bd2-jcpenney-individual-1

December 2, 2023

1 University of Stirling

2 ITNPBD2 Representing and Manipulating Data

3 Assignment Autumn 2023

4 A Consultancy Job for JC Penney

This notebook forms the assignment instructions and submission document of the assignment for ITNPBD2. Read the instructions carefully and enter code into the cells as indicated.

You will need these five files, which were in the Zip file you downloaded from the course webpage:

- jcpenney_reviewers.json
- jcpenney_products.json
- products.csv
- reviews.csv
- users.csv

The data in these files describes products that have been sold by the American retail giant, JC Penney, and reviews by customers who bought them. Note that the product data is real, but the customer data is synthetic.

Your job is to process the data, as requested in the instructions in the markdown cells in this notebook.

5 Completing the Assignment

Rename this file to be xxxxxx_BD2 where xxxxxx is your student number, then type your code and narrative description into the boxes provided. Add as many code and markdown cells as you need. The cells should contain:

- **Text narrative describing what you did with the data**
- **The code that performs the task you have described**
- **Comments that explain your code**

6 Marking Scheme

The assessment will be marked against the university Common Marking Scheme (CMS)

Here is a summary of what you need to achieve to gain a grade in the major grade bands:

Grade	Requirement
Fail	You will fail if your code does not run or does not achieve even the basics of the task. You may also fail if you submit code without either comments or a text explanation of what the code does.
Pass	To pass, you must submit sufficient working code to show that you have mastered the basics of the task, even if not everything works completely. You must include some justifications for your choice of methods, but without mentioning alternatives.
Merit	For a merit, your code must be mostly correct, with only small problems or parts missing, and your comments must be useful rather than simply re-stating the code in English. Most choices for methods and structures should be explained and alternatives mentioned.
Distinction	For a distinction, your code must be working, correct, and well commented and shows an appreciation of style, efficiency and reliability. All choices for methods and structures are concisely justified and alternatives are given well thought considerations. For a distinction, your work should be good enough to present to executives at the company.

The full details of the CMS can be found here

https://www.stir.ac.uk/about/professional-services/student-academic-and-corporate-services/academic-registry/academic-policy-and-practice/quality-handbook/assessment-policy-and-procedure/appendix-2-postgraduate-common-marking-scheme/

Note that this means there are not certain numbers of marks allocated to each stage of the assignment. Your grade will reflect how well your solutions and comments demonstrate that you have achieved the learning outcomes of the task.

6.1 Submission

When you are ready to submit, **print** your notebook as PDF (go to File -> Print Preview) in the Jupyter menu. Make sure you have run all the cells and that their output is displayed. Any lines of code or comments that are not visible in the pdf should be broken across several lines. You can then submit the file online.

Late penalties will apply at a rate of three marks per day, up to a maximum of 7 days. After 7

days you will be given a mark of 0. Extensions will be considered under acceptable circumstances outside your control.

6.2 Academic Integrity

This is an individual assignment, and so all submitted work must be fully your own work.

The University of Stirling is committed to protecting the quality and standards of its awards. Consequently, the University seeks to promote and nurture academic integrity, support staff academic integrity, and support students to understand and develop good academic skills that facilitate academic integrity.

In addition, the University deals decisively with all forms of Academic Misconduct.

Where a student does not act with academic integrity, their work or behaviour may demonstrate Poor Academic Practice or it may represent Academic Misconduct.

6.2.1 Poor Academic Practice

Poor Academic Practice is defined as: "The submission of any type of assessment with a lack of referencing or inadequate referencing which does not effectively acknowledge the origin of words, ideas, images, tables, diagrams, maps, code, sound and any other sources used in the assessment."

6.2.2 Academic Misconduct

Academic Misconduct is defined as: "any act or attempted act that does not demonstrate academic integrity and that may result in creating an unfair academic advantage for you or another person, or an academic disadvantage for any other member or member of the academic community."

Plagiarism is presenting somebody else's work as your own **and includes the use of artificial intelligence tools such as GPT or CoPilot**. Plagiarism is a form of academic misconduct and is taken very seriously by the University. Students found to have plagiarised work can have marks deducted and, in serious cases, even be expelled from the University. Do not submit any work that is not entirely your own. Do not collaborate with or get help from anybody else with this assignment.

The University of Stirling's full policy on Academic Integrity can be found at:

https://www.stir.ac.uk/about/professional-services/student-academic-and-corporate-services/academic-registry/academic-policy-and-practice/quality-handbook/academic-integrity-policy-and-academic-misconduct-procedure/

6.3 The Assignment

Your task with this assignment is to use the data provided to demonstrate your Python data manipulation skills.

There are three `.csv` files and two `.json` files so you can process different types of data. The files also contain unstructured data in the form of natural language in English and links to images that you can access from the JC Penney website (use the field called `product_image_urls`).

Start with easy tasks to show you can read in a file, create some variables and data structures, and manipulate their contents. Then move onto something more interesting.

Look at the data that we provided with this assessment and think of something interesting to do with it using whatever libraries you like. Describe what you decide to do with the data and why it might be interesting or useful to the company to do it.

You can add additional data if you need to - either download it or access it using `requests`. Produce working code to implement your ideas in as many cells as you need below. There is no single right answer, the aim is to simply show you are competent in using python for data analysis. Exactly how you do that is up to you.

For a distinction class grade, this must show originality, creative thinking, and insights beyond what you've been taught directly on the module.

6.4 Structure

You may structure the project how you wish, but here is a suggested guideline to help you organise your work:

1. Data Exploration - Explore the data and show you understand its structure and relations
2. Data Validation - Check the quality of the data. Is it complete? Are there obvious errors?
3. Data Visualisation - Gain an overall understanding of the data with visualisations
4. Data Analysis = Set some questions and use the data to answer them
5. Data Augmentation - Add new data from another source to bring new insights to the data you already have

7 Remember to make sure you are working completely on your own.

8 Don't work in a group or with a friend

You may NOT use any automated code generation or analytics tools for this assignment, so do not use tools like GPT. You can look up the syntax for the functions you use, but you must write the code yourself and the comments must provide an insightful analysis of the results.

8.1 Data Importing

Here i am reading the "products.csv" dataset through pandas.

This data has 6 features including Uniq_id, SKU, Name, Description, Price, and Av_Score.

Only five Data records are displayed.

