import requests

import csv

from bs4 import BeautifulSoup

# URL of the product listing page

base\_url = "https://www.amazon.in/s?k=bags&crid=2M096C61O4MLT&qid=1653308124&sprefix=ba%2Caps%2C283&ref=sr\_pg\_"

# Number of pages to scrape

num\_pages = 20

# Initialize an empty list to store the scraped data

products = []

# Scrape data from each page

for page in range(1, num\_pages + 1):

url = base\_url + str(page)

# Send a GET request to the URL

response = requests.get(url)

# Create a BeautifulSoup object to parse the HTML content

soup = BeautifulSoup(response.content, "html.parser")

# Find the elements containing the product information

product\_elements = soup.find\_all("div", class\_="sg-col-inner")

# Iterate over the product elements and extract relevant information

for product in product\_elements:

# Extract the product URL

product\_url = product.find("a", class\_="a-link-normal s-no-outline").get("href")

# Extract the product name

product\_name = product.find("span", class\_="a-size-medium a-color-base a-text-normal").text

# Extract the product price

product\_price = product.find("span", class\_="a-offscreen").text

# Extract the product rating

product\_rating = product.find("span", class\_="a-icon-alt").text

# Extract the number of reviews

product\_reviews = product.find("span", class\_="a-size-base").text

# Append the product data to the list

products.append([product\_url, product\_name, product\_price, product\_rating, product\_reviews])

# Scrape additional information from each product URL

for product in products:

url = product[0]

# Send a GET request to the product URL

response = requests.get(url)

# Create a BeautifulSoup object to parse the HTML content

soup = BeautifulSoup(response.content, "html.parser")

# Extract additional information

product\_description = soup.find("div", id="productDescription").text.strip()

product\_asin = soup.find("th", text="ASIN").find\_next\_sibling("td").text.strip()

product\_manufacturer = soup.find("th", text="Manufacturer").find\_next\_sibling("td").text.strip()

# Update the product data with additional information

product.extend([product\_description, product\_asin, product\_manufacturer])

# Export the data to a CSV file

csv\_filename = "product\_data.csv"

with open(csv\_filename, "w", newline="", encoding="utf-8") as csvfile:

writer = csv.writer(csvfile)

writer.writerow(["Product URL", "Product Name", "Product Price", "Rating", "Number of Reviews", "Description", "ASIN", "Manufacturer"])

writer.writerows(products)

print("Scraping complete. Data exported to", csv\_filename)