

ACTIVITY-10 : WEBSITE SCRAPING

```
In [31]: from bs4 import BeautifulSoup
import requests
```

```
In [32]: # Function to extract Product Title
def get_title(soup):
    try:
        title = soup.find("span", attrs={"id": "productTitle"})
        title_string = title.get_text(strip=True)
    except AttributeError:
        title_string = ""
    return title_string
```

```
# Function to extract Product Rating
def get_rating(soup):
    try:
        rating = soup.find("span", attrs={"class": "a-icon-alt"})
        if rating is not None:
            rating_string = rating.get_text(strip=True)
        else:
            rating_string = ""
    except AttributeError:
        rating_string = ""
    return rating_string
```

```
In [33]: # Function to extract Product Price
def get_price(soup):
    try:
        price = soup.find("span", attrs={"class": "a-offscreen"})
        if price is not None:
            price_string = price.get_text(strip=True)
        else:
            price_string = ""
    except AttributeError:
        price_string = ""
    return price_string
```

```
# Function to extract Number of User Reviews
def get_review_count(soup):
    try:
        review_count = soup.find("span", attrs={"id": "acrCustomerReviewText"})
        if review_count is not None:
            review_count_string = review_count.get_text(strip=True)
        else:
            review_count_string = ""
    except AttributeError:
        review_count_string = ""
    return review_count_string
```

```
In [36]: # Function to extract Availability Status
def get_availability(soup):
    try:
        availability = soup.find("div", attrs={"id": "availability"})
        if availability is not None:
            availability_string = availability.get_text(strip=True)
        else:
            availability_string = ""
    except AttributeError:
        availability_string = ""
    return availability_string
```

```
In [37]: if __name__ == "__main__":
# Headers for request
headers = {
    "User-Agent": "Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/44.0.2403.157 Safari/537.36",
    "Accept-Language": "en-US, en;q=0.5"
}

# The webpage URL
url = "https://www.amazon.in/Airdopes-141-Bluetooth-Wireless-Playtime/dp/B09N3ZLB3T?pd_rd_w=SaZb7&content-id=amzn1.sym.9477bc"
# HTTP Request
webpage = requests.get(url, headers=headers)

# Soup Object containing all data
soup = BeautifulSoup(webpage.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))
print("Availability =", get_availability(soup))
```

```
Product Title = boAt Airdopes 141 Bluetooth Truly Wireless in Ear Headphones with 42H Playtime, Beast Mode(Low Latency Upto 80ms) for Gaming, ENx Tech, IWP, IPX4 Water Resistance, Smooth Touch Controls(Pure White)
Product Price = ₹1,399.00
Product Rating = 3.9 out of 5 stars
Number of Product Reviews = 184,660 ratings
Availability = In stock
```