B0 - HUB

B-INN-000

Learn to scrap with scrappy in python

Introduction to web-scraping

Learn to scrap with s

- The totality of your source files, except all useless files (binary, temp files, obj files,...), must be included in your delivery.
- All the bonus files (including a potential specific Makefile) should be in a directory named *bonus*.
- Error messages have to be written on the error output, and the program should then exit with the 84 error code (0 if there is no error).

1. SETUP THE PROJECT

- create a folder for your project
- install scrapy : `pip install scrapy`
- create a scrapy project in your folder : `scrapy startproject animals`
- create a spider :
  - `cd animals`
  - `scrapy genspider paresseux https://www.monde-animal.fr/fiches-animaux/bradypus-variegatus/`
- run your spider and redirect it to see the output clearly: `scrapy crawl paresseux > tmp.txt`

2. NOW ITS YOUR TURN !

you already created a spider, now code it !

- get the name of the animal
- get the description of the animal
- get the size of the animal
- get the weight of the animal
  and more if you want

go to Scrappy doc and read the tutorial.
Don't hesitate to ask for help if you are stuck.
print the result in the terminal

3. MULTIPLE ANIMALS

now that you can scrap one animal, scrap all of them and their informations.
in fact you're not going to do page by page like you do for one animal, with scrappy you can acces to all the pages of the website.
create a new spider
reminder: `scrapy genspider allanimals https://www.monde-animal.fr/liste-animaux-a-z/`
Don't hesitate to ask for help if you are stuck.
print the result in the terminal

To go further

when you do a scrapper we have to store our data somewhere, you can store it in a json file, in a csv file, in a database, etc...

- collect more informations about the animals
- store your data in a csv file
- try to scrap others websites !