In [1]:
```
import pandas as pd
import time
import yfinance as yf
from random import randint
import os
os.chdir('C:\\Users\\utpala mohapatra\\Documents\\python_folder\\python datasets')
```

Lets load the data

In [2]:
```
df = pd.read_csv('nifty50_tickers - ind_nifty50list.csv')
df.head()
```

Out[2]:

	Symbol
0	ADANIPORTS.NS
1	ASIANPAINT.NS
2	AXISBANK.NS
3	BAJAJ-AUTO.NS
4	BAJFINANCE.NS

Convert the symbols to list

In [3]:
```
index_tickers = df.Symbol.to_list()
index_tickers
```

Out[3]:
```
['ADANIPORTS.NS',
 'ASIANPAINT.NS',
 'AXISBANK.NS',
 'BAJAJ-AUTO.NS',
 'BAJFINANCE.NS',
 'BAJAJFINSV.NS',
 'BPCL.NS',
 'BHARTIARTL.NS',
 'BRITANNIA.NS',
 'CIPLA.NS',
 'COALINDIA.NS',
 'DIVISLAB.NS',
 'DRREDDY.NS',
 'EICHERMOT.NS',
 'GAIL.NS',
 'GRASIM.NS',
 'HCLTECH.NS',
 'HDFCBANK.NS',
 'HDFCLIFE.NS',
 'HEROMOTOCO.NS',
 'HINDALCO.NS',
 'HINDUNILVR.NS',
 'HDFC.NS',
 'ICICIBANK.NS',
 'ITC.NS',
 'IOC.NS',
 'INDUSINDBK.NS',
 'INFY.NS',
 'JSWSTEEL.NS',
 'KOTAKBANK.NS',
 'LT.NS',
 'M&M.NS',
 'MARUTI.NS',
 'NTPC.NS',
 'NESTLEIND.NS',
 'ONGC.NS',
 'POWERGRID.NS',
 'RELIANCE.NS',
 'SBILIFE.NS',
 'SHREECEM.NS',
 'SBIN.NS',
 'SUNPHARMA.NS',
 'TCS.NS',
 'TATAMOTORS.NS',
 'TATASTEEL.NS',
 'TECHM.NS',
 'TITAN.NS',
 'UPL.NS',
 'ULTRACEMCO.NS',
 'WIPRO.NS']
```

Lets check for the Reliance

In [4]:
```
reliance = yf.Ticker('RELIANCE.NS')
reliance_df = reliance.sustainability.T
reliance_df
```

Out[4]:

2022-8	palmOil	controversialWeapons	gambling	socialScore	nuclear	furLeather	alcoholic	gmo	catholic	socialPercentile	...	esgPerformance
Value	False	False	False	10.1	False	False	False	False	True	None	...	OUT_PER

1 rows × 27 columns

A new dataframe esg_data is created. All the tickers' sustainability from the index_tickers list are collected and added to this new dataframe.

In [5]:
```
esg_data = pd.DataFrame()

for ticker in index_tickers:
    print(ticker)
    ticker_name = yf.Ticker(ticker)

    try:
        if ticker_name.sustainability is not None:
            ticker_df = ticker_name.sustainability.T
            ticker_df['Symbol']= ticker
            esg_data = esg_data.append(ticker_df)
            time.sleep(randint(2,8))
    except(IndexError, ValueError) as e:
        print(f'{ticker} did not run')
        pass
```

```
ADANIPORTS.NS
ASIANPAINT.NS
AXISBANK.NS
BAJAJ-AUTO.NS
BAJFINANCE.NS
BAJAJFINSV.NS
BPCL.NS
BHARTIARTL.NS
BRITANNIA.NS
CIPLA.NS
COALINDIA.NS
DIVISLAB.NS
DRREDDY.NS
EICHERMOT.NS
GAIL.NS
GRASIM.NS
HCLTECH.NS
HDFCBANK.NS
HDFCLIFE.NS
HEROMOTOCO.NS
HINDALCO.NS
HINDUNILVR.NS
HDFC.NS
ICICIBANK.NS
ITC.NS
IOC.NS
INDUSINDBK.NS
INFY.NS
JSWSTEEL.NS
KOTAKBANK.NS
LT.NS
M&M.NS
MARUTI.NS
NTPC.NS
NESTLEIND.NS
ONGC.NS
POWERGRID.NS
RELIANCE.NS
SBILIFE.NS
SHREECEM.NS
SBIN.NS
SUNPHARMA.NS
TCS.NS
TATAMOTORS.NS
TATASTEEL.NS
TECHM.NS
TITAN.NS
UPL.NS
ULTRACEMCO.NS
WIPRO.NS
```

