Module 1: Scrapy Shell:

1. Module overview:

Hello, and welcome, on this module I'll talk about scrapy shell and how to use it. I'll talk also about the differences between the two more known scrapping methods, web scrapping and web crawling and then I'll define scrapy, how to prepare the environment and how we can extract data from websites with Scrapy shell using XPath and CSS selectors.

2. Scrapy

"An open source and collaborative framework for extracting or crawling structured data that you need from websites. In a fast, simple, yet extensible way."

« https://scrapy.org/»

Scrapy is used for web Crawling, between web crawling and web scrapping we can find a variance:

Web Crawling	Web Scrapping
Download Websites and then index it.	Get and extract Data directly from websites
Duplication is essential on web Crawling	There is no duplication
General crawl all the website	Specific scrape only the Data that you want to get.

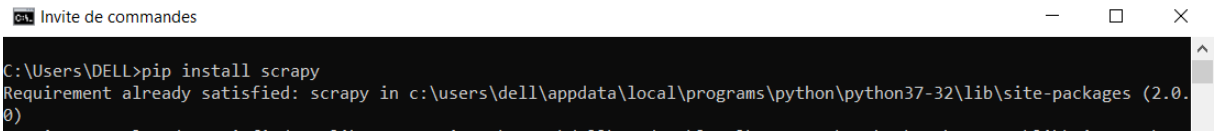
Important: we can scrape data from websites using Beautiful Soup a python library, but on this course, we are going to use Scrapy, it's a python framework to collect data from websites not a library.

3. Preparing Scrappy environment

In this part I'll talk about how to prepare the environment by installing Scrapy and we will explore the scrapy shell and how we can use it to download websites and work with the HTML.

To install the Scrapy package in a local machine we must open the terminal and type

Terminal → Pip install scrapy

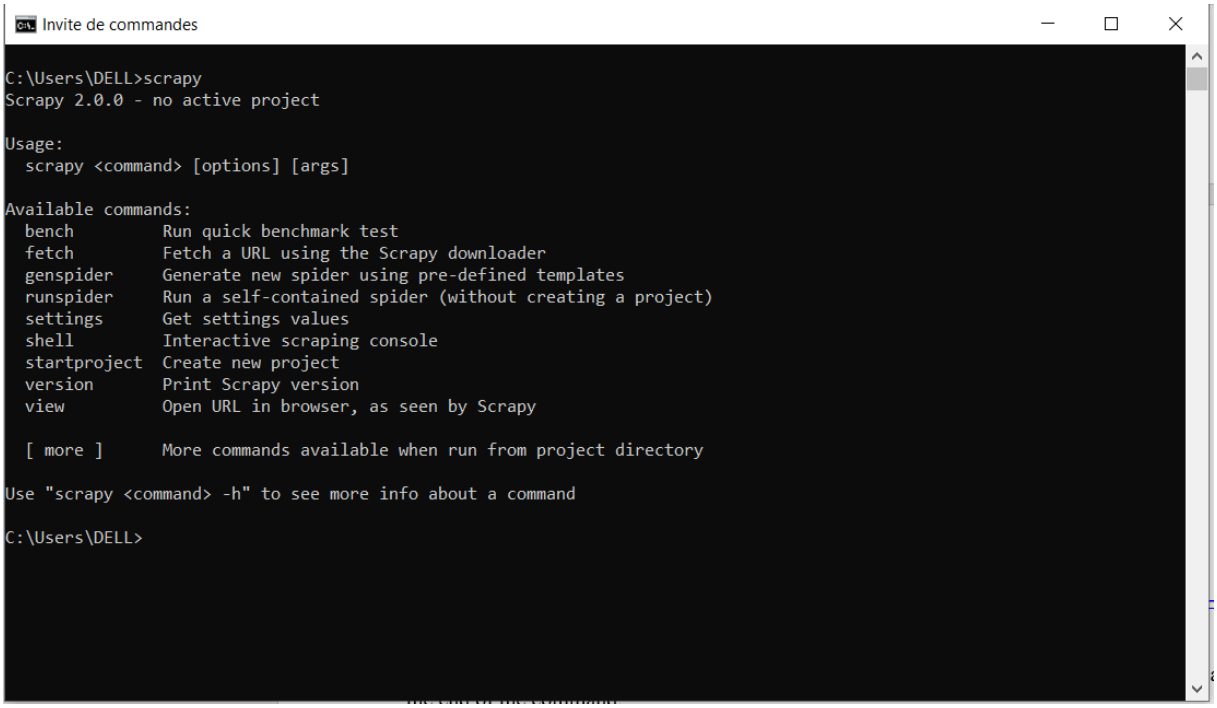


And after you must check that you installed the latest version 2.0.1

Once scrapy is successfully installed in your local machine you can run on your Terminal

Terminal → scrapy

To see the scrapy information's page wich composed by the version That You have and the available commands, we are going to see a few of those commands

