

IT VEDANT INSTITUTE, THANE.
MASTER IN DATA SCIENCE & ANALYTICS WITH
ARTIFICIAL INTELLIGENCE



PROJECT FOR PYTHON MODULE
ON
SNAPDEAL WEBSITE SCRAPPING
BY

Ayush Dnyaneshwar Gunjal

UNDER THE GUIDENCE OF
Mr. Sameer Warsolkar

Academic year: 2024-2025

Acknowledgment

I would like to express my deepest gratitude to Mr. Sameer Warsolkar for his invaluable guidance and support throughout the completion of this project, titled "Website Scraping on Snapdeal Website". His expertise and insights have been crucial to my understanding of Python and data science concepts. I would also like to extend my sincere thanks to ITvedant for providing the platform and resources necessary for my learning and growth in this domain. This project marks a significant milestone in my journey toward becoming a proficient data scientist.

Name: Ayush Dnyaneshwar Gunjal

Module: Python for Data Science Module

Institute: ITvedant

Abstraction

This project involves building a web scraping tool using Python to extract key information from the Snapdeal e-commerce website. The primary goal is to collect data such as product names, prices, discounts, and off-prices. By employing Python libraries like BeautifulSoup and requests, the project efficiently parses HTML content to retrieve structured data from product listings. The extracted data can then be further analysed or used for applications like price monitoring, trend analysis, or competitive market research.

The project demonstrates key concepts of web scraping, data extraction, and cleaning, along with handling challenges like dynamic content and pagination. The tool was developed as part of the Python for Data Science module under the guidance of Mr. Sameer Warsolkar at ITvedant, emphasizing practical applications of Python in data science workflows

Keywords:

Web Scraping
Python
BeautifulSoup
Data Extraction
E-commerce

TABLE OF CONTENTS

Sr. no.	Title	Page no.
1.	Acknowledgment	2
2.	Abstraction	3
3.	Introduction	5
4.	Snapdeal website description	6
5.	Methodology	7
6.	Code explanation	8
7.	Result and analysis	14
8.	Conclusion	15

INTRODUCTION

In the age of digital commerce, e-commerce websites like Snapdeal hold vast amounts of product data that can offer critical insights into market trends, pricing strategies, and consumer preferences. The ability to efficiently extract and analyse this data is crucial for businesses, researchers, and analysts alike. This project, titled "Website Scraping on Snapdeal Website," leverages the power of Python to automate the process of extracting essential information such as product names, prices, discounts, and off-price values from the Snapdeal platform.

By utilizing popular Python libraries such as BeautifulSoup and requests, this project demonstrates how data can be programmatically retrieved from web pages and structured for further analysis. The scraped data has potential applications in price comparison, market research, and understanding consumer behavior.

This project was undertaken as part of the Python for Data Science module at ITvedant, under the mentorship of Mr. Sameer Warsolkar. It not only enhances proficiency in web scraping techniques but also provides a hands-on understanding of the real-world applications of Python in data science.

SNAPDEAL WEBSITE DESCRIPTION

Snapdeal is one of India's leading online e-commerce platforms, offering a vast range of products across multiple categories such as electronics, fashion, home goods, health, and beauty. Launched in 2010, Snapdeal has grown rapidly and has become a prominent marketplace, connecting buyers and sellers from all over India. The platform provides users with access to millions of products, including mobile phones, home appliances, clothing, accessories, and more, catering to diverse consumer needs.

Snapdeal is known for offering competitive pricing and frequent discounts, making it a popular choice for price-conscious customers. The website features a user-friendly interface, allowing customers to easily search for products, compare prices, read reviews, and make secure purchases. Snapdeal also provides various payment options, including cash on delivery, credit/debit cards, and digital wallets, ensuring convenience for its users.

Snapdeal's mission is to provide value-driven products to the Indian market and to offer a seamless shopping experience for customers across urban and rural areas alike.

Outline:

From this Website we are going to grab the following details.

