

Introduction to APIs and JSONs

INTERMEDIATE IMPORTING DATA IN PYTHON



Hugo Bowne-Anderson
Data Scientist at DataCamp

APIs

- Application Programming Interface
- Protocols and routines
 - Building and interacting with software applications

OMDb API

The Open Movie Database

APIs

- Application Programming Interface
- Protocols and routines
 - Building and interacting with software applications

OMDb API

The Open Movie Database

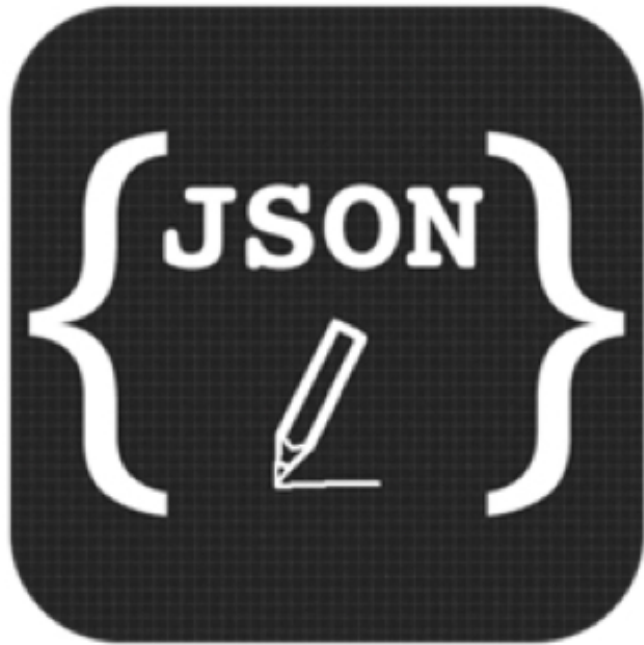


2. APIs

learning how to interact with APIs, or Application Programming Interfaces. An API is a set of protocols and routines for building and interacting with software applications. In particular, you'll learn how to use the Open Movie Database API

JSONs

- JavaScript Object Notation
- Real-time server-to-browser communication
- Douglas Crockford
- Human readable



JSONs

```
{'Actors': 'Samuel L. Jackson, Julianna Margulies, Nathan Phillips, Rachel Blanchard',  
'Awards': '3 wins & 7 nominations.',  
'Country': 'Germany, USA, Canada',  
'Director': 'David R. Ellis',  
'Genre': 'Action, Adventure, Crime',  
'Language': 'English',  
'Rated': 'R',  
'Released': '18 Aug 2006',  
'Runtime': '105 min',  
'Title': 'Snakes on a Plane',  
'Type': 'movie',  
'Writer': 'John Heffernan (screenplay), Sebastian Gutierrez (screenplay), David Dalessandro (story), John Heffernan (story)',  
'Year': '2006',  
'imdbID': 'tt0417148',  
'imdbRating': '5.6',  
'imdbVotes': '114,668'}
```

JSONs

```
{'Actors': 'Samuel L. Jackson, Julianna Margulies, Nathan Phillips, Rachel Blanchard',  
'Awards': '3 wins & 7 nominations.',  
'Country': 'Germany, USA, Canada',  
'Director': 'David R. Ellis',  
'Genre': 'Action, Adventure, Crime',  
'Language': 'English',  
'Rated': 'R',  
'Released': '18 Aug 2006',  
'Runtime': '105 min',  
'Title': 'Snakes on a Plane',    <--  
'Type': 'movie',  
'Writer': 'John Heffernan (screenplay), Sebastian Gutierrez (screenplay), David Dalessandro (story), John Heffernan (story)',  
'Year': '2006',  
'imdbID': 'tt0417148',  
'imdbRating': '5.6',  
'imdbVotes': '114,668'}
```

JSONs

```
{'Actors': 'Samuel L. Jackson, Julianna Margulies, Nathan Phillips,  
Rachel Blanchard',  
'Awards': '3 wins & 7 nominations.',  
'Country': 'Germany, USA, Canada',  
'Director': 'David R. Ellis',  
'Genre': 'Action, Adventure, Crime',  
'Language': 'English',  
'Rated': 'R',  
'Released': '18 Aug 2006',  
'Runtime': '105 min',  
'Title': 'Snakes on a Plane',    <--  
'Type': 'movie',  
'Writer': 'John Heffernan (screenplay), Sebastian Gutierrez (screenplay),  
David Dalessandro (story), John Heffernan (story)',  
'Year': '2006',                <--  
'imdbID': 'tt0417148',  
'imdbRating': '5.6',  
'imdbVotes': '114,668'}
```

Loading JSONs in Python

```
import json
with open('snakes.json', 'r') as json_file:
    json_data = json.load(json_file)
```

```
type(json_data)
```

```
dict
```

8. Loading JSONs in Python

Lets say that I had the JSON stored in my working directory as 'snakes dot json'. To load the JSON into my Python environment, I would first import the package json and then open a connection to the file and use the function json dot load to load the JSON. If I then check the datatype of json_data by executing type(json_data), I see that Python cleverly imported the JSON as a dictionary!

