In [19]:
```
import pandas as pd
```

In [20]:
```
import numpy as np
```

In [21]:
```
fortune=pd.read_csv(r"C:\Users\rajkp\Downloads\fortune1000 (3).csv")
```

In [22]:
```
fortune
```

Out[22]:

	Rank	Company	Sector	Industry	Revenue	Profits	Employees
0	1	Walmart	Retailing	General Merchandisers	482130	14694	2300000
1	2	Exxon Mobil	Energy	Petroleum Refining	246204	16150	75600
2	3	Apple	Technology	Computers, Office Equipment	233715	53394	110000
3	4	Berkshire Hathaway	Financials	Insurance: Property and Casualty (Stock)	210821	24083	331000
4	5	McKesson	Health Care	Wholesalers: Health Care	181241	1476	70400
...	...	...	...	...	...	...	...
995	996	New York Community Bancorp	Financials	Commercial Banks	1902	-47	3448
996	997	Portland General Electric	Energy	Utilities: Gas and Electric	1898	172	2646
997	997	Portland General Electric	Energy	Utilities: Gas and Electric	1898	172	2646
998	999	Wendy's	Hotels, Resturants & Leisure	Food Services	1896	161	21200
999	1000	Briggs & Stratton	Industrials	Industrial Machinery	1895	46	5480

1000 rows × 7 columns

In [24]:
```
fortune=pd.read_csv(r"C:\Users\rajkp\Downloads\fortune1000 (3).csv",index_col="R
```

In [25]:
```
fortune
```

Out[25]:

Rank	Company	Sector	Industry	Revenue	Profits	Employees
1	Walmart	Retailing	General Merchandisers	482130	14694	2300000
2	Exxon Mobil	Energy	Petroleum Refining	246204	16150	75600
3	Apple	Technology	Computers, Office Equipment	233715	53394	110000
4	Berkshire Hathaway	Financials	Insurance: Property and Casualty (Stock)	210821	24083	331000
5	McKesson	Health Care	Wholesalers: Health Care	181241	1476	70400
...	...	...	...	...	...	...
996	New York Community Bancorp	Financials	Commercial Banks	1902	-47	3448
997	Portland General Electric	Energy	Utilities: Gas and Electric	1898	172	2646
997	Portland General Electric	Energy	Utilities: Gas and Electric	1898	172	2646
999	Wendy's	Hotels, Resturants & Leisure	Food Services	1896	161	21200
1000	Briggs & Stratton	Industrials	Industrial Machinery	1895	46	5480

1000 rows × 6 columns

In [35]:
```
sectors=fortune.groupby("Sector")
```

In [36]:
```
sectors
```

Out[36]:
```
<pandas.core.groupby.generic.DataFrameGroupBy object at 0x00000251B4BF7F50>
```

In [37]:
```
len(sectors)
```

Out[37]:
```
21
```

In [38]:
```
sectors.first()
```

Out[38]:

Sector	Company	Industry	Revenue	Profits	Employees
Aerospace & Defense	Boeing	Aerospace and Defense	96114	5176	161400
Apparel	Nike	Apparel	30601	3273	62600
Business Services	ManpowerGroup	Temporary Help	19330	419	27000
Chemicals	Dow Chemical	Chemicals	48778	7685	49495
Energy	Exxon Mobil	Petroleum Refining	246204	16150	75600
Engineering & Construction	Fluor	Engineering, Construction	18114	413	38758
Financials	Berkshire Hathaway	Insurance: Property and Casualty (Stock)	210821	24083	331000
Food and Drug Stores	CVS Health	Food and Drug Stores	153290	5237	199000
Food, Beverages & Tobacco	Archer Daniels Midland	Food Production	67702	1849	32300
Health Care	McKesson	Wholesalers: Health Care	181241	1476	70400
Hotels, Resturants & Leisure	McDonald's	Food Services	25413	4529	420000
Household Products	Procter & Gamble	Household and Personal Products	78756	7036	110000
Industrials	General Electric	Industrial Machinery	140389	-6126	333000
Materials	Alcoa	Metals	22534	-322	60000
Media	Disney	Entertainment	52465	8382	185000
Motor Vehicles & Parts	General Motors	Motor Vehicles and Parts	152356	9687	215000
Retailing	Walmart	General Merchandisers	482130	14694	2300000
Technology	Apple	Computers, Office Equipment	233715	53394	110000
Telecommunications	AT&T	Telecommunications	146801	13345	281450
Transportation	UPS	Mail, Package, and Freight Delivery	58363	4844	341240
Wholesalers	Sysco	Wholesalers: Food and Grocery	48681	687	51700