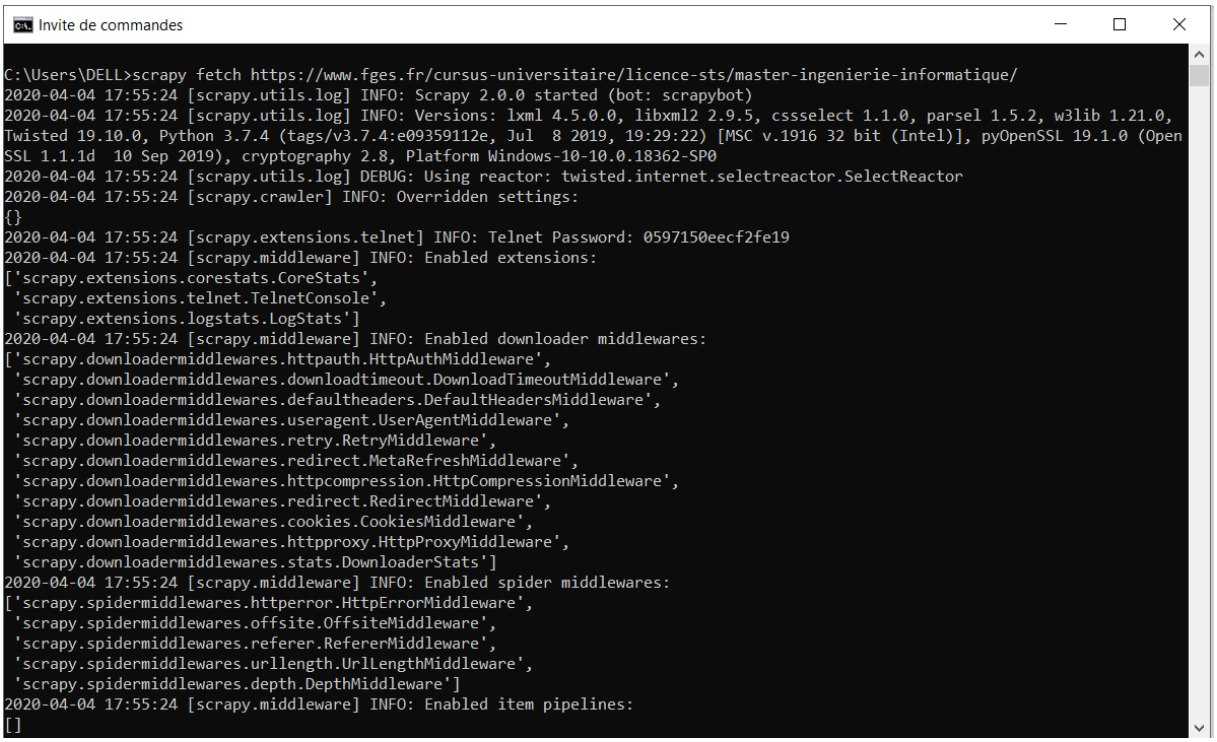


To print the scrapy version we can type: Terminal → scrapy version

To fetch an URL using the scrapy downloader we can type scrapy fetch, we are going to download the contents of our master and write on the terminal Window.

URL: Fges.fr → Master III

Terminal → scrapy fetch +URL



If you want to save the data collected from the webpage you have to add > and the file name at the end of the command.

Terminal → scrapy fetch +URL > fges.html

4. Scrappy Shell

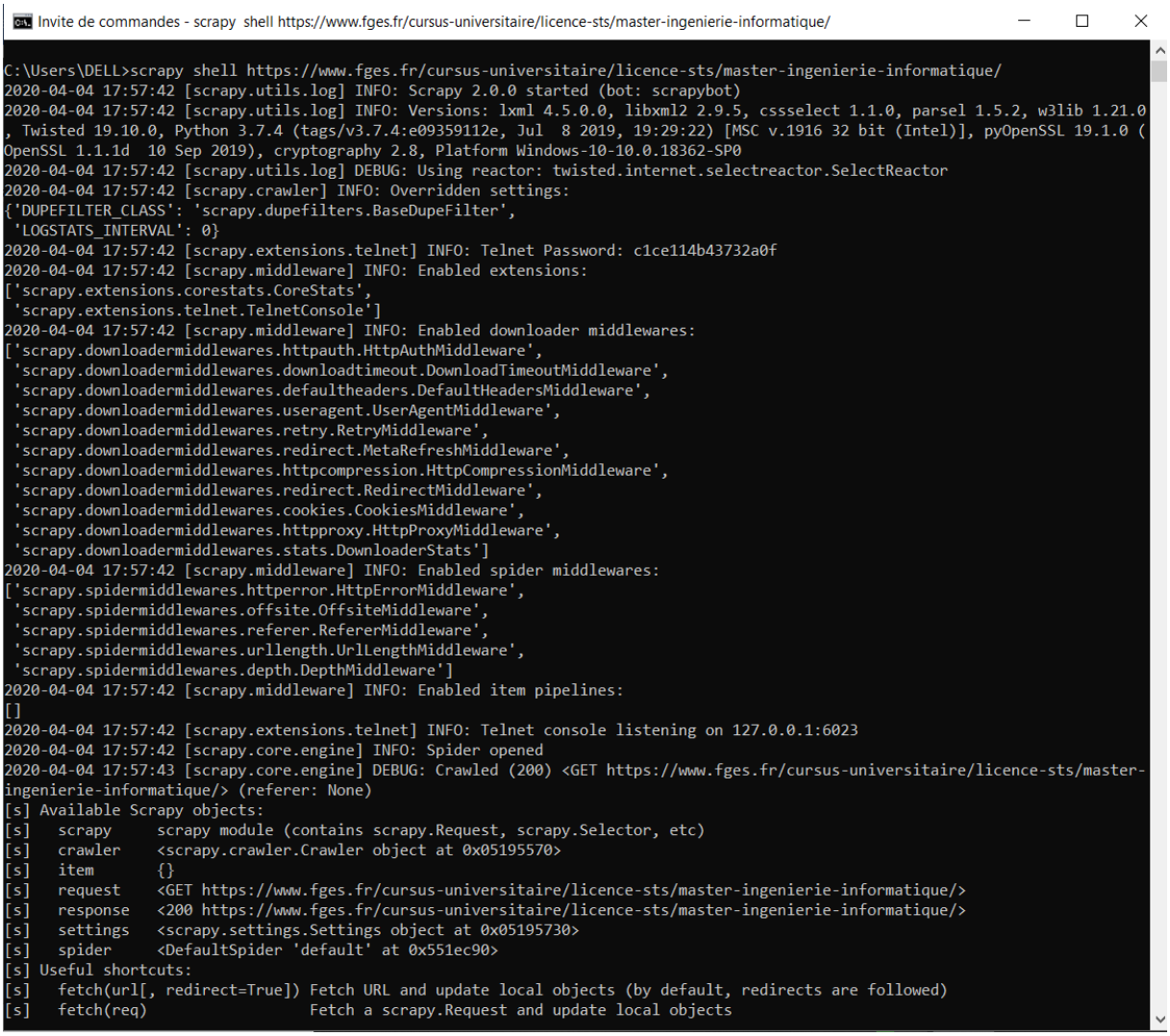
"The Scrapy shell is an interactive shell where you can try and debug your scraping code very quickly, without having to run the spider."

Spiders are classes with them we can describe how websites will be scrapped.

