



## **Web Scrapping Eyeglasses from Lenskart**

**Project By - Miss. Pooja Sudam Pawar**

**Guided By - Mr. Sameer Warsolkar Sir**



# Web Scraping Eyeglasses from Lenskart

## 1. Aim:

To extract key product information including the product title, size, models/variants, price, ratings, and special offers. This data will facilitate market analysis, product comparison, and consumer trend insights.

## 2. Objectives:

- **Data Collection:** Gather detailed information on eyeglasses, including product titles, sizes, models/variants, prices, ratings, and special offers.
- **Database Creation:** Compile the scraped data into a structured format to create a comprehensive database for easy access and analysis.
- **Price Monitoring and Analysis:** Track and analyze price trends over time to identify the best times to buy and to monitor pricing strategies.
- **Customer Feedback Analysis:** Gather and analyze customer ratings and reviews to assess product quality and customer satisfaction.
- **Structured Data Frame Generation:** Convert the extracted data into a well-organized dataframe format for efficient analysis.

### 3. Outline:

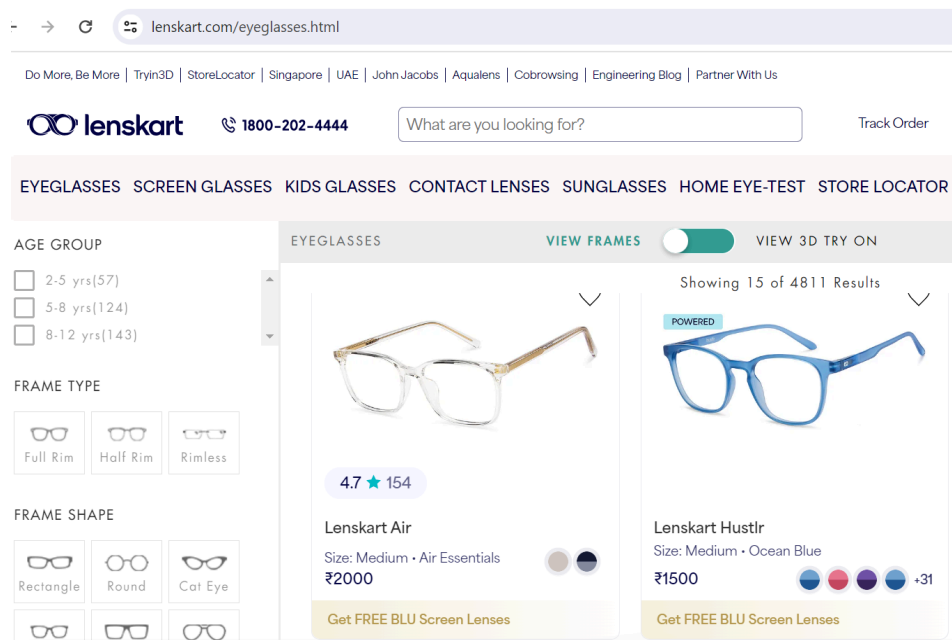
From this site, we are going to grab the following information:

- Product Title
- Size
- Model/Variant
- Price
- Ratings
- Special Offer

### 4. Steps:

#### Choose the Website and Webpage URL:

- The first step is to select the website you want to scrape. We will extract data of Eyeglasses from lenskart.com



#### 1. Inspect the website:

➤ Now the next step is to understand the website structure. Understand what the attributes of the elements that are of your interest are. Right click on the website to select “Inspect”. This will open HTML code. Use the inspector tool to see the name of all the elements to use in the code.



## 2. Installing the important libraries:

Python has several web scraping libraries. We will use the following libraries:

- Requests – for making HTTP requests to website
- BeautifulSoup – for parsing the HTML code
- Pandas – for storing the scraped data in data frame

## 3. Write the Python source code:

We'll write the main python code. The code will perform the following steps:

- Using requests to send an HTTP GET requests
- Using BeautifulSoup to parse the HTML code
- Extracting the required data from the HTML code
- Store the information in a pandas DataFrame

## 4. Exporting the extracted data:

We'll export the data as a CSV file.

We'll use the pandas library.