The new Dataframe esg_data is shown.

In [6]:
```
esg_data.head()
esg_data.reset_index(drop=True)
```

Out[6]:

2022-8	palmOil	controversialWeapons	gambling	socialScore	nuclear	furLeather	alcoholic	gmo	catholic	socialPercentile	...	coal	pesticid
0	False	False	False	3.15	False	False	False	False	False	None	...	False	Fal
1	False	False	False	9.03	False	False	False	False	False	None	...	False	Fal
2	False	False	False	12.07	False	False	False	False	False	None	...	False	Fal
3	False	False	False	5.58	False	False	False	False	False	None	...	False	Fal
4	False	False	False	11.44	False	False	False	False	False	None	...	False	Fal
5	False	False	False	8.4	False	False	False	False	False	None	...	False	Fal
6	False	False	False	7.92	False	False	False	False	False	None	...	False	Fal
7	False	False	False	18.23	False	False	False	False	True	None	...	False	Fal
8	False	False	False	11.85	False	False	False	False	False	None	...	True	Fal
9	False	False	False	17.16	False	False	False	False	True	None	...	False	Fal
10	False	False	False	3.28	False	False	False	False	False	None	...	False	Fal
11	False	False	False	11.17	False	False	False	False	False	None	...	False	Fal
12	False	False	False	11.21	False	False	False	False	False	None	...	False	Fal
13	False	False	False	6.1	False	False	False	False	False	None	...	False	Fal
14	False	False	False	16.31	False	False	False	False	False	None	...	False	Fal
15	False	False	False	2.97	False	False	False	False	False	None	...	False	Fal
16	False	False	False	9.56	False	False	False	False	False	None	...	False	Fal
17	False	False	False	8.7	False	False	False	False	False	None	...	False	Fal
18	False	False	False	7.89	False	False	False	False	False	None	...	False	Fal
19	False	False	False	11.95	False	False	False	False	False	None	...	False	Fal
20	False	False	False	14.3	False	False	False	False	False	None	...	False	Fal
21	False	False	False	9.19	False	False	False	False	False	None	...	False	Fal
22	False	False	False	7.41	False	False	False	False	False	None	...	False	Fal
23	False	False	False	11.98	False	False	False	False	False	0	...	False	Fal
24	False	False	False	10.91	False	False	False	False	False	0	...	False	Fal
25	False	True	False	13.86	False	False	False	False	True	0	...	False	Fal
26	False	False	False	13.41	False	False	False	False	False	0	...	False	Fal
27	False	False	False	10.85	False	False	False	False	False	0	...	False	Fal
28	False	False	False	13.45	False	False	False	False	False	0	...	True	Fal
29	False	False	False	10.63	False	False	False	False	False	0	...	False	Fal
30	False	False	False	17.74	False	False	False	False	False	0	...	False	Fal
31	False	False	False	10.1	False	False	False	False	True	0	...	False	Fal
32	False	False	False	4.19	False	False	False	False	False	0	...	True	Fal
33	False	False	False	13.74	False	False	False	False	False	0	...	False	Fal
34	False	False	False	20.16	False	False	False	False	True	0	...	False	Fal
35	False	False	False	6.58	False	False	False	False	False	0	...	False	Fal
36	False	False	False	14.01	False	False	False	False	False	0	...	False	Fal
37	False	False	False	15.4	False	False	False	False	True	0	...	True	Fal
38	False	False	False	6.13	False	False	False	False	False	0	...	False	Fal
39	False	False	False	7.09	False	False	False	False	False	0	...	False	Fal
40	False	False	False	6.92	False	False	False	False	False	0	...	False	Tr
41	False	False	False	6.67	False	False	False	False	False	0	...	False	Fal
42	False	False	False	6.75	False	False	False	False	False	None	...	False	Fal

43 rows × 28 columns

Some of the tickers from the Index tickers lists could not added to the esg_data because of no sustainabilty information , those tickers are shown below.

