DATA EXPLORATION AND VISUALISATION

Coursework1:

Accessing OxCGRT Data via Web API

Dr. Maysson Ibrahim

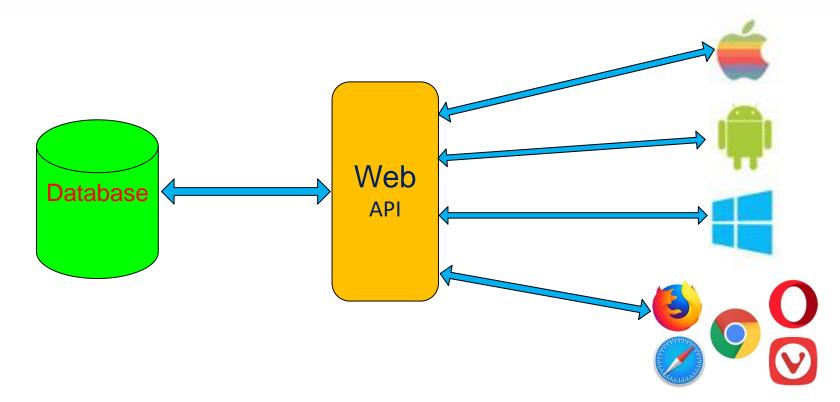






Web API

- A Web application programming interface consists of one or more endpoints (URLs accessed by a client application)
- The request–response messages are typically expressed in JSON or XML.





JSON- JavaScript Object Notation

- JSON is a syntax for storing and exchanging data objects.
- It consists of a set of attribute—value pairs

```
"firstName": "John",
"lastName": "Smith",
"age": "45",
"phoneNumbers": "0123 1234"
}
```

 JSON file has .json extension, which is a text written with JavaScript object notation.



Json Format

 Like object-oriented programming, you can have object inside a bigger one

```
"firstName": "John",
"lastName": "Smith",
"age": "45",
"address": {
 "houseNo": "100",
 "street": "Hunter Street",
 "postCode": "MK181EG"
"phoneNumbers": "0123 1234"
```



Json Format

You can also have an array of objects



Json Format

You can have both

```
"firstName": "John",
"lastName": "Smith",
"age": "45",
"address": {
  "houseNo": "100",
  "street": "Hunter Street",
  "postCode": "MK181EG"
},
"employees":[
    { "firstName": "John", "lastName": "Doe" },
    { "firstName": "Anna", "lastName": "Smith" },
    { "firstName": "Peter", "lastName": "Jones" }
"phoneNumbers": "0123 1234"
```



JSON vs XML

Same example of the list of employees

```
<employees>
   <employee>
        <firstName>John</firstName>
       <lastName>Doe</lastName>
   </employee>
   <employée>
       <firstName>Anna</firstName>
        <lastName>Smith</lastName>
   </employee>
   <employée>
       <firstName>Peter</firstName>
       <lastName>Jones
    </employee>
</employees>
```

Json: 2013

XML: 1970s

Reading live data (**OxCGRT**) on Covid-19 From Json- based API



The Oxford COVID-19 Government Response Tracker (OxCGRT)

- OxCGRT provides a systematic way to track the stringency of government responses to COVID-19 across countries and time.
- It uses a novel index that combines various measures of government responses.
- Data is collected and updated in real time by a team of dozens of students and staff at Oxford University.
- OxCGRT provides two APIs:
 - Request data for one country for a specific day
 - Request data for all countries over period of time (startDate-endDate)

https://www.bsg.ox.ac.uk/research/research-projects/oxford-covid-19-government-response-tracker https://www.bsg.ox.ac.uk/sites/default/files/2020-04/BSG-WP-2020-031-v4.0 0.pdf



OxCGRT- Country data for a specific day

- https://covidtrackerapi.bsg.ox.ac.uk/api/v2/stringency/actions/{ALPHA-3}/{YYYY-MM-DD}
- ALPHA-3 is the ISO 3166-1 alpha-3 country code

Response:

```
policyActions: {
0...n: { //Numerical key
  policy type code: String, //Policy type S1..Sn
  policy_type_display: String, //String describing policy value,
  policyvalue: Integer, //Represents policy status
  isgeneral: Boolean, //If this is a general policy
  notes: String, //Notes entered by contributors
```

```
stringencyData: {
  date value: String, //YYYY-MM-DD date of record
  country code: String, //ALPHA-3 country code
  confirmed: Integer, //Recorded confirmed cases,
  deaths: Integer, //Recorded deaths,
  stringency actual: Integer, //Calculated stringency
  stringency: Integer, //Display stringency
                                                10
```



OxCGRT- all countries over period of time

https://covidtrackerapi.bsg.ox.ac.uk/api/v2/stringency/date-range/{YYYY-MM-DD}/{YYYY-MM-DD}

Response:

```
{ scale: Object //Scale for visualisations
   deaths: Object, //Min and max int values in returned data
   confirmed: Object, //Min and max int values in returned data
   stringency: Object, //Min and max int values in returned data
countries: [
   Array of Strings //3 letter country codes in returned data],
data: {
 YYYY-MM-DD: { //Key is date in YYYY-MM-DD format
   BRA: { //3 letter country code
   date value: YYYY-MM-DD, //Repeat of date in key
    country code: BRA, //Repeat of country code in key
    confirmed: Integer, //Value of confirmed cases
   deaths: Integer, //Value of deaths
    stringency_actual: Integer, //Calculated value for day. Can be null if no value available
   stringency: Integer, //Display value for day **1.
      **1 Will be actual value if available. For previous 7 days will take last available value. Otherwise null.
```



Access data in python Example 1

#Accessing the UK data on 6th April

import json, urllib.request

print(x)

```
# open a request to read the data
data = urllib.request.urlopen("https://covidtrackerapi.bsg.ox.ac.uk/api/v2/stringency/actions/GBR/2020-04-06").read()
# load the data (will be saved as dictionary in output)
output = json.loads(data)
# let's see what is inside "stringencyData"
sd=output["stringencyData"]
for x in sd:
```



Access data in python Example 2

```
# Accessing the number of confirmed cases in the UK data between 1st April and 10th April cases=[0]*10

for d in range(1,11):
    url="https://covidtrackerapi.bsg.ox.ac.uk/api/v2/stringency/actions/GBR/2020-04-"+str(d) data = urllib.request.urlopen(url).read()
    output = json.loads(data)
    cases[d-1]=output["stringencyData"]["confirmed"]
```

print(cases)



Access data in python Example 3

Accessing data of all countries between two dates e.g. 1st and 10th April

```
data = urllib.request.urlopen("https://covidtrackerapi.bsg.ox.ac.uk/api/v2/stringency/date-range/2020-04-01/2020-04-
10").read()
output = json.loads(data)
Note that this API is different from Example 1 and 2
#We can filter the output to choose certain countries (e.g. UK, USA)
i=0
casesUK=[0]*10
casesUSA =[0]*10
for d in output["data"]:
  casesUK[i]=output["data"][d]["GBR"]["confirmed"]
  casesUSA[i]=output["data"][d]["USA"]["confirmed"]
  i=i+1
```