In [39]:
```
sector.last()
```

Out[39]:

Sector	Company	Industry	Revenue	Profits	Employees
Aerospace & Defense	Delta Tucker Holdings	Aerospace and Defense	1923	-133	12000
Apparel	Guess	Apparel	2204	82	13500
Business Services	DeVry Education Group	Education	1910	140	11770
Chemicals	H.B. Fuller	Chemicals	2084	87	4425
Energy	Portland General Electric	Utilities: Gas and Electric	1898	172	2646
Engineering & Construction	MDC Holdings	Homebuilders	1909	66	1225
Financials	New York Community Bancorp	Commercial Banks	1902	-47	3448
Food and Drug Stores	Fred's	Food and Drug Stores	2151	-7	7103
Food, Beverages & Tobacco	Alliance One International	Tobacco	2066	-15	6835
Health Care	Providence Service	Health Care: Pharmacy and Other Services	1987	84	9072
Hotels, Resturants & Leisure	Wendy's	Food Services	1896	161	21200
Household Products	Revlon	Household and Personal Products	1914	56	5700
Industrials	Briggs & Stratton	Industrial Machinery	1895	46	5480
Materials	Schnitzer Steel Industries	Metals	1924	-197	2955
Media	Cenveo	Publishing, Printing	1921	-31	7300
Motor Vehicles & Parts	Allison Transmission Holdings	Motor Vehicles and Parts	1986	182	2700
Retailing	99 Cents Only Stores	Specialty Retailers: Other	1999	-232	18200
Technology	F5 Networks	Network and Other Communications Equipment	1920	365	4178
Telecommunications	Equinix	Telecommunications	2726	188	5042

Sector	Company	Industry	Revenue	Profits	Employees
Transportation	Roadrunner Transportation Systems	Transportation and Logistics	1995	48	4502
Wholesalers	BlueLinx Holdings	Wholesalers: Diversified	1917	-12	1600

In [43]:
```
sectors.get_group
```

Out[43]:
```
<bound method BaseGroupBy.get_group of <pandas.core.groupby.generic.DataFrameGr
oupBy object at 0x00000251B4BF7F50>>
```

In [45]:
```
sectors.size()
```

Out[45]:
```
Sector
Aerospace & Defense              20
Apparel                          15
Business Services                51
Chemicals                        30
Energy                          122
Engineering & Construction       26
Financials                      139
Food and Drug Stores             15
Food, Beverages & Tobacco        43
Health Care                      75
Hotels, Resturants & Leisure     25
Household Products               28
Industrials                      46
Materials                        43
Media                            25
Motor Vehicles & Parts           24
Retailing                        80
Technology                      102
Telecommunications               15
Transportation                   36
Wholesalers                      40
dtype: int64
```

In [48]:
```
sector.get_group("Energy")
```

Out[48]:

Rank	Company	Sector	Industry	Revenue	Profits	Employees
2	Exxon Mobil	Energy	Petroleum Refining	246204	16150	75600
14	Chevron	Energy	Petroleum Refining	131118	4587	61500
30	Phillips 66	Energy	Petroleum Refining	87169	4227	14000
32	Valero Energy	Energy	Petroleum Refining	81824	3990	10103
42	Marathon Petroleum	Energy	Petroleum Refining	64566	2852	45440
...	...	...	...	...	...	...
981	WPX Energy	Energy	Mining, Crude-Oil Production	1958	-1727	1040
983	Adams Resources & Energy	Energy	Petroleum Refining	1944	-1	809
995	EP Energy	Energy	Mining, Crude-Oil Production	1908	-3748	665
997	Portland General Electric	Energy	Utilities: Gas and Electric	1898	172	2646
997	Portland General Electric	Energy	Utilities: Gas and Electric	1898	172	2646

122 rows × 6 columns

In [55]:
```
sector["Profits"].sum()
```

Out[55]:
```
Sector
Aerospace & Defense              28742
Apparel                           8236
Business Services                28227
Chemicals                        22628
Energy                          -73447
Engineering & Construction        5304
Financials                      260209
Food and Drug Stores             16759
Food, Beverages & Tobacco        51417
Health Care                     106114
Hotels, Resturants & Leisure     20697
Household Products               14428
Industrials                      20764
Materials                         4428
Media                            24347
Motor Vehicles & Parts           25898
Retailing                        47830
Technology                      180473
Telecommunications               48637
Transportation                   44169
Wholesalers                       8233
Name: Profits, dtype: int64
```