In [7]:
```
esg_tickers = esg_data['Symbol']
no_esg_data = list(set(index_tickers) - set(esg_tickers))
print(no_esg_data)
```

```
['BAJFINANCE.NS', 'DIVISLAB.NS', 'HDFCLIFE.NS', 'SBILIFE.NS', 'POWERGRID.NS', 'INDUSINDBK.NS', 'BRITANNIA.NS']
```

In [8]:
```
esg_data.columns
```

Out[8]:
```
Index(['palmOil', 'controversialWeapons', 'gambling', 'socialScore', 'nuclear',
       'furLeather', 'alcoholic', 'gmo', 'catholic', 'socialPercentile',
       'peerCount', 'governanceScore', 'environmentPercentile',
       'animalTesting', 'tobacco', 'totalEsg', 'highestControversy',
       'esgPerformance', 'coal', 'pesticides', 'adult', 'percentile',
       'peerGroup', 'smallArms', 'environmentScore', 'governancePercentile',
       'militaryContract', 'Symbol'],
      dtype='object', name='2022-8')
```

In [9]:
```
new_esg_df = esg_data[['Symbol','socialScore','governanceScore','totalEsg','environmentScore']]
new_esg_df.reset_index(drop=True).head()
```

Out[9]:

2022-8	Symbol	socialScore	governanceScore	totalEsg	environmentScore
0	ADANIPORTS.NS	3.15	5.62	12.37	3.61
1	ASIANPAINT.NS	9.03	8.07	30.56	13.46
2	AXISBANK.NS	12.07	12.13	26.15	1.94
3	BAJAJ-AUTO.NS	5.58	7.35	16.31	3.38
4	BAJAJFINSV.NS	11.44	13.59	27.31	2.27

In [19]:
```
new_esg_df.rename(columns={'Symbol':'symbol'},inplace=True)
```

```
C:\Users\utpala mohapatra\anaconda3\lib\site-packages\pandas\core\frame.py:4441: 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#ret
urning-a-view-versus-a-copy
  return super().rename(
```

new_esg_df is the dataframe created from esg_data showing our useful columns only like socialscore, governencescore,totalesg and environmentscore.

In [20]:
```
new_esg_df.head()
```

Out[20]:

2022-8	symbol	socialScore	governanceScore	totalEsg	environmentScore
Value	ADANIPORTS.NS	3.15	5.62	12.37	3.61
Value	ASIANPAINT.NS	9.03	8.07	30.56	13.46
Value	AXISBANK.NS	12.07	12.13	26.15	1.94
Value	BAJAJ-AUTO.NS	5.58	7.35	16.31	3.38
Value	BAJAJFINSV.NS	11.44	13.59	27.31	2.27

Showing the Reliance ticker information.

