

Name: Pranav Vispute

Email: pranavvispute671@gmail.com

In [1]:

```
import pandas as pd
import numpy as np
```

Case Study 1: IPL

Question-1: Suppose in 'ipl18', you want to filter out the teams that have an NRR greater than zero, and for which the 'For' score exceeds the 'Against' score, i.e. both the conditions should be satisfied. Which teams will be left after you perform the above filtration?

In [6]:

```
ipl18 = pd.DataFrame({'Team': ['SRH', 'CSK', 'KKR', 'RR', 'MI', 'RCB', 'KXIP', 'DD'],
                        'Matches': [14, 14, 14, 14, 14, 14, 14, 14],
                        'Won': [9, 9, 8, 7, 6, 6, 6, 5],
                        'Lost': [5, 5, 6, 7, 8, 8, 8, 9],
                        'Tied': [0, 0, 0, 0, 0, 0, 0, 0],
                        'N/R': [0, 0, 0, 0, 0, 0, 0, 0],
                        'Points': [18, 18, 16, 14, 12, 12, 12, 10],
                        'NRR': [0.284, 0.253, -0.070, -0.250, 0.317, 0.129, -0.502, -0.222],
                        'For': [2230, 2488, 2363, 2130, 2380, 2322, 2210, 2297],
                        'Against': [2193, 2433, 2425, 2141, 2282, 2383, 2259, 2304]},
                        index = range(1,9))
```

In [3]:

```
#To print names of teams that satisfy both conditions,
ipl18[(ipl18["NRR"]>0)&(ipl18["For"]>ipl18["Against"])]
```

Out[3]:

	Team	Matches	Won	Lost	Tied	N/R	Points	NRR	For	Against
1	SRH	14	9	5	0	0	18	0.284	2230	2193
2	CSK	14	9	5	0	0	18	0.253	2488	2433
5	MI	14	6	8	0	0	12	0.317	2380	2282

Question-2: If all the stats are taken for both 'ipl17' and 'ipl18', which team with its total points greater than 25 will have the highest win percentage?

In [5]:

```
ipl17 = pd.DataFrame({'Team': ['MI', 'RPS', 'SRH', 'KKR', 'KXIP', 'DD', 'GL', 'RCB'],
                      'Matches': [14, 14, 14, 14, 14, 14, 14, 14],
                      'Won': [10, 9, 8, 8, 7, 6, 4, 3],
                      'Lost': [4, 5, 5, 6, 7, 8, 10, 10],
                      'Tied': [0, 0, 0, 0, 0, 0, 0, 0],
                      'N/R': [0, 0, 1, 0, 0, 0, 0, 1],
                      'Points': [20, 18, 17, 16, 14, 12, 8, 7],
                      'NRR': [0.784, 0.176, 0.469, 0.641, 0.123, -0.512, -0.412, -1.299],
                      'For': [2407, 2180, 2221, 2329, 2207, 2219, 2406, 1845],
                      'Against': [2242, 2165, 2118, 2300, 2229, 2255, 2472, 2033]},
                      index = range(1,9))
```

In [15]:

```
both=ipl17.append(ipl18)
```

...

In [14]:

```
combined=both[["Team","Won","Lost"]].groupby(["Team"]).sum()
combined.sort_values(by=["Won"],inplace=True, ascending = False)
(combined["Won"]/(combined["Won"]+combined["Lost"]))*100
new_combined=combined[combined["Won"]+combined["Lost"]>25]
final_combined=(new_combined["Won"]/(new_combined["Won"]+new_combined["Lost"]))*100
final_combined
```

Out[14]:

```
Team
SRH      62.962963
KKR      57.142857
MI       57.142857
KXIP     46.428571
DD       39.285714
RCB      33.333333
dtype: float64
```

Case Study 2: Wine Review

In [4]:

```
import pandas as pd
import numpy as np
```

In [5]:

```
Csv_File=pd.read_csv("wine.csv", index_col=0)
pd.set_option("display.max_rows", 5)
```

Questions

Q.1: Select the description column from reviews and assign the result to the variable desc .

