# 2306AML112 - Renduchintala Navya ML Assignment

https://www.amazon.com/dp/B0BYV51VBN/ref=sspa_dk_detail_3?pd_rd_i=B0BYV51VBN&pd_rd_w=V7rwe&content-id=amzn1.sym.386c274b-4bfe-4421-9052-a1a56db557ab&pf_rd_p=386c274b-4bfe-4421-9052-a1a56db557ab&pf_rd_r=C5Y7NS61W32HJP49EEYF&pd_rd_wg=wqkQC&pd_rd_r=392669cb-7640-41a9-9b66-2b9047c51e93&s=electronics&sp_csd=d2lkZ2V0TmFtZT1zcF9kZXRhaWxfdGhlbWF0aWM&smid=A257BP83UN2F6Y&th=1

Web Scraping on Amazon Products using above link

https://www.digitalocean.com/community/tutorials/scrape-amazon-product-information-beautiful-soup

this is reference

In [1]:
```
# pip install bs4 - installed
```

In [2]:
```
HEADERS = ({'User-Agent':
            'Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/44.0.2403.157 Safari
            'Accept-Language': 'en-US, en;q=0.5'})
```

In [3]:
```
import requests

page = requests.get("https://www.amazon.com/dp/B0BYV51VBN/ref=sspa_dk_detail_3?pd_rd_i=B0BYV51VBN&pd_rd_w=V7rwe

page
```

Out[3]:
```
<Response [200]>
```

In [4]:
```
page.status_code
```

Out[4]:
```
200
```

In [5]:
```
# page.content
```

In [6]:
```
from bs4 import BeautifulSoup
soup = BeautifulSoup(page.content,"html.parser")
```

In [7]:
```
# print(soup.prettify())
```

In [8]:
```
soup = BeautifulSoup(page.content, 'html.parser')
# soup.find_all('span')
```

In [9]:
```
soup = BeautifulSoup(page.text, 'lxml')
```

In [10]:
```
# Outer Tag Object
title = soup.find("span", attrs={"id":'productTitle'})
print(title)
```

```
<span class="a-size-large product-title-word-break" id="productTitle">        3 in 1 Magnetic Wireless Travel F
oldable Charging Station with Light for Apple Multiple Devices, Mag-Safe Charger Portable Stand, Compatible wit
h iPhone 14/13/12 Series | Apple Watch | Airpods       </span>
```

In [11]:
```
# Inner NavigableString Object
title_value = title.string
print(title_value)
```

```
        3 in 1 Magnetic Wireless Travel Foldable Charging Station with Light for Apple Multiple Devices, Mag-Sa
fe Charger Portable Stand, Compatible with iPhone 14/13/12 Series | Apple Watch | Airpods
```

In [12]:
```
# Title as a string value
title_string = title_value.strip()
print(title_string)
```

```
3 in 1 Magnetic Wireless Travel Foldable Charging Station with Light for Apple Multiple Devices, Mag-Safe Charg
er Portable Stand, Compatible with iPhone 14/13/12 Series | Apple Watch | Airpods
```

In [13]:
```
# Printing types of values for efficient understanding
print(type(title))
print(type(title_value))
print(type(title_string))
print()

# Printing Product Title
print("Product Title = ", title_string)
```

```
<class 'bs4.element.Tag'>
<class 'bs4.element.NavigableString'>
<class 'str'>

Product Title =  3 in 1 Magnetic Wireless Travel Foldable Charging Station with Light for Apple Multiple Device
s, Mag-Safe Charger Portable Stand, Compatible with iPhone 14/13/12 Series | Apple Watch | Airpods
```

In [14]:
```
price = soup.find("span", attrs={'id':'price_inside_buybox'})
print(price)
```

```
<span class="a-size-medium a-color-price" id="price_inside_buybox"> $38.99 </span>
```

In [15]:
```
rating = soup.find("i", attrs={'class':'a-icon a-icon-star a-star-4-5'})
print(rating)
```

```
<i class="a-icon a-icon-star a-star-4-5"></i>
```

In [16]:
```
    from bs4 import BeautifulSoup
    import requests

    # Function to extract Product Title
    def get_title(soup):

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

            # Inner NavigableString Object
            title_value = title.string

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

            # # Printing types of values for efficient understanding
            # print(type(title))
            # print(type(title_value))
            # print(type(title_string))
            # print()

        except AttributeError:
            title_string = ""

        return title_string

    # Function to extract Product Price
    def get_price(soup):

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

        except AttributeError:
            price = ""

        return price

    # 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

    # 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

    # Function to extract Availability Status
    def get_availability(soup):
        try:
            available = soup.find("div", attrs={'id':'twisterAvailability'})
            available = available.find("span").string.strip()

        except AttributeError:
            available = ""
```

```
        return available

    if __name__ == '__main__':

        # Headers for request
        HEADERS = ({'User-Agent':
                    'Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/44.0.2403.15
                    'Accept-Language': 'en-US, en;q=0.5'})

        # The webpage URL
        URL = "https://www.amazon.com/dp/B0BYV51VBN/ref=sspa_dk_detail_3?pd_rd_i=B0BYV51VBN&pd_rd_w=V7rwe&conte

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

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

        # Function calls to display all necessary product information
        print("Product Title =", get_title(soup))
        print("Product Price =", get_price(soup))
        print("Product Rating =", get_rating(soup))
        print("Number of Product Reviews =", get_review_count(soup))
```

```
Product Title = 3 in 1 Magnetic Wireless Travel Foldable Charging Station with Light for Apple Multiple Devices
, Mag-Safe Charger Portable Stand, Compatible with iPhone 14/13/12 Series | Apple Watch | Airpods
Product Price = $38.99
Product Rating = 4.7 out of 5 stars
Number of Product Reviews = 35 ratings
```