[1]:
```
#reading first csv data

import pandas as pd

products_data = pd.read_csv("JCPenneyFiles/products.csv")
print(products_data.head())
```

```
                           Uniq_id           SKU  \
0  b6c0b6bea69c722939585baeac73c13d  pp5006380337
```

```
1  93e5272c51d8cce02597e3ce67b7ad0a  pp5006380337
2  013e320f2f2ec0cf5b3ff5418d688528  pp5006380337
3  505e6633d81f2cb7400c0cfa0394c427  pp5006380337
4  d969a8542122e1331e304b09f81a83f6  pp5006380337

                                          Name  \
0  Alfred Dunner® Essential Pull On Capri Pant
1  Alfred Dunner® Essential Pull On Capri Pant
2  Alfred Dunner® Essential Pull On Capri Pant
3  Alfred Dunner® Essential Pull On Capri Pant
4  Alfred Dunner® Essential Pull On Capri Pant

                                       Description  Price  Av_Score
0  Youll return to our Alfred Dunner pull-on capr…  41.09     2.625
1  Youll return to our Alfred Dunner pull-on capr…  41.09     3.000
2  Youll return to our Alfred Dunner pull-on capr…  41.09     2.625
3  Youll return to our Alfred Dunner pull-on capr…  41.09     3.500
4  Youll return to our Alfred Dunner pull-on capr…  41.09     3.125
```

Here i am reading the "reviews.csv" dataset through pandas.

This data has only 4 features including Uniq_id, Username, Score, and Review.

Only five Data records are displayed.

[2]:
```
#reading second csv data

reviews_data = pd.read_csv("JCPenneyFiles/reviews.csv")
print(reviews_data.head())
```

```
                           Uniq_id  Username  Score  \
0  b6c0b6bea69c722939585baeac73c13d  fsdv4141      2
1  b6c0b6bea69c722939585baeac73c13d  krpz1113      1
2  b6c0b6bea69c722939585baeac73c13d  mbmg3241      2
3  b6c0b6bea69c722939585baeac73c13d  zeqg1222      0
4  b6c0b6bea69c722939585baeac73c13d  nvfn3212      3

                                              Review
0  You never have to worry about the fit…Alfred…
1  Good quality fabric. Perfect fit. Washed very …
2  I do not normally wear pants or capris that ha…
3  I love these capris! They fit true to size and…
4  This product is very comfortable and the fabri…
```

Here i am reading the "users.csv" dataset through pandas.

This data has 3 features including Username, DOB, and State.

Only five Data records are displayed.

[3]:
```
#reading third csv data

users_data = pd.read_csv("JCPenneyFiles/users.csv")
print(users_data.head())
```

```
   Username         DOB          State
0  bkpn1412  31.07.1983         Oregon
1  gqjs4414  27.07.1998  Massachusetts
2  eehe1434  08.08.1950          Idaho
3  hkxj1334  03.08.1969        Florida
4  jjbd1412  26.07.2001        Georgia
```

Here i am reading "jcpenney_products.json" data file.

This json has multiple jsons in it. So each json has assigned a unique ID because i want to merge all data in one pandas dataframe.

[4]:
```
# reading first json data
# Json data has multiple jsons in it, each json has been assign a unique key (e.
 ↪g., 'json_1', 'json_2'), So it can be used to create the pandas datafrmae,␣
 ↪where each key will be the column name

import json

products_data_json = {}
with open("JCPenneyFiles/jcpenney_products.json", "r") as json_file:
    val = 0
    for line in json_file:
        data = json.loads(line)
        val = val + 1
        json_index = 'json_' + str(val)
        products_data_json[json_index] = data

print(len(products_data_json))
```

```
7982
```

Here i am reading "jcpenney_reviews.json" data file.

This json has also multiple jsons in it. So each json has assigned a unique ID because here again, I want to merge all data in one pandas dataframe.

[5]:
```
# reading second json data
# Json data has multiple jsons in it, each json has been assign a unique key (e.
 ↪g., 'json_1', 'json_2'), So it can be used to create the pandas datafrmae,␣
 ↪where each key will be the column name

reviewers_data_json = {}
with open("JCPenneyFiles/jcpenney_reviewers.json", "r") as json_file:
```

```
    val = 0
    for line in json_file:
        data = json.loads(line)
        val = val + 1
        json_index = 'json_' + str(val)
        reviewers_data_json[json_index] = data

print(len(reviewers_data_json))
```

```
5000
```

Here 4th Pandas dataframe is created from the New Product json, here we used the keys of each json data to define columns of pandas dataframe. So data here in the final form can be merged with all other dataframes.

[ ]:
```
# first json data to pandas dataframe
# column name is the json key

products_json_keys = list(products_data_json.keys())
print(products_json_keys[0])

df_products_data_json = pd.DataFrame()
for j_key in products_json_keys:
    df_products_data_json = df_products_data_json.append(pd.
 ↪json_normalize(products_data_json[j_key]), ignore_index=True)
```

[7]:
```
print(df_products_data_json.head())
```

```
                           uniq_id           sku  \
0  b6c0b6bea69c722939585baeac73c13d  pp5006380337
1  93e5272c51d8cce02597e3ce67b7ad0a  pp5006380337
2  013e320f2f2ec0cf5b3ff5418d688528  pp5006380337
3  505e6633d81f2cb7400c0cfa0394c427  pp5006380337
4  d969a8542122e1331e304b09f81a83f6  pp5006380337

                                  name_title  \
0  Alfred Dunner® Essential Pull On Capri Pant
1  Alfred Dunner® Essential Pull On Capri Pant
2  Alfred Dunner® Essential Pull On Capri Pant
3  Alfred Dunner® Essential Pull On Capri Pant
4  Alfred Dunner® Essential Pull On Capri Pant

                                       description list_price sale_price  \
0  You'll return to our Alfred Dunner pull-on cap…      41.09      24.16
1  You'll return to our Alfred Dunner pull-on cap…      41.09      24.16
2  You'll return to our Alfred Dunner pull-on cap…      41.09      24.16
3  You'll return to our Alfred Dunner pull-on cap…      41.09      24.16
4  You'll return to our Alfred Dunner pull-on cap…      41.09      24.16
```

```
        category                 category_tree  average_product_rating  \
0  alfred dunner  jcpenney|women|alfred dunner                   2.625
1  alfred dunner  jcpenney|women|alfred dunner                   3.000
2       view all       jcpenney|women|view all                   2.625
3       view all       jcpenney|women|view all                   3.500
4       view all       jcpenney|women|view all                   3.125

                                        product_url  \
0  http://www.jcpenney.com/alfred-dunner-essentia…
1  http://www.jcpenney.com/alfred-dunner-essentia…
2  http://www.jcpenney.com/alfred-dunner-essentia…
3  http://www.jcpenney.com/alfred-dunner-essentia…
4  http://www.jcpenney.com/alfred-dunner-essentia…

                                  product_image_urls          brand  \
0  http://s7d9.scene7.com/is/image/JCPenney/DP122…  Alfred Dunner
1  http://s7d9.scene7.com/is/image/JCPenney/DP122…  Alfred Dunner
2  http://s7d9.scene7.com/is/image/JCPenney/DP122…  Alfred Dunner
3  http://s7d9.scene7.com/is/image/JCPenney/DP122…  Alfred Dunner
4  http://s7d9.scene7.com/is/image/JCPenney/DP122…  Alfred Dunner

   total_number_reviews                                            Reviews  \
0                     8  [{'User': 'fsdv4141', 'Review': 'You never hav…
1                     8  [{'User': 'tpcu2211', 'Review': 'You never hav…
2                     8  [{'User': 'pcfg3234', 'Review': 'You never hav…
3                     8  [{'User': 'ngrq4411', 'Review': 'You never hav…
4                     8  [{'User': 'nbmi2334', 'Review': 'You never hav…

                                        Bought With
0  [898e42fe937a33e8ce5e900ca7a4d924, 8c02c262567…
1  [bc9ab3406dcaa84a123b9da862e6367d, 18eb69e8fc2…
2  [3ce70f519a9cfdd85cdbdecd358e5347, b0295c96d2b…
3  [efcd811edccbeb5e67eaa8ef0d991f7c, 7b2cc00171e…
4  [0ca5ad2a218f59eb83eec1e248a0782d, 9869fc8da14…
```