In [32]:

```
#Answer:
desc=Csv_File["description"]
desc
```

Out[32]:

```
0      Aromas include tropical fruit, broom, brimston...
1      This is ripe and fruity, a wine that is smooth...
...
129969  A dry style of Pinot Gris, this is crisp with ...
129970  Big, rich and off-dry, this is powered by inte...
Name: description, Length: 129971, dtype: object
```

Q.2: Select the first value from the description column of reviews , assigning it to variable first_description .

In [24]:

```
#Answer:
first_description=Csv_File.loc[0,"description"]
first_description
```

Out[24]:

"Aromas include tropical fruit, broom, brimstone and dried herb. The palate isn't overly expressive, offering unripened apple, citrus and dried sage alongside brisk acidity."

Q.3:Select the first row of data (the first record) from reviews , assigning it to the variable first_row .

In [9]:

```
#Answer:
first_row=Csv_File[:1]
first_row
```

Out[9]:

	country	description	designation	points	price	province	region_1	region_2	taster_name
0	Italy	Aromas include tropical fruit, broom, brimston...	Vulkà Bianco	87	NaN	Sicily & Sardinia	Etna	NaN	Kerin O'Keefe

Q.4:Select the first 10 values from the description column in reviews , assigning the result to variable first_descriptions .

In [26]:

```
#Answer:
first_descriptions= Csv_File.loc[0:9,"description"]
first_descriptions
```

Out[26]:

```
0    Aromas include tropical fruit, broom, brimston...
1    This is ripe and fruity, a wine that is smooth...
   ...
8    Savory dried thyme notes accent sunnier flavor...
9    This has great depth of flavor with its fresh ...
Name: description, Length: 10, dtype: object
```

Q.5:Select the records with index labels 1 , 2 , 3 , 5 , and 8 , assigning the result to the variable sample_reviews .

In [27]:

```
#Answer:
sample_reviews=Csv_File.loc[[1,2,3,5,8]]
sample_reviews
```

Out[27]:

	country	description	designation	points	price	province	region_1	region_2	taster_r
1	Portugal	This is ripe and fruity, a wine that is smooth...	Avidagos	87	15.0	Douro	NaN	NaN	Roger
2	US	Tart and snappy, the flavors of lime flesh and...	NaN	87	14.0	Oregon	Willamette Valley	Willamette Valley	Paul Gr
3	US	Pineapple rind, lemon pith and orange blossom ...	Reserve Late Harvest	87	13.0	Michigan	Lake Michigan Shore	NaN	Alexa Pez
5	Spain	Blackberry and raspberry aromas show a typical...	Ars In Vitro	87	15.0	Northern Spain	Navarra	NaN	Mi Schae
8	Germany	Savory dried thyme notes accent sunnier flavor...	Shine	87	12.0	Rheinhessen	NaN	NaN	Anna Li

Q.6:Create a variable df containing the country , province , region_1 ,and region_2 columns of the records with the index labels 0 , 1 , 10 , and 100 .

In [29]:

```
#Answer:
df=Csv_File.loc[[0,1,10,100],["country","province","region_1","region_2"]]
df
```

Out[29]:

	country	province	region_1	region_2
0	Italy	Sicily & Sardinia	Etna	NaN
1	Portugal	Douro	NaN	NaN
10	US	California	Napa Valley	Napa
100	US	New York	Finger Lakes	Finger Lakes

Q.7:Create a variable df containing the country and variety columns of the first 100 records.

In [30]:

```
#Answer:
df=Csv_File.loc[0:99,["country", "variety"]]
df
```

Out[30]:

	country	variety
0	Italy	White Blend
1	Portugal	Portuguese Red
...
98	Italy	Sangiovese
99	US	Bordeaux-style Red Blend

100 rows × 2 columns

Q.8:Create a DataFrame italian_wines containing reviews of wines made in Italy .

In [31]:

```
#Answer:
italian_wines=Csv_File["country"]=="Italy"
italian_wines
```

Out[31]:

```
0      True
1     False
...
129969  False
129970  False
Name: country, Length: 129971, dtype: bool
```

End of the PDF