In [10]:
```
reliance.info
```

Out[10]:

```
{'zip': '400021',
 'sector': 'Energy',
 'fullTimeEmployees': 236334,
 'longBusinessSummary': 'Reliance Industries Limited engages in hydrocarbon exploration and production, petrole
um refining and marketing, petrochemicals, textile, retail, digital, and financial services businesses worldwid
e. The company produces and markets petroleum products, such as liquefied petroleum gas, propylene, naphtha, ga
soline, jet/aviation turbine fuel, kerosene oil, diesel, Sulphur, and petroleum coke. It also provides petroche
micals, including high-density and low-density polyethylene (PE), linear low density PE, polyester fibers and y
arns, polypropylene, polyvinyl chloride, polyester yarn, purified terephthalic acid, ethylene glycols and oxid
e, paraxylene, ortho xylene, benzene, linear alkyl benzene and paraffin, poly butadiene rubber, styrene butadie
ne rubber, butyl rubber, and polyethylene terephthalate. In addition, the company manufactures and markets yarn
s, fabrics, apparel, and auto furnishings; explores, develops, and produces crude oil and natural gas; and oper
ates various stores comprising neighborhood, supermarket, hypermarket, wholesale cash and carry, specialty, onl
ine, pharmacy, and grocery stores, as well as stores that offer apparel, beauty and cosmetics, accessories, foo
twear, consumer electronics, connectivity products, and others. As of March 31, 2021, it operated 12,711 retail
 stores. Further, it provides range of digital services under the Jio brand name; and non-banking financial and
 insurance broking services. Further, it operates news and entertainment platforms, finance portals, fashion and
 lifestyle portals, and online ticket booking platforms, as well as Network18 and television channels; publishes
 magazines; and offers highway hospitality and fleet management services. The company was incorporated in 1973 a
nd is based in Mumbai, India.',
 'city': 'Mumbai',
 'phone': '91 22 3555 5000',
 'country': 'India',
 'companyOfficers': [],
 'website': 'https://www.ril.com',
 'maxAge': 1,
 'address1': 'Maker Chambers IV',
 'fax': '91 22 2204 2268',
 'industry': 'Oil & Gas Refining & Marketing',
 'address2': '3rd Floor 222 Nariman Point',
 'ebitdaMargins': 0.15483,
 'profitMargins': 0.089899994,
 'grossMargins': 0.32514,
 'operatingCashflow': None,
 'revenueGrowth': 0.57,
 'operatingMargins': 0.1137,
 'ebitda': 994277523456,
 'targetLowPrice': 1350,
 'recommendationKey': 'buy',
 'grossProfits': 1722910000000,
 'freeCashflow': None,
 'targetMedianPrice': 2280,
 'currentPrice': 2598,
 'earningsGrowth': 0.393,
 'currentRatio': None,
 'returnOnAssets': None,
 'numberOfAnalystOpinions': 29,
 'targetMeanPrice': 2272.1,
 'debtToEquity': 31.21,
 'returnOnEquity': None,
 'targetHighPrice': 2830,
 'totalCash': 2197629960192,
 'totalDebt': 2639180070912,
 'totalRevenue': 6421619933184,
 'totalCashPerShare': 332.683,
 'financialCurrency': 'INR',
 'revenuePerShare': 987.63,
 'quickRatio': None,
 'recommendationMean': 2.1,
 'exchange': 'NSI',