- Product name
- Price
- Discount percentage
- Off price after description

METHODOLOGY

Tools and Technologies Used:

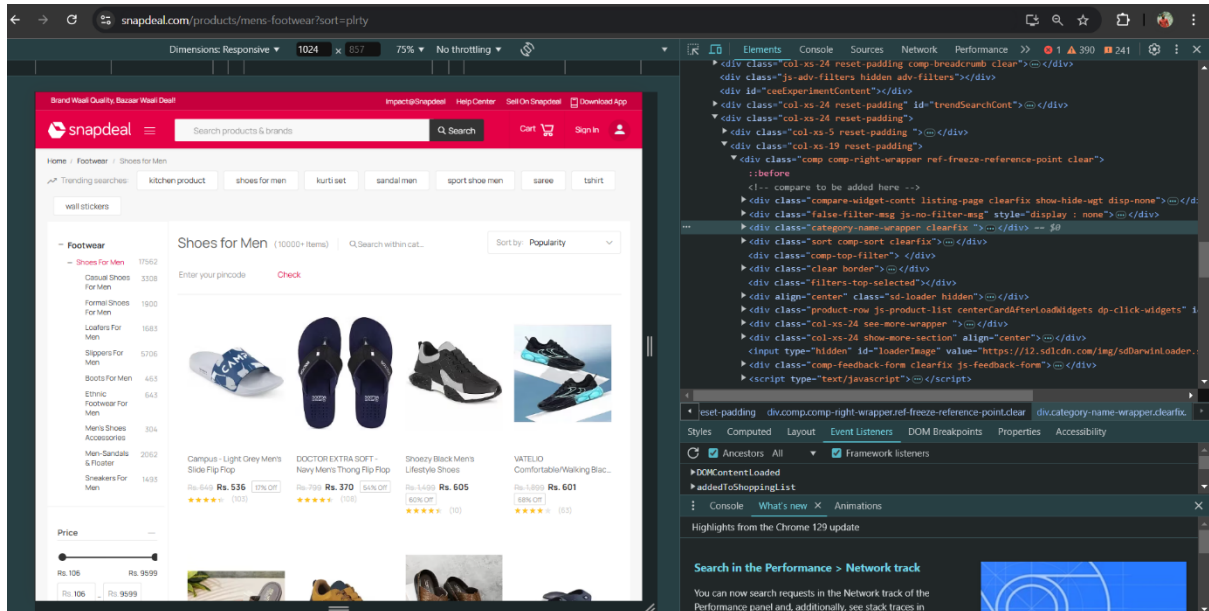
- Python: Programming language
- BeautifulSoup: Python library used for web scraping purposes to extract data from HTML and XML files
- Requests: Requests is a Python library used for sending HTTP requests to websites and web services.
- Pandas: it provides data structures and functions needed to work with structured data, particularly data that is stored in tables or time series.

Steps Involved in Web Scraping:

- Import the libraries need to access the website
- Accessing the website.
- Inspecting the HTML structure of Snapdeal's product pages.
- Extracting product details (name, price, discount).
- Storing the extracted data in a structured format (e.g., CSV or DataFrame).
- Extract the data for the analysis.

CODE EXPLANATION

Step 1: Choose the Website and Webpage URL Inspect the Website



Step 2: installing or importing the essential libraries for scrapping process.

- requests
- BeautifulSoup
- pandas

```
import requests
from bs4 import BeautifulSoup
import pandas as pd
```


Step 3: using requests library, requests the website for providing data.

```
page = requests.get("https://www.snapdeal.com/products/mens-footwear?sort=plrty")
page

<Response [200]>
```

