API en JSON

Wat is een API?

Een API (Application Programming Interface) is een toegangspoort vanuit een website om informatie op te vragen.

Het is eigenlijk een soort database (zoals onze SQL-database) maar dan op een andere server. Je kunt als gebruiker die informatie opvragen, maar de informatie die op een website staat si door een programma lastig te verwerken. Daarom is er een soort standaard formaat om gegevens te delen en dat formaat heet JSON. De meeste API's gebruiken dan ook JSON als een formaat om gegevens van de ene site naar de andere te sturen.

Zo maakt de Canvas Monitor gebruik van de Canvas API en kan het daarmee gegevens ophalen uit Canvas over jouw voortgang. De reden dat de Canvas Monitor niet altijd up-to-date is, is omdat er een apart (Python) programma is dat om bepaalde tijden via de API vraagt of er updates zijn.

Via een API kan je meestal gegevens opvragen, maar ook veranderen. Voorlopig gaan wij kijken naar het lezen van gegevens via een API. Er zijn verschillende soorten API's.

Wil je nog wat meer voorbeeld van API;s dan wordt er in dit korte YouTube filmpje nog een keer uitgelegd wat een API is.

https://www.youtube.com/watch?v=AwXoCmz4yMg&pp=ygUKd2F0IGlzIEFQSQ%3D%3D

Een API gebruiken in Python

Ga naar:

https://www.wijs.ovh/c-api/

dit is een API interface waarbij je alle gegevens van alle landen van de wereld kan opvragen.

Check de JSON die de API geeft.

We gaan met een Python programma een lijstje maken van alle landen en hun hoofdsteden.

Omdat te doen hebben we een Python Library nodig. Deze zorgt ervoor dat je een webpagina kan opvragen. Deze library heet *requests* en met dit commando importeren we de library (die is vergelijkbaar met *require* in PHP).

Na de import lezen we de informatie uit de API op.

```
import requests


response = requests.get(f"https://www.wijs.ovh/c-api/")
if response.status_code == 200:
  countries_data = response.json()
else:
  print("No Luck, I cannot get data from the API")

# print( countries_data[0]['name']['common'], countries_data[0]['capital'][0], countries_data[0]['car']['side'] )
# print( countries_data[113]['name']['common'], countries_data[113]['capital'][0],
countries_data[113]['car']['side'] )
```

Try it out!

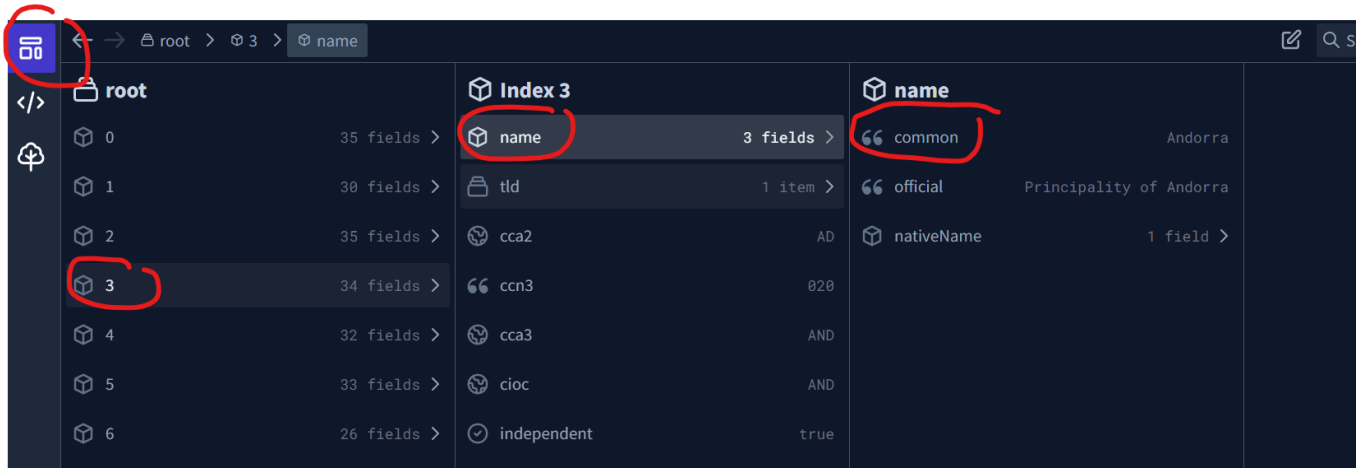
Deze code leest voert een API-call uit en krijgt JSON terug waarin informatie over alle landen van de wereld staat.

Je kunt in je browser de JSO oproepen door de URL te gebruiken die in de code staat.

Die JSON is een heleboel data en hoe weten we nu wat en hoe we daar gegevens uit kunnen halen?

JSON ontcijferen

Je hebt hiervoor een hulpmiddel nodig. Er zijn er meerdere, maar ik raad https://jsonhero.io/ aan.



Als je het JSON-bestand inleest (dat kan door de URL in de tool mee te geven dan zie je een interface zo als in het plaatje.

In de eerste kolom staat 0,1,2,3,4, ... dat zijn de record. In dit geval staat alle informatie over één land in een zo'n record.

Stel ons JSON bestand staat in de variabele countries_data, dan kunnen we het derde (of eigenlijk vierde want we beginnen weer met 0 te tellen) land selecteren met:

    countries_data[3]

Nu hebben we alle data van het derde land In de volgende kolom staan alle items die in een land staan. Als eerste zien we name, we kiezen nu de name door:

    countries_data[3][name]

In de derde kolom staan nu weer drie items, common, official, en nativeName. Als we de common naam willen, dan akn dat door:

    countries_data[3][name][common]

In de code staan op de laatste twee regels een paar voorbeelden. Loop deze na en controleer of deze werken.

Zie je wat countries_data[0]['car']['side'] dit afdrukt? Probeer het uit!

Stel we willen alle namen van alle landen afdrukken. Dat kan met:

```
print(countries_data[0][name][common])
print(countries_data[1][name][common])
print(countries_data[2][name][common])
....
....
```

Niet handig dus we gebruiken een loop in Python. Net als we in PHP foreach zouden gebruiken, doen we dit in Python op de volgende manier:

```
for item in countries_data
  print(print(item[name][comon]))
```

De variabele item wordt in de loop telkens countries_data[0], countries_data[1], countries_data[2], countries_data[3], etc. etc.

Opdracht

Maak een txt bestand en beantwoord de volgende vraag:

1. Als je de hoofdstad afdrukt van een land dan gebruik je countries_data[0]['capital'][0] Waarom staat die [0] erachter en waarom denk je dat dat nodig is?

Opdracht

Druk alle namen van alle landen af met daarachter de munteenheid (currency).

Gebruik niet de afkorting maar de volledige naam, zoals:

- Jordan - Jordian Dinar
- Northern Mariana Islands - United States Dollar
- Serbia - Serbian Dinar

Inleveren

1. Python code (py bestand) en;
2. een schermafdruk van de CMD waarin je de code hebt gedraaid.

Idee voor later.....

```
import requests
import random

def get_random_countries(num_countries):
    response = requests.get(f"https://restcountries.com/v3.1/all")
    if response.status_code == 200:
        countries_data = response.json()
        random_countries = random.sample(countries_data, num_countries)
        return random_countries
    else:
        return None
```

```
def play_game():
    random_countries = get_random_countries(10)
    if random_countries is None:
        print("Failed to retrieve country data. Please try again later.")
        return

    total_points = 0

    for country in random_countries:
        name = country['name']['common']
        population = country['population']

        print(f"Guess the population of {name}: ")
        guess = input()

        try:
            guess = int(guess)
            difference = abs(guess - population)
            points = max(0, 100 - difference // 1000000)  # Calculate points based on difference

            print(f"The population of {name} is {population}")
            print(f"You guessed {guess}. You scored {points} points.\n")

            total_points += points
        except ValueError:
            print("Invalid input. Skipping the country.")

    print(f"Game over! Your total score is {total_points}.")

play_game()
```

Revision #12
Created 24 June 2023 19:12:28 by Max
Updated 5 July 2023 18:07:10 by Max