Exploring JSONs in Python

```
for key, value in json_data.items():  
    print(key + ': ', value)
```

Note: you can access a value in a dictionary using the syntax: `dictionary[key]`.

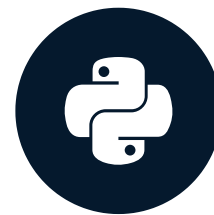
```
Title: Snakes on a Plane  
Country: Germany, USA, Canada  
Response: True  
Language: English  
Awards: 3 wins & 7 nominations.  
Year: 2006  
Actors: Samuel L. Jackson, Julianna Margulies  
Runtime: 105 min  
Genre: Action, Adventure, Crime  
imdbID: tt0417148  
Director: David R. Ellis  
imdbRating: 5.6  
Rated: R  
Released: 18 Aug 2006
```

Let's practice!

INTERMEDIATE IMPORTING DATA IN PYTHON

APIs and interacting with the world wide web

INTERMEDIATE IMPORTING DATA IN PYTHON



Hugo Bowne-Anderson
Data Scientist at DataCamp

Herein, you'll learn

- What APIs are
- Why APIs are important
- In the exercises:
 - Connecting to APIs
 - Pulling data from APIs
 - Parsing data from APIs

What is an API?

- Set of protocols and routines
- Bunch of code
 - Allows two software programs to communicate with each other



What is an API?

- Set of protocols and routines
- Bunch of code
 - Allows two software programs to communicate with each other



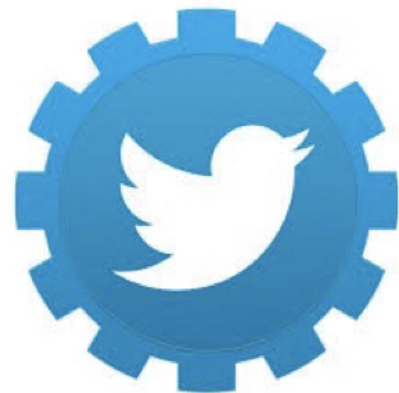
3. What is an API?

API is a set of protocols and routines for building and interacting with software applications. Another way to think of it is that an API is a bunch of code that allows two software programs to communicate with each other. For example, if you wanted to stream twitter data by writing some Python code, you would use the Twitter API. If you wanted to automate pulling and processing information

APIs are everywhere



APIs are everywhere



UBER Developers

APIs are everywhere



UBER Developers



APIs are everywhere



UBER Developers



Instagram
API

Connecting to an API in Python

```
import requests
url = 'http://www.omdbapi.com/?t=hackers'
r = requests.get(url)
json_data = r.json()
for key, value in json_data.items():
    print(key + ': ', value)
```

9. Connecting to an API in Python

In this example, we'll pull movie data from the Open Movie Database, or OMDb, API. Once again, you'll use the ever-elegant requests library. You import requests and assign the URL of interest to the variable url. You then package and send the request to the URL, which describes your API query, and catch the response in one line of code. Thanks again, requests package! Another really cool aspect of the requests package is that the Response objects, such as r, have an associate method json, which is a built-in JSON decoder for when we're dealing with JSON data. This returns a dictionary and we can then print all the key-value pairs to check out what we pulled from the OMBD API!

What was that URL?

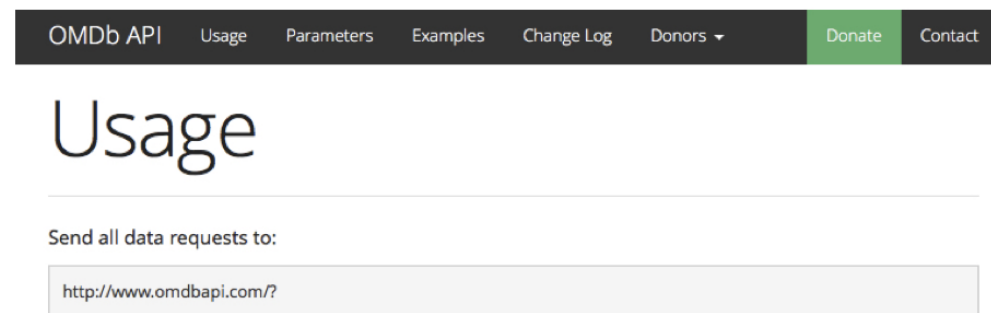
- http - making an HTTP request
- www.omdbapi.com - querying the OMDB API
- `?t=hackers`
 - Query string
 - Return data for a movie with title (t) 'Hackers'

```
'http://www.omdbapi.com/?t=hackers'
```