## 5. Benefits:

- Access to Valuable Data
- Automation of Data Collection
- Real-Time Insights

## 6. Risks:

- Legal Issues
- IP Blocking and Rate Limiting
- Data Accuracy and Integrity

## 5. Lenskart Eyeglasses Web Scrapping Coding Commands:

### 1. Accessing eyeglasses on the Lenskart website using the requests and BeautifulSoup libraries:

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

```
page = requests.get("https://www.lenskart.com/eyeglasses.html")
soup = BeautifulSoup(page.content, 'html.parser')
soup
```

```
<!DOCTYPE html>
<html dir="ltr" lang="en-in"><head><meta charset="utf-8"/><meta conte
```

### 2. Accessing the Eyeglasses Names:

```
Prod_Title = soup.select(".ProductTitle--xakon1.dZrMkC")
Prod_Title
```

```
[<p class="ProductTitle--xakon1 dZrMkC">Lenskart Air</p>,
<p class="ProductTitle--xakon1 dZrMkC">Lenskart Hustlr</p>,
<p class="ProductTitle--xakon1 dZrMkC">Lenskart Hustlr</p>,
<p class="ProductTitle--xakon1 dZrMkC">Vincent Chase</p>,
...]
```

```
Product_Title=[]
for i in Prod_Title:
    Product_Title.append(i.get_text())
Product_Title
#Len(Product_Title)
```

```
['Lenskart Air',
 'Lenskart Hustlr',
 'Lenskart Hustlr',
 'Vincent Chase',
 'Vincent Chase',
 'Lenskart Hustlr',
 'Lenskart Hustlr',
 'Lenskart Hustlr',
 'Lenskart Hustlr',
 'Lenskart Hustlr',
 'Lenskart Air',
 'Lenskart Air',
 'Vincent Chase',
 'Lenskart Air',
 'Lenskart Hustlr',
 'Lenskart STUDIO']
```

### 3. Accessing the Eyeglasses Size, Model/Variant:

```
Prod_S = soup.select(".ProductSize--1gq7vf9.bTR1xU ")
Prod_S
```

```
[<span class="ProductSize--1gq7vf9 bTR1xU">Size<!-- -->
```

```
ProductS=[]
for i in Prod_S :
    ProductS.append(i.get_text())
ProductS
#Len(Product_Strike_Price)
```

```
['Size: Medium • Air Essentials',
 'Size: Medium • Ocean Blue',
 'Size: Medium • Flamingo',
 'Size: Medium • Sleek Steel',
 'Size: Medium • Classic Acetate',
 'Size: Medium • Navy',
 'Size: Medium • Hustlr ACE-Olive',
 'Size: Medium • Hustlr ACE-Denim',
 'Size: Medium • Electric Green',
 'Size: Wide • Matte Essentials',
 'Size: Medium • Air Fusion',
 'Size: Medium • Sleek Steel',
 'Size: Wide • Air Fusion',
 'Size: Medium • Greystone',
 'Size: Medium • KJO']
```

### 4. Combined 'Size' and 'Model/Variant' data has been split into two separate columns:

#### ➤ Accessing Product Size:

```
product_elements = soup.select(".ProductSize--1gq7vf9.bTR1xU")
sizes = [element.get_text().split(' • ')[0].replace('Size: ', '') for element in product_elements]

# Display the extracted sizes
print("Sizes:", sizes)

#Len(sizes)
```

```
Sizes: ['Medium', 'Medium', 'Medium', 'Medium', 'Medium', 'Medium', 'Medium', 'Medium', 'Medium', 'Wide', 'Medium', 'Medium', 'Wide', 'Medium', 'Medium']
```

## ➤ Accessing Product Model/Variation:

```
: # Select the elements with the desired class
product_elements = soup.select(".ProductSize--1gq7vf9.bTR1xU")

# Extract the model/variant information
models = [element.get_text().split(' • ')[1] for element in product_elements]

# Display the model/variant information
print("Models/Variants:", models)
#Len(models)
```