Here 5th Pandas dataframe is created from the New Reviews json, here as well we used the keys of each json data to define columns of pandas dataframe. So data here in the final form can be merged with all other dataframes

[ ]:
```
# second json data to pandas dataframe
# column name is the json key

reviewers_json_keys = list(reviewers_data_json.keys())
print(reviewers_json_keys[0])

df_reviewers_data_json = pd.DataFrame()
for j_key in reviewers_json_keys:
```

```
    df_reviewers_data_json = df_reviewers_data_json.append(pd.
↪json_normalize(reviewers_data_json[j_key]), ignore_index=True)
```

[9]:
```
print(df_reviewers_data_json.head())
```

```
   Username         DOB          State  \
0  bkpn1412  31.07.1983         Oregon
1  gqjs4414  27.07.1998  Massachusetts
2  eehe1434  08.08.1950          Idaho
3  hkxj1334  03.08.1969        Florida
4  jjbd1412  26.07.2001        Georgia

                                            Reviewed
0                 [cea76118f6a9110a893de2b7654319c0]
1                 [fa04fe6c0dd5189f54fe600838da43d3]
2                                                 []
3  [f129b1803f447c2b1ce43508fb822810, 3b0c9bc0be6…
4                                                 []
```

8.2 Data Preprocessing

Here i merged the first two pandas dataframe in single dataframe based on Uniq_id.

[10]:
```
# merge product data and review data in one datafrmae
merged_df_1 = pd.merge(products_data, reviews_data, on='Uniq_id', how='inner')
print(merged_df_1.shape)
merged_df_1.head()
```

```
(39063, 9)
```

[10]:
```
                           Uniq_id           SKU  \
0  b6c0b6bea69c722939585baeac73c13d  pp5006380337
1  b6c0b6bea69c722939585baeac73c13d  pp5006380337
2  b6c0b6bea69c722939585baeac73c13d  pp5006380337
3  b6c0b6bea69c722939585baeac73c13d  pp5006380337
4  b6c0b6bea69c722939585baeac73c13d  pp5006380337

                                    Name  \
0  Alfred Dunner® Essential Pull On Capri Pant
1  Alfred Dunner® Essential Pull On Capri Pant
2  Alfred Dunner® Essential Pull On Capri Pant
3  Alfred Dunner® Essential Pull On Capri Pant
4  Alfred Dunner® Essential Pull On Capri Pant

                                     Description  Price  Av_Score  \
0  Youll return to our Alfred Dunner pull-on capr…  41.09     2.625
1  Youll return to our Alfred Dunner pull-on capr…  41.09     2.625
2  Youll return to our Alfred Dunner pull-on capr…  41.09     2.625
```

```
3  Youll return to our Alfred Dunner pull-on capr…  41.09     2.625
4  Youll return to our Alfred Dunner pull-on capr…  41.09     2.625

   Username  Score                                              Review
0  fsdv4141      2  You never have to worry about the fit…Alfred…
1  krpz1113      1  Good quality fabric. Perfect fit. Washed very …
2  mbmg3241      2  I do not normally wear pants or capris that ha…
3  zeqg1222      0  I love these capris! They fit true to size and…
4  nvfn3212      3  This product is very comfortable and the fabri…
```

Here i merged the 3rd pandas dataframe with the merged dataframe based on Username.

[11]:
```
# merge product data, review data, and user data in one dataframe
merged_df_2 = pd.merge(merged_df_1, users_data, on='Username', how='inner')
merged_df_2.head()
```

[11]:
```
                           Uniq_id           SKU  \
0  b6c0b6bea69c722939585baeac73c13d  pp5006380337
1  cbe8d131628ec67e803c47d3dd6f2529  pp5005090739
2  5ea5f53bbb750106865a044634404dd7  pp5006020188
3  0144d2094668b42ae7c674915806f5f3  pp5005690566
4  99141a2b164cf257c96bcb4593915b50  pp5005740454

                                          Name  \
0       Alfred Dunner® Essential Pull On Capri Pant
1        Xersion  Quick-Dri Short-Sleeve Polo Shirt
2       Liz Claiborne® Serenity Bath Rug Collection
3  bareMinerals Matte Foundation Broad Spectrum S…
4               Zoomers Maternity Knit Gaucho Pants

                                      Description  Price  Av_Score  \
0  Youll return to our Alfred Dunner pull-on capr…  41.09  2.625000
1  Collar up and cool down in this soft polo from… -42.30  3.250000
2  Defined by its tufted ogee pattern and calming…  36.26  3.500000
3  What it is:A mineral-based foundation that del…    NaN  3.454545
4  Keep your maternity style simple and polished …  36.26  2.200000

   Username  Score                                              Review  \
0  fsdv4141      2  You never have to worry about the fit…Alfred…
1  fsdv4141      2  These are great shirts, looks great all day. W…
2  fsdv4141      5  I purchase three rugs to replace twenty-year-o…
3  fsdv4141      1  I am a huge user of BE original fromula. I hap…
4  fsdv4141      1  Very soft and stretchy! They arent as dressy a…

          DOB           State
0  31.07.1980  American Samoa
1  31.07.1980  American Samoa
```

```
2  31.07.1980  American Samoa
3  31.07.1980  American Samoa
4  31.07.1980  American Samoa
```

Here the first character of the column name "uniq_id" was in lower case, so, renamed it to the first letter capital. So that it can be used to merged both dataframes.

