Scraping Multiple Pages in Python

WORKSHOP 3 | CREATOR: CHARLOTTE LLOYD

Outline

I. Recap
II. Workshop Example
III. Verify Data
IV. Celebration, Back-slapping

Recap

PART I

Three Major Ways to Use Python

1. Command Line
2. “IDE”
3. Notebook

Scraping Process // Battle Plan

- 1. Surveillance
  - Evaluate the page, learn the terrain.
- 2. Plan of Attack
  - Brainstorm ways to approach the enemy.
- 3. Write code
  - Be willing to change your strategy if you encounter obstacles or see another "weakness" to exploit.
- 4. Emerge bloodied, yet victorious.
  - Verify the data before all that syntax evaporates from your short term memory.

Workshop Example

PART IV

GOAL

- http://www.bfi.org.uk/films-tv-people/sightandsoundpoll2012/voters
- Scrape all information about all voters
- Scrape “film details” (except ”featuring”) for all films chosen by voters in their “top ten"
- Save data as 2 different csv files

1. Surveillance

- Voter: http://www.bfi.org.uk/films-tv-people/sightandsoundpoll2012/voter/94
  - special case: http://www.bfi.org.uk/films-tv-people/sightandsoundpoll2012/voter/6
- Film: http://www.bfi.org.uk/films-tv-people/4ce2b6a7a801b
  - special case: http://www.bfi.org.uk/films-tv-people/4ce2b8bb6b693
  - special case: http://www.bfi.org.uk/films-tv-people/4ce2b7d2993a2

2. Plan of Attack: Voters

- What is our strategy to get the judge URLs?
  - exploit the "class=sas-poll" feature to scrape URLs from each of 25 tables
- What is our strategy to get the data for each judge?
  - scrape the name, type, info and country from the main page
  - scrape the 10 films and comment from the judge's individual page
- How can we handle the special cases?
  - manually create filmIDs for films without webpages

2. Plan of Attack: Films

- What is our strategy for getting the film URLs?
  - save them to a list while we're scraping the judges
- What is our strategy to get the data for each film? Why do we have to incorporate the special cases directly into the strategy?
  - we need to separately search for cells containing the director, country, year, genre, type, and category info
  - the number of cells in the table varies, so we have to know what they are based on their content and not their position

3. Let's look at the code together

- available at: https://github.com/charlloyd/film-gaze
- First let's run it in Spyder.
- Then let's download the jupyter notebook.