Models/Variants: ['Air Essentials', 'Ocean Blue', 'Flamingo', 'Sleek Steel', 'Classic Acetate', 'Navy', 'Hustlr ACE-Olive', 'Hustlr ACE-Denim', 'Electric Green', 'Matte Essentials', 'Air Fusion', 'Sleek Steel', 'Air Fusion', 'Greystone', 'KJO']

## 5. Accessing the Eyeglasses Price:

```
Prod_Price = soup.select(".SpecialPriceSpan--1mh26ry.ddtLHH")
Prod_Price
```

```
[<span class="SpecialPriceSpan--1mh26ry ddtLHH"><span class="CurrencySpan--50r8yw bMy0qR">₹</span><span>2000</span></span>,
<span class="SpecialPriceSpan--1mh26ry ddtLHH"><span class="CurrencySpan--50r8yw bMy0qR">₹</span><span>1500</span></span></span>,
<span class="SpecialPriceSpan--1mh26ry ddtLHH"><span class="CurrencySpan--50r8yw bMy0qR">₹</span><span>1500</span></span></span>]
```

```
Product_Pri=[]
for i in Prod_Price:
    Product_Pri.append(i.get_text())
Product_Pri
#Len(Prod_Price)
```

```
['₹2000',
'₹1500',
'₹1500',
'₹2000',
'₹2000',
'₹1500',
'₹1500',
'₹1500',
'₹1500',
'₹1500',
'₹2000',
'₹2000',
'₹2000',
'₹2000',
'₹1500',
'₹2000']
```



## 6. Using RegEx to remove the '₹' symbol from the product price list.

```
: import re
Prices = [re.sub(r'[\u20B9]', '', price) for price in Product_Pri]
Prices

: ['2000',
  '1500',
  '1500',
  '2000',
  '2000',
  '1500',
  '1500',
  '1500',
  '1500',
  '1500',
  '2000',
  '2000',
  '2000',
  '2000',
  '1500',
  '2000']
```

## 7. Accessing the Eyeglasses Ratings:

```
: Prod_Rating = soup.select(".NumberedRatingSpan--kts3v6.gA-DJlF")
Prod_Rating
```

```
: [<span class="NumberedRatingSpan--kts3v6 gA-DJlF">4.7</span>,
  <span class="NumberedRatingSpan--kts3v6 gA-DJlF">0</span>,
  <span class="NumberedRatingSpan--kts3v6 gA-DJlF">4.5</span>]
```

```
: Rating_list=[]
for i in Prod_Rating:
    Rating_list.append(i.get_text())
Rating_list
#Len(Prod_Rating)
```

```
: ['4.7',
  '0',
  '4.5',
  '4.8',
  '4.5',
  '4.6',
  '5',
  '0',
  '5',
  '4.8',
  '4.6',
  '4.8',
  '4.8',
  '4.6',
  '0']
```

## 8. Accessing the Eyeglasses Special Offer:

```
offer = soup.select(".OfferContainer--gvtaaw.dQgfyT")
offer
```

```
[<div class="OfferContainer--gvtaaw dQgfyT">Get FREE BLU Screen Lenses</div>,<br><div class="OfferContainer--gvtaaw dQgfyT">Get FREE BLU Screen Lenses</div>,<br><div class="OfferContainer--gvtaaw dQgfyT">Get FREE BLU Screen Lenses</div>,<br>
```

```
P_Offer=[]
for i in offer:
    P_Offer.append(i.get_text())
P_Offer
#len(offer)
```

[illegible]

## 9.Importing the pandas and creating DataFrame:

```
import pandas as pd
Eyeglasses = pd.DataFrame({
    "Product Name":Product_Title,
    "Size": sizes,
    "Model/Variant": models,
    "Price(Rs)":Prices,
    "Ratings(*)": Rating_list,
    "Special Offer": P_Offer
})
Eyeglasses
```

