**II- Choisissez n'importe quelle page Produit sur le site de Books to Scrape.**

Écrivez un script Python qui visite cette page et en extrait les informations suivantes :

● product\_page\_url

● universal\_ product\_code (upc)

● title ● price\_including\_tax

● price\_excluding\_tax

● number\_available

● product\_description

● category

● review\_rating

● image\_url

Écrivez les données dans un fichier CSV qui utilise les champs ci-dessus comme en-têtes de colonnes.

#-1-Extrait les informations d'une page produit

import requests

from bs4 import BeautifulSoup

import csv

# Fonction pour extraire les informations de la page produit

def extract\_product\_info(url):

    response = requests.get(url)

    if response.status\_code != 200:

        return None

    soup = BeautifulSoup(response.text, 'html.parser')

    # Extraire les informations nécessaires

    product\_page\_url = url

    title = soup.h1.text

    upc = soup.select('th')[0].find\_next('td').text

    price\_including\_tax = soup.select('th')[3].find\_next('td').text.replace('Â','')

    price\_excluding\_tax = soup.select('th')[2].find\_next('td').text.replace('Â','')

    number\_available = soup.select('th')[5].find\_next('td').text

    product\_description = soup.find('meta', attrs={'name': 'description'})['content']

    category = soup.select('a')[3].text

    review\_rating = soup.select('p')[2]['class'][1]

    image\_url = soup.select('img')[0]['src']

    return {

        "product\_page\_url": product\_page\_url,

        "upc": upc,

        "title": title,

        "price\_including\_tax": price\_including\_tax,

        "price\_excluding\_tax": price\_excluding\_tax,

        "number\_available": number\_available,

        "product\_description": product\_description,

        "category": category,

        "review\_rating": review\_rating,

        "image\_url": image\_url

    }

# URL de la page produit que vous souhaitez extraire

product\_url = "https://books.toscrape.com/catalogue/a-light-in-the-attic\_1000/index.html"

# Appeler la fonction pour extraire les informations

product\_info = extract\_product\_info(product\_url)

# Vérifier si les informations ont été extraites avec succès

if product\_info:

    # Écrire les données dans un fichier CSV

    with open('unepage30.csv', 'w', newline='', encoding='utf-8') as csv\_file:

        fieldnames = product\_info.keys()

        writer = csv.DictWriter(csv\_file, fieldnames=fieldnames)

        writer.writeheader()

        writer.writerow(product\_info)

    print("Données extraites avec succès et enregistrées dans unepage30.csv")

else:

    print("Impossible d'extraire les données de la page produit.")