# Web scraping and social media scraping: Scraping single static page

Przemysław Kurek, Maciej Wysocki

Chair of Political Economy
Faculty of Economic Sciences
University of Warsaw

Class 04

**We already know:**

- How to download a web page file with python.
- Tools for data extraction from HTML file.

**Today we will cover:**

- An application of this tools to real world static pages.
- Storing data in Pandas Dataframe.

Uniwersytet Warszawski
Wydział Nauk Ekonomicznych

## Static and dynamic pages:

- A *static web page* is a page that is delivered to the user's web browser exactly as stored.
- A *server-side dynamic web page* is a page whose construction is controlled by a server processing server-side scripts.
- A *client-side dynamic web page* processes the page using HTML scripting running in the browser. JavaScript and other scripting languages determine the site construction.
- However we will refer to all sites without javascript or other dynamics as static - because they do not change after being loaded. Also, when a page has some javascript, but all interesting content is static (like Wikipedia), lets consider it for our purposes static too.
- True static pages are quite rare nowadays. You can find them mostly as micro business or personal sites.

### Scraping single static page:

- Scraping one is the first step to scraping many. We do it hoping, that we can loop it for pages with similar layout.
- We should make sure, that the code will work at least for few other pages we want to scrap.
- It depends on project size, but at this moment we **could** think of handling some exceptions and not using for example too many `loc`'s and `iloc`'s, as they are terribly inefficient...
- ...however **we should not optimize code too early** (and yes, now is too early).
- In general - we want to use good, *pythonic* practices, but we also want to save coders time, which is precious :)
- We may run *profiler* later and try address real problems, which cause loss of crawling time, not imaginary ones, that we can think of them now.
- It is good idea to cover the whole process - from downloading a page to exporting data into memory or file.

**When you know, that our scraper will go through many sites:**

- Many means rather tens of thousands. Smaller scrapers are not worth too much trouble.
- You might want to extract data directly to hard drive, as next csv line. This both addresses problem of unexpected crash and potential problem of memory (connected to dynamically extending Pandas data frame which slows in time).
- You should definitively handle exceptions well, or have plan to divide project into many parts.

Uniwersytet Warszawski
Wydział Nauk Ekonomicznych

## Storing data:

- With Beautiful Soup we will be exporting data to memory, to Pandas data frame, then to csv file.
- Pandas (or Numpy) are definitively the most basic Data Science tools and putting every data there feels natural.
- However it is not the best idea!
- In Pandas you should make as little single cell operations as possible. If possible operate on series. At least add rows, not single cells.
- Do not dynamically extend data frame. Create full size at the beginning. If you do not know how big it should be you can extend it dynamically twice or by thousand(s) rows.
- When going towards big data you may learn other, more adequate techniques of reading, storing, analysing and saving data. Use them, when your project will be bigger.

Uniwersytet Warszawski
Wydział Nauk Ekonomicznych

### Example:

- The main part of this class is to finally attack real world jungle!
- Run and analyze codes from files:
  - 04a_link_list.py
  - 04b_links.py
  - 04c_painter.py

### Classroom activity:

- Read and solve exercises in 04_exercises.pdf file.