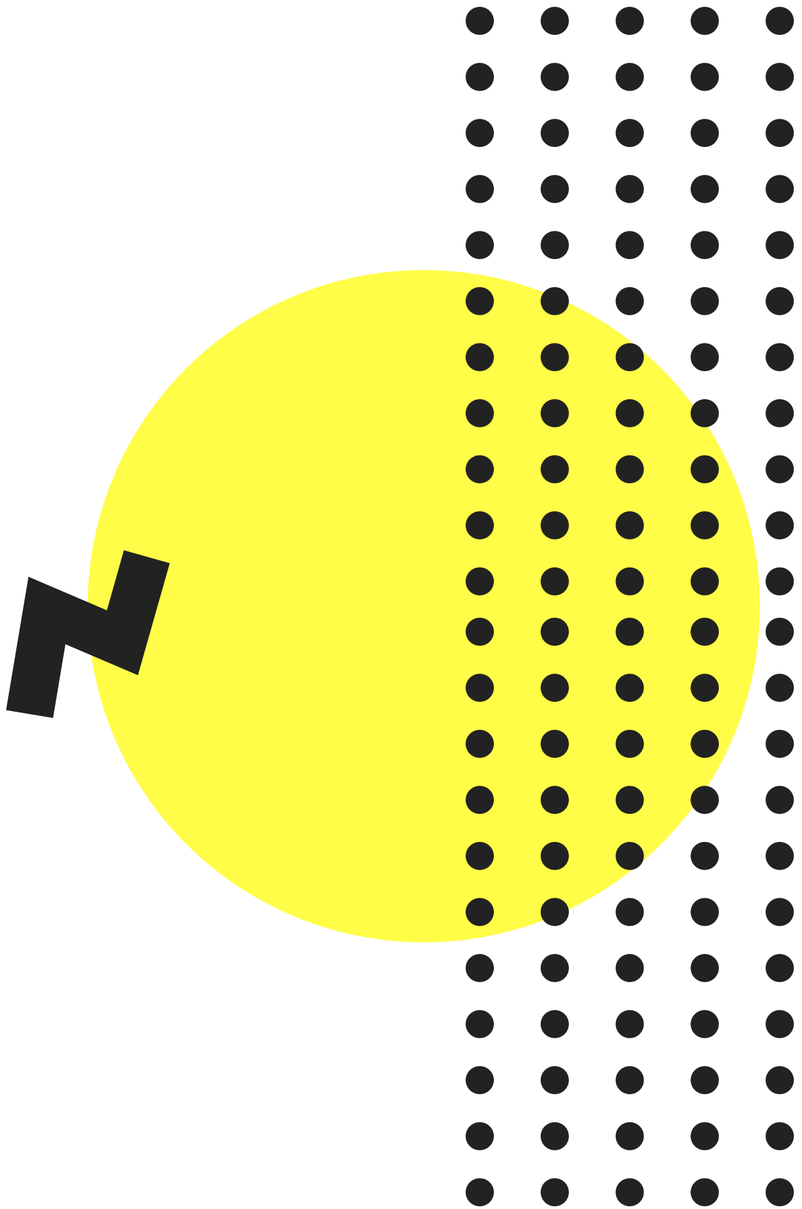
Парсинг данных

YISAEVA@HSE.RU
VIGNATOVA@HSE.RU

Структура типового сайта

```
<!DOCTYPE html>
<html>
  <head>
    <title> Web page name
    </title>
    .....
  </head>
  <body>
    <h1> Post title
    </h1>
     .....
  </body>
</html>
```