10. What was that URL?

Now the last thing to discuss is how the URL we used actually pulled data from the API. To do so, let's break it up into chunks. The http signifies that we're making an HTTP request, the 'www dot omdb dot api' that we're querying the OMDB API, then there's the "?t equals hackers". This string that begins with a question mark is called a Query String. Query Strings are parts of URLs that do not necessarily fit into conventional a hierarchical path structure. What follows the question mark in the query string is the query we are making to the OMDB API. The query we just made was simple : querying 't equals hackers' asked the API to return the data about the movie with the title Hackers. The 't' in the query stood for title.

OMDb API



11. OMDb API

We knew that this was how to perform such a query from the documentation on the OMDb API's homepage. Under "Usage" here, they state explicitly that 'Send all data requests to: http:// www dot omdbapi dot com /?'.

OMDb API

Usage

Send all data requests to:

http://www.omdbapi.com/?

Parameters

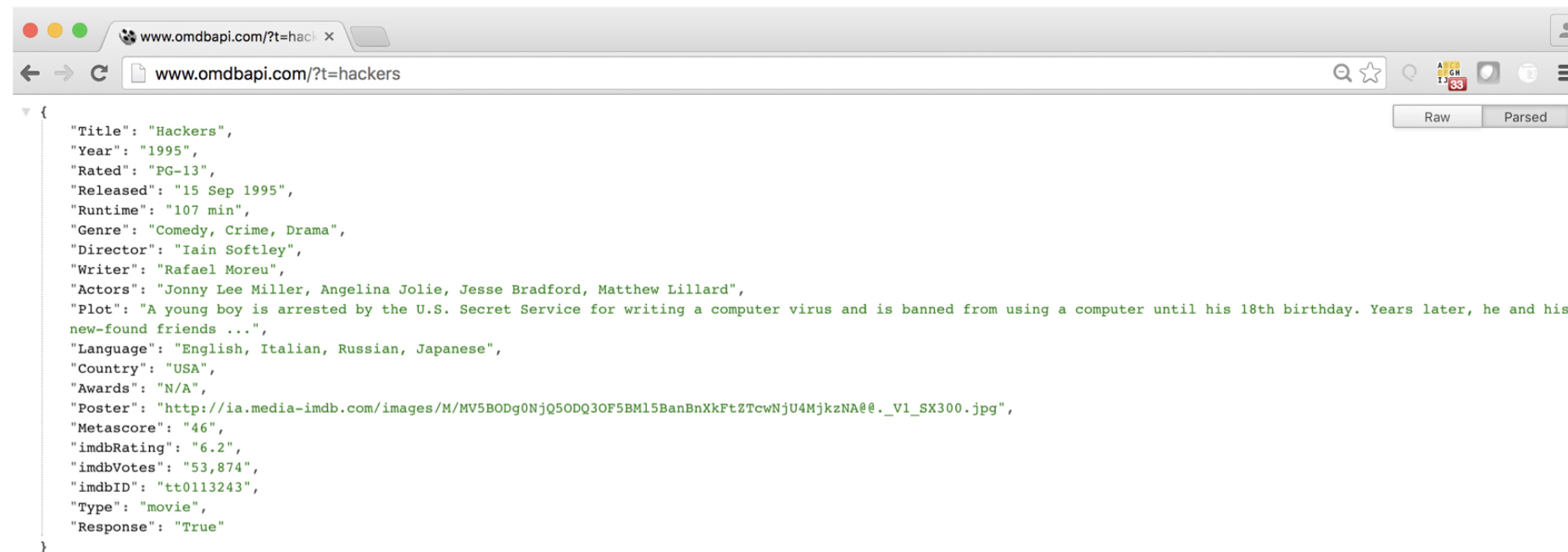
By ID or Title

Parameter	Required	Valid Options	Default Value	Description
i	Optional*		<empty>	A valid IMDb ID (e.g. tt1285016)
t	Optional*		<empty>	Movie title to search for.
type	No	movie, series, episode	<empty>	Type of result to return.

12. OMDb API

They also have a query string parameters table that shows how to query a particular title or a particular movie ID.

It's a regular URL!



13. It's a regular URL!

It is also worth mentioning that there is nothing special about this URL and so you can also navigate to it in your browser of choice. It will generally look like this. I like to use a Chrome extension called JSON formatter to make it a bit prettier. Alright. Now you know all about APIs and have a basic practical understanding of how to query them,

Let's practice!

INTERMEDIATE IMPORTING DATA IN PYTHON