« https://docs.scrapy.org/en/latest/topics/shell.html »

To use the scrappy Shell, you must add scrappy shell before the URL in your terminal

Terminal → scrapy shell +URL



You can type shelp() command to see the available scrapy objects

```
>>> shelp()
[s] Available Scrapy objects:
[s]   scrapy     scrapy module (contains scrapy.Request, scrapy.Selector, etc)
[s]   crawler    <scrapy.crawler.Crawler object at 0x05195570>
[s]   item       {}
[s]   request    <GET https://www.fges.fr/cursus-universitaire/licence-sts/master-ingenierie-informatique/>
[s]   response   <200 https://www.fges.fr/cursus-universitaire/licence-sts/master-ingenierie-informatique/>
[s]   settings   <scrapy.settings.Settings object at 0x05195730>
[s]   spider     <DefaultSpider 'default' at 0x551ec90>
[s] Useful shortcuts:
[s]   fetch(url[, redirect=True]) Fetch URL and update local objects (by default, redirects are followed)
[s]   fetch(req)                  Fetch a scrapy.Request and update local objects
[s]   shelp()           Shell help (print this help)
[s]   view(response)    View response in a browser
>>>
```

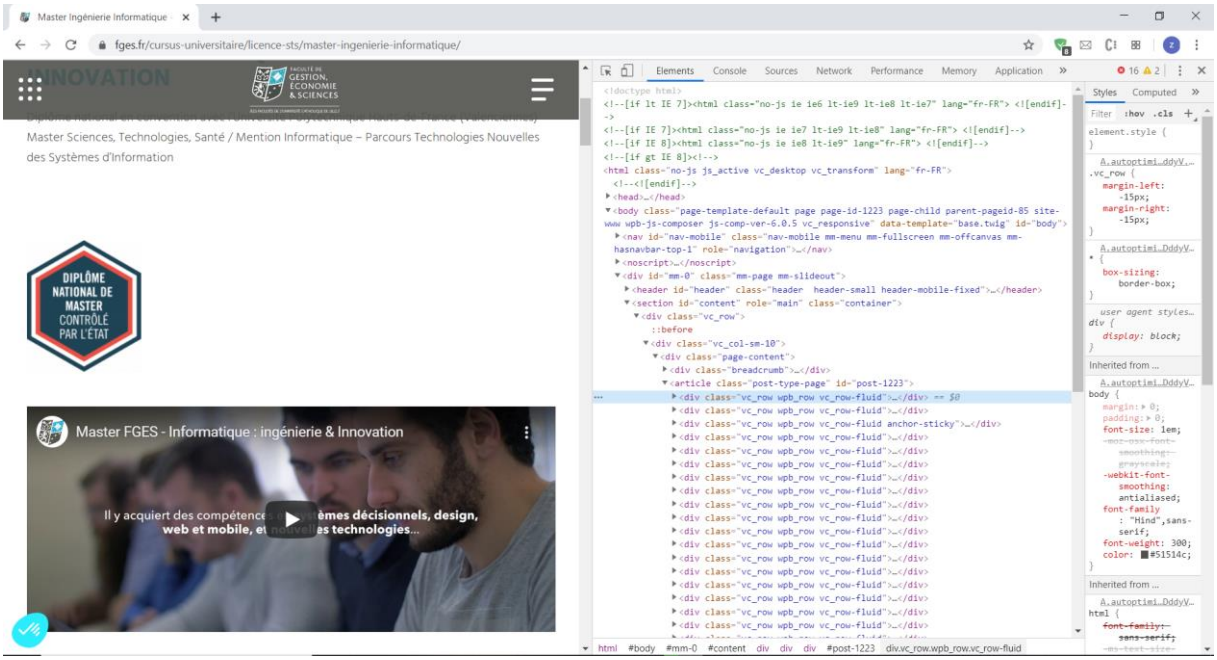
5. Select Elements using CSS selector

In this part I'll talk about how to use the famous scrapy CSS selector to get or collect data from webpage. So, we will parse the page of our master using CSS selectors:

We have first to download the webpage using

Terminal → scrapy shell +URL

You have to go to the site and write click on webpage and after you can use your navigator inspector



If we want to get the title data, we have to click on head, then you can get the title tag if you want to access to it with scrapy we have just to type on the terminal:

```
▼ <head>
    <meta charset="UTF-8">
    <link type="text/css" media="all" href="https://www.fges.fr/wp-content/cache/autoptimize/1/css/A.autoptimize_b6d8bf4….css.pagespeed.cf.COycWDddyV.css" rel="stylesheet">
    <title>Master Ingénierie Informatique - FGES Lille</title>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge,chrome=1">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <link rel="pingback" href="https://www.fges.fr/xmlrpc.php">
    <link href="https://fonts.googleapis.com/css?family=Hind:300,400,500,600,700" rel="stylesheet">
```

Terminal → response.css('title') → and the result will be a selector object

```
>>> response.css('title')
[<Selector xpath='descendant-or-self::title' data='<title>Master Ingénierie Informatique...'>]
>>>
```

To get only the text you must add

Terminal → response.css('title::text').get()

```
>>> response.css('title::text').get()
'Master Ingénierie Informatique - FGES Lille'
>>>
```

To select something else from the webpage you have just to click "right mouse button" on the element and then use the navigator inspector to see the html element to extract the paragraph text with scrappy shell, you have just to type

Terminal → response.css('p::text').get()

```
>>> response.css('p::text').get()
' '
>>>
```