[12]:
```
# rename the name of column
df_products_data_json_df = df_products_data_json.rename(columns={'uniq_id':␣
 ↪'Uniq_id'})
```

Here i merged the 4th pandas dataframe with the latest merged dataframe by using Uniq_id column.

[13]:
```
# merge product data, review data, and user data with product json data in one␣
 ↪dataframe
merged_df_3 = pd.merge(merged_df_2, df_products_data_json_df, on='Uniq_id',␣
 ↪how='inner')
merged_df_3.head()
```

[13]:
```
                           Uniq_id           SKU  \
0  b6c0b6bea69c722939585baeac73c13d  pp5006380337
1  b6c0b6bea69c722939585baeac73c13d  pp5006380337
2  b6c0b6bea69c722939585baeac73c13d  pp5006380337
3  b6c0b6bea69c722939585baeac73c13d  pp5006380337
4  b6c0b6bea69c722939585baeac73c13d  pp5006380337

                                    Name  \
0  Alfred Dunner® Essential Pull On Capri Pant
1  Alfred Dunner® Essential Pull On Capri Pant
2  Alfred Dunner® Essential Pull On Capri Pant
3  Alfred Dunner® Essential Pull On Capri Pant
4  Alfred Dunner® Essential Pull On Capri Pant

                                     Description  Price  Av_Score  \
0  Youll return to our Alfred Dunner pull-on capr…  41.09     2.625
1  Youll return to our Alfred Dunner pull-on capr…  41.09     2.625
2  Youll return to our Alfred Dunner pull-on capr…  41.09     2.625
3  Youll return to our Alfred Dunner pull-on capr…  41.09     2.625
4  Youll return to our Alfred Dunner pull-on capr…  41.09     2.625

   Username  Score                                             Review  \
0  fsdv4141      2  You never have to worry about the fit…Alfred…
1  krpz1113      1  Good quality fabric. Perfect fit. Washed very …
2  mbmg3241      2  I do not normally wear pants or capris that ha…
3  zeqg1222      0  I love these capris! They fit true to size and…
4  nvfn3212      3  This product is very comfortable and the fabri…
```

```
          DOB  … sale_price       category                 category_tree  \
0  31.07.1980  …      24.16  alfred dunner  jcpenney|women|alfred dunner
1  30.07.1987  …      24.16  alfred dunner  jcpenney|women|alfred dunner
2  08.08.1951  …      24.16  alfred dunner  jcpenney|women|alfred dunner
3  28.07.1994  …      24.16  alfred dunner  jcpenney|women|alfred dunner
4  31.07.1980  …      24.16  alfred dunner  jcpenney|women|alfred dunner

  average_product_rating                                        product_url  \
0                  2.625  http://www.jcpenney.com/alfred-dunner-essentia…
1                  2.625  http://www.jcpenney.com/alfred-dunner-essentia…
2                  2.625  http://www.jcpenney.com/alfred-dunner-essentia…
3                  2.625  http://www.jcpenney.com/alfred-dunner-essentia…
4                  2.625  http://www.jcpenney.com/alfred-dunner-essentia…

                                product_image_urls          brand  \
0  http://s7d9.scene7.com/is/image/JCPenney/DP122…  Alfred Dunner
1  http://s7d9.scene7.com/is/image/JCPenney/DP122…  Alfred Dunner
2  http://s7d9.scene7.com/is/image/JCPenney/DP122…  Alfred Dunner
3  http://s7d9.scene7.com/is/image/JCPenney/DP122…  Alfred Dunner
4  http://s7d9.scene7.com/is/image/JCPenney/DP122…  Alfred Dunner

  total_number_reviews                                            Reviews  \
0                    8  [{'User': 'fsdv4141', 'Review': 'You never hav…
1                    8  [{'User': 'fsdv4141', 'Review': 'You never hav…
2                    8  [{'User': 'fsdv4141', 'Review': 'You never hav…
3                    8  [{'User': 'fsdv4141', 'Review': 'You never hav…
4                    8  [{'User': 'fsdv4141', 'Review': 'You never hav…

                                      Bought With
0  [898e42fe937a33e8ce5e900ca7a4d924, 8c02c262567…
1  [898e42fe937a33e8ce5e900ca7a4d924, 8c02c262567…
2  [898e42fe937a33e8ce5e900ca7a4d924, 8c02c262567…
3  [898e42fe937a33e8ce5e900ca7a4d924, 8c02c262567…
4  [898e42fe937a33e8ce5e900ca7a4d924, 8c02c262567…

[5 rows x 25 columns]
```

Here i merged the 5th pandas dataframe with the latest merged dataframe by using Username column. So all five datasets has been merged in a single pandas dataframe.

[14]:
```
# merge product data, review data, and user data with product json data in one␣
 ↪dataframe
final_merged_df = pd.merge(merged_df_3, df_reviewers_data_json, on='Username',␣
 ↪how='inner')
final_merged_df.head()
```

[14]:

