In [1]:
```
import pandas as pd
import numpy as np
```

In [5]:
```
df = pd.read_csv \
(r'C:\Users\CSE-A102-Sys36\Downloads\gapminder-FiveYearData.csv')
```

In [6]:
```
df
```

Out[6]:

	country	year	pop	continent	lifeExp	gdpPercap
0	Afghanistan	1952	8425333.0	Asia	28.801	779.445314
1	Afghanistan	1957	9240934.0	Asia	30.332	820.853030
2	Afghanistan	1962	10267083.0	Asia	31.997	853.100710
3	Afghanistan	1967	11537966.0	Asia	34.020	836.197138
4	Afghanistan	1972	13079460.0	Asia	36.088	739.981106
...	...	...	...	...	...	...
1699	Zimbabwe	1987	9216418.0	Africa	62.351	706.157306
1700	Zimbabwe	1992	10704340.0	Africa	60.377	693.420786
1701	Zimbabwe	1997	11404948.0	Africa	46.809	792.449960
1702	Zimbabwe	2002	11926563.0	Africa	39.989	672.038623
1703	Zimbabwe	2007	12311143.0	Africa	43.487	469.709298

1704 rows × 6 columns

In [49]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1704 entries, 0 to 1703
Data columns (total 6 columns):
 #   Column     Non-Null Count  Dtype  
---  ------     --------------  -----  
 0   country    1704 non-null   object 
 1   year       1704 non-null   int64  
 2   pop        1704 non-null   float64
 3   continent  1704 non-null   object 
 4   lifeExp    1704 non-null   float64
 5   gdpPercap  1704 non-null   float64
dtypes: float64(3), int64(1), object(2)
memory usage: 80.0+ KB
```

In [50]:
```
# Changing dtype
df['pop'] = df['pop'].astype(np.int64)
```

In [51]:
```
1 df
```

Out[51]:

	country	year	pop	continent	lifeExp	gdpPercap
0	Afghanistan	1952	8425333	Asia	28.80100	779.445314
1	Afghanistan	1957	9240934	Asia	30.33200	820.853030
2	Afghanistan	1962	10267083	Asia	31.99700	853.100710
3	Afghanistan	1967	11537966	Asia	34.02000	836.197138
4	Afghanistan	1972	13079460	Asia	36.08800	739.981106
5	Afghanistan	1977	14880372	Asia	38.43800	786.113360
6	Afghanistan	1982	12881816	Asia	39.85400	978.011439
7	Afghanistan	1987	13867957	Asia	40.82200	852.395945
8	Afghanistan	1992	16317921	Asia	41.67400	649.341395
9	Afghanistan	1997	22227415	Asia	41.76300	635.341351
10	Afghanistan	2002	25268405	Asia	42.12900	726.734055
11	Afghanistan	2007	31889923	Asia	43.82800	974.580338

In [52]:
```
1 # Check the number of maximum returned rows:
2 import pandas as pd
3 print(pd.options.display.max_rows)
```

```
1704
```

In [53]:
```
1 # to see the entire dataset
2 pd.set_option('display.max_rows', 1704)
3 df
```

Out[53]:

	country	year	pop	continent	lifeExp	gdpPercap
0	Afghanistan	1952	8425333	Asia	28.80100	779.445314
1	Afghanistan	1957	9240934	Asia	30.33200	820.853030
2	Afghanistan	1962	10267083	Asia	31.99700	853.100710
3	Afghanistan	1967	11537966	Asia	34.02000	836.197138
4	Afghanistan	1972	13079460	Asia	36.08800	739.981106
5	Afghanistan	1977	14880372	Asia	38.43800	786.113360
6	Afghanistan	1982	12881816	Asia	39.85400	978.011439
7	Afghanistan	1987	13867957	Asia	40.82200	852.395945
8	Afghanistan	1992	16317921	Asia	41.67400	649.341395
9	Afghanistan	1997	22227415	Asia	41.76300	635.341351
10	Afghanistan	2002	25268405	Asia	42.12900	726.734055
11	Afghanistan	2007	31889923	Asia	43.82800	974.580338

In [54]:
```
1 print(df.head())
```