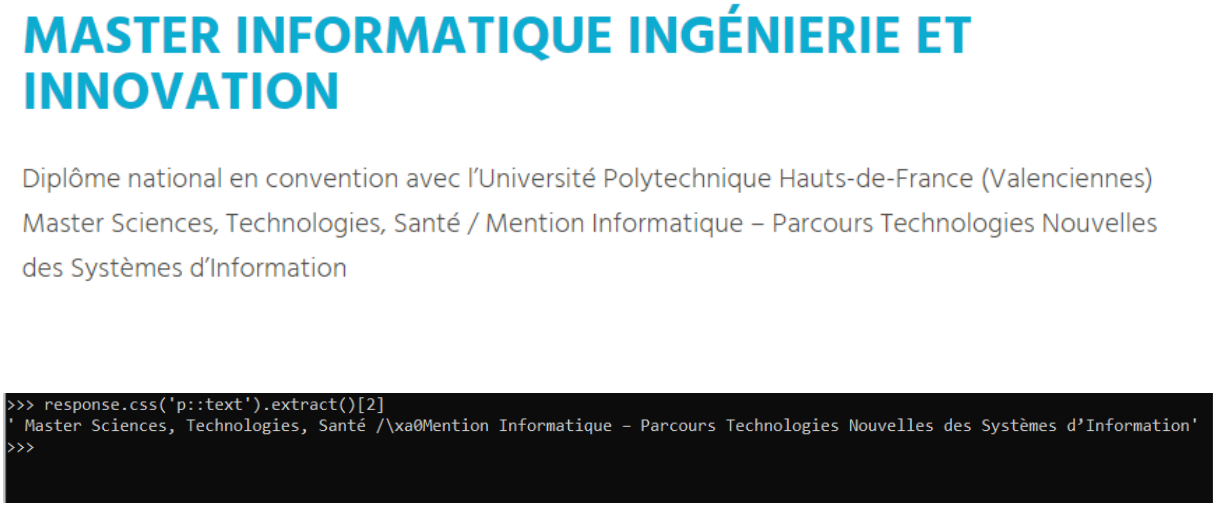
The response is null because on the webpage we have a lot of 'p' elements and scrappy shell will send for us the first one, you can see here that there is no data on the first 'p' element so for that we can use

Terminal → response.css('p::text').getall() to get all the 'p' elements text on the webpage

```
>>> response.css('p::text').getall()
[' ', 'Diplôme national en convention avec l'Université Polytechnique Hauts-de-France (Valenciennes)', ' Master Sciences, Techn
ologies, Santé /\xa0Mention Informatique – Parcours Technologies Nouvelles des Systèmes d'Information', '\xa0', 'Des étudiants
titulaires d'une licence (informatique ou équivalent) ou d'un titre RNCP niveau 2 dans le domaine du numérique qui veulent maît
riser les dernières technologies mais aussi la gestion de projets. Ils préfèrent la pratique à la théorie et souhaitent intégre
r un cursus en alternance.', 'Les matières et les crédits présentés dans ce programme sont susceptibles d'évoluer sans remettre
 en cause les contenus et orientations essentiels de la formation.', 'En alternance de septembre à mai. Du lundi au mercredi en
 entreprise + le jeudi et vendredi en cours.', ' En entreprise de juin à août.', 'En alternance de septembre à mai. Du lundi au
 mercredi en entreprise + le jeudi et vendredi en cours.', ' En entreprise de juin à août.', '*Projet de Master (300 heures) :
conduire de la conception à la mise en production un projet grandeur nature en lien avec le parcours et mettant en oeuvre les t
echnologies du Master.', 'L'intelligence artificielle (IA) est certainement la révolution technologique de ce siècle.', 'Elle p
ermettra à terme aux machines de percevoir, de comprendre, d'apprendre et d'agir. Et parce que l'IA s'insinue déjà dans tous le
s métiers et secteurs d'activité, elle changera le monde.', 'Le parcours « Intelligence Artificielle » permettra aux étudiants
d'approfondir leurs connaissances sur les algorithmes et les modèles dédiés à l'IA et de les pratiquer.', 'Choisir ce parcours
c'est\xa0:', 'Le progrès technologique aidant, il est aujourd hui assez commun pour un ordinateur de calculer en temps réel des
 graphismes proches du photoréalisme. Dès lors de nouvelles formes d'interactions homme / machine beaucoup plus immersives sont
 possibles.', 'On distingue la réalité augmentée qui superpose de l'information calculée au monde réel percu par l'utilisateur,
 de la réalité virtuelle qui immerge totalement l'utilisateur en le coupant du monde réel.', 'Les applications de ces réalités
numériques sont multiples et vont de la formation au jeu vidéo, en passant par la chirurgie, la robotique ou l'exploration spat
iale !', 'Choisir ce parcours c'est\xa0:', 'Le nombre d'objets connectés à internet qui nous entoure est en croissance exponent
ielle depuis plusieurs années. En effet, on ne conçoit plus qu'un nouveau service ou produit ne soit pas accessible sur le web
ou sur une application.', 'L'être humain est désormais minoritaire sur internet, cerné par des machines «\xa0intelligentes\xa0»
, et communiquant entre elles et gérant son environnement et données.', 'Choisir ce parcours c'est\xa0:', '\xa0', 'Le master In
formatique a été une réelle opportunité pour ma future vie professionnelle.', ' Les cours m'ont apporté toutes les connaissance
s nécessaires pour avoir de la maîtrise en développement informatique. Les projets ont renforcé mon esprit d'équipe, mon goût d
u challenge, mon intérêt pour l'innovation. L'alternance en entreprise m'a donné l'expérience nécessaire pour donner confiance
aux employeurs dans ma capacité d'adaptation.', ' Résultat : j'ai intégré dès l'obtention de mon diplôme un cabinet d'ingénieri
e informatique où je m'épanouis pleinement.', ' ', ' ', 'Ce master m'a permis de me forger une véritable expérience professionn
elle reconnue par les entreprises\xa0; c'est un réel "plus" par rapport à d'autres formations. L'alternance est une formidable
façon d'apprendre, car cela permet d'acquérir des compétences indispensables pendant les cours et de les appliquer très concrèt
ement durant les périodes en entreprise. Grâce au contrat de professionnalisation j'ai trouvé un emploi avant même l'obtention
du diplôme.', ' ', ' ', ' 60, Boulevard Vauban, CS\xa040109, 59016 Lille', ' Tél. : 03 20 13 40 20 (Licences + ISEA)', ' Tél. :
 03 28 38 48 94 (Masters)', 'Nous rejoindre sur', ' ', ' ', ' ']
>>>
```

And to get the second element for example we can type

Terminal → response.css('p::text').extract()[2]




```
>>> response.css('p::text').extract()[2]
' Master Sciences, Technologies, Santé /\xa0Mention Informatique – Parcours Technologies Nouvelles des Systèmes d'Information'
>>>
```

