0. google drive mount 및 경로 설정

In [31]:
```
import pandas as pd
```

In [32]:
```
from google.colab import drive
drive.mount('/content/drive')
```

```
Drive already mounted at /content/drive; to attempt to forcibly remount, call drive.mount("/content/drive", force_remount=True).
```

In [33]:
```
dpath = "/content/drive/MyDrive/.../"   # 경로 설정 필요
```

1. FURNITURE 전처리

Furniture category in IKEA Saudi Arabia as of 4/20/2020

In [34]:
```
FURNITURE = pd.read_csv(dpath + "IKEA_FURNITURE_raw.csv")
```

In [35]:
```
FURNITURE.head()
```

Out[35]:

	Unnamed: 0	item_id	name	category	price	old_price	sellable_online	link	other_colors	short_description	designer	depth	height	width
0	0	90420332	FREKVENS	Bar furniture	265.0	No old price	True	https://www.ikea.com/sa/en/p/frekvens-bar-tabl...	No	Bar table, in/outdoor, 51x51 cm	Nicholai Wiig Hansen	NaN	99.0	51.0
1	1	368814	NORDVIKEN	Bar furniture	995.0	No old price	False	https://www.ikea.com/sa/en/p/nordviken-bar-tab...	No	Bar table, 140x80 cm	Francis Cayouette	NaN	105.0	80.0
2	2	9333523	NORDVIKEN / NORDVIKEN	Bar furniture	2095.0	No old price	False	https://www.ikea.com/sa/en/p/nordviken-nordvik...	No	Bar table and 4 bar stools	Francis Cayouette	NaN	NaN	NaN
3	3	80155205	STIG	Bar furniture	69.0	No old price	True	https://www.ikea.com/sa/en/p/stig-bar-stool-wi...	Yes	Bar stool with backrest, 74 cm	Henrik Preutz	50.0	100.0	60.0
4	4	30180504	NORBERG	Bar furniture	225.0	No old price	True	https://www.ikea.com/sa/en/p/norberg-wall-moun...	No	Wall-mounted drop-leaf table, ...	Marcus Arvonen	60.0	43.0	74.0

In [36]:
```
# 필요한 칼럼 추출 및 순서 재배치
FURNITURE = FURNITURE[['name', 'price', 'category', 'designer', 'sellable_online', 'depth', 'height', 'width','short_description']]
```

In [37]:
```
# 결측치 확인
FURNITURE.isna().sum()
```

Out[37]:
```
name                  0
price                 0
category              0
designer              0
sellable_online       0
depth              1463
height              988
width               589
short_description     0
dtype: int64
```

In [38]:
```
# 결측치 제거
FURNITURE = FURNITURE.dropna()
```

In [39]:
```
# 특정 data에 , 포함된 행 제거 => csv파일 읽을 시 오류 발생(description 칼럼은 제외)
drop_idx_category = FURNITURE[FURNITURE['category'].str.contains(',')].index
FURNITURE.drop(index = drop_idx_category, inplace = True)

drop_idx_designer = FURNITURE[FURNITURE['designer'].str.contains(',')].index
FURNITURE.drop(index = drop_idx_designer, inplace = True)
```

In [40]:
```
# 전처리 완료된 데이터파일 저장
FURNITURE.to_csv(dpath + "full_FURNITURE.csv", encoding = 'utf8', index= False)
```

2. GROCERY 전처리

The dataset contains the products listed on the website of online grocery store Big Basket.

In [41]:
```
GROCERY = pd.read_csv(dpath + "DMart_GROCERY_raw.csv")
```

In [42]:
```
GROCERY.head()
```

Out[42]:

	Name	Brand	Price	DiscountedPrice	Category	SubCategory	Quantity	Description	BreadCrumbs
0	Premia Badam (Almonds)	Premia	451.0	329.0	Grocery	Grocery/Dry Fruits	500 gm	India	Grocery > Grocery/Dry Fruits
1	Premia Badam (Almonds)	Premia	109.0	85.0	Grocery	Grocery/Dry Fruits	100 gm	India	Grocery > Grocery/Dry Fruits
2	Premia Badam (Almonds)	Premia	202.0	175.0	Grocery	Grocery/Dry Fruits	200 gm	India	Grocery > Grocery/Dry Fruits
3	Nutraj California Almonds (Badam)	Nutraj	599.0	349.0	Grocery	Dry Fruits	500 gm	USA	Grocery > Dry Fruits
4	Nutraj California Almonds (Badam)	Nutraj	1549.0	659.0	Grocery	Dry Fruits	1 kg	USA	Grocery > Dry Fruits

In [43]:
```
# 필요한 칼럼 추출 및 순서 재배치
GROCERY = GROCERY[['Name', 'Price', 'DiscountedPrice', 'SubCategory','Brand', 'Quantity' ,'Description']]
```

In [44]:
```
# 결측치 확인
GROCERY.isna().sum()
```

Out[44]:
```
Name                1
Price               1
DiscountedPrice     1
SubCategory         3
Brand             400
Quantity            1
Description         2
dtype: int64
```

In [45]:
```
# 결측치 제거
GROCERY = GROCERY.dropna()
```

In [46]:
```
# 특정 data에 \n 포함된 행 제거 => csv파일 읽을 시 오류 발생
drop_idx_Description = GROCERY[GROCERY['Description'].str.contains('\n')].index
GROCERY.drop(index = drop_idx_Description, inplace = True)
```

```
/usr/local/lib/python3.8/dist-packages/pandas/core/frame.py:4906: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  return super().drop(
```

In [47]:
```
# 전처리 완료된 데이터파일 저장
GROCERY.to_csv(dpath + "full_GROCERY.csv", encoding = 'utf8', index= False)
```

3. PHONE 전처리

The dataset set contains data about the mobile phones which were released in past 4 years and which can be bought in Ukraine.