 'shortName': 'RELIANCE INDS',
 'longName': 'Reliance Industries Limited',
 'exchangeTimezoneName': 'Asia/Kolkata',
 'exchangeTimezoneShortName': 'IST',
 'isEsgPopulated': False,
 'gmtOffSetMilliseconds': '19800000',
 'quoteType': 'EQUITY',
 'symbol': 'RELIANCE.NS',
 'messageBoardId': 'finmb_878373',
 'market': 'in_market',
 'annualHoldingsTurnover': None,
 'enterpriseToRevenue': 2.855,
 'beta3Year': None,
 'enterpriseToEbitda': 18.441,
 '52WeekChange': 0.20880055,
 'morningStarRiskRating': None,
 'forwardEps': 71.85,
 'revenueQuarterlyGrowth': None,
 'sharesOutstanding': 6765989888,
 'fundInceptionDate': None,
 'annualReportExpenseRatio': None,
 'totalAssets': None,
 'bookValue': 1132.233,
 'sharesShort': None,
 'sharesPercentSharesOut': None,
 'fundFamily': None,
 'lastFiscalYearEnd': 1617148800,
 'heldPercentInstitutions': 0.25726998,
 'netIncomeToCommon': 540959997952,
 'trailingEps': 87.323,
 'lastDividendValue': 8,
 'SandP52WeekChange': -0.056043804,
 'priceToBook': 2.2945807,
 'heldPercentInsiders': 0.49661,
 'nextFiscalYearEnd': 1680220800,
 'yield': None,
 'mostRecentQuarter': 1640908800,
 'shortRatio': None,
 'sharesShortPreviousMonthDate': None,
 'floatShares': 3024055096,
 'beta': 0.955614,
 'enterpriseValue': 18335190220800,
 'priceHint': 2,
 'threeYearAverageReturn': None,
 'lastSplitDate': 1504742400,
 'lastSplitFactor': '2:1',
 'legalType': None,
 'lastDividendDate': 1660780800,
 'morningStarOverallRating': None,
 'earningsQuarterlyGrowth': 0.416,
 'priceToSalesTrailing12Months': 2.737322,
 'dateShortInterest': None,
 'pegRatio': None,
 'ytdReturn': None,
 'forwardPE': 36.158665,
 'lastCapGain': None,
 'shortPercentOfFloat': None,
 'sharesShortPriorMonth': None,
 'impliedSharesOutstanding': 0,
 'category': None,
 'fiveYearAverageReturn': None,
 'previousClose': 2613.85,
 'regularMarketOpen': 2617,
 'twoHundredDayAverage': 2499.7485,
 'trailingAnnualDividendYield': 0,
 'payoutRatio': 0.0802,
 'volume24Hr': None,
 'regularMarketDayHigh': 2631.3,
 'navPrice': None,
 'averageDailyVolume10Day': 4142698,
 'regularMarketPreviousClose': 2613.85,
 'fiftyDayAverage': 2518.314,
 'trailingAnnualDividendRate': 0,
 'open': 2617,
 'toCurrency': None,
 'averageVolume10days': 4142698,
 'expireDate': None,
 'algorithm': None,
 'dividendRate': 7,
 'exDividendDate': 1623369600,
 'circulatingSupply': None,
 'startDate': None,
 'regularMarketDayLow': 2586,
 'currency': 'INR',
 'trailingPE': 29.751612,
 'regularMarketVolume': 1950159,
 'lastMarket': None,
 'maxSupply': None,
 'openInterest': None,
 'marketCap': 17578041802752,
 'volumeAllCurrencies': None,
 'strikePrice': None,
 'averageVolume': 7933843,
 'dayLow': 2586,
 'ask': 2598,
 'askSize': 0,
 'volume': 1950159,
 'fiftyTwoWeekHigh': 2856.15,
 'fromCurrency': None,
 'fiveYearAvgDividendYield': None,
 'fiftyTwoWeekLow': 2132.3,
 'bid': 2597.9,
 'tradeable': False,
 'dividendYield': 0.0027,
 'bidSize': 0,
 'dayHigh': 2631.3,
 'coinMarketCapLink': None,
 'regularMarketPrice': 2598,
 'preMarketPrice': None,
 'logo_url': 'https://logo.clearbit.com/ril.com'}
```