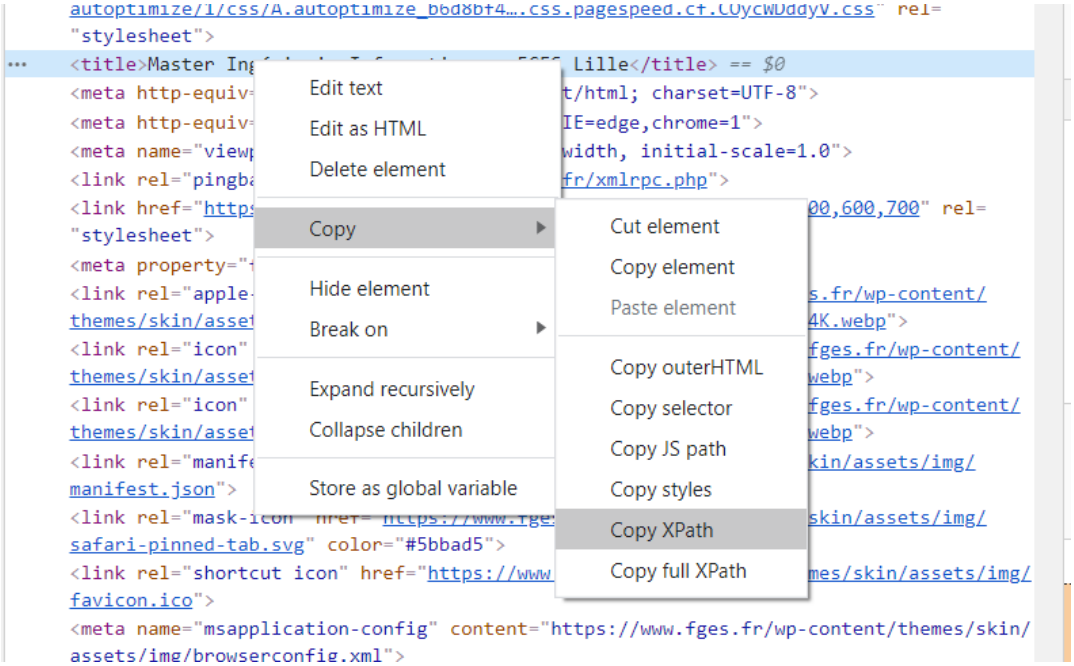
And we can applicate the same logic with any element on the page.

6. Select Elements using xPATH selector

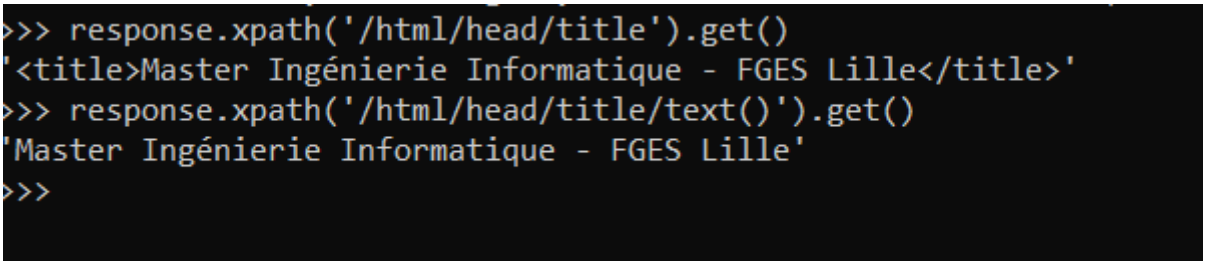
In this part I'll talk about how to use the second element provided by scrapy the scrapy XPATH selector allowing extracting data from webpage

First, we are going to restart our scrappy shell and then we must open again the webpage on navigator and right click on it to inspect the page.

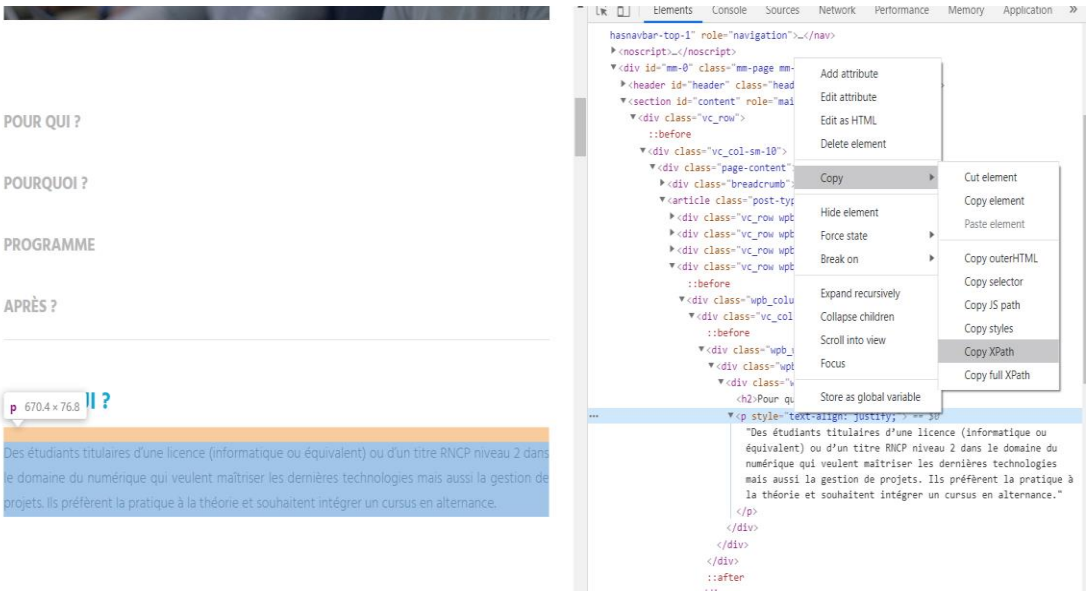
You must click on head and after right click on title then copy option and copy the Xpath and after we can use the xpath selected to get data with scrappy shell



we can add /text() to get only the element text



We can select a paragraph element with the xpath selector we have just to do the same thing and get the xpath from chrome inspector then



Terminal → response.xpath('XpathID+/text()').get()

```
>>> response.xpath('//*[@id="post-1223"]/div[4]/div/div/div/div/div/p').get()
'<p style="text-align: justify;">Des étudiants titulaires d’une licence (informatique ou équivalent) ou d’un titre RNCP niveau
2 dans le domaine du numérique qui veulent maîtriser les dernières technologies mais aussi la gestion de projets. Ils préfèrent
 la pratique à la théorie et souhaitent intégrer un cursus en alternance.</p>'
>>> response.xpath('//*[@id="post-1223"]/div[4]/div/div/div/div/div/p/text()').get()
'Des étudiants titulaires d’une licence (informatique ou équivalent) ou d’un titre RNCP niveau 2 dans le domaine du numérique q
ui veulent maîtriser les dernières technologies mais aussi la gestion de projets. Ils préfèrent la pratique à la théorie et sou
haitent intégrer un cursus en alternance.'
>>>
```

And we can applicate the same logic with any element on the page.