	Product Name	Size	Model/Variant	Price(Rs)	Ratings(*)	Special Offer
0	Lenskart Air	Medium	Air Essentials	2000	4.7	Get FREE BLU Screen Lenses
1	Lenskart Hustlr	Medium	Ocean Blue	1500	0	Get FREE BLU Screen Lenses
2	Lenskart Hustlr	Medium	Flamingo	1500	4.5	Get FREE BLU Screen Lenses
3	Vincent Chase	Medium	Sleek Steel	2000	4.8	Get FREE BLU Screen Lenses
4	Vincent Chase	Medium	Classic Acetate	2000	4.5	Get FREE BLU Screen Lenses
5	Lenskart Hustlr	Medium	Navy	1500	4.6	Get FREE BLU Screen Lenses
6	Lenskart Hustlr	Medium	Hustlr ACE-Olive	1500	5	Get FREE BLU Screen Lenses
7	Lenskart Hustlr	Medium	Hustlr ACE-Denim	1500	0	Get FREE BLU Screen Lenses
8	Lenskart Hustlr	Medium	Electric Green	1500	5	Get FREE BLU Screen Lenses
9	Lenskart Air	Wide	Matte Essentials	2000	4.8	Get FREE BLU Screen Lenses
10	Lenskart Air	Medium	Air Fusion	2000	4.6	Get FREE BLU Screen Lenses
11	Vincent Chase	Medium	Sleek Steel	2000	4.8	Get FREE BLU Screen Lenses
12	Lenskart Air	Wide	Air Fusion	2000	4.8	Get FREE BLU Screen Lenses
13	Lenskart Hustlr	Medium	Greystone	1500	4.6	Get FREE BLU Screen Lenses
14	Lenskart STUDIO	Medium	KJO	2000	0	Get FREE BLU Screen Lenses

## 10. Converting and Storing DataFrame in the form of a CSV file and opening the file in application:

```
: Eyeglasses.to_csv("Eyeglasses_Web_Scraping.csv",index=False)
```

```
: Read=pd.read_csv("Eyeglasses_Web_Scraping.csv")  
Read
```

	Product Name	Size	Model/Variant	Price(Rs)	Ratings(*)	Special Offer
0	Lenskart Air	Medium	Air Essentials	2000	4.7	Get FREE BLU Screen Lenses
1	Lenskart Hustlr	Medium	Ocean Blue	1500	0.0	Get FREE BLU Screen Lenses
2	Lenskart Hustlr	Medium	Flamingo	1500	4.5	Get FREE BLU Screen Lenses
3	Vincent Chase	Medium	Sleek Steel	2000	4.8	Get FREE BLU Screen Lenses
4	Vincent Chase	Medium	Classic Acetate	2000	4.5	Get FREE BLU Screen Lenses
5	Lenskart Hustlr	Medium	Navy	1500	4.6	Get FREE BLU Screen Lenses
6	Lenskart Hustlr	Medium	Hustlr ACE-Olive	1500	5.0	Get FREE BLU Screen Lenses
7	Lenskart Hustlr	Medium	Hustlr ACE-Denim	1500	0.0	Get FREE BLU Screen Lenses
8	Lenskart Hustlr	Medium	Electric Green	1500	5.0	Get FREE BLU Screen Lenses
9	Lenskart Air	Wide	Matte Essentials	2000	4.8	Get FREE BLU Screen Lenses
10	Lenskart Air	Medium	Air Fusion	2000	4.6	Get FREE BLU Screen Lenses
11	Vincent Chase	Medium	Sleek Steel	2000	4.8	Get FREE BLU Screen Lenses
12	Lenskart Air	Wide	Air Fusion	2000	4.8	Get FREE BLU Screen Lenses
13	Lenskart Hustlr	Medium	Greystone	1500	4.6	Get FREE BLU Screen Lenses
14	Lenskart STUDIO	Medium	KJO	2000	0.0	Get FREE BLU Screen Lenses

## **6. Conclusion:**

In conclusion, the web scraping project focused on Lenskart.com efficiently collected comprehensive data on eyeglasses, encompassing product titles, sizes, models/variants, prices, ratings, and special offers. Adherence to stringent ethical guidelines guaranteed the accuracy and reliability of the dataset, while maintaining compliance with website terms of service and privacy policies.

The comparative analysis against industry competitors revealed valuable insights into Lenskart's market position and consumer preferences. This analysis provided strategic intelligence on pricing strategies and market trends, offering a clear path for informed decision-making and future business initiatives in the eyewear sector.

***Thank You.***

