

```
In [ ]: from bs4 import BeautifulSoup
import requests
import pandas as pd
import numpy as np
```

```
In [ ]: # Function to extract Product Title
def get_title(soup):

    try:
        # Outer Tag Object
        title = soup.find("span", attrs={"id": 'productTitle'})

        # Inner NavigatableString Object
        title_value = title.text

        # Title as a string value
        title_string = title_value.strip()

    except AttributeError:
        title_string = ""

    return title_string
```

```
In [ ]: # Function to extract Product Price
def get_price(soup):

    try:
        price = soup.find("span", attrs={'id': 'priceblock_ourprice'}).string.strip()

    except AttributeError:

        try:
            # If there is some deal price
            price = soup.find("span", attrs={'id': 'priceblock_dealprice'}).string.strip()

        except:
            price = ""

    return price
```

```
In [ ]: # Function to extract Product Rating
def get_rating(soup):

    try:
        rating = soup.find("i", attrs={'class': 'a-icon a-icon-star a-star-4-5'}).string.strip()

    except AttributeError:
        try:
            rating = soup.find("span", attrs={'class': 'a-icon-alt'}).string.strip()
        except:
            rating = ""

    return rating
```

```
In [ ]: # Function to extract Number of User Reviews
def get_review_count(soup):

    try:
        review_count = soup.find("span", attrs={'id': 'acrCustomerReviewText'}).string.strip()

    except AttributeError:
        review_count = ""

    return review_count
```

```
In [ ]: # Function to extract Availability Status
def get_availability(soup):

    try:
        available = soup.find("div", attrs={'id': 'availability'})
        available = available.find("span").string.strip()

    except AttributeError:
        available = "Not Available"

    return available
```

```
In [ ]: if __name__ == '__main__':

    # add your user agent
    HEADERS = ({'User-Agent': '', 'Accept-Language': 'en-US, en;q=0.5'})

    # The webpage URL
    URL = "https://www.amazon.com/s?k=playstation+4&ref=nb_sb_noss_2"

    # HTTP Request
    webpage = requests.get(URL, headers=HEADERS)

    # Soup Object containing all data
    soup = BeautifulSoup(webpage.content, "html.parser")

    # Fetch links as List of Tag Objects
    links = soup.find_all("a", attrs={'class': 'a-link-normal s-no-outline'})

    # Store the links
    links_list = []

    # Loop for extracting links from Tag Objects
    for link in links:
        links_list.append(link.get('href'))

    d = {"title": [], "price": [], "rating": [], "reviews": [], "availability": []}
```

```
In [ ]: # Loop for extracting links from Tag Objects
for link in links:
    links_list.append(link.get('href'))

d = {"title": [], "price": [], "rating": [], "reviews": [], "availability": []}
```

```
In [ ]: # Loop for extracting product details from each link
for link in links_list:
    new_webpage = requests.get("https://www.amazon.com" + link, headers=HEADERS)

    new_soup = BeautifulSoup(new_webpage.content, "html.parser")

    # Function calls to display all necessary product information
    d['title'].append(get_title(new_soup))
    d['price'].append(get_price(new_soup))
    d['rating'].append(get_rating(new_soup))
    d['reviews'].append(get_review_count(new_soup))
    d['availability'].append(get_availability(new_soup))
```

```
In [ ]: amazon_df = pd.DataFrame.from_dict(d)
amazon_df = amazon_df.dropna(subset=['title'])
amazon_df.to_csv("amazon_data.csv", header=True, index=False)
```

```
In [ ]: amazon_df
```

```
Out[ ]:      title price rating reviews availability
0  Sony Playstation PS4 1TB Black Console  4.6 out of 5 stars  1,399 ratings  In Stock
1  Flagship Newest Play Station 4 1TB HDD Only on...  4.5 out of 5 stars  199 ratings  Not Available
2  Newest Sony Playstation 4 Slim 1TB SSD Console...  4.3 out of 5 stars  313 ratings  Not Available
3  PlayStation®5 Digital Edition (slim)  4.7 out of 5 stars  3,504 ratings  Only 5 left in stock - order soon.
4  PlayStation 4 Slim 500GB Console - Uncharted 4...  4.8 out of 5 stars  6,453 ratings  Only 1 left in stock - order soon.
...  ...  ...  ...  ...  ...
75  Playstation SONY 4, 500GB Slim System [CUH-221...  4.5 out of 5 stars  409 ratings  Only 5 left in stock - order soon.
76  PlayStation®4 Console – Call of Duty® Modern W...  4.4 out of 5 stars  48 ratings  In Stock
77  Replacement Astro A40 A10 A30 A50 Gaming Heads...  3.9 out of 5 stars  12 ratings
78  Rolling Universal Gaming Backpack for Xbox One...  3.8 out of 5 stars  51 ratings  In Stock
79  PS4 Stand Cooling Fan Station for Playstation ...  4.3 out of 5 stars  234 ratings  In Stock
```

80 rows × 5 columns

```
In [ ]:
```