```
       country  year       pop continent  lifeExp   gdpPercap
0  Afghanistan  1952   8425333      Asia   28.801  779.445314
1  Afghanistan  1957   9240934      Asia   30.332  820.853030
2  Afghanistan  1962  10267083      Asia   31.997  853.100710
3  Afghanistan  1967  11537966      Asia   34.020  836.197138
4  Afghanistan  1972  13079460      Asia   36.088  739.981106
```

In [55]:
```
1 print(df.shape)
```

```
(1704, 6)
```

In [56]:
```
1 print(df.columns)
```

```
Index(['country', 'year', 'pop', 'continent', 'lifeExp', 'gdpPercap'], dty
pe='object')
```

In [57]:
```
1 print(df.dtypes)
```

```
country       object
year           int64
pop            int64
continent     object
lifeExp      float64
gdpPercap    float64
dtype: object
```

In [58]:
```
1 # retrieve country column
2 c = df['country']
3 c
```

Out[58]:
```
0             Afghanistan
1             Afghanistan
2             Afghanistan
3             Afghanistan
4             Afghanistan
5             Afghanistan
6             Afghanistan
7             Afghanistan
8             Afghanistan
9             Afghanistan
10            Afghanistan
11            Afghanistan
12                Albania
13                Albania
14                Albania
15                Albania
16                Albania
17                Albania
18                Albania
```

In [59]:
```
# retrieve country and year column
sub = df[['country', 'year']]
print(sub)
```

```
                country  year
0           Afghanistan  1952
1           Afghanistan  1957
2           Afghanistan  1962
3           Afghanistan  1967
4           Afghanistan  1972
5           Afghanistan  1977
6           Afghanistan  1982
7           Afghanistan  1987
8           Afghanistan  1992
9           Afghanistan  1997
10          Afghanistan  2002
11          Afghanistan  2007
12              Albania  1952
13              Albania  1957
14              Albania  1962
15              Albania  1967
16              Albania  1972
17              Albania  1977
```

In [60]:
```
print(df.tail())
```

```
       country  year       pop continent  lifeExp   gdpPercap
1699  Zimbabwe  1987   9216418    Africa   62.351  706.157306
1700  Zimbabwe  1992  10704340    Africa   60.377  693.420786
1701  Zimbabwe  1997  11404948    Africa   46.809  792.449960
1702  Zimbabwe  2002  11926563    Africa   39.989  672.038623
1703  Zimbabwe  2007  12311143    Africa   43.487  469.709298
```

In [61]:
```
# Retrieve first row. loc means location
print(df.loc[0])
```

```
country      Afghanistan
year                1952
pop              8425333
continent           Asia
lifeExp           28.801
gdpPercap     779.445314
Name: 0, dtype: object
```

In [62]:
```
# R etrieve last row
print(df.tail(n=1))
```

```
       country  year       pop continent  lifeExp   gdpPercap
1703  Zimbabwe  2007  12311143    Africa   43.487  469.709298
```

In [63]:
```
#retrieve last two records
print(df.tail(n=2))
```

```
       country  year       pop continent  lifeExp   gdpPercap
1702  Zimbabwe  2002  11926563    Africa   39.989  672.038623
1703  Zimbabwe  2007  12311143    Africa   43.487  469.709298
```

In [64]:
```
# Retrieve rows
print(df.loc[[0,1,3]])
```

```
       country  year       pop continent  lifeExp   gdpPercap
0  Afghanistan  1952   8425333      Asia   28.801  779.445314
1  Afghanistan  1957   9240934      Asia   30.332  820.853030
3  Afghanistan  1967  11537966      Asia   34.020  836.197138
```

In [65]:
```
# another way of subsetting rows- details about 2nd row with iloc
print(df.iloc[1])
```

```
country      Afghanistan
year                1957
pop              9240934
continent           Asia
lifeExp            30.332
gdpPercap       820.85303
Name: 1, dtype: object
```

In [66]:
```
print(df.iloc[99])
```