	Uniq_id	SKU	Name	Description	Price	Av_Score	Username	Score	Review	DOB_x	…	average_product_rating	product_url	product_image_urls	brand
0	b6c0b6bea69c722939585baeac73c13d	pp5006380337	Alfred Dunner® Essential Pull On Capri Pant	Youll return to our Alfred Dunner pull-on capr…	41.09	2.625000	fsdv4141	2	You never have to worry about the fit…Alfred…	31.07.1980	…	2.625000	http://www.jcpenney.com/alfred-dunner-essentia…	http://s7d9.scene7.com/is/image/JCPenney/DP122…	Alfred Dunner
1	cbe8d131628ec67e803c47d3dd6f2529	pp5005090739	Xersion Quick-Dri Short-Sleeve Polo Shirt	Collar up and cool down in this soft polo from…	-42.30	3.250000	fsdv4141	2	These are great shirts, looks great all day. W…	31.07.1980	…	3.250000	http://www.jcpenney.com/xersion-quick-dri-shor…	http://s7d9.scene7.com/is/image/JCPenney/DP021…	Xersion
2	5ea5f53bbb750106865a044634404dd7	pp5006020188	Liz Claiborne® Serenity Bath Rug Collection	Defined by its tufted ogee pattern and calming…	36.26	3.500000	fsdv4141	5	I purchase three rugs to replace twenty-year-o…	31.07.1980	…	3.500000	http://www.jcpenney.com/liz-claiborne-serenity…	http://s7d9.scene7.com/is/image/JCPenney/DP100…	LIZ CLAIBORNE
3	0144d2094668b42ae7c674915806f5f3	pp5005690566	bareMinerals Matte Foundation Broad Spectrum S…	What it is:A mineral-based foundation that del…	NaN	3.454545	fsdv4141	1	I am a huge user of BE original fromula. I hap…	31.07.1980	…	3.454545	http://www.jcpenney.com/bareminerals-matte-fou…	http://s7d2.scene7.com/is/image/JCPenney/DP061…	BAREMINERALS
4	99141a2b164cf257c96bcb4593915b50	pp5005740454	Zoomers Maternity Knit Gaucho Pants	Keep your maternity style simple and polished …	36.26	2.200000	fsdv4141	1	Very soft and stretchy! They arent as dressy a…	31.07.1980	…	2.200000	http://www.jcpenney.com/zoomers-maternity-knit…		

```
4  http://s7d2.scene7.com/is/image/JCPenney/DP062…  Asstd National Brand

  total_number_reviews                                                Reviews  \
0                    8  [{'User': 'fsdv4141', 'Review': 'You never hav…
1                    8  [{'User': 'mlvi1412', 'Review': 'I am very hap…
2                    8  [{'User': 'seeu4332', 'Review': 'Very plush an…
3                   11  [{'User': 'srym3234', 'Review': 'This is the o…
4                    5  [{'User': 'hmfw1233', 'Review': 'Really comfor…

                                      Bought With       DOB_y  \
0  [898e42fe937a33e8ce5e900ca7a4d924, 8c02c262567…  31.07.1980
1  [bf9cce92e57c76c94725c3c90e736209, 27b03b9241a…  31.07.1980
2  [1524d45ea3054089c6a1fd8089ab4c5b, 66aaaea6976…  31.07.1980
3  [4113b59b6a3cf107b836a551edfc21bf, e4978a4d098…  31.07.1980
4  [d98c99c360657fc69a4b44f423f64ed9, 4f3c67ce08b…  31.07.1980

          State_y                                           Reviewed
0  American Samoa  [0144d2094668b42ae7c674915806f5f3, 7c27ffd820c…
1  American Samoa  [0144d2094668b42ae7c674915806f5f3, 7c27ffd820c…
2  American Samoa  [0144d2094668b42ae7c674915806f5f3, 7c27ffd820c…
3  American Samoa  [0144d2094668b42ae7c674915806f5f3, 7c27ffd820c…
4  American Samoa  [0144d2094668b42ae7c674915806f5f3, 7c27ffd820c…

[5 rows x 28 columns]
```

8.3 Data Exploration

Here data exploration is being performed.

1. Final combined dataset shape, and features.
2. Dataset column list
3. Dataset describe property, to get the statistics of the dataset
4. Dataset info to find the datatypes of each feature

[15]:
```
# final combined data shape
final_merged_df.shape
```

[15]:
```
(39114, 28)
```

[16]:
```
# dataset columns
final_merged_df.columns
```

[16]:
```
Index(['Uniq_id', 'SKU', 'Name', 'Description', 'Price', 'Av_Score',
       'Username', 'Score', 'Review', 'DOB_x', 'State_x', 'sku', 'name_title',
       'description', 'list_price', 'sale_price', 'category', 'category_tree',
       'average_product_rating', 'product_url', 'product_image_urls', 'brand',
```

```
       'total_number_reviews', 'Reviews', 'Bought With', 'DOB_y', 'State_y',
       'Reviewed'],
      dtype='object')
```

[17]:
```
# data summary statistics
final_merged_df.describe()
```

[17]:

	Price	Av_Score	Score	average_product_rating	\
count	26904.000000	39114.000000	39114.000000	39114.000000	
mean	147.821308	2.990919	1.488009	2.990919	
std	444.543713	0.643796	1.400579	0.643796	
min	-65.270000	1.000000	0.000000	1.000000	
25%	41.380000	2.625000	0.000000	2.625000	
50%	58.010000	3.000000	1.000000	3.000000	
75%	90.650000	3.375000	2.000000	3.375000	
max	17122.170000	5.000000	5.000000	5.000000	

	total_number_reviews
count	39114.000000
mean	7.138186
std	2.932272
min	1.000000
25%	6.000000
50%	8.000000
75%	8.000000
max	23.000000

[18]:
```
# data types and missing values
final_merged_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 39114 entries, 0 to 39113
Data columns (total 28 columns):
 #   Column                  Non-Null Count  Dtype
---  ------                  --------------  -----
 0   Uniq_id                 39114 non-null  object
 1   SKU                     38950 non-null  object
 2   Name                    39114 non-null  object
 3   Description             35766 non-null  object
 4   Price                   26904 non-null  float64
 5   Av_Score                39114 non-null  float64
 6   Username                39114 non-null  object
 7   Score                   39114 non-null  int64
 8   Review                  39114 non-null  object
 9   DOB_x                   39114 non-null  object
 10  State_x                 39114 non-null  object
 11  sku                     39114 non-null  object
```

```
 12  name_title              39114 non-null  object
 13  description             39114 non-null  object
 14  list_price              39114 non-null  object
 15  sale_price              39114 non-null  object
 16  category                39114 non-null  object
 17  category_tree           39114 non-null  object
 18  average_product_rating  39114 non-null  float64
 19  product_url             39114 non-null  object
 20  product_image_urls      39114 non-null  object
 21  brand                   39114 non-null  object
 22  total_number_reviews    39114 non-null  int64
 23  Reviews                 39114 non-null  object
 24  Bought With             39114 non-null  object
 25  DOB_y                   39114 non-null  object
 26  State_y                 39114 non-null  object
 27  Reviewed                39114 non-null  object
dtypes: float64(3), int64(2), object(23)
memory usage: 8.7+ MB
```

