Resources of Open Data: Exercise Solutions

Maksim Misin

November 3, 2017

Exercise 1 *OpenAIRE*

a) Root URL: `http://api.openaire.eu/search`. Endpoints are: `publications`, `datasets`, `projects`.

By default response is in XML format. To obtain JSON response add `format=json` parameter to URL.

b) The number of returned datasets is specified in `response/header/query/size` part of the response. `response/header/query/total` shows the total number of results satisfying search criteria. To narrow the results add `size=300` parameter to query.

c) To get results in CSV format use `format=csv`.

Exercise 2 *Figshare*

a) 15 most viewed articles:

```
https://api.figshare.com/v2/articles?
    page_size=15&
    order=views&
    order_direction=desc
```

and collections:

```
https://api.figshare.com/v2/collections?
    page_size=15&
    order=views&
    order_direction=desc
```

b) For example, to search for "social media" datasets we send a POST request to `https://api.figshare.com/v2/articles/search` endpoint with the following body:

```
{
    "item_type": 3,
    "search_for": "social media",
    "page": 1,
    "page_size": 1000,
    "order": "views",
    "order_direction": "desc",
    "published_since": "2016-01-01"
}
```

c) An example of article details endpoint: `https://api.figshare.com/v2/articles/5539834`. Download links are associated with `download_url` keys. To download all files you can for example use python `urllib` and `json` libraries.

Exercise 3 *Plos ONE*

a)
```
http://api.plos.org/search?
    q=abstract:economic AND body:Estonia&
    wt=json
```

b)
```
http://api.plos.org/search?
    q=subject:"social sciences" AND
        publication_date:[2014-01-01T00:00:00Z TO 2015-01-01T00:00:00Z] AND
        doc_type:full&
    wt=json&
    fl=title,pagecount,counter_total_all,alm_scopusCiteCount&
    rows=500
```

c) Assuming that we got the following response:

```
{
    "response": {
        "numFound": 4521,
        "start": 0,
        "docs": [
            {
                "alm_scopusCiteCount": 6,
                "counter_total_all": 47605,
                "pagecount": 7,
                "title": "Globalization and Econ..."
            },
            {
                "alm_scopusCiteCount": 18,
```

```
                "counter_total_all": 24113,
                "pagecount": 15,
                "title": "Gender on the Brain: A Cas..."
            },
            {
                "alm_scopusCiteCount": 2,
                "counter_total_all": 4961,
                "pagecount": 8,
                "title": "Evidence for the Identifi..."
            }
        ]
    }
}
```


we delete everything unrelated to the articles:


```
[
    {
        "alm_scopusCiteCount": 6,
        "counter_total_all": 47605,
        "pagecount": 7,
        "title": "Globalization and Econ..."
    },
    {
        "alm_scopusCiteCount": 18,
        "counter_total_all": 24113,
        "pagecount": 15,
        "title": "Gender on the Brain: A Cas..."
    },
    {
        "alm_scopusCiteCount": 2,
        "counter_total_all": 4961,
        "pagecount": 8,
        "title": "Evidence for the Identifi..."
    }
]
```


and after conversion with `https://sqlify.io` obtain the following CSV:

```
alm_scopusCiteCount,counter_total_all,pagecount,title
6,47605,7,Globalization and Econ...
18,24113,15,Gender on the Brain: A Cas...
2,4961,8,Evidence for the Identifi...
```

The subsequent analyses can be done in a number of ways depending on your method of choice for tabulated data manipulations.

Exercise 4 *Figshare*

a) Assuming we picked Swiss medieval manuscripts repository, the formatted manifest file is going to look like this:


```
{
    "@context": "http://iiif.io/api/presentation/2/context.json",
    "@id": "http://www.e-codices.unifr.ch/metadata/iiif/kba-0...",
    "@type": "sc:Manifest",
    "label": "Aarau, Aargauer Kantonsbibliothek, MsMurF 3",
    "metadata": [
        {
            "label": "Location",
            "value": "Aarau"
        },
        {
            "label": "Date",
            "value": [
                {
                    "@value": "1508",
                    "@language": "de"
                },
                {
                    "@value": "1508",
                    "@language": "en"
                },
                {
                    "@value": "1508",
                    "@language": "fr"
                },
                {
                    "@value": "1508",
                    "@language": "it"
                }
            ]
        }
    ],
    "description": [
        {
            "@value": "Pontificale für Johannes Feie...",
            "@language": "de"
        ...
}
```


b) The first image URL in case of the above manuscript is the following:

```
http://www.e-codices.unifr.ch/loris/kba/kba-0003/
        kba-0003_e001.jp2/full/full/0/default.jpg
```

c) The JSONPath returning links to images for the above manifest is:

```
$.sequences[*].canvases[*].images[*].resource.@id
```

After copying links and removing all commas and quotes the resulting file should look like this:

```
http://www.e-codices.unifr.ch:80/l...03__e001.jp2/full/full/0/default.jpg
http://www.e-codices.unifr.ch:80/l...03__e005.jp2/full/full/0/default.jpg
http://www.e-codices.unifr.ch:80/l...03__000a.jp2/full/full/0/default.jpg
http://www.e-codices.unifr.ch:80/l...03__000b.jp2/full/full/0/default.jpg
http://www.e-codices.unifr.ch:80/l...03__000c.jp2/full/full/0/default.jpg
http://www.e-codices.unifr.ch:80/l...03__000d.jp2/full/full/0/default.jpg
...
```

Then we can download all links with `wget` via

```
wget -i links.txt
```