Step 4: using BeautifulSoup library import the HTML content of the website

```
soup = BeautifulSoup(page.content, "html.parser")
soup

<!DOCTYPE html>

<!--[if IE 8]><html lang="en" class="ie ie8 lt-ie9 lt-ie10"> <![endif]-->
<!--[if IE 9]><html lang="en" class="ie ie9 lt-ie10"> <![endif]-->
<!--[if IE]><html lang="en" class="ie"> <![endif]-->
<!--[if gt IE 9]><!--><html lang="en"><!--><![endif]-->
<head prefix="og: https://ogp.me/ns# fb: https://ogp.me/ns/fb# snapdeallog: https://ogp.me/ns/fb/snapdeallog#"
<link href="https://i1.sd1cdn.com" rel="dns-prefetch"/>
<link href="https://i2.sd1cdn.com" rel="dns-prefetch"/>
<link href="https://i3.sd1cdn.com" rel="dns-prefetch"/>
<link href="https://i4.sd1cdn.com" rel="dns-prefetch"/>
<link href="https://n1.sd1cdn.com" rel="dns-prefetch"/>
<link href="https://n2.sd1cdn.com" rel="dns-prefetch"/>
<link href="https://n3.sd1cdn.com" rel="dns-prefetch"/>
<link href="https://n4.sd1cdn.com" rel="dns-prefetch"/>
<link href="https://sa.snapdeal.com" rel="dns-prefetch"/>
<link href="https://search-suggester.snapdeal.com" rel="dns-prefetch"/>
<link href="https://mobileapi.snapdeal.com" rel="dns-prefetch"/>
```

Step 5: accessing the product name.

```
period = soup.find(class_="product-title")
period

<p class="product-title" title="lyvi - Black Rubber Daily Slipper">lyvi - Black Rubber Daily Slipper</p>
```

```
period_tags = soup.select(".product-title")
period_tags

[<p class="product-title" title="lyvi - Black Rubber Daily Slipper">lyvi - Black Rubber Daily Slipper</p>,
<p class="product-title" title="atom Spring Edge Alpha 1 Beige Men's Lifestyle Shoes">atom Spring Edge Alpha 1 Beige Men's Lifestyle Shoes</p>,
<p class="product-title" title="Xtoon White Men's Lifestyle Shoes">Xtoon White Men's Lifestyle Shoes</p>,
<p class="product-title" title="Crazy Bunny - Red Men's Thong Flip Flop">Crazy Bunny - Red Men's Thong Flip Flop</p>,
<p class="product-title" title="Fashion Victim Lifestyle Tan Casual Shoes">Fashion Victim Lifestyle Tan Casual Shoes</p>,
<p class="product-title" title="DOCTOR EXTRA SOFT - Dark Grey Men's Daily Slipper">DOCTOR EXTRA SOFT - Dark Grey Men's Daily Slipper</p>,
<p class="product-title" title="Figor Black Lifestyle Shoes">Figor Black Lifestyle Shoes</p>,
<p class="product-title" title="Campus AGR-003 - Black Men's Slip-on Shoes">Campus AGR-003 - Black Men's Slip-on Shoes</p>,
<p class="product-title" title="Fashion Victim - White Men's Sneakers">Fashion Victim - White Men's Sneakers</p>,
<p class="product-title" title="Campus - Mustard Men's Slide Flip Flop">Campus - Mustard Men's Slide Flip Flop</p>,
<p class="product-title" title="Secritas Black Men's Slide Flip Flop">Secritas Black Men's Slide Flip Flop</p>,
<p class="product-title" title="Aerowalk - Tan Men's Leather Slipper">Aerowalk - Tan Men's Leather Slipper</p>,
<p class="product-title" title="Xtoon Grey Men's Outdoor Shoes">Xtoon Grey Men's Outdoor Shoes</p>,
<p class="product-title" title="lyvi - Grey Men's Daily Slipper">lyvi - Grey Men's Daily Slipper</p>,
<p class="product-title" title="Aadi - White Men's Sneakers">Aadi - White Men's Sneakers</p>,
<p class="product-title" title="Neemans Black Men's Daily Slipper">Neemans Black Men's Daily Slipper</p>,
<p class="product-title" title="UniStar Outdoor Olive Casual Shoes">UniStar Outdoor Olive Casual Shoes</p>,
<p class="product-title" title="Campus Blue Men's Daily Slipper (Pair of 1)">Campus Blue Men's Daily Slipper (Pair of 1)</p>,
<p class="product-title" title="Big Fox Blue Flip Flops">Big Fox Blue Flip Flops</p>,
<p class="product-title" title="Aadi Beige Outdoor Shoes">Aadi Beige Outdoor Shoes</p>]
```