```
country      Bangladesh
year               1967
pop            62821884
continent          Asia
lifeExp          43.453
gdpPercap    721.186086
Name: 99, dtype: object
```

In [67]:
```
#with loc we cant use - indexing but with iloc we can use
print(df.loc[1])
```

```
country      Afghanistan
year                1957
pop              9240934
continent           Asia
lifeExp            30.332
gdpPercap       820.85303
Name: 1, dtype: object
```

In [68]:
```
print(df.iloc[-1])
```

```
country        Zimbabwe
year               2007
pop            12311143
continent        Africa
lifeExp           43.487
gdpPercap    469.709298
Name: 1703, dtype: object
```

In [69]:
```
print(df.iloc[[0,1,2]])
```

```
       country  year       pop continent  lifeExp   gdpPercap
0  Afghanistan  1952   8425333      Asia   28.801  779.445314
1  Afghanistan  1957   9240934      Asia   30.332  820.853030
2  Afghanistan  1962  10267083      Asia   31.997  853.100710
```

In [70]:
```
#: all rows
sub = df.loc[:,['country', 'year']]
print(sub)
```

```
       country  year
0  Afghanistan  1952
1  Afghanistan  1957
2  Afghanistan  1962
3  Afghanistan  1967
4  Afghanistan  1972
5  Afghanistan  1977
6  Afghanistan  1982
7  Afghanistan  1987
8  Afghanistan  1992
9  Afghanistan  1997
10 Afghanistan  2002
11 Afghanistan  2007
12     Albania  1952
13     Albania  1957
14     Albania  1962
15     Albania  1967
16     Albania  1972
17     Albania  1977
```

In [71]:
```
sub1 = df.iloc[:,[1,2,-1]]
print(sub1)
```

```
    year      pop     gdpPercap
0   1952  8425333    779.445314
1   1957  9240934    820.853030
2   1962 10267083    853.100710
3   1967 11537966    836.197138
4   1972 13079460    739.981106
5   1977 14880372    786.113360
6   1982 12881816    978.011439
7   1987 13867957    852.395945
8   1992 16317921    649.341395
9   1997 22227415    635.341351
10  2002 25268405    726.734055
11  2007 31889923    974.580338
12  1952  1282697   1601.056136
13  1957  1476505   1942.284244
14  1962  1728137   2312.888958
15  1967  1984060   2760.196931
16  1972  2263554   3313.422188
17  1977  2509048   3533.003910
```

In [72]:
```
sub1 = df.loc[:,['year','pop','gdpPercap']]
print(sub1)
```

```
    year       pop     gdpPercap
0   1952   8425333    779.445314
1   1957   9240934    820.853030
2   1962  10267083    853.100710
3   1967  11537966    836.197138
4   1972  13079460    739.981106
5   1977  14880372    786.113360
6   1982  12881816    978.011439
7   1987  13867957    852.395945
8   1992  16317921    649.341395
9   1997  22227415    635.341351
10  2002  25268405    726.734055
11  2007  31889923    974.580338
12  1952   1282697   1601.056136
13  1957   1476505   1942.284244
14  1962   1728137   2312.888958
15  1967   1984060   2760.196931
16  1972   2263554   3313.422188
17  1977   2509048   3533.003910
```

In [73]:
```
#The range() function in Python is used to generate a sequence of numbe
s = list(range(5))
print(s)
```

```
[0, 1, 2, 3, 4]
```

In [74]:
```
#subsetting rows and columns togther
print(df.loc[1,'country'])
```

```
Afghanistan
```

In [75]:
```
print(df.iloc[1,1])
```

```
1957
```

In [76]:
```
print(df.iloc[[0,1,2],[0,1,2]])
```

```
       country  year       pop
0  Afghanistan  1952   8425333
1  Afghanistan  1957   9240934
2  Afghanistan  1962  10267083
```

In [77]:
```
print(df.loc[[0,1,2], ['country', 'year', 'pop']])
```

```
       country  year       pop
0  Afghanistan  1952   8425333
1  Afghanistan  1957   9240934
2  Afghanistan  1962  10267083
```

In [78]:
```
print(df.loc[0:2,['country','year']])
```

