Scraping Companies/Brands for Categories on Trustpilot

(Trustpilot is a review platform)

Here are the steps we'll follow:

- We're going to scrape https://www.trustpilot.com/categories
- We'll get a list of categories. For each category, we'll get category name and category page URL
- For each category, we'll get the top 20 company reviews and other information from the category page
- For each company, we'll grab the company name, star rating, location and tags
- For each category we'll create a CSV file in the following format:

```
Company Name,Stars,Location,Tags Boomerang Pet ID Tags,5.0,"Pismo Beach, United States",Pet Supply Store
```

Scrape the list of categories

How to do?

- use requests to downlaod the page
- use BS4 to parse and extract information
- convert to a Pandas dataframe

Let's write a function to download the page.

In [1]:
```
import requests
```

In [2]:
```
category_url = 'https://www.trustpilot.com/categories'
```

In [3]:
```
response = requests.get(category_url)
```

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

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

In [5]:
```
len(response.text)
```

Out[5]:
```
353199
```

In [6]:
```
from bs4 import BeautifulSoup
```

In [7]:
```
doc = BeautifulSoup(response.text, 'html.parser')
```

In [8]:
```
type(doc)
```

Out[8]:
```
bs4.BeautifulSoup
```

In [9]:
```
def get_category_names(doc):
    a_class = 'typography_heading-xs__jSwUz typography_appearance-default__AAY17 style
    category_name_ = doc.find_all('h2', {'class': a_class})
    category_name=[]
    for name in category_name_:
        category_name.append(name.text)
    return category_name
```

In [10]:
```
categories_names = get_category_names(doc)
```

In [11]:
```
len(categories_names)
```

Out[11]:
```
22
```

In [12]:
```
categories_names[:5]
```

Out[12]:
```
['Animals & Pets',
 'Beauty & Well-being',
 'Business Services',
 'Construction & Manufacturing',
 'Education & Training']
```

Similarly we have defined functions for URLs.

In [13]:
```
def get_category_urls(doc):
    category_link_ = doc.find_all('a', {'class': 'link_internal__7XN06 link_wrapper__5
    category_urls=[]
    base_url = "https://www.trustpilot.com"
    for category_url in category_link_:
        category_urls.append(base_url + category_url['href'])
    return category_urls
```

In [14]:
```
categories_urls = get_category_urls(doc)
```

In [15]:
```
categories_urls[:5]
```

Out[15]:
```
['https://www.trustpilot.com/categories/animals_pets',
 'https://www.trustpilot.com/categories/beauty_wellbeing',
 'https://www.trustpilot.com/categories/business_services',
 'https://www.trustpilot.com/categories/construction_manufactoring',
 'https://www.trustpilot.com/categories/education_training']
```

Let's put this all together into a single function

In [16]:
```
import pandas as pd

def scrap_categorys():
    category_url = 'https://www.trustpilot.com/categories'
    response = requests.get(category_url)
    if response.status_code != 200:
```

```
        raise Exception('Failed to load page {}'.format(category_url))
    category_dict= {
    'Category': get_category_names(doc),
    'Category_URL' : get_category_urls(doc)
    }
    return pd.DataFrame(category_dict)
```

In [17]:
```
scrap_categorys()
```

Out[17]:

	Category	Category_URL
0	Animals & Pets	https://www.trustpilot.com/categories/animals_...
1	Beauty & Well-being	https://www.trustpilot.com/categories/beauty_w...
2	Business Services	https://www.trustpilot.com/categories/business...
3	Construction & Manufacturing	https://www.trustpilot.com/categories/construc...
4	Education & Training	https://www.trustpilot.com/categories/educatio...
5	Electronics & Technology	https://www.trustpilot.com/categories/electron...
6	Events & Entertainment	https://www.trustpilot.com/categories/events_e...
7	Food, Beverages & Tobacco	https://www.trustpilot.com/categories/food_bev...
8	Health & Medical	https://www.trustpilot.com/categories/health_m...
9	Hobbies & Crafts	https://www.trustpilot.com/categories/hobbies_...
10	Home & Garden	https://www.trustpilot.com/categories/home_garden
11	Home Services	https://www.trustpilot.com/categories/home_ser...
12	Legal Services & Government	https://www.trustpilot.com/categories/legal_se...
13	Media & Publishing	https://www.trustpilot.com/categories/media_pu...
14	Money & Insurance	https://www.trustpilot.com/categories/money_in...
15	Public & Local Services	https://www.trustpilot.com/categories/public_l...
16	Restaurants & Bars	https://www.trustpilot.com/categories/restaura...
17	Shopping & Fashion	https://www.trustpilot.com/categories/shopping...
18	Sports	https://www.trustpilot.com/categories/sports
19	Travel & Vacation	https://www.trustpilot.com/categories/travel_v...
20	Utilities	https://www.trustpilot.com/categories/utilities
21	Vehicles & Transportation	https://www.trustpilot.com/categories/vehicles...

