API Fundamentals Workshop

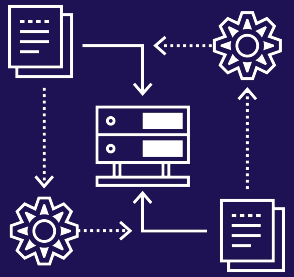
by Facundo Goiriz

dyvenia

Agenda

Hello 👋

1. What's an API? 🤔
2. Basics 🏫
3. How to use Python to interact with APIs🔲
4. Practice! 😱



dyvenia

What's an API?

Application Programming Interface (API)

dyvenia

Application Programming Interface (API)

Characteristics of APIs

- They are like normal interfaces, but for computers
- They simplify programming by abstracting the underlying implementation
- They only expose objects or actions that developers need.

dyvenia

Understanding with a metaphor

The Manual Restaurant:

Imagine a restaurant where you had to specify every action the chef has to do to prepare your food, something like:

- Welcome to the Manual Restaurant, what would you like to order?
+ Yes, please tell the chef to first grab tomato, chop it, add salt, then grab milk, stir it, etc, etc, etc.

IT WOULD BE HORRIBLE.

dyvenia

Solution:

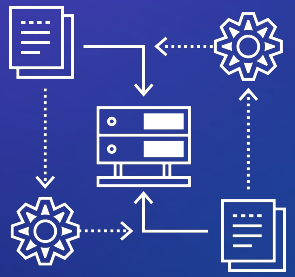
Create a Restaurant “API” so the user just has to order what they want

---

Now you just have to look at the docs (Menu) to see the supported commands (Dishes) and tell the waiter the dish you want, he will transmit your order to the kitchen, and then come back with your dish.

Notice that with this system you don't need to know the steps that are needed to prepare your dish

dyvenia



Basics

dyvenia

Real example:

Fetching a WEB API hosted in a server

In this workshop we are going to focus on WEB APIs.
Now we are going to fetch an API that returns random facts about numbers.
We are using cURL to make the request,

Go to your linux command line and run:
curl http://numbersapi.com/random/math

Finally, be amazed by the wonderful fact about a random number that you just received.
Congrats, you just made a GET request to a WEB API using cURL.

dyvenia

Hypertext Transfer Protocol (HTTP)

HTTP is the protocol we use to make requests between client and server.

HTTP defines a set of request methods to indicate the desired action to be performed for a given resource.

Most used methods:
GET: It is used to request data from a specified resource.
POST: It is used to send data to create/update a resource.

Before when we requested a fact about a random number we did a GET request.
In this workshop we are going to focus on GET requests to APIs.

dyvenia

GET parameters

Parameters to requests a particular resource from an API using the HTTP protocol

What do URL parameters look like?

An example URL could look like this:
https://www.example.com/index.html?name1=value1&name2=value2

what comes after the “?” sign are the parameters, if there are more than 1 parameter, they are delimited with “&” sign

dyvenia

GET parameters example:

Fetching an API that predicts nationality based on a name

https://api.nationalize.io is the URL of the API.
We could request it with:
curl https://api.nationalize.io

but how is the API going to guess our nationality based on a name if we don't provide a name? Think of a way of providing a name to this API.

Solution: curl https://api.nationalize.io?name=<your_name>
replace <your_name> with your name, in my case it would be:
curl https://api.nationalize.io?name=facundo

dyvenia

API HTTP responses

JSON response from the API that predicts nationality based on a name

In my case after running
curl https://api.nationalize.io?name=facundo

I get the following response:
{"country":[{"country_id":"AR","probability":0.715},{"country_id":"UY","probability":0.196},{"country_id":"MX","probability":0.009},{"country_id":"PY","probability":0.008},{"country_id":"CL","probability":0.008}],"name":"facundo"}

This response format is very common and it is called JSON, very similar to a python dictionary
Notice how the API predicted that I am from Argentina xD

dyvenia

HTTP response status codes

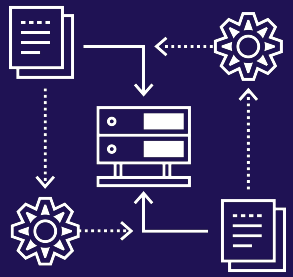
HTTP response status codes indicate whether a specific HTTP request has been successfully completed.

Responses are grouped in five classes:

1. **Informational responses (100 – 199)**
2. **Successful responses (200 – 299)**
3. **Redirection messages (300 – 399)**
4. **Client error responses (400 – 499)**
5. **Server error responses (500 – 599)**

The most common one is 200 OK status code. If you get a 200 as status code of your request, it means that everything went OK.
But if the status code is 400 or higher then something went wrong.
A classic example is the 404 Not Found error.

dyvenia



How to use Python to interact with APIs

dyvenia

Requests: HTTP for Humans

Requests is an elegant and simple HTTP library for Python, built for human beings.

Go to your linux command line and run:
git clone https://github.com/dyvenia/api-workshop

Then follow README

Thanks!

Contact us:

ul. Czyżówka 14/0.3,
30-526 Kraków
Poland

hello@dyvenia.com
www.dyvenia.com