```
       country  year
0  Afghanistan  1952
1  Afghanistan  1957
2  Afghanistan  1962
```

In [79]:
```
print(df.head(n=10))
```

```
       country  year       pop continent  lifeExp   gdpPercap
0  Afghanistan  1952   8425333      Asia   28.801  779.445314
1  Afghanistan  1957   9240934      Asia   30.332  820.853030
2  Afghanistan  1962  10267083      Asia   31.997  853.100710
3  Afghanistan  1967  11537966      Asia   34.020  836.197138
4  Afghanistan  1972  13079460      Asia   36.088  739.981106
5  Afghanistan  1977  14880372      Asia   38.438  786.113360
6  Afghanistan  1982  12881816      Asia   39.854  978.011439
7  Afghanistan  1987  13867957      Asia   40.822  852.395945
8  Afghanistan  1992  16317921      Asia   41.674  649.341395
9  Afghanistan  1997  22227415      Asia   41.763  635.341351
```

The groupby() method in pandas is a powerful tool used to group data based on the values of one or more columns and then apply various aggregation or transformation operations on these groups. It is commonly used in data analysis to summarize or analyze large datasets by categories.

In [80]:
```
print(df.groupby('year') ['lifeExp'].mean())
```

```
year
1952    49.057620
1957    51.507401
1962    53.609249
1967    55.678290
1972    57.647386
1977    59.570157
1982    61.533197
1987    63.212613
1992    64.160338
1997    65.014676
2002    65.694923
2007    67.007423
Name: lifeExp, dtype: float64
```

In [81]:
```
#The nunique() method returns the number of unique values for each colu
print(df.groupby('continent') ['country'].nunique())
```

```
continent
Africa      52
Americas    25
Asia        33
Europe      30
Oceania      2
Name: country, dtype: int64
```

In [84]:
```
print(df.groupby('continent') ['country'].count())
```

```
continent
Africa      624
Americas    300
Asia        396
Europe      360
Oceania      24
Name: country, dtype: int64
```

In [82]:
```
multi_group_var = df.groupby(['year', 'continent'])[['lifeExp', 'gdpPer
print(multi_group_var)
```

year	continent	lifeExp	gdpPercap
1952	Africa	39.135500	1252.572466
	Americas	53.279840	4079.062552
	Asia	46.314394	5195.484004
	Europe	64.408500	5661.057435
	Oceania	69.255000	10298.085650
1957	Africa	41.266346	1385.236062
	Americas	55.960280	4616.043733
	Asia	49.318544	5787.732940
	Europe	66.703067	6963.012816
	Oceania	70.295000	11598.522455
1962	Africa	43.319442	1598.078825
	Americas	58.398760	4901.541870
	Asia	51.563223	5729.369625
	Europe	68.539233	8365.486814
	Oceania	71.085000	12696.452430
1967	Africa	45.334538	2050.363801
	Americas	60.410920	5668.253496
	Asia	54.663640	5971.173374
	Europe	69.737600	10143.823757
	Oceania	71.310000	14495.021790
1972	Africa	47.450942	2339.615674
	Americas	62.394920	6491.334139
	Asia	57.319269	8187.468699
	Europe	70.775033	12479.575246
	Oceania	71.910000	16417.333380
1977	Africa	49.580423	2585.938508
	Americas	64.391560	7352.007126
	Asia	59.610556	7791.314020
	Europe	71.937767	14283.979110
	Oceania	72.855000	17283.957605
1982	Africa	51.592865	2481.592960
	Americas	66.228840	7506.737088
	Asia	62.617939	7434.135157
	Europe	72.806400	15617.896551
	Oceania	74.290000	18554.709840
1987	Africa	53.344788	2282.668991
	Americas	68.090720	7793.400261
	Asia	64.851182	7608.226508
	Europe	73.642167	17214.310727
	Oceania	75.320000	20448.040160
1992	Africa	53.629577	2281.810333
	Americas	69.568360	8044.934406
	Asia	66.537212	8639.690248
	Europe	74.440100	17061.568084
	Oceania	76.945000	20894.045885
1997	Africa	53.598269	2378.759555
	Americas	71.150480	8889.300863
	Asia	68.020515	9834.093295
	Europe	75.505167	19076.781802
	Oceania	78.190000	24024.175170
2002	Africa	53.325231	2599.385159
	Americas	72.422040	9287.677107
	Asia	69.233879	10174.090397
	Europe	76.700600	21711.732422
	Oceania	79.740000	26938.778040
2007	Africa	54.806038	3089.032605
	Americas	73.608120	11003.031625
	Asia	70.728485	12473.026870

```
Europe     77.648600  25054.481636
Oceania    80.719500  29810.188275
```

The reset_index() method in pandas is used to reset the index of a DataFrame, which can be particularly useful when you've performed operations that alter the index, like filtering or grouping data. By resetting the index, you can convert the current index into a regular column, and/or create a new default integer index.

In [83]:
```
1 multi_group_var.reset_index()
```

Out[83]:

	year	continent	lifeExp	gdpPercap
0	1952	Africa	39.135500	1252.572466
1	1952	Americas	53.279840	4079.062552
2	1952	Asia	46.314394	5195.484004
3	1952	Europe	64.408500	5661.057435
4	1952	Oceania	69.255000	10298.085650
5	1957	Africa	41.266346	1385.236062
6	1957	Americas	55.960280	4616.043733
7	1957	Asia	49.318544	5787.732940
8	1957	Europe	66.703067	6963.012816
9	1957	Oceania	70.295000	11598.522455
10	1962	Africa	43.319442	1598.078825
11	1962	Americas	58.398760	4901.541870
12	1962	Asia	51.563223	5729.369625
13	1962	Europe	68.539233	8365.486814
14	1962	Oceania	71.085000	12696.452430
15	1967	Africa	45.334538	2050.363801
16	1967	Americas	60.410920	5668.253496
17	1967	Asia	54.663640	5971.173374
18	1967	Europe	69.737600	10143.823757
19	1967	Oceania	71.310000	14495.021790
20	1972	Africa	47.450942	2339.615674
21	1972	Americas	62.394920	6491.334139
22	1972	Asia	57.319269	8187.468699
23	1972	Europe	70.775033	12479.575246
24	1972	Oceania	71.910000	16417.333380
25	1977	Africa	49.580423	2585.938508
26	1977	Americas	64.391560	7352.007126
27	1977	Asia	59.610556	7791.314020
28	1977	Europe	71.937767	14283.979110
29	1977	Oceania	72.855000	17283.957605
30	1982	Africa	51.592865	2481.592960
31	1982	Americas	66.228840	7506.737088
32	1982	Asia	62.617939	7434.135157
33	1982	Europe	72.806400	15617.896551
34	1982	Oceania	74.290000	18554.709840
35	1987	Africa	53.344788	2282.668991
36	1987	Americas	68.090720	7793.400261
37	1987	Asia	64.851182	7608.226508
38	1987	Europe	73.642167	17214.310727

	year	continent	lifeExp	gdpPercap
39	1987	Oceania	75.320000	20448.040160
40	1992	Africa	53.629577	2281.810333
41	1992	Americas	69.568360	8044.934406
42	1992	Asia	66.537212	8639.690248
43	1992	Europe	74.440100	17061.568084
44	1992	Oceania	76.945000	20894.045885
45	1997	Africa	53.598269	2378.759555
46	1997	Americas	71.150480	8889.300863
47	1997	Asia	68.020515	9834.093295
48	1997	Europe	75.505167	19076.781802
49	1997	Oceania	78.190000	24024.175170
50	2002	Africa	53.325231	2599.385159
51	2002	Americas	72.422040	9287.677107
52	2002	Asia	69.233879	10174.090397
53	2002	Europe	76.700600	21711.732422
54	2002	Oceania	79.740000	26938.778040
55	2007	Africa	54.806038	3089.032605
56	2007	Americas	73.608120	11003.031625
57	2007	Asia	70.728485	12473.026870
58	2007	Europe	77.648600	25054.481636
59	2007	Oceania	80.719500	29810.188275

In [ ]:
```
1
```