML is rich, extensible, and easy to validate and transform by machines. It is widely used by document formats and standards, such as HTML, RSS, and SOAP, as it can handle metadata, schemas, and namespaces

```
<?xml version="1.0" encoding="UTF-8"?>
- <EmployeeData>
  - <employee id="34594">
      <firstName>Heather</firstName>
      <lastName>Banks</lastName>
      <hireDate>1/19/1998</hireDate>
      <deptCode>BB001</deptCode>
      <salary>72000</salary>
    </employee>
  - <employee id="34593">
      <firstName>Tina</firstName>
      <lastName>Young</lastName>
      <hireDate>4/1/2010</hireDate>
      <deptCode>BB001</deptCode>
      <salary>65000</salary>
    </employee>
</EmployeeData>
```

CSV

CSV stands for comma-separated values, and it is a plain text format that stores data in rows and columns, separated by commas. CSV is simple, universal, and easy to read and write by humans and machines. Another common variation of CSVs are TSVs, which is a data file where the data is separated by tabs rather than commas

	A	B	C
1	Name	Food	Age
2	Sarah	Cheese	17
3	Dexter	Chicken	46
4	Li	Burgers	24
5	Amy	Ice Cream	38
6	Ahmed	Cauliflower	44
7	Portia	Steak	22

```
Name,Food,Age
Sarah,Cheese,17
Dexter,Chicken,46
Li,Burgers,24
Amy,Ice Cream,38
Ahmed,Cauliflower,44
Portia,Steak,22
```

Pros/Cons

	JSON	XML	CSV
File Size	Medium - some structure for each point	Largest - rigid and large structure for each point	Smallest - one line per data point
Supported Data Types	Bools, numbers, strings, arrays, other objects, etc	Everything JSON has, plus more like date/time	3 built in - number, string, date
Nested Data	Nested data achieved with arrays	Nested data from hierarchy of file	CANNOT handle nested data
Readability	Also easy to read, slightly harder than CSV	Hard to understand w/o background knowledge	Simplest to read, simple data structure
Ease of Use	Built in support in JS, easy to parse in other languages	Hardest, need to build/use built in XML parser	Easiest, universal support in all languages