使用这个爬虫的原因：我自己一开始是想爬取这个NASA的数据库，但是这个网站的数据较为复杂，用xpath较为复杂，需要操作的很多。更主要的是他的数据在后端，用xpath无法解决这个问题，所以我便放弃了用scrapy和xpath爬取这个网站信息的方式，转而在csdn上寻找可以直接爬取后端数据的方式，我发现了我们可以直接使用requests库来发送HTTP请求，并使用json方法解析响应数据。这样就可以解决我之前的获取不到后端数据的问题了！

def get\_page\_data(page\_number):

url = "https://osdr.nasa.gov/geode-py/ws/repo/search"

params = {

"term": "kidney",

"source": "cgene,alsda",

"type": "study",

"from": str(page\_number),

"size": "100",

"sort": "Study Public Release Date",

"order": "desc"

}

response = requests.get(url, params=params).json()

hits = response['hits']['hits']

for hit in hits:

Title = hit['\_source']['Study Title']

Identifier = hit['\_source']['Accession']

Organisms = hit['\_source']['organism']

Factors = hit['\_source']['Study Factor Name']

assayTypes = hit['\_source']['Study Assay Measurement Type']

releaseDate = int(hit['\_source']['Study Public Release Date'])

timeDate = datetime.fromtimestamp(releaseDate).strftime('%Y-%m-%d')

Description = hit['\_source']['Study Description']

study\_link = f"https://osdr.nasa.gov/bio/repo/data/studies/{Identifier}"

write\_to\_excel([Title, Identifier, Organisms, Factors, assayTypes, timeDate, Description, study\_link], 'nasa')

logger.info(Title, Identifier, Organisms, Factors, assayTypes, timeDate, Description, study\_link)

定义了爬取页面数据的函数 get\_page\_data，接受一个参数 page\_number，用于指定页码。

设置请求的URL和参数。

使用 requests.get 发送HTTP GET请求，并将响应解析为JSON格式。

提取响应中的数据（hits），并遍历每个数据项，将其各个字段提取出来。

使用 write\_to\_excel 函数将提取的数据写入Excel文件。

使用 logger.info 记录提取的数据。

def write\_to\_excel(val, title):

file\_path = os.path.join(os.getcwd(), f'{title}.xlsx')

if os.path.exists(file\_path) is False:

wb = Workbook()

wb.create\_sheet(title, 0)

sheet = wb[title]

sheet.append(['Title', 'Identifier', 'Organisms', 'Factors', 'assayTypes', 'timeDate', 'Description','Link'])

wb.save(file\_path)

wb = load\_workbook(file\_path)

sheet = wb[title]

sheet.\_current\_row = sheet.max\_row

sheet.append(val)

wb.save(file\_path)

定义了将数据写入Excel文件的函数 write\_to\_excel，接受两个参数 val（要写入的数据列表）和 title（文件名）。

构建文件路径，并检查文件是否存在。

如果文件不存在，则创建一个新的Excel工作簿，并在第一个工作表中添加表头。

加载现有的工作簿，并获取指定工作表。

将数据追加到工作表中，并保存工作簿。

最后自己也是通过这个方式获取到了NASA数据库的信息。