A new dataframe is created called main_df. All the tickers' info from the index_tickers list is added to this dataframe.

In [11]:
```
main_df = pd.DataFrame()

for ticker in index_tickers:
    ticker_name = yf.Ticker(ticker)

    try:
        ticker_info = ticker_name.info
        ticker_df = pd.DataFrame.from_dict(ticker_info.items()).T
        ticker_df.columns = ticker_df.iloc[0]
        ticker_df = ticker_df.drop(ticker_df.index[0])
        main_df = main_df.append(ticker_df)
        time.sleep(randint(2,8))
        print(f'{ticker} + complete')
    except(IndexError,ValueError) as e:
        print(f'{ticker} + data not found')
```

```
ADANIPORTS.NS + complete
ASIANPAINT.NS + complete
AXISBANK.NS + complete
BAJAJ-AUTO.NS + complete
BAJFINANCE.NS + complete
BAJAJFINSV.NS + complete
BPCL.NS + complete
BHARTIARTL.NS + complete
BRITANNIA.NS + complete
CIPLA.NS + complete
COALINDIA.NS + complete
DIVISLAB.NS + complete
DRREDDY.NS + complete
EICHERMOT.NS + complete
GAIL.NS + complete
GRASIM.NS + complete
HCLTECH.NS + complete
HDFCBANK.NS + complete
HDFCLIFE.NS + complete
HEROMOTOCO.NS + complete
HINDALCO.NS + complete
HINDUNILVR.NS + complete
HDFC.NS + complete
ICICIBANK.NS + complete
ITC.NS + complete
IOC.NS + complete
INDUSINDBK.NS + complete
INFY.NS + complete
JSWSTEEL.NS + complete
KOTAKBANK.NS + complete
LT.NS + complete
M&M.NS + complete
MARUTI.NS + complete
NTPC.NS + complete
NESTLEIND.NS + complete
ONGC.NS + complete
POWERGRID.NS + complete
RELIANCE.NS + complete
SBILIFE.NS + complete
SHREECEM.NS + complete
SBIN.NS + complete
SUNPHARMA.NS + complete
TCS.NS + complete
TATAMOTORS.NS + complete
TATASTEEL.NS + complete
TECHM.NS + complete
TITAN.NS + complete
UPL.NS + complete
ULTRACEMCO.NS + complete
WIPRO.NS + complete
```

A new dataframe filtered_df is created from main_df showing only the useful columns like symbol,sector,previousclose and shreoutstanding.

In [12]:
```
filtered_df = main_df[['symbol','sector','previousClose','sharesOutstanding']]
filtered_df.head()
```

Out[12]:

	symbol	sector	previousClose	sharesOutstanding
1	ADANIPORTS.NS	Industrials	871.8	2112370048
1	ASIANPAINT.NS	Basic Materials	3482.55	959198016
1	AXISBANK.NS	Financial Services	757.5	3071589888
1	BAJAJ-AUTO.NS	Consumer Cyclical	4074.55	286739008
1	BAJFINANCE.NS	Financial Services	7301.7	605428992

A new column newMarketCap is created showing the product of the previousClose and shareOutstanding.Then another column marketWeight is created showing the percentage of newMarketCap.

In [15]:
```
filtered_df['newMarketCap'] = filtered_df['previousClose'] * filtered_df['sharesOutstanding']
total_index_mcap = filtered_df['newMarketCap'].sum()
filtered_df['marketWeight'] = (filtered_df['newMarketCap']/total_index_mcap) * 100
```

```
<ipython-input-15-66fdf2382c41>:1: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#ret
urning-a-view-versus-a-copy
  filtered_df['newMarketCap'] = filtered_df['previousClose'] * filtered_df['sharesOutstanding']
```

In [17]:
```
filtered_df.head()
```

Out[17]:

	symbol	sector	previousClose	sharesOutstanding	newMarketCap	marketWeight
1	ADANIPORTS.NS	Industrials	871.8	2112370048	1841564207846.399902	1.332042
1	ASIANPAINT.NS	Basic Materials	3482.55	959198016	3340455050620.800293	2.41622
1	AXISBANK.NS	Financial Services	757.5	3071589888	2326729340160.0	1.682971
1	BAJAJ-AUTO.NS	Consumer Cyclical	4074.55	286739008	1168332425046.400146	0.845079
1	BAJFINANCE.NS	Financial Services	7301.7	605428992	4420660870886.399414	3.197555

Both the filtered_df and new_esg_df is merged where all the data of filtered_df is added and a new dataframe is created called final_df.

In [21]:
```
final_df = filtered_df.merge(new_esg_df,how='left',on='symbol')
final_df.head()
```

Out[21]:

	symbol	sector	previousClose	sharesOutstanding	newMarketCap	marketWeight	socialScore	governanceScore	totalEsg
0	ADANIPORTS.NS	Industrials	871.8	2112370048	1841564207846.399902	1.332042	3.15	5.62	12.37
1	ASIANPAINT.NS	Basic Materials	3482.55	959198016	3340455050620.800293	2.41622	9.03	8.07	30.56
2	AXISBANK.NS	Financial Services	757.5	3071589888	2326729340160.0	1.682971	12.07	12.13	26.15
3	BAJAJ-AUTO.NS	Consumer Cyclical	4074.55	286739008	1168332425046.400146	0.845079	5.58	7.35	16.31
4	BAJFINANCE.NS	Financial Services	7301.7	605428992	4420660870886.399414	3.197555	NaN	NaN	NaN

Showing the data whose totalEsg is missing.

In [22]:
```
final_df[~final_df['totalEsg'].notnull()]
```

Out[22]:

	symbol	sector	previousClose	sharesOutstanding	newMarketCap	marketWeight	socialScore	governanceScore	totalEsg
4	BAJFINANCE.NS	Financial Services	7301.7	605428992	4420660870886.399414	3.197555	NaN	NaN	NaN
8	BRITANNIA.NS	Consumer Defensive	3662.55	240868000	882191093400.0	0.638107	NaN	NaN	NaN
11	DIVISLAB.NS	Healthcare	3696.2	265468992	981226488230.399902	0.709741	NaN	NaN	NaN
18	HDFCLIFE.NS	Financial Services	576.65	2111810048	1217775264179.199951	0.880842	NaN	NaN	NaN
26	INDUSINDBK.NS	Financial Services	1063.3	774548992	823577943193.599976	0.595711	NaN	NaN	NaN
36	POWERGRID.NS	Utilities	228.15	6975450112	1591448943052.800049	1.151128	NaN	NaN	NaN
38	SBILIFE.NS	Financial Services	1296.95	1000280000	1297313146000.0	0.938373	NaN	NaN	NaN

We can see above missing values for some of the company.Those values can be imputed by finding average of socialscore, governancescore, totalesg and environmentscore sectorwise.

Next the marketweightedEsg, marketweightedenvironmentScore,marketweightedSocialScore and marketweightedGovernanceScore is calculated with respect to the marketweight calulated before.

In [24]:
```
final_esg_df= pd.DataFrame()
sector_list = final_df['sector'].unique().tolist()

for sector in sector_list:
    sector_df = final_df[final_df['sector']==sector]
    sector_df['socialScore'].fillna(round(sector_df['socialScore'].mean(),2),inplace= True)
    sector_df['governanceScore'].fillna(round(sector_df['governanceScore'].mean(),2),inplace= True)
    sector_df['totalEsg'].fillna(round(sector_df['totalEsg'].mean(),2),inplace= True)
    sector_df['environmentScore'].fillna(round(sector_df['environmentScore'].mean(),2),inplace= True)

    final_esg_df = final_esg_df.append(sector_df)

final_esg_df['mktweightedEsg'] = (final_esg_df['marketWeight'] * final_esg_df['totalEsg'])/100
final_esg_df['mktweightedEnvScore'] = (final_esg_df['marketWeight'] * final_esg_df['environmentScore'])/100
final_esg_df['mktweightedSocScore'] = (final_esg_df['marketWeight'] * final_esg_df['socialScore'])/100
final_esg_df['mktweightedGovScore'] = (final_esg_df['marketWeight'] * final_esg_df['governanceScore'])/100
```

```
C:\Users\utpala mohapatra\anaconda3\lib\site-packages\pandas\core\series.py:4463: 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#ret
urning-a-view-versus-a-copy
  return super().fillna(
```

Showing the Total Environment score, Governance score, Social score and ESG score.

In [25]:
```
print('Total Environment Score : {}'.format(round(final_esg_df['mktweightedEnvScore'].sum(),2)))
print('Total Governance Score : {}'.format(round(final_esg_df['mktweightedGovScore'].sum(),2)))
print('Total Social Score : {}'.format(round(final_esg_df['mktweightedSocScore'].sum(),2)))
print('Total EsgScore : {}'.format(round(final_esg_df['mktweightedEsg'].sum(),2)))
```

```
Total Environment Score : 7.26
Total Governance Score : 8.89
Total Social Score : 10.47
Total EsgScore : 26.63
```

In [ ]:

In [ ]:

In [ ]:

In [ ]: