File Edit Selection View Go Run Terminal Help

imdb_spider.py - weathervsphenology - Visual Studio Code

EXPLORER

WEATHERVSPHENOLOGY
- data
- extractingData.ipynb
- proposal notes.txt
- README.md

README.md | __init__.py C:\...\spiders | imdb_spider.py 1, M | extractingData.ipynb | __init__.py C:\...\IMDB_scraper | settings.py

C: > Users > ianm2 > OneDrive > Documents > 1UCLA > Winter_2022 > PIC16B > github > IMDB_scraper > IMDB_scraper > IMDB_scraper > spiders > imdb_spider.py > ...

```python
# This worksheet was originally designed by [Erin George]
# (https://www.math.ucla.edu/~egeo/classes/spr21_pic16b-1/)
# (Department of Mathematics, UCLA), with subsequent
# revisions by John Zhang (Department of Mathematics, UCLA).

import scrapy

class ImdbSpider(scrapy.Spider):
    name = 'imdb_spider'

    start_urls = [
        'https://www.imdb.com/title/tt8714904/'
    ]

    def parse(self, response):#filmo-head-actor. head a

        topCst = response.css("div.ipc-title__wrapper a").attrib["href"]

        if topCst:
            topCst = response.urljoin(topCst)

            yield scrapy.Request(topCst, callback = self.parse_full_credits)

    def parse_full_credits(self, response):

        actLst = [a.attrib["href"] for a in response.css("td.primary_photo a")]

        for actor in actLst:
            actrPg = response.urljoin(actor)
            yield scrapy.Request(actrPg, callback = self.parse_actor_page)

    def parse_actor_page(self, response):

        actNam = response.css('title::text').get()[0:-7]

        filmID = response.css("::attr(id)").re('filmography')
        tvName = response.css("a::text").getall()

        albums = response.css('div.storytext li em::text').getall()
        songs = response.css('div.storytext li::text').re('Featured Song: "(.+)"')
        yield {"actor" : actNam, "movie_or_TV_name" : movie_or_TV_name}
```

OUTLINE

TIMELINE

main* | Python 3.8.12 ('PIC16B': conda) | 0 1 | Connect | Ln 1, Col 1 | Spaces: 4 | UTF-8 | LF | Python

9:49 PM
2/7/2022