Get the top 20 company reviews & information from a category page

In [18]:
```
import os
import pandas as pd

def get_category(category_urls):
```

```
    response = requests.get(category_urls)
    if response.status_code != 200:
        raise Exception('Failed to load page {}'.format(category_page_url))
    category_doc = BeautifulSoup(response.text, 'html.parser')
    return category_doc

def get_company_info(company_name, stars_, location_, tag_):
    #returns all the required info about company
    name = company_name
    star = parse_star(stars_.text)
    location =location_
    tags  = tag_
    return name,star,location, tags

def get_category_info(category_doc):
    p_tag1 = 'typography_heading-xs__jSwUz typography_appearance-default__AAY17 styles
    company_name = category_doc.find_all('p',{'class': p_tag1})

    b_class = 'typography_body-m__xgxZ_ typography_appearance-subtle__8_H2l styles_tru
    stars_ = category_doc.find_all('span',{'class': b_class})

    c_class = 'typography_body-m__xgxZ_ typography_appearance-subtle__8_H2l styles_met
    location_ = category_doc.find_all('span',{'class': c_class})

    d_class = 'styles_wrapper___E6__ styles_categoriesLabels__FiWQ4 styles_desktop__U5
    tag_ = category_doc.find_all('div',{'class': d_class})

    company_dict = {'Name' : [],'Stars' : [],'Location' : [],'Tags' : []}

    for i in range(len(location_)):
        com_info = get_company_info(company_name[i].text,stars_[i],location_[i].text,
        company_dict['Name'].append(com_info[0])
        company_dict['Stars'].append(com_info[1])
        company_dict['Location'].append(com_info[2])
        company_dict['Tags'].append(com_info[3])

    return pd.DataFrame(company_dict)

def scrape_category(category_urls, path):
    if os.path.exists(path):
        print("The file {} already exists. Skipping...".format(path))
        return

    category_df = get_category_info(get_category(category_urls))
    category_df.to_csv(path, index = None)
```

In [19]:

```
def parse_star(stars_str):
    if stars_str[:11] == 'TrustScore ':
        return float(stars_str[11:])
    return float(stars_str)
```

Putting it all together

- We have a funciton to get the list of categories
- We have a function to create a CSV file for scraped companies from a category page

- Let's create a function to put them together

In [26]:
```
def scrap_category_company():
    print('Scraping list of categories')
    company_df = scrap_categorys()

    ## Create a folder
    os.makedirs('data', exist_ok=True)
    for index, row in company_df.iterrows():
        print('Scraping companies for "{}"'.format(row['Category']))
        scrape_category(row['Category_URL'], 'data/{}.csv'. format(row['Category']))
```

In [27]:
```
scrap_category_company()
```

```
Scraping list of categories
Scraping companies for "Animals & Pets"
Scraping companies for "Beauty & Well-being"
Scraping companies for "Business Services"
Scraping companies for "Construction & Manufacturing"
Scraping companies for "Education & Training"
Scraping companies for "Electronics & Technology"
Scraping companies for "Events & Entertainment"
Scraping companies for "Food, Beverages & Tobacco"
Scraping companies for "Health & Medical"
Scraping companies for "Hobbies & Crafts"
Scraping companies for "Home & Garden"
Scraping companies for "Home Services"
Scraping companies for "Legal Services & Government"
Scraping companies for "Media & Publishing"
Scraping companies for "Money & Insurance"
Scraping companies for "Public & Local Services"
Scraping companies for "Restaurants & Bars"
Scraping companies for "Shopping & Fashion"
Scraping companies for "Sports"
Scraping companies for "Travel & Vacation"
Scraping companies for "Utilities"
Scraping companies for "Vehicles & Transportation"
```

Summary

- I have scraped the website, got data about categories.
- In that categories, i scraped information about top company/brands.
- Collected informations such as company name, stars, location and tags.
- The collected informations are saves in csv files.