8.4 Data Validate

In data validation

1. Checked the duplicated
2. Displayed missing values heatmap

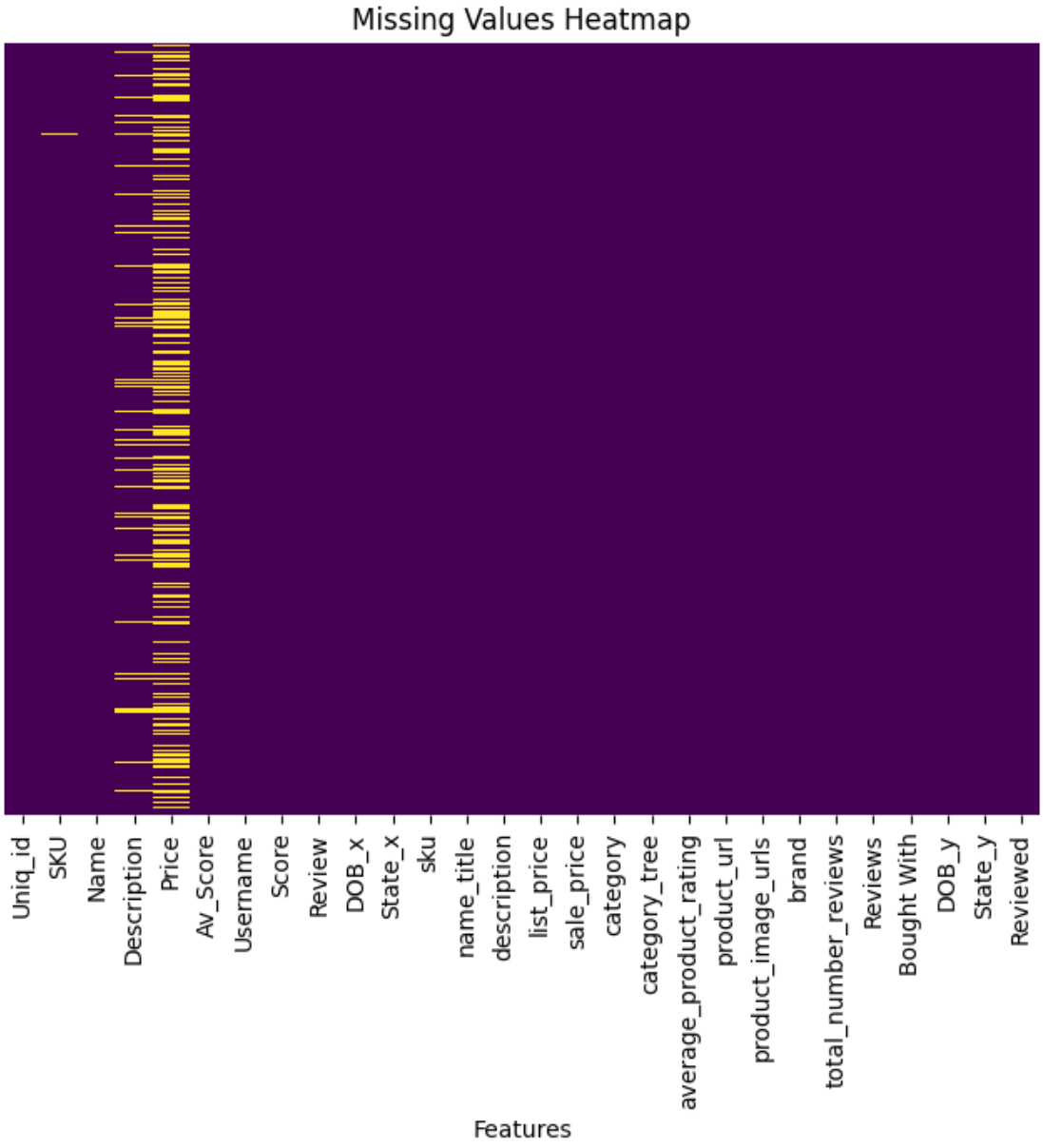
[19]:
```
final_merged_df['Uniq_id'].duplicated()
```

[19]:
```
0        False
1        False
2        False
3        False
4        False
         …
39109    False
39110     True
39111     True
39112    False
39113    False
Name: Uniq_id, Length: 39114, dtype: bool
```

[20]:
```
import matplotlib.pyplot as plt
import seaborn as sns
```

[21]:
```
# missing values headmap
plt.figure(figsize=(8, 6))
sns.heatmap(final_merged_df.isnull(), cmap='viridis', cbar=False,␣
 ↪yticklabels=False)
```

```
plt.title('Missing Values Heatmap')
plt.xlabel("Features")
plt.show()
```



8.5 Data Visualize and Data Analysis

Graph displays the total number of reviews for each product (Top 10)

[23]:

```
# Count the number of reviews for each Uniq_id (Product) with the description␣
 ↪of the product name

reviews_count = final_merged_df.groupby('Name')['Uniq_id'].nunique()

# top 10 products with the most reviews
top_products_10 = reviews_count.sort_values(ascending=False).head(10)

# Visualization
plt.figure(figsize=(10, 6))
top_products_10.plot(kind='bar', color='orange')
plt.title('Top 10 Products with Most Reviews')
plt.xlabel('Product Name')
plt.ylabel('Number of Reviews')
plt.xticks(rotation=90)
plt.show()
```