7. Summary

On this module we talked about scrappy and the differences between web crawling and web scrapping then we saw how to install scrappy and how to use the scrappy shell to fetch data from webpages using css and xpath selectors.

Module 2: Scrapy Spiders:

1. Module overview

Hello, and welcome, on this module we are going to see how to use scrapy spiders. In the last module we saw how to crawl Data from a webpage using scrapy shell but if you want to productionize your code you have to use scrapy spiders.

We will define in classes the crawling job using spiders to robot data crawling.

2. Create custom spider

“Spiders are classes which define how a certain site will be scraped, including how to perform the crawl and how to extract structured data from their pages.”

https://docs.scrapy.org/en/latest/topics/spiders.html

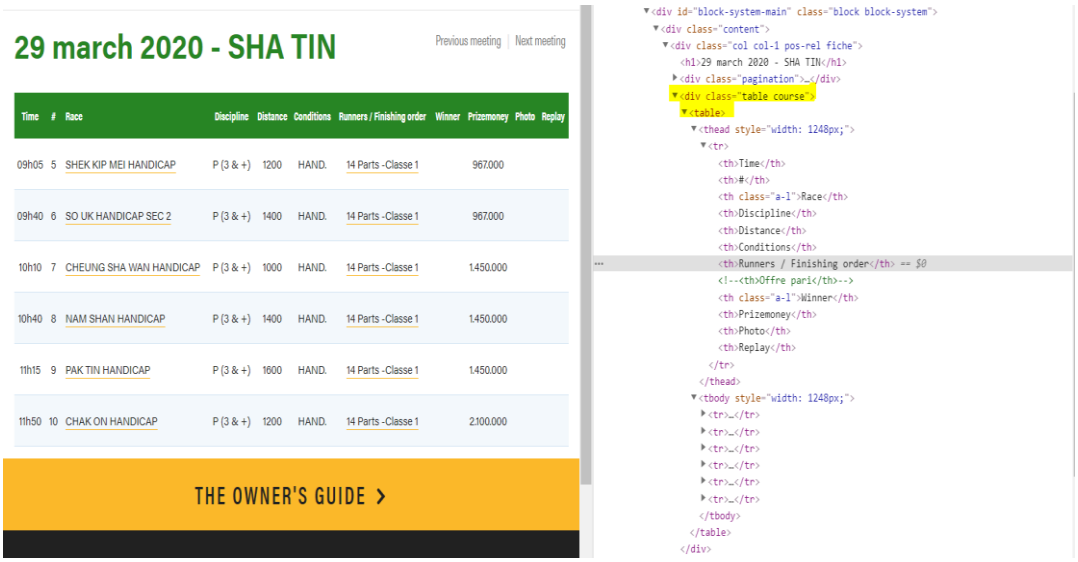
in this I’ll talk about how to initialize a spider to extract the contents of this page

URL: France Galop 29 march 2020 - SHA TIN

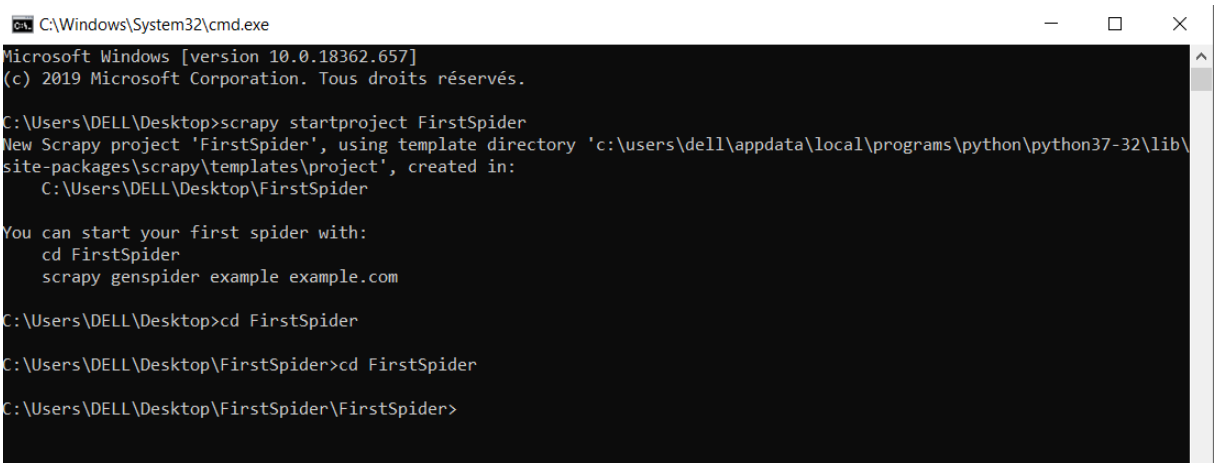
this page gives us the ranking of a horse race.

There is a table with the race distance and Prizemoney before extracting data we must inspect this webpage.

We can note here we have a div class ‘table courses’ and inside it we have a table element with all information.



To start a project, we must type on terminal



We can note here our project is created successfully to create our first spider we have to type

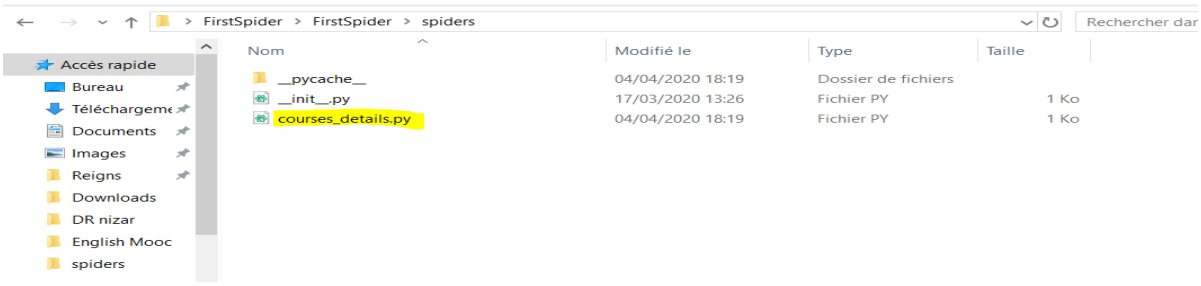
Terminal → cd spiders

Then:

scrapy genspider courses_details +URL

```
C:\Users\DELL\Desktop>scrapy startproject FirstSpider
New Scrapy project 'FirstSpider', using template directory 'c:\users\dell\appdata\local\programs\python\python37-32\lib\
site-packages\scrapy\templates\project', created in:
    C:\Users\DELL\Desktop\FirstSpider

You can start your first spider with:
    cd FirstSpider
    scrapy genspider example example.com

C:\Users\DELL\Desktop>cd FirstSpider

C:\Users\DELL\Desktop\FirstSpider>cd FirstSpider

C:\Users\DELL\Desktop\FirstSpider\FirstSpider>cd spiders

C:\Users\DELL\Desktop\FirstSpider\FirstSpider\spiders>scrapy genspider courses_details http://www.france-galop.com/
Created spider 'courses_details' using template 'basic' in module:
  FirstSpider.spiders.courses_details

C:\Users\DELL\Desktop\FirstSpider\FirstSpider\spiders>
```

→ we can see here a new python file is generated.



We are going to edit the code and add

```
def parse(self, response):
    courses_names = response.xpath(
    '//*[@id="block-system-main"]/div/div/div[2]/table/tbody/tr[*]/td[3]/a/text()').extract()
    courses_distance = response.xpath(
    '//*[@id="block-system-main"]/div/div/div[2]/table/tbody/tr[*]/td[5]/text()').extract()
    course_prizeMoney = response.xpath(
    '//*[@id="block-system-main"]/div/div/div[2]/table/tbody/tr[*]/td[9]/text()').extract()
    count = len(courses_names)
```

* we have to change the xpath, just we have to put * to select all the tr in the table

And then to print the result you must add

```
for i in range (0,count) :

 print(courses_names[i] , courses_distance[i] , course_prizeMoney[i])
```

Result:

courses_details.py

```python
# -*- coding: utf-8 -*-
import scrapy

filename = 'courses_details.txt'

class CoursesDetailsSpider(scrapy.Spider):
    name = 'courses_details'
    start_urls = ['http://www.france-galop.com/en/racing/meeting/20200329/QzRsYU1WMUhSb05sUzQ4UzZVdGVXZz09']

    def parse(self, response):
        courses_names = response.xpath(
        '//*[@id="block-system-main"]/div/div/div[2]/table/tbody/tr[*]/td[3]/a/text()').extract()
        courses_distance = response.xpath(
        '//*[@id="block-system-main"]/div/div/div[2]/table/tbody/tr[*]/td[5]/text()').extract()
        course_prizeMoney = response.xpath(
        '//*[@id="block-system-main"]/div/div/div[2]/table/tbody/tr[*]/td[9]/text()').extract()
        count = len(courses_names)

        with open(filename, '+a') as file:
            for i in range (0,count) :
                file.write(courses_names[i] +' , ' + courses_distance[i] +' , ' + course_prizeMoney[i])
                print(courses_names[i] +' , ' + courses_distance[i] +' , ' + course_prizeMoney[i])
```

finally we have to move to the project directory by typing Terminal → cd ../..

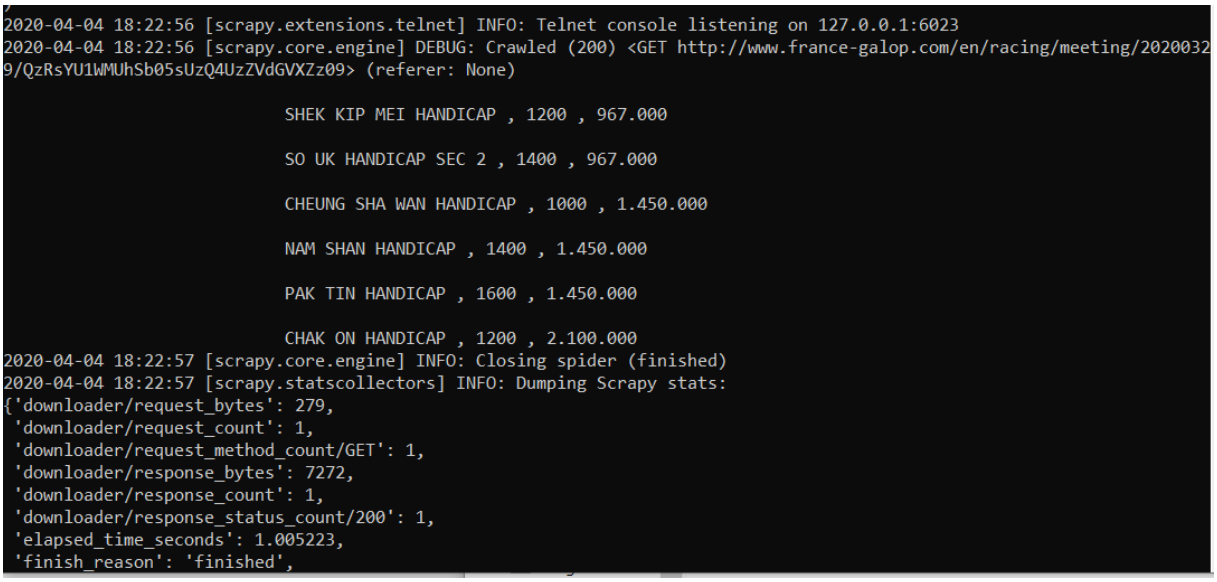
```
C:\Users\DELL\Desktop\FirstSpider\FirstSpider\spiders>cd ../..

C:\Users\DELL\Desktop\FirstSpider>
```

and then we run our script

Terminal → scrapy crawl courses_details

Time	#	Race	Discipline	Distance	Conditions	Runners / Finishing order	Winner	Prizemoney	Photo	Replay
09h05	5	SHEK KIP MEI HANDICAP	P (3 & +)	1200	HAND.	14 Parts -Classe 1		967.000		
09h40	6	SO UK HANDICAP SEC 2	P (3 & +)	1400	HAND.	14 Parts -Classe 1		967.000		
10h10	7	CHEUNG SHA WAN HANDICAP	P (3 & +)	1000	HAND.	14 Parts -Classe 1		1.450.000		
10h40	8	NAM SHAN HANDICAP	P (3 & +)	1400	HAND.	14 Parts -Classe 1		1.450.000		
11h15	9	PAK TIN HANDICAP	P (3 & +)	1600	HAND.	14 Parts -Classe 1		1.450.000		
11h50	10	CHAK ON HANDICAP	P (3 & +)	1200	HAND.	14 Parts -Classe 1		2.100.000		

We can see the Table Data is printed on the terminal



3. Writing extracted data into a file

To write the data extracted into a specific file we have just to edit our code by adding:

1/ filename = 'courses.txt'

The name of the file that will be generated after crawling Data

2/ with open(filename, '+a') as file:

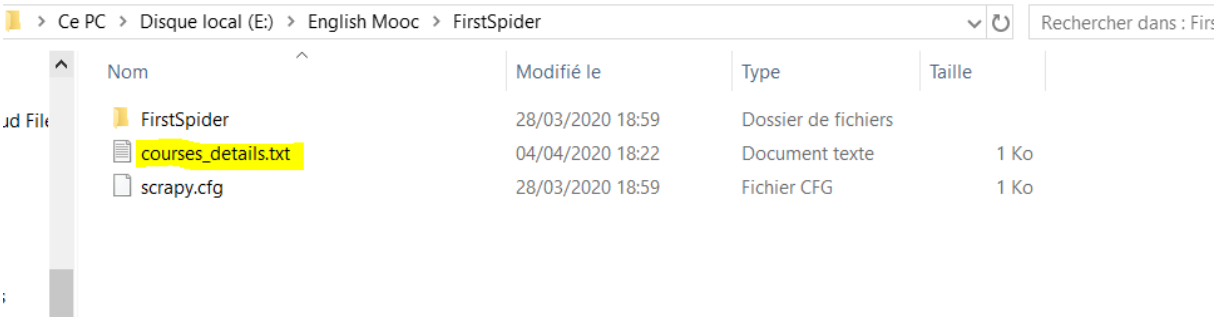
We use +a to write data into the file on append mode

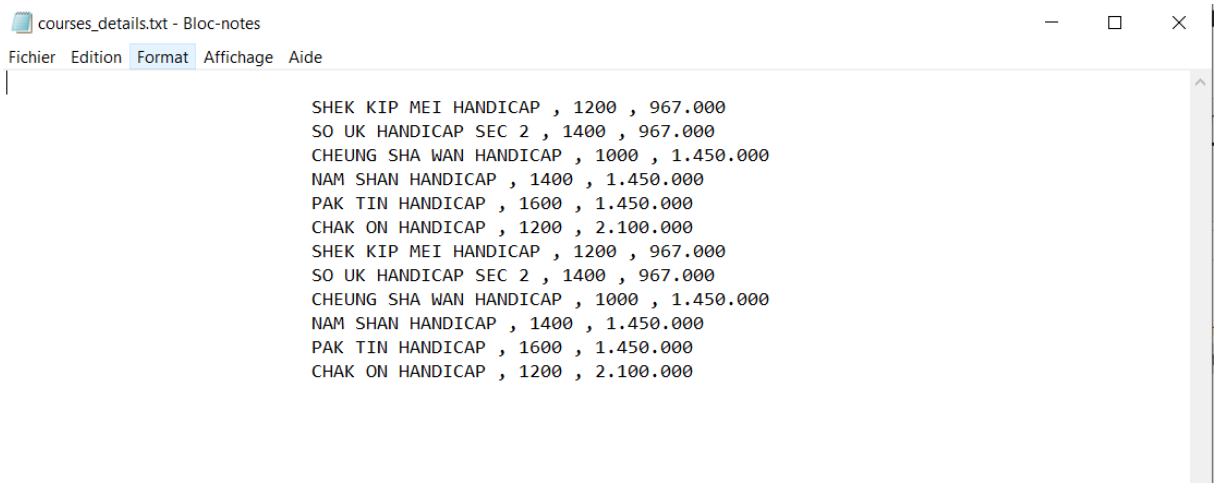
```
for i in range (0,count) :
    file.write(courses_names[i] +' , ' + courses_distance[i] +' , ' +
course_prizeMoney[i])
```

courses_details.py

```python
# -*- coding: utf-8 -*-
import scrapy

filename = 'courses_details.txt'

class CoursesDetailsSpider(scrapy.Spider):
    name = 'courses_details'
    start_urls = ['http://www.france-galop.com/en/racing/meeting/20200329/QzRsYU1WMUhSb05sUzQ4UzZVdGVXZz09']

    def parse(self, response):
        courses_names = response.xpath(
        '//*[@id="block-system-main"]/div/div/div[2]/table/tbody/tr[*]/td[3]/a/text()').extract()
        courses_distance = response.xpath(
        '//*[@id="block-system-main"]/div/div/div[2]/table/tbody/tr[*]/td[5]/text()').extract()
        course_prizeMoney = response.xpath(
        '//*[@id="block-system-main"]/div/div/div[2]/table/tbody/tr[*]/td[9]/text()').extract()
        count = len(courses_names)

        with open(filename, '+a') as file:
            for i in range (0,count) :
                file.write(courses_names[i] +' , ' + courses_distance[i] +' , ' + course_prizeMoney[i])
                print(courses_names[i] +' , ' + courses_distance[i] +' , ' + course_prizeMoney[i])
```

and you can observe that the file is successfully generated with the course data





4. Summary

On this module I talked about how to use scrapy spiders and how to productionize the code to crawl specific data from webpage and save it into a file.

We come to the end of this course, in which we saw the basics of crawling a webpage using scrapy a python framework, if you are interested and after having a very good base on crawling basics you can see scrapy items containers used to collect the scraped data.

« https://docs.scrapy.org/en/latest/topics/items.html »