Step 6: using regular expressions access the name of the product

```
period_tags = soup.select(".product-title")
period_tags
import re
name_list=[]
for i in range(0,len(period_tags)):
    text = period_tags[i].get_text()
    text_1=re.split("\n",text)
    a=text_1[0]
    name_list.append(a)
name_list

['lyvi - Black Rubber Daily Slipper',
'atom Spring Edge Alpha 1 Beige Men's Lifestyle Shoes',
'Xtoon White Men's Lifestyle Shoes',
'Crazy Bunny - Red Men's Thong Flip Flop',
'Fashion Victim Lifestyle Tan Casual Shoes',
'DOCTOR EXTRA SOFT - Dark Grey Men's Daily Slipper',
'Figor Black Lifestyle Shoes',
'Campus AGR-003 - Black Men's Slip-on Shoes',
'Fashion Victim - White Men's Sneakers',
'Campus - Mustard Men's Slide Flip Flop',
'Secritas Black Men's Slide Flip Flop',
'Aerowalk - Tan Men's Leather Slipper',
'Xtoon Grey Men's Outdoor Shoes',
'lyvi - Grey Men's Daily Slipper',
'Aadi - White Men's Sneakers',
'Neemans Black Men's Daily Slipper',
'UniStar Outdoor Olive Casual Shoes',
'Campus Blue Men's Daily Slipper (Pair of 1)',
'Big Fox Blue Flip Flops',
'Aadi Beige Outdoor Shoes']
```

Step 7: accessing the price of the product

```
period_tags = soup.select(".lfloat.product-price")
period_tags

import re
price_list=[]
for i in range(0,len(period_tags)):
    text = period_tags[i].get_text()
    text_1=re.findall("[0-9]+",text)
    a=text_1[0]
    price_list.append(a)

price_list

for p in price_list:
    print("Rs.",p)
```

```
Rs. 157
Rs. 1
Rs. 496
Rs. 183
Rs. 628
Rs. 399
Rs. 575
Rs. 748
Rs. 596
Rs. 649
Rs. 228
Rs. 384
Rs. 675
Rs. 161
Rs. 340
Rs. 352
Rs. 1
Rs. 748
Rs. 747
Rs. 604
```

Step 8: accessing the discount of the product.

```
period_tags = soup.select(".product-discount")
period_tags

import re
discount_list=[]
for i in range(-1,len(period_tags)):
    text = period_tags[i].get_text()
    text_1=re.findall("[0-9]+",text)
    a=text_1[0]
    discount_list.append(a)

discount_list

for D in discount_list:
    print(D,"% Off")
```

```
40 % Off
84 % Off
60 % Off
67 % Off
63 % Off
37 % Off
60 % Off
70 % Off
47 % Off
83 % Off
77 % Off
10 % Off
55 % Off
84 % Off
66 % Off
50 % Off
56 % Off
6 % Off
70 % Off
40 % Off
```

Step 9: accessing the off price of the product.

```
period_tags = soup.select(".lfloat.product-desc-price.strike ")
period_tags

import re
Off_price_list=[]
for i in range(0,len(period_tags)):
    text = period_tags[i].get_text()
    text_1=re.findall("[A-Z a-z]+[0-9]+",text)
    a=text_1[0]
    Off_price_list.append(a)

Off_price_list

for O in Off_price_list:
    print(O)
```

```
999
3
1
499
999
999
1
1
3
649
999
429
1
999
999
699
2
799
2
999
```