In [57]:
```
sectors.agg({})
```

```
---------------------------------------------------------------------------
ValueError                                Traceback (most recent call last)
Cell In[57], line 1
----> 1 sectors.agg({})

File ~\anaconda3\Lib\site-packages\pandas\core\groupby\generic.py:1432, in DataFrameGroupBy.aggregate(self, func, engine, engine_kwargs, *args, **kwargs)
   1429     kwargs["engine_kwargs"] = engine_kwargs
   1431 op = GroupByApply(self, func, args=args, kwargs=kwargs)
-> 1432 result = op.agg()
   1433 if not is_dict_like(func) and result is not None:
   1434     # GH #52849
   1435     if not self.as_index and is_list_like(func):

File ~\anaconda3\Lib\site-packages\pandas\core\apply.py:190, in Apply.agg(self)
    187     return self.apply_str()
    189 if is_dict_like(func):
--> 190     return self.agg_dict_like()
    191 elif is_list_like(func):
    192     # we require a list, but not a 'str'
    193     return self.agg_list_like()

File ~\anaconda3\Lib\site-packages\pandas\core\apply.py:423, in Apply.agg_dict_like(self)
    415 def agg_dict_like(self) -> DataFrame | Series:
    416     """
    417     Compute aggregation in the case of a dict-like argument.
    418
   (...)
    421     Result of aggregation.
    422     """
--> 423     return self.agg_or_apply_dict_like(op_name="agg")

File ~\anaconda3\Lib\site-packages\pandas\core\apply.py:1611, in GroupByApply.agg_or_apply_dict_like(self, op_name)
   1605 with com.temp_setattr(
   1606     obj, "as_index", True, condition=hasattr(obj, "as_index")
   1607 ):
   1608     result_index, result_data = self.compute_dict_like(
   1609         op_name, selected_obj, selection, kwargs
   1610     )
-> 1611 result = self.wrap_results_dict_like(selected_obj, result_index, result_data)
   1612 return result

File ~\anaconda3\Lib\site-packages\pandas\core\apply.py:531, in Apply.wrap_results_dict_like(self, selected_obj, result_index, result_data)
    528         keys_to_use = ktu
    530     axis: AxisInt = 0 if isinstance(obj, ABCSeries) else 1
--> 531     result = concat(
    532         {k: results[k] for k in keys_to_use},
    533         axis=axis,
    534         keys=keys_to_use,
    535     )
    536 elif any(is_ndframe):
    537     # There is a mix of NDFrames and scalars
    538     raise ValueError(
    539         "cannot perform both aggregation "
    540         "and transformation operations "
    541         "simultaneously"
```

```
    542     )

File ~\anaconda3\Lib\site-packages\pandas\core\reshape\concat.py:382, in concat(objs, axis, join, ignore_index, keys, levels, names, verify_integrity, sort, copy)
    379 elif copy and using_copy_on_write():
    380     copy = False
--> 382 op = _Concatenator(
    383     objs,
    384     axis=axis,
    385     ignore_index=ignore_index,
    386     join=join,
    387     keys=keys,
    388     levels=levels,
    389     names=names,
    390     verify_integrity=verify_integrity,
    391     copy=copy,
    392     sort=sort,
    393 )
    395 return op.get_result()

File ~\anaconda3\Lib\site-packages\pandas\core\reshape\concat.py:445, in _Concatenator.__init__(self, objs, axis, join, keys, levels, names, ignore_index, verify_integrity, copy, sort)
    442 self.verify_integrity = verify_integrity
    443 self.copy = copy
--> 445 objs, keys = self._clean_keys_and_objs(objs, keys)
    447 # figure out what our result ndim is going to be
    448 ndims = self._get_ndims(objs)

File ~\anaconda3\Lib\site-packages\pandas\core\reshape\concat.py:507, in _Concatenator._clean_keys_and_objs(self, objs, keys)
    504     objs_list = list(objs)
    506 if len(objs_list) == 0:
--> 507     raise ValueError("No objects to concatenate")
    509 if keys is None:
    510     objs_list = list(com.not_none(*objs_list))

ValueError: No objects to concatenate
```

In [ ]: