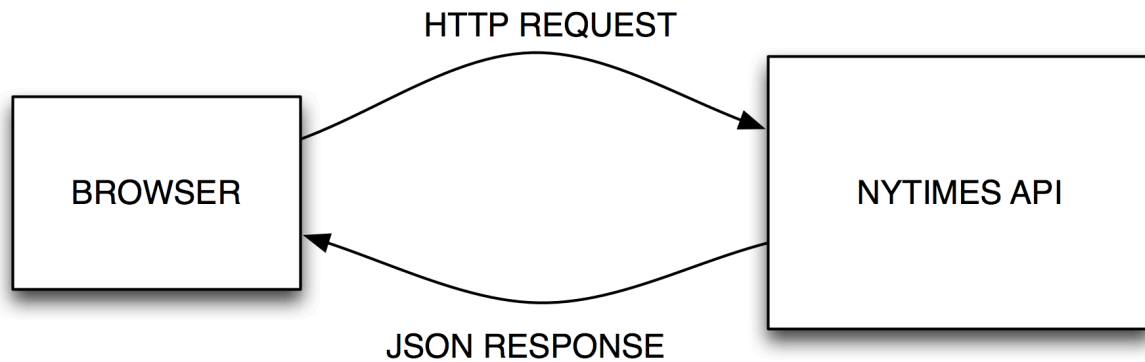


## Assignment 2: The New York Times article search API + processing

### API: Application Programming Interface

A set of rules and specifications that a software program can follow to access and make use of a (web) service



### NYTimes Article search API

The request:



```
http://api.nytimes.com/svc/search/v1/article?query=(field:)  
keywords (facet:[value])(&params)&api-key=your-API-key
```

## Making queries: examples

```
http://api.nytimes.com/svc/search/v1/article?query=text:CO2&api-key=your-API-key
```

The default rank is newest. This means that if you don't specify a period, you will only get articles of the last period

```
http://api.nytimes.com/svc/search/v1/article?query=text:CO2&begin_date:20000101&end_date:20110101 &api-key=your-API-key
```

Even when you specify a begin/end date you will only see 10 articles.

Specifying which fields you want to see:

```
http://api.nytimes.com/svc/search/v1/article?query=text:CO2&begin_date:20000101&end_date:20110101&fields=title &api-key=your-API-key
```

### Facets

- geo\_facet
- per\_facet
- des\_facet
- desk\_facet
- org\_facet

### Different ways to use them

1 - Using field=geo\_facet: you want to get the facet related to the article as a field, in this example the geo\_facet:

```
http://api.nytimes.com/svc/search/v1/article?query=text:CO2&begin_date:20000101&end_date:20110101&fields=title,geo_facet &api-key=your-API-key
```

In the response you'll see the facets names.

Note: By default (unless you include a fields list in your request), the following fields are returned for each result: body, byline, date, title, url

2 - Using facets=geo\_facet

```
http://api.nytimes.com/svc/search/v1/article?query=text:CO2&facets=geo_facet&begin_date:20000101&end_date:20110101&api-key=your-API-key
```

In the response you will see the facet name and how many times that facet is related to articles searched in the query

3 - Specify the name of the facet you are interested in: geo\_facet=[ROTTERDAM (NETHERLANDS)]

```
http://api.nytimes.com/svc/search/v1/article?query=text:CO2
geo_facet:[ROTTERDAM
(NETHERLANDS)]&begin_date:20000101&end_date:20110101&api-key=your-API-key
```

IMPORTANT: If you want to know which categories are available (to use them in the facet search) go to <http://prototype.nytimes.com/gst/apitool/index.html>, enter the term you want to look for in "search query", and the kind of facet you are looking for.

## Examples using the NYTimes api

<http://www.flickr.com/photos/blprnt/3257243248/sizes/o/in/photostream/>  
[http://developer.nytimes.com/trans\\_viz\\_app](http://developer.nytimes.com/trans_viz_app)

other:

<http://mlmckenna.com/news/>  
<http://www.flickr.com/photos/paulmmay/sets/72157626087196057/with/5503956357/>

See more at: [http://developer.nytimes.com/gallery\\_by\\_api](http://developer.nytimes.com/gallery_by_api)

## Getting started

Get an api key (sign up first)

<http://developer.nytimes.com/docs>

Download Jer Thorp's **NYTimes API Processing Library** ("download library" link)

<http://www.blprnt.com/libraries/nytimes>

and eventually the bonus examples

Install the library: just copy the library to a folder called libraries inside of your sketch-book. If you don't manage see also:

[http://wiki.processing.org/w/How\\_to\\_Install\\_a\\_Contributed\\_Library](http://wiki.processing.org/w/How_to_Install_a_Contributed_Library)

Download the examples:

<http://www.torre.nl/nyt.zip> : **Don't forget to include your own api key in the examples, otherwise they will not work**

Read more about the NYtimes api at:

[http://developer.nytimes.com/docs/read/article\\_search\\_api](http://developer.nytimes.com/docs/read/article_search_api)

Test your queries at:

<http://prototype.nytimes.com/gst/apitool/index.html>

Description of Jer Thorp's library

<http://www.blprnt.com/processing/libraries/nytimes/docs/blprnt/nytimes/package-summary.html>

Reading:

Beautiful Visualization

Chapter 15: This was 1994: Data Exploration with the NYTimes Article Search API

Here you can find some more examples using the library.