Step 10: make a dictionary of the list you have get by using above statement.

```
final_one = {"product name":name_list,"discount percentage":discount_list, "price":price_list,"Off price":Off_price_list}
final_one
```

```
{'product name': ['lyvi - Black Rubber Daily Slipper',
'atom Spring Edge Alpha 1 Beige Men's Lifestyle Shoes',
'Xtoon White Men's Lifestyle Shoes',
'Crazy Bunny - Red Men's Thong Flip Flop',
'Fashion Victim Lifestyle Tan Casual Shoes',
'DOCTOR EXTRA SOFT - Dark Grey Men's Daily Slipper',
'Figor Black Lifestyle Shoes',
'Campus AGR-003 - Black Men's Slip-on Shoes',
'Fashion Victim - White Men's Sneakers',
'Campus - Mustard Men's Slide Flip Flop',
'Secritas Black Men's Slide Flip Flop',
'Aerowalk - Tan Men's Leather Slipper',
'Xtoon Grey Men's Outdoor Shoes',
'lyvi - Grey Men's Daily Slipper',
'Aadi - White Men's Sneakers',
'Neemans Black Men's Daily Slipper',
'UniStar Outdoor Olive Casual Shoes',
'Campus Blue Men's Daily Slipper (Pair of 1)',
'Big Fox Blue Flip Flops',
'Aadi Beige Outdoor Shoes'],
'discount percentage': ['40',
'84',
'60',
'67',
'63',
'37',
'60',
'70',
'47',
'83',
'77',
'10',
'55',
'84',
'66',
'50',
'56',
'6',
'70',
'40'],
```

```
'157',
'1',
'496',
'183',
'628',
'399',
'575',
'748',
'596',
'649',
'228',
'384',
'675',
'161',
'340',
'352',
'1',
'748',
'747',
'604'],
'Off price': [' 999',
' 3',
' 1',
' 499',
' 999',
' 999',
' 1',
' 1',
' 3',
' 649',
' 999',
' 429',
' 1',
' 999',
' 999',
' 699',
' 2',
' 799',
' 2',
' 999']}]
```

Step 11: using pandas library make a data frame of the dictionary you have created.

```
dict = pd.DataFrame(final_one)
dict
```

	product name	discount percentage	price	Off price
0	lyvi - Black Rubber Daily Slipper	40	157	999
1	atom Spring Edge Alpha 1 Beige Men's Lifestyle...	84	1	3
2	Xtoon White Men's Lifestyle Shoes	60	496	1
3	Crazy Bunny - Red Men's Thong Flip Flop	67	183	499
4	Fashion Victim Lifestyle Tan Casual Shoes	63	628	999
5	DOCTOR EXTRA SOFT - Dark Grey Men's Daily Slipper	37	399	999
6	Figor Black Lifestyle Shoes	60	575	1
7	Campus AGR-003 - Black Men's Slip-on Shoes	70	748	1
8	Fashion Victim - White Men's Sneakers	47	596	3
9	Campus - Mustard Men's Slide Flip Flop	83	649	649
10	Secritas Black Men's Slide Flip Flop	77	228	999
11	Aerowalk - Tan Men's Leather Slipper	10	384	429
12	Xtoon Grey Men's Outdoor Shoes	55	675	1
13	lyvi - Grey Men's Daily Slipper	84	161	999
14	Aadi - White Men's Sneakers	66	340	999
15	Neemans Black Men's Daily Slipper	50	352	699
16	UniStar Outdoor Olive Casual Shoes	56	1	2
17	Campus Blue Men's Daily Slipper (Pair of 1)	6	748	799
18	Big Fox Blue Flip Flops	70	747	2
19	Aadi Beige Outdoor Shoes	40	604	999

Step 12: save the file to csv mode in your computer and read that file.

```
dict.to_csv("snapdeal online shopping.csv")

read = pd.read_csv("snapdeal online shopping.csv")
read
```

Unnamed: 0	product name	discount percentage	price	Off price
0	lyvi - Black Rubber Daily Slipper	40	157	999
1	atom Spring Edge Alpha 1 Beige Men's Lifestyle...	84	1	3
2	Xtoon White Men's Lifestyle Shoes	60	496	1
3	Crazy Bunny - Red Men's Thong Flip Flop	67	183	499
4	Fashion Victim Lifestyle Tan Casual Shoes	63	628	999
5	DOCTOR EXTRA SOFT - Dark Grey Men's Daily Slipper	37	399	999
6	Figor Black Lifestyle Shoes	60	575	1
7	Campus AGR-003 - Black Men's Slip-on Shoes	70	748	1
8	Fashion Victim - White Men's Sneakers	47	596	3
9	Campus - Mustard Men's Slide Flip Flop	83	649	649
10	Secritas Black Men's Slide Flip Flop	77	228	999
11	Aerowalk - Tan Men's Leather Slipper	10	384	429
12	Xtoon Grey Men's Outdoor Shoes	55	675	1
13	lyvi - Grey Men's Daily Slipper	84	161	999
14	Aadi - White Men's Sneakers	66	340	999
15	Neemans Black Men's Daily Slipper	50	352	699
16	UniStar Outdoor Olive Casual Shoes	56	1	2
17	Campus Blue Men's Daily Slipper (Pair of 1)	6	748	799
18	Big Fox Blue Flip Flops	70	747	2
19	Aadi Beige Outdoor Shoes	40	604	999

RESULT AND ANALYSIS

- check the file by opening from the computer into MS excel

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
1		product name	discount percentage	price	Off price															
2	0	lyvi - Black Rubber Daily Slipper	40	157	999															
3	1	atom Spring Edge Alpha 1 Beige Men's Lifestyle Shoes	84	1	3															
4	2	Xtoon White Men's Lifestyle Shoes	60	496	1															
5	3	Crazy Bunny - Red Men's Thong Flip Flop	67	183	499															
6	4	Fashion Victim Lifestyle Tan Casual Shoes	63	628	999															
7	5	DOCTOR EXTRA SOFT - Dark Grey Men's Daily Slipper	37	399	999															
8	6	Figor Black Lifestyle Shoes	60	575	1															
9	7	Campus AGR-003 - Black Men's Slip-on Shoes	70	748	1															
10	8	Fashion Victim - White Men's Sneakers	47	596	3															
11	9	Campus - Mustard Men's Slide Flip Flop	83	649	649															
12	10	Secritas Black Men's Slide Flip Flop	77	228	999															
13	11	Aerowalk - Tan Men's Leather Slipper	10	384	429															
14	12	Xtoon Grey Men's Outdoor Shoes	55	675	1															
15	13	lyvi - Grey Men's Daily Slipper	84	161	999															
16	14	Aadi - White Men's Sneakers	66	340	999															
17	15	Neemans Black Men's Daily Slipper	50	352	699															
18	16	UniStar Outdoor Olive Casual Shoes	56	1	2															
19	17	Campus Blue Men's Daily Slipper (Pair of 1)	6	748	799															
20	18	Big Fox Blue Flip Flops	70	747	2															
21	19	Aadi Beige Outdoor Shoes	40	604	999															
22																				

- **ANALYSIS FROM THE DATA.**

Price Patterns: Identify the most common price ranges and spot any outliers.

Discount Effectiveness: Determine which products/categories are most discounted and if higher-priced items have better deals.

Market Trends: Discover which product categories are most competitive or in demand, and evaluate pricing trends over time.

Customer Savings: Analyse how much customers actually save on average and which products offer the largest reductions.

CONCLUSION

In the project titled "Website Scraping on Snapdeal Website," we successfully utilized Python libraries, specifically Requests and BeautifulSoup, to automate the extraction of valuable product data from the Snapdeal platform. This process allowed us to collect detailed information such as product names, prices, discounts, and off-prices, providing a comprehensive overview of Snapdeal's offerings. Through careful analysis of the scraped data, we uncovered significant insights regarding price distributions, discount trends, and the performance of various product categories. This demonstrated the practical applications of web scraping in obtaining market insights and understanding consumer behavior.

The insights gained highlight the competitive landscape of e-commerce, revealing opportunities for customers to find the best deals and informing businesses on potential pricing strategies. This project enhanced my understanding of data extraction and analysis techniques in Python, showcasing the critical role data plays in decision-making within the online retail environment. Future enhancements could include scraping additional data points and comparing insights across multiple e-commerce platforms, further expanding the potential for in-depth market analysis.