**Create a program that extracts product information, such as names, prices, and ratings, from an online e- commerce website and stores the data in a structured format like a CSV file**

import requests

from bs4 import BeautifulSoup

import pandas as pd

# Function to extract product information from the website

def extract\_product\_info(url):

headers = {

'User-Agent': 'Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/58.0.3029.110 Safari/537.3'}

# Send a GET request to the webpage

response = requests.get(url, headers=headers)

if response.status\_code != 200:

print(f"Failed to retrieve webpage. Status code: {response.status\_code}")

return None

# Parse the HTML content of the page with BeautifulSoup

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

# Extract product information (this example is for a generic structure)

products = []

# You need to adjust these tags and classes according to the website structure you are scraping

for product in soup.find\_all('div', class\_='product-item'): # Replace 'product-item' with actual class

name = product.find('h2', class\_='product-title').get\_text(strip=True) # Replace with actual class

price = product.find('span', class\_='product-price').get\_text(strip=True) # Replace with actual class

rating = product.find('div', class\_='product-rating').get\_text(strip=True) # Replace with actual class

products.append({

'Name': name,

'Price': price,

'Rating': rating

})

return products

# Function to save extracted data to a CSV file

def save\_to\_csv(products, filename='products.csv'):

df = pd.DataFrame(products)

df.to\_csv(filename, index=False)

print(f"Data saved to {filename}")

def main():

# Replace with the URL of the e-commerce website and adjust selectors accordingly

url = 'https://example-ecommerce-site.com/products'

products = extract\_product\_info(url)

if products:

save\_to\_csv(products)

if \_\_name\_\_ == "\_\_main\_\_":

main()