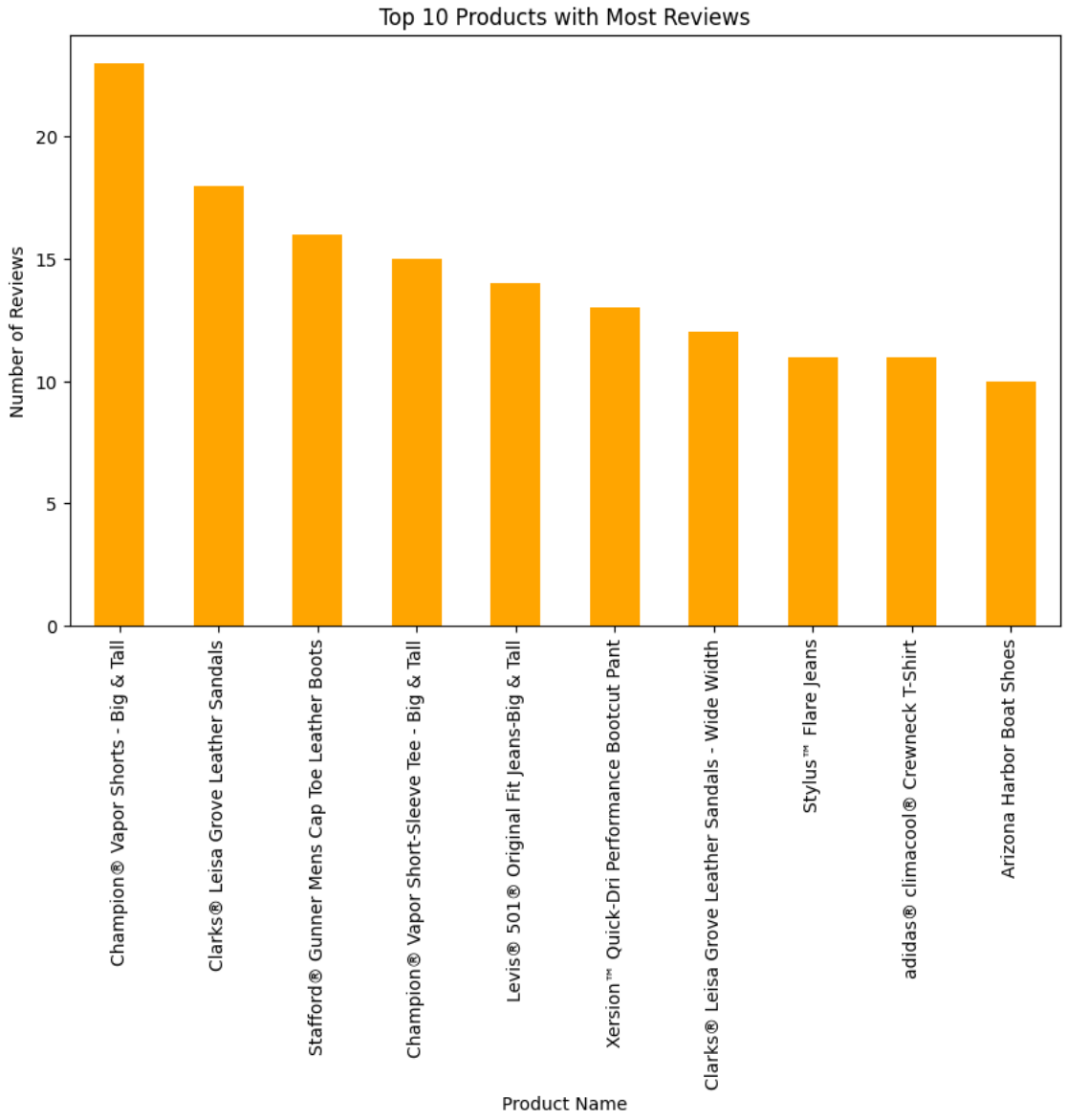
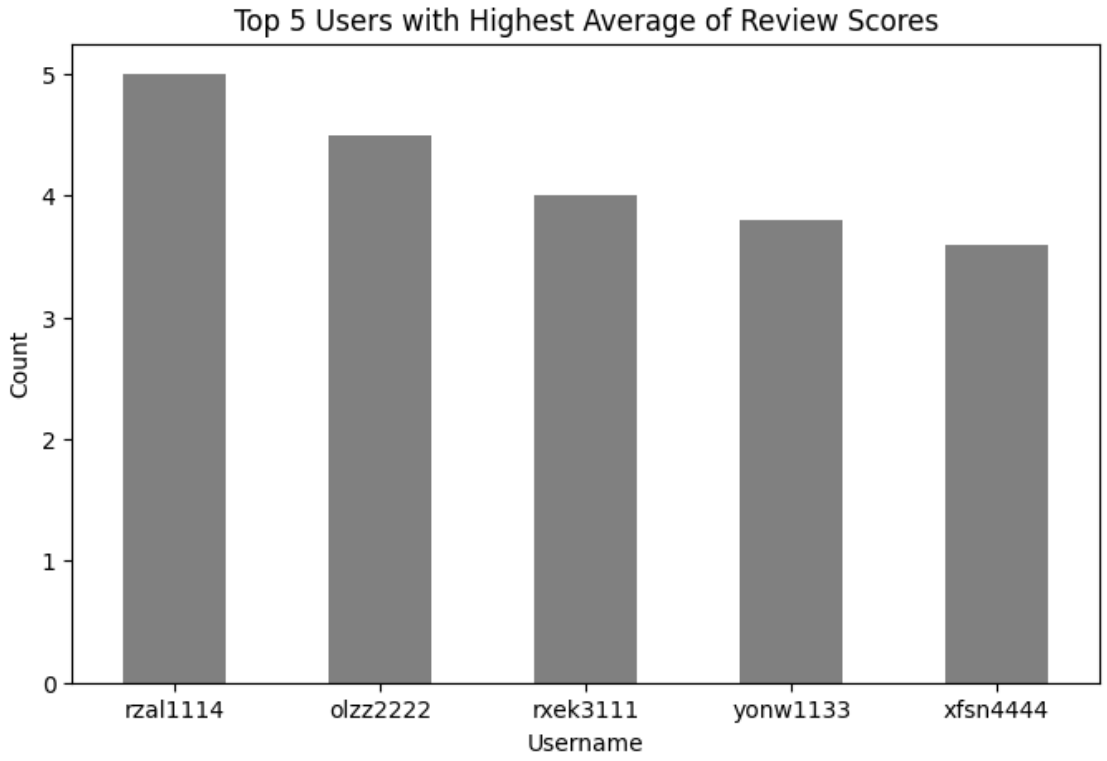


Chart shows the top 5 users that gave the highest review scores (top 5)

[24]:
```
# Users with have highest average of review scores
avg_score_username = final_merged_df.groupby(['Username'])['Score'].mean()

# Top 5 users
top_users_5 = avg_score_username.nlargest(5)

# Visualiazation
plt.figure(figsize=(8, 5))
top_users_5.plot(kind='bar', color='gray')
```

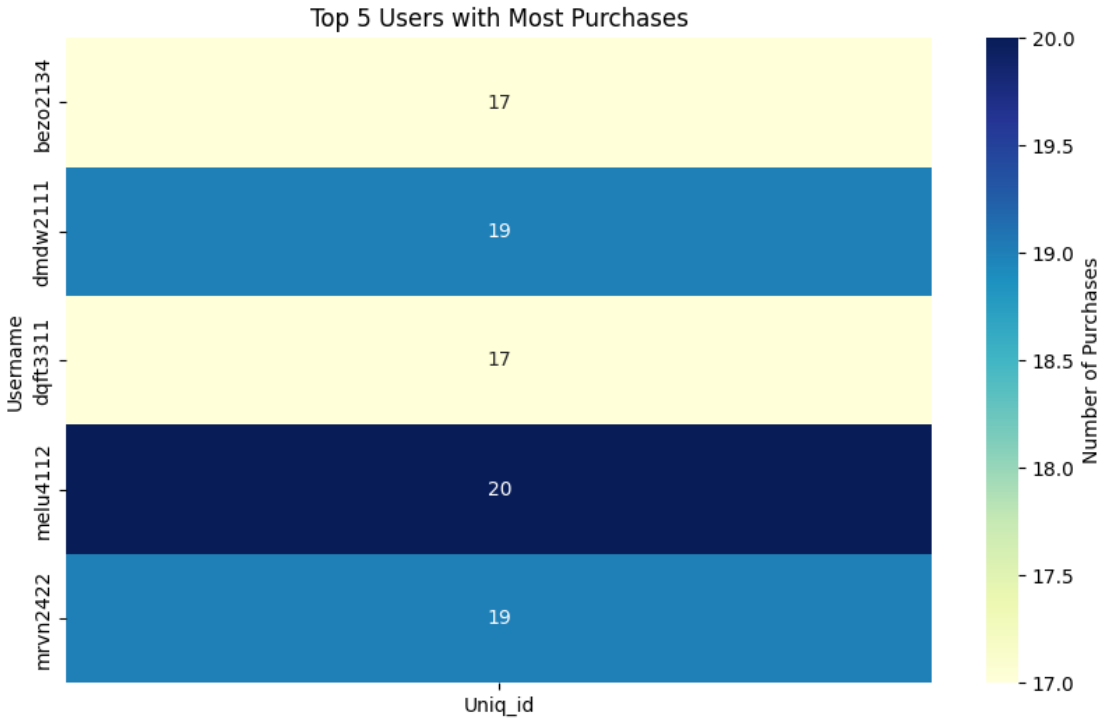
```
plt.title('Top 5 Users with Highest Average of Review Scores')
plt.xlabel('Username')
plt.ylabel('Count')
plt.xticks(rotation=0)
plt.show()
```