Задача: написать пути (xpath) к элементам сайта

КАК ЭТО ВЫГЛЯДИТ

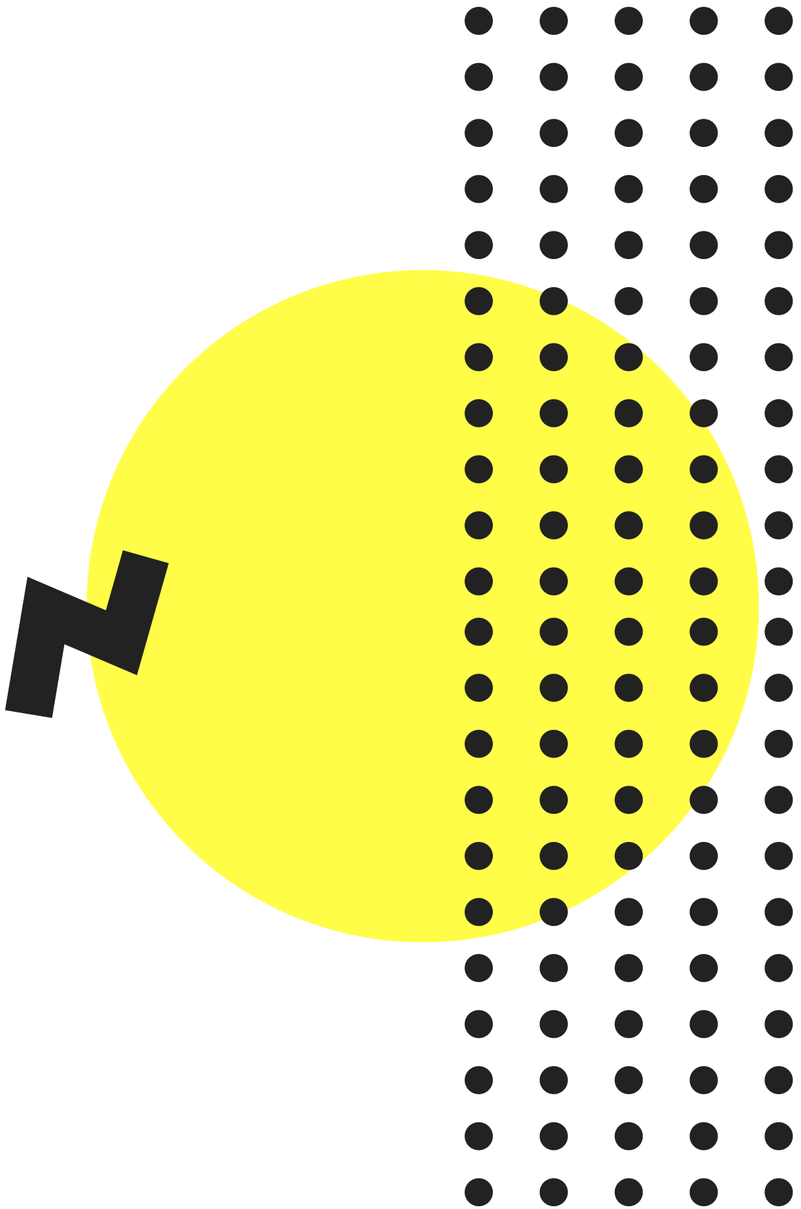
HEADER_XPATH = ['//h1/text()']
AUTHOR_XPATH = ['//span[@class="author"]/a/text()']
PUBDATE_XPATH = ['//span[@class="date"]/text()']
TAGS_XPATH = ['//span[@class="tag"]/text()']
CATEGORY_XPATH = ['//span[@class="category"]/text()']
TEXT_XPATH = ['//div[@class="article_body"]/p//text()']
INTERLINKS = ['//div[@class="article_body"]//p//a/@href']
DATE_FORMAT_STRING = '%d %b %Y'

КРАСНЫМ ОТМЕЧЕНЫ ОБЯЗАТЕЛЬНЫЕ ЭЛЕМЕНТЫ

Типовой парсер

ПОРЯДОК ДЕЙСТВИЙ

1. Активируем среду: **source ve/your_folder_name/bin/activate**
2. Запускаем шэлл: **scrapy shell**
3. Устанавливаем соединение с сайтом:
   fetch('https://www.yoursite.ru/news/news-title-whatever').
   Crawled (200) означает, что все хорошо.
4. Пишем пути к элементам и проверям, подставляя в конструкцию: **response.xpath('path/to/element').extract()**



Типовой парсер

https://www.brokernews.com.au/

КАК ПИСАТЬ ПУТИ К ЭЛЕМЕНТАМ

1. Зайти на сайт через браузер, открыть какую-то новость
2. Выделить нужный элемент и щекнуть правой кнопкой мыши
3. В выпадающем меню найти инспектор страниц / просмотр кода
4. Сначала указать название контейнера, в котором лежит нужный фрагмент (прим. <h1 ...>, <div ....>,)
5. Назавание контейнера часто неуникальное, поэтому в квадратных скобках можно дополнительно указать атрибуты (прим. //h1[@**class="page_title"**])
6. Далее указать, что именно надо извлечь - обычно текст (прим. //h1[@class="page_title"]**/text()**), ссылку (//div[@class="article_body"]//p//a/@**href**)

! Ссылка в примере указывается через @, так как является атрибутом и лежит внутри контейнера

Title, pubdate

Иногда случается, что в качестве заголовка или даты извлекается целый список элементов, а **правильный заголовок и дата** могут быть только **одни**. Можно указать порядковый номер элемент (нумерация с 1)

HEADER_XPATH = ['(//h1/text())[2]']

Прим. s_helicoptersmagazine

Pubdate

Дата указывается как строка

DATE_FORMAT_STRING = '%Y-%m-%d'

2009-01-14T19:54:26+00:00 - очень удобный формат для извлечения даты. Искать нужно в контейнерах meta поиском по коду страницы по году (прим. **s_canadiansecuritymag**)

Для указания даты в другом формате: **http://strftime.org/**

Избавление от мусорных элементов

```
from newsfeeds.item_functions import clear_text
.....
item['author'] = process_array_item(self, response, AUTHOR_XPATH, single=False)
if item['author'] is not None:
   authors = []
   for author in item['author']:
         authors.append(clear_text(author).replace('what to replace', 'with what'))
.....
```

Прим. sm_airwaysmag

Наследование

Необходимо, чтобы текст и ссылки не содержали в себе "наследников" script и style

Предположить их наличие можно, посмотрев на то, что находится за последним контейнером, содержащим текст

```
TEXT_XPATH = ['//p[not(@id="articledate")]//text()[not(ancestor::style)]']
INTERLINKS = ['//p[not(@id="articledate")]//a[not(ancestor::style)]/@href']
```

Прим. sm_controlengeurope

Несколько путей

Бывает, что структура новостных страниц на одном сайте различается. В таких случаях необходимо указывать альтернативные пути к элементам через логический оператор **|**

TEXT_XPATH = ['(//div[@class="field-item even"]//p//text())|(//div[@class="field-items"]//div/text())']
INTERLINKS = ['(//div[@class="field-item even"]//p/a/@href)|(//div[@class="field-items"]//div//@href)']
TEXT_XPATH = ['//div[@class="articletext"]//p//text()|
//div[@class="articletext"]/ul/li/text()']

Прим. sm_greenbiz, s_progressiveforage