In [48]:
```
PHONE = pd.read_csv(dpath + "PHONE_raw.csv")
```

In [49]:
```
PHONE.columns
```

Out[49]:
```
Index(['Unnamed: 0', 'brand_name', 'model_name', 'os', 'popularity',
       'best_price', 'lowest_price', 'highest_price', 'sellers_amount',
       'screen_size', 'memory_size', 'battery_size', 'release_date'],
      dtype='object')
```

In [50]:
```
PHONE.head()
```

Out[50]:

	Unnamed: 0	brand_name	model_name	os	popularity	best_price	lowest_price	highest_price	sellers_amount	screen_size	memory_size	battery_size	release_date
0	0	ALCATEL	1 1/8GB Bluish Black (5033D-2JALUAA)	Android	422	1690.0	1529.0	1819.0	36	5.00	8.0	2000.0	10-2020
1	1	ALCATEL	1 5033D 1/16GB Volcano Black (5033D-2LALUAF)	Android	323	1803.0	1659.0	2489.0	36	5.00	16.0	2000.0	9-2020
2	2	ALCATEL	1 5033D 1/16GB Volcano Black (5033D-2LALUAF)	Android	299	1803.0	1659.0	2489.0	36	5.00	16.0	2000.0	9-2020
3	3	ALCATEL	1 5033D 1/16GB Volcano Black (5033D-2LALUAF)	Android	287	1803.0	1659.0	2489.0	36	5.00	16.0	2000.0	9-2020
4	4	Nokia	1.3 1/16GB Charcoal	Android	1047	1999.0	NaN	NaN	10	5.71	16.0	3000.0	4-2020

In [51]:
```
# 필요한 칼럼 추출 및 순서 재배치
PHONE = PHONE[['model_name', 'best_price', 'lowest_price', 'highest_price', 'brand_name', 'os', 'popularity', 'screen_size','memory_size' ,'battery_size','release_date']]
```

In [52]:
```
PHONE
```

Out[52]:

	model_name	best_price	lowest_price	highest_price	brand_name	os	popularity	screen_size	memory_size	battery_size	release_date
0	1 1/8GB Bluish Black (5033D-2JALUAA)	1690.0	1529.0	1819.0	ALCATEL	Android	422	5.00	8.0	2000.0	10-2020
1	1 5033D 1/16GB Volcano Black (5033D-2LALUAF)	1803.0	1659.0	2489.0	ALCATEL	Android	323	5.00	16.0	2000.0	9-2020
2	1 5033D 1/16GB Volcano Black (5033D-2LALUAF)	1803.0	1659.0	2489.0	ALCATEL	Android	299	5.00	16.0	2000.0	9-2020
3	1 5033D 1/16GB Volcano Black (5033D-2LALUAF)	1803.0	1659.0	2489.0	ALCATEL	Android	287	5.00	16.0	2000.0	9-2020
4	1.3 1/16GB Charcoal	1999.0	NaN	NaN	Nokia	Android	1047	5.71	16.0	3000.0	4-2020
...	...	...	...	...	...	...	...	...	...	...	...
1219	iPhone XS Max 64GB Gold (MT522)	22685.0	16018.0	27900.0	Apple	iOS	1101	6.50	64.0	3174.0	9-2018
1220	iPhone XS Max Dual Sim 64GB Gold (MT732)	24600.0	21939.0	33720.0	Apple	iOS	530	6.50	64.0	3174.0	9-2018
1221	nova 5T 6/128GB Black (51094MEU)	8804.0	7999.0	9999.0	HUAWEI	Android	1174	6.26	128.0	3750.0	11-2019
1222	nubia Red Magic 5G 8/128GB Black	18755.0	18500.0	19010.0	ZTE	Android	752	6.65	128.0	4500.0	10-2020
1223	x-style 35 Screen	907.0	785.0	944.0	Sigma mobile	NaN	952	3.50	NaN	1750.0	1-2020

1224 rows × 11 columns

In [53]:
```
# 결측치 확인
PHONE.isna().sum()
```

Out[53]:
```
model_name          0
best_price          0
lowest_price      260
highest_price     260
brand_name          0
os                197
popularity          0
screen_size         2
memory_size       112
battery_size       10
release_date        0
dtype: int64
```

In [54]:
```
# 결측치 제거
PHONE = PHONE.dropna()
```

In [55]:
```
# 전처리 완료된 데이터파일 저장
PHONE.to_csv(dpath + "full_PHONE.csv", encoding = 'utf8', index= False)
```

4. 최종 전처리 완료된 데이터셋

In [56]:
```
# 전처리 완료된 데이터파일 불러오기
full_FURNITURE = pd.read_csv(dpath + "full_FURNITURE.csv")
full_GROCERY = pd.read_csv(dpath + "full_GROCERY.csv")
full_PHONE = pd.read_csv(dpath + "full_PHONE.csv")
```

In [57]:
```
# test용의 작은 datafile 생성
test_FURNITURE = full_FURNITURE.iloc[:30, :]
test_GROCERY = full_GROCERY.iloc[:30, :]
test_PHONE = full_PHONE.iloc[:30, :]
```

In [58]:
```
# test용의 작은 datafile 저장
test_FURNITURE.to_csv(dpath + "test_FURNITURE.csv", encoding = 'utf8', index= False)
test_GROCERY.to_csv(dpath + "test_GROCERY.csv", encoding = 'utf8', index= False)
test_PHONE.to_csv(dpath + "test_PHONE.csv", encoding = 'utf8', index= False)
```

END