Below Graph displays the highest number of purchases by user (top 5)

[25]:
```
# Highest number of purchases per user
user_purchases_count = final_merged_df.groupby('Username')['Uniq_id'].nunique()

# top 5
top_users_5 = user_purchases_count.nlargest(5)

# Visualization
plt.figure(figsize=(10, 6))
sns.heatmap(top_users_5.reset_index().pivot_table(index='Username',␣
 ↪values='Uniq_id', aggfunc='sum'), annot=True, fmt=".0f", cmap="YlGnBu",␣
 ↪cbar_kws={'label': 'Number of Purchases'})
plt.title('Top 5 Users with Most Purchases')
plt.xlabel('')
plt.ylabel('Username')
```

```
plt.show()
```



[25]:
```
from PIL import Image
import urllib.request
import io
```

Below are the most expensive products, their images are used in plots using their provided url (top 4)

[26]:
```
# Most Expensive Products with their images extracted from given url

final_merged_df11 = final_merged_df.copy()
a = final_merged_df11.drop_duplicates(subset='Uniq_id')
b = a.sort_values(by='Price', ascending=False).
  ↪head(4)[['Uniq_id','Price','product_image_urls','Name']]


# Visualization
fig, axes = plt.subplots(1, len(b), figsize=(30, 4))

for i, (index, product) in enumerate(b.iterrows()):
    img_url = product['product_image_urls']
    price = product['Price']
    uniq_id = product['Uniq_id']
```

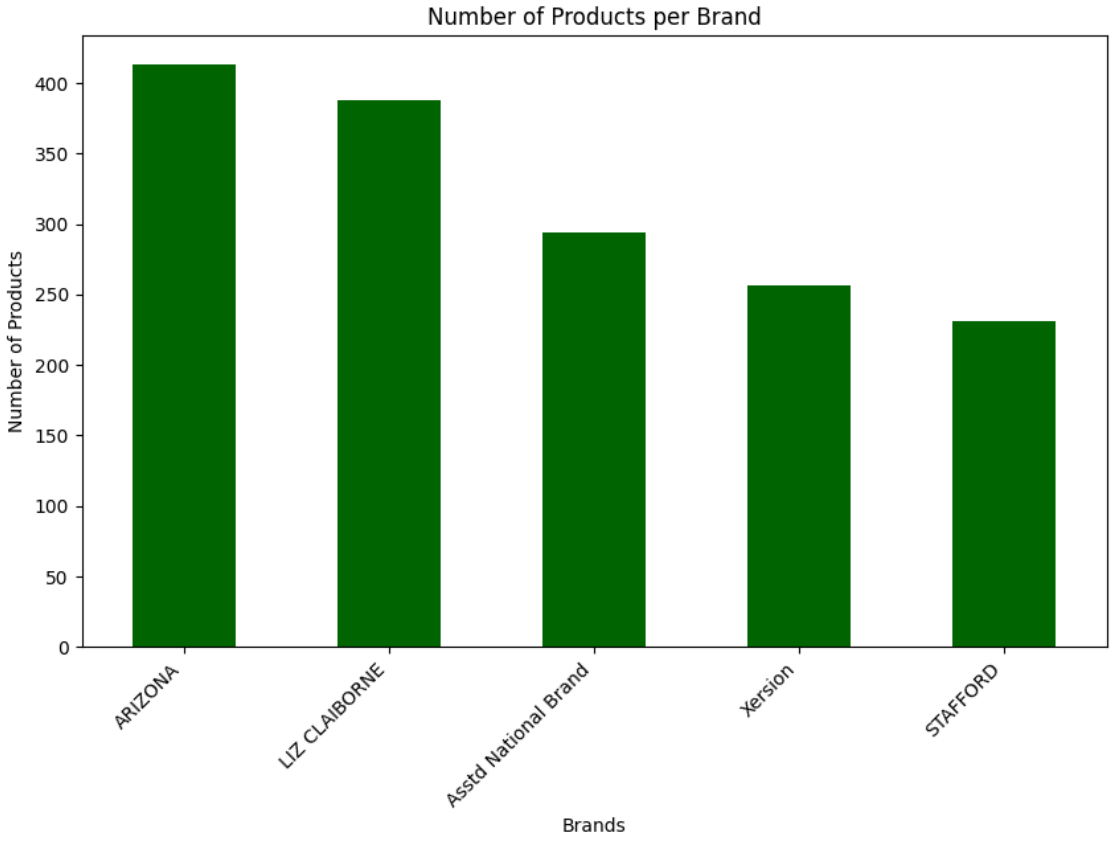
```
    product_name = product['Name']

    # read image url
    img_data = urllib.request.urlopen(img_url).read()
    img = Image.open(io.BytesIO(img_data))

    # ploting image
    axes[i].imshow(img)
    axes[i].set_title(f'\n\nPrice: ${price}\nProduct: {product_name}', pad = 20)
    axes[i].axis('off')

plt.suptitle('Top Most Expensive Products', fontweight='bold', fontsize=24)
plt.subplots_adjust(top=0.7)
plt.show()
```

Last diagram presents the top brands having most products (top 5)

[ ]:
```
# Highest number of products per Brand

final_merged_df11 = final_merged_df.copy()
a = final_merged_df11.drop_duplicates(subset=['Uniq_id', 'brand'])

# top 5
brands_count = a['brand'].value_counts().head(5)
print(brands_count)

# Visualization
plt.figure(figsize=(10, 6))
brands_count.plot(kind='bar', color='darkgreen')
plt.title('Number of Products per Brand')
plt.xlabel('Brands')
plt.ylabel('Number of Products')
plt.xticks(rotation=45, ha='right')
plt.show()
```

```
ARIZONA                 413
LIZ CLAIBORNE           388
Asstd National Brand    294
Xersion                 256
STAFFORD                231
```

```
Name: brand, dtype: int64
```



[ ]: