

QB:32

```
In [1]: 1 import pandas as pd
        2 df = pd.read_csv("ipl-matches.csv")
        3 df
```

Out[1]:

	ID	City	Date	Season	MatchNumber	Team1	Team2	Venue	TossWinner	TossDecision	SuperOver	WinningTeam
0	1312200	Ahmedabad	2022-05-29	2022	Final	Rajasthan Royals	Gujarat Titans	Narendra Modi Stadium, Ahmedabad	Rajasthan Royals	bat	N	Gujarat Titans
1	1312199	Ahmedabad	2022-05-27	2022	Qualifier 2	Royal Challengers Bangalore	Rajasthan Royals	Narendra Modi Stadium, Ahmedabad	Rajasthan Royals	field	N	Rajasthan Royals
2	1312198	Kolkata	2022-05-25	2022	Eliminator	Royal Challengers Bangalore	Lucknow Super Giants	Eden Gardens, Kolkata	Lucknow Super Giants	field	N	Royal Challengers Bangalore
3	1312197	Kolkata	2022-05-24	2022	Qualifier 1	Rajasthan Royals	Gujarat Titans	Eden Gardens, Kolkata	Gujarat Titans	field	N	Gujarat Titans
4	1304116	Mumbai	2022-05-22	2022	70	Sunrisers Hyderabad	Punjab Kings	Wankhede Stadium, Mumbai	Sunrisers Hyderabad	bat	N	Punjab Kings
...
945	335986	Kolkata	2008-04-20	2007/08	4	Kolkata Knight Riders	Deccan Chargers	Eden Gardens	Deccan Chargers	bat	N	Kolkata Knight Riders
946	335985	Mumbai	2008-04-20	2007/08	5	Mumbai Indians	Royal Challengers Bangalore	Wankhede Stadium	Mumbai Indians	bat	N	Royal Challengers Bangalore
947	335984	Delhi	2008-04-19	2007/08	3	Delhi Daredevils	Rajasthan Royals	Feroz Shah Kotla	Rajasthan Royals	bat	N	Delhi Daredevils
948	335983	Chandigarh	2008-04-19	2007/08	2	Kings XI Punjab	Chennai Super Kings	Punjab Cricket Association Stadium, Mohali	Chennai Super Kings	bat	N	Chennai Super Kings
949	335982	Bangalore	2008-04-18	2007/08	1	Royal Challengers Bangalore	Kolkata Knight Riders	M Chinnaswamy Stadium	Royal Challengers Bangalore	field	N	Kolkata Knight Riders

950 rows × 20 columns

In [2]:

1 df.info()

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 950 entries, 0 to 949
Data columns (total 20 columns):
#   Column                Non-Null Count  Dtype
---  -
0   ID                    950 non-null   int64
1   City                  899 non-null   object
2   Date                  950 non-null   object
3   Season                950 non-null   object
4   MatchNumber           950 non-null   object
5   Team1                 950 non-null   object
6   Team2                 950 non-null   object
7   Venue                 950 non-null   object
8   TossWinner            950 non-null   object
9   TossDecision          950 non-null   object
10  SuperOver             946 non-null   object
11  WinningTeam           946 non-null   object
12  WonBy                 950 non-null   object
13  Margin                932 non-null   float64
14  method                19 non-null    object
15  Player_of_Match       946 non-null   object
16  Team1Players           950 non-null   object
17  Team2Players           950 non-null   object
18  Umpire1               950 non-null   object
19  Umpire2               950 non-null   object
dtypes: float64(1), int64(1), object(18)
memory usage: 148.6+ KB
```

In [5]:

1df[df["SuperOver"]=="Y"]

Out[5]:

	ID	City	Date	Season	MatchNumber	Team1	Team2	Venue	TossWinner	TossDecision	SuperOver	WinningTeam
114	1254077	Chennai	2021-04-25	2021	20	Delhi Capitals	Sunrisers Hyderabad	MA Chidambaram Stadium, Chepauk, Chennai	Delhi Capitals	bat	Y	Delhi Capitals
158	1216512	Abu Dhabi	2020-10-18	2020/21	35	Kolkata Knight Riders	Sunrisers Hyderabad	Sheikh Zayed Stadium	Sunrisers Hyderabad	field	Y	Kolkata Knight Riders
159	1216517	NaN	2020-10-18	2020/21	36	Mumbai Indians	Kings XI Punjab	Dubai International Cricket Stadium	Mumbai Indians	bat	Y	Kings XI Punjab
184	1216547	NaN	2020-09-28	2020/21	10	Royal Challengers Bangalore	Mumbai Indians	Dubai International Cricket Stadium	Mumbai Indians	field	Y	Royal Challengers Bangalore
192	1216493	NaN	2020-09-20	2020/21	2	Delhi Capitals	Kings XI Punjab	Dubai International Cricket Stadium	Kings XI Punjab	field	Y	Delhi Capitals
203	1178426	Mumbai	2019-05-02	2019	51	Mumbai Indians	Sunrisers Hyderabad	Wankhede Stadium	Mumbai Indians	bat	Y	Mumbai Indians
244	1175365	Delhi	2019-03-30	2019	10	Kolkata Knight Riders	Delhi Capitals	Arun Jaitley Stadium	Delhi Capitals	field	Y	Delhi Capitals
339	1082625	Rajkot	2017-04-29	2017	35	Gujarat Lions	Mumbai Indians	Saurashtra Cricket Association Stadium	Gujarat Lions	bat	Y	Mumbai Indians
474	829741	Ahmedabad	2015-04-21	2015	18	Rajasthan Royals	Kings XI Punjab	Sardar Patel Stadium, Motera	Kings XI Punjab	field	Y	Kings XI Punjab
533	729315	Abu Dhabi	2014-04-29	2014	19	Kolkata Knight Riders	Rajasthan Royals	Sheikh Zayed Stadium	Rajasthan Royals	bat	Y	Rajasthan Royals
608	598017	Bangalore	2013-04-16	2013	21	Royal Challengers Bangalore	Delhi Daredevils	M Chinnaswamy Stadium	Royal Challengers Bangalore	field	Y	Royal Challengers Bangalore
621	598004	Hyderabad	2013-04-07	2013	7	Sunrisers Hyderabad	Royal Challengers Bangalore	Rajiv Gandhi International Stadium, Uppal	Royal Challengers Bangalore	bat	Y	Sunrisers Hyderabad
819	419121	Chennai	2010-03-21	2009/10	16	Chennai Super Kings	Kings XI Punjab	MA Chidambaram Stadium, Chepauk	Chennai Super Kings	field	Y	Kings XI Punjab
883	392190	Cape Town	2009-04-23	2009	10	Kolkata Knight Riders	Rajasthan Royals	Newlands	Kolkata Knight Riders	field	Y	Rajasthan Royals

In [13]:

1 df[(df["WinningTeam"]=="Chennai Super Kings")&(df["City"]=="Kolkata")]

Out[13]:

	ID	City	Date	Season	MatchNumber	Team1	Team2	Venue	TossWinner	TossDecision	SuperOver	WinningTeam	WonBy	Margir
224	1178404	Kolkata	2019-04-14	2019	29	Kolkata Knight Riders	Chennai Super Kings	Eden Gardens	Chennai Super Kings	field	N	Chennai Super Kings	Wickets	5.0
602	598022	Kolkata	2013-04-20	2013	26	Kolkata Knight Riders	Chennai Super Kings	Eden Gardens	Kolkata Knight Riders	bat	N	Chennai Super Kings	Wickets	4.0
641	548368	Kolkata	2012-05-14	2012	63	Kolkata Knight Riders	Chennai Super Kings	Eden Gardens	Chennai Super Kings	field	N	Chennai Super Kings	Wickets	5.0
827	419113	Kolkata	2010-03-16	2009/10	8	Kolkata Knight Riders	Chennai Super Kings	Eden Gardens	Chennai Super Kings	bat	N	Chennai Super Kings	Runs	55.0
908	336025	Kolkata	2008-05-18	2007/08	41	Kolkata Knight Riders	Chennai Super Kings	Eden Gardens	Kolkata Knight Riders	bat	N	Chennai Super Kings	Runs	3.0

In [15]:

1 df[(df["Player_of_Match"]=="MS Dhoni")&((df["Team1"]=="Mumbai Indians")|(df["Team2"]=="Mumbai Indians"))]

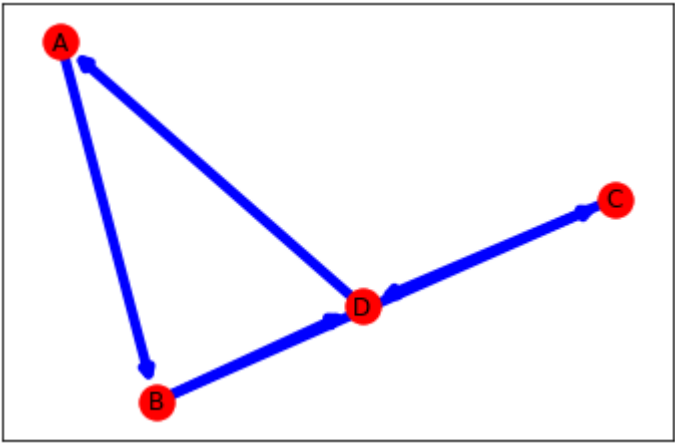
Out[15]:

	ID	City	Date	Season	MatchNumber	Team1	Team2	Venue	TossWinner	TossDecision	SuperOver	WinningTeam	WonBy
630	548379	Bangalore	2012-05-23	2012	Elimination Final	Chennai Super Kings	Mumbai Indians	M Chinnaswamy Stadium	Mumbai Indians	field	N	Chennai Super Kings	Runs

QB:-70

In [3]:

1 import networkx as nx
2 import matplotlib.pyplot as plt
3 G = nx.DiGraph()
4
5 G.add_edges_from([("A","B"),("B","C"),("C","D"),("D","A"),("B","D")])
6 nx.draw_networkx(G,node_size=300,node_color="red",edge_color="blue",width=5)
7 plt.show()



QB:-72

The following dictionary shows how five people follow each other on Instagram:

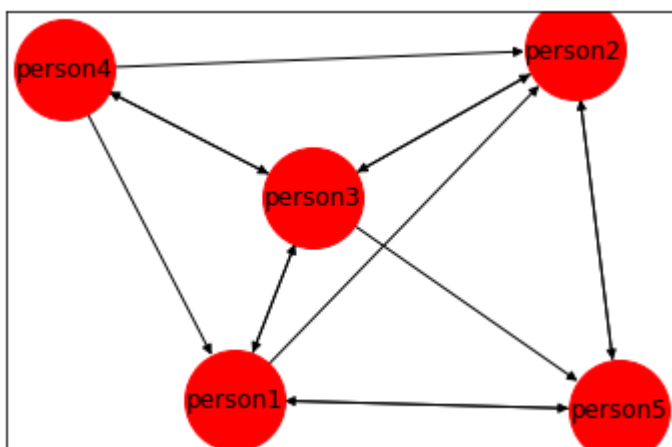
```
instagram = {'person1': [0,1,1,0,1], 'person2': [0,0,1,0,1], 'person3': [1,1,0,1,1], 'person4': [1,1,1,0,0], 'person5': [1,1,0,0,0]}
```

E.g., the list for person1 has the value on index 2 as 1 which means person1 follows person3 and a directed edge should be

added from person1 to person3.

Using networkx library, create a directed graph.

```
In [20]: 1 import networkx as nx
2 import matplotlib.pyplot as plt
3
4 instagram = {'person1': [0,1,1,0,1],
5             'person2': [0,0,1,0,1],
6             'person3': [1,1,0,1,1],
7             'person4': [1,1,1,0,0],
8             'person5': [1,1,0,0,0]}
9 G = nx.DiGraph()
10 k = list(instagram.keys())
11
12 for i in k:
13     for j in range(len(instagram)):
14         if instagram[i][j]==1:
15             G.add_edge(i,k[j])
16
17 nx.draw_networkx(G,node_size=2500,node_color="red")
18 plt.figure(figsize=[150,150])
19 plt.show()
```

**Old PB Mcq:**

```
In [7]: 1 import re
2 x = re.split('(a)(t)', "maths is difficult subject.")
3 print(x)
```

```
['m', 'a', 't', 'hs is difficult subject.']
```

```
In [9]: 1 import re
2 x = re.split('a',"maths is difficult subject.")
3 print(x)
```

```
['m', 'ths is difficult subject.']
```

```
In [10]: 1 import re
2 x = re.split('t',"maths is difficult subject.")
3 print(x)
```

```
['ma', 'hs is difficul', ' subject.', '.']
```

```
In [11]: 1 import re
2 x = re.split('at',"maths is difficult subject.")
3 print(x)
```

```
['m', 'hs is difficult subject.']
```

```
In [12]: 1 import re
2 x = re.split('(a)(t)', "maths is difficut subject.")
3 print(x)
```

```
['m', 'a', 't', 'hs is difficu', 'a', 't', ' subject.']
```

```
In [14]: 1 import re
2 x = re.split('(a)(t)', "maths is difficult subjeat")
3 print(x)
```

```
['m', 'a', 't', 'hs is difficult subje', 'a', 't', '']
```

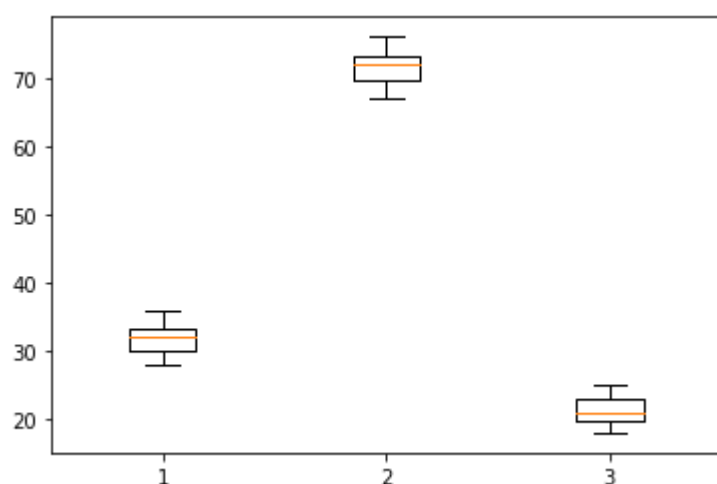
PB:71

Create a boxplot of the distribution of temperatures in different cities. Take data from 'temperatures.csv' from below:

<https://raw.githubusercontent.com/kavit88/Data-Sets/main/temperatures.csv>
(<https://raw.githubusercontent.com/kavit88/Data-Sets/main/temperatures.csv>)

```
In [24]: 1 import numpy as np
2 import matplotlib.pyplot as plt
3 import pandas as pd
4 df = pd.read_csv("https://raw.githubusercontent.com/kavit88/Data-Sets/main/temperatures.csv")
5 df
6 plt.boxplot(df[['New York', 'Los Angeles', 'Chicago']])
```

```
Out[24]: {'whiskers': [<matplotlib.lines.Line2D at 0x22603bc4670>,
<matplotlib.lines.Line2D at 0x22603bc49d0>,
<matplotlib.lines.Line2D at 0x22603bd5e50>,
<matplotlib.lines.Line2D at 0x22603be31f0>,
<matplotlib.lines.Line2D at 0x22603bec670>,
<matplotlib.lines.Line2D at 0x22603bec9d0>],
'caps': [<matplotlib.lines.Line2D at 0x22603bc4d30>,
<matplotlib.lines.Line2D at 0x22603bd50d0>,
<matplotlib.lines.Line2D at 0x22603be3550>,
<matplotlib.lines.Line2D at 0x22603be38b0>,
<matplotlib.lines.Line2D at 0x22603becd30>,
<matplotlib.lines.Line2D at 0x22603bf80d0>],
'boxes': [<matplotlib.lines.Line2D at 0x22603bc4310>,
<matplotlib.lines.Line2D at 0x22603bd5af0>,
<matplotlib.lines.Line2D at 0x22603bec310>],
'medians': [<matplotlib.lines.Line2D at 0x22603bd5430>,
<matplotlib.lines.Line2D at 0x22603be3c10>,
<matplotlib.lines.Line2D at 0x22603bf8430>],
'fliers': [<matplotlib.lines.Line2D at 0x22603bd5790>,
<matplotlib.lines.Line2D at 0x22603be3f70>,
<matplotlib.lines.Line2D at 0x22603bf8790>],
'means': []}
```



PB:73

You have been given a dataset of car prices and their respective horsepower, mileage, and weight. You have been

tasked to analyze the relationship between these variables and create a scatter plot to visualize the patterns.

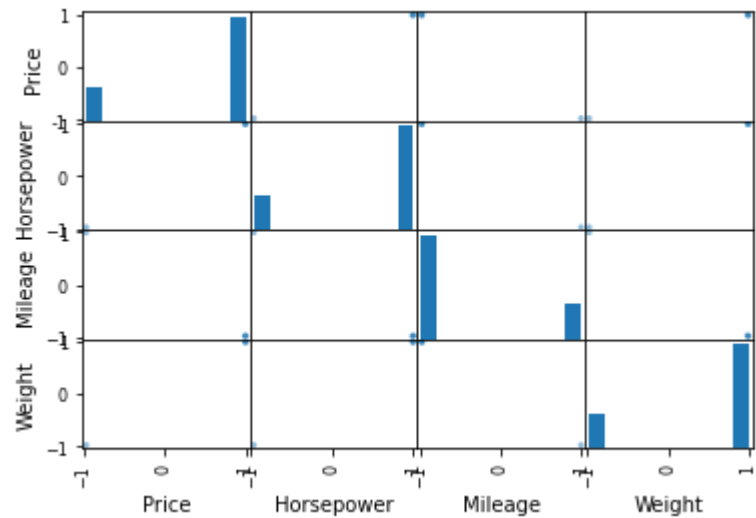
Dataset:

The dataset, named "car_data.csv" :

https://raw.githubusercontent.com/kavit88/Data-Sets/main/car_data.csv

(https://raw.githubusercontent.com/kavit88/Data-Sets/main/car_data.csv)

```
In [30]: 1 import numpy as np
2 import matplotlib.pyplot as plt
3 import pandas as pd
4 df = pd.read_csv("https://raw.githubusercontent.com/kavit88/Data-Sets/main/car_data.csv")
5 pd.plotting.scatter_matrix(df.corr())
6 plt.show()
7
```



PB:85

You have been hired as a network analyst by a company to analyze the social network of their employees. The company has

provided you with the following data:

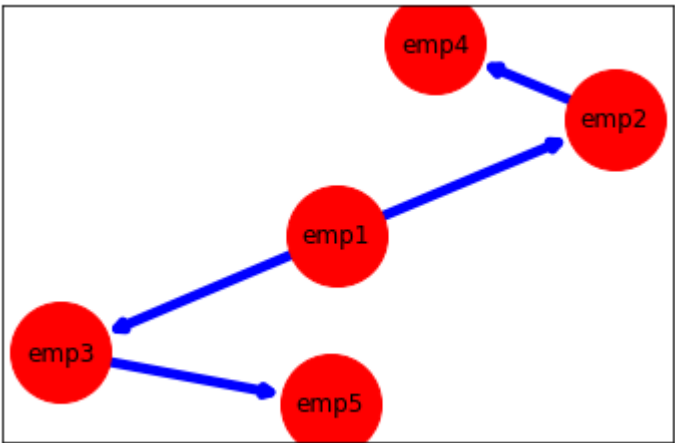
There are 5 employees in the company, each identified by a unique ID from 1 to 5.

The following relationships exist between the employees:

- 1. Employee 1 is friends with Employee 2 and Employee 3.
- 2. Employee 2 is friends with Employee 4.
- 3. Employee 3 is friends with Employee 5.

Your task is to create a NetworkX graph representing this social network and display it.

```
In [32]: 1 import matplotlib.pyplot as plt
2 import networkx as nx
3 G = nx.DiGraph()
4 G.add_edges_from([("emp1", "emp2"), ("emp1", "emp3"), ("emp2", "emp4"), ("emp3", "emp5")])
5 nx.draw_networkx(G, node_size=2500, node_color="red", edge_color="blue", width=5)
6 plt.show()
7
```



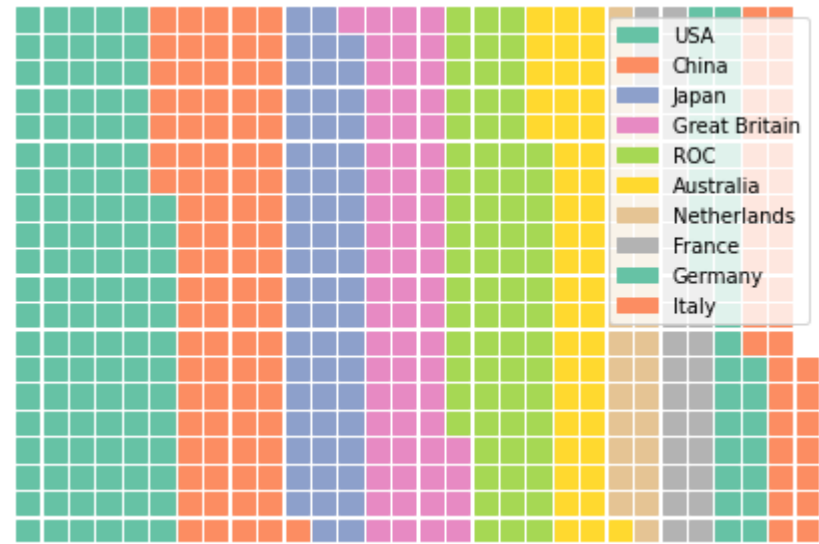
PB:83

Suppose you have data on the number of medals won by a country in the 2020 Tokyo Olympics. You want to visualize this data

using a waffle chart to show the proportional representation of each country's medal count.

Data={'USA': 113, 'China': 88, 'Japan': 58, 'Great Britain': 65, 'ROC': 71, 'Australia': 46, 'Netherlands': 36, 'France': 33, 'Germany': 37, 'Italy': 40}

```
In [37]: 1 import numpy as np
2 import pandas as pd
3 import matplotlib.pyplot as plt
4 import seaborn as sns
5 from pywaffle import Waffle
6
7 Data={'USA': 113, 'China': 88, 'Japan': 58, 'Great Britain': 65, 'ROC': 71, 'Australia': 46, 'Netherlands': 36, 'France': 33, 'Germany': 37, 'Italy': 40}
8
9
10 fig = plt.figure(FigureClass=Waffle,rows=20,values=list(Data.values()),labels=list(Data.keys()))
11 plt.show()
```



PB:79

The file "student_scores.csv" contains the marks scored by a group of students in three subjects: Maths, Science, and

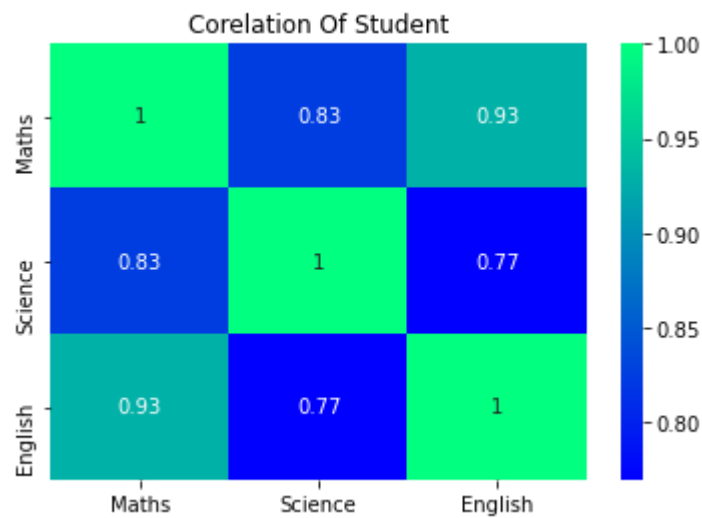
English. Each row contains the name of the student, their score in Maths, Science, and English. Create a pandas

DataFrame from this data and create a heatmap to visualize the correlations between the scores in these three

subjects. Take Dataset from below:

https://raw.githubusercontent.com/kavit88/Data-Sets/main/student_scores.csv
(https://raw.githubusercontent.com/kavit88/Data-Sets/main/student_scores.csv)

```
In [40]: 1 import pandas as pd
2 import matplotlib.pyplot as plt
3 import numpy as np
4 import seaborn as sns
5
6 df = pd.read_csv("https://raw.githubusercontent.com/kavit88/Data-Sets/main/student_scores.csv")
7 score = df[["Maths", "Science", "English"]].corr()
8 sns.heatmap(score, annot=True, cmap="winter")
9 plt.title("Corelation Of Student")
10 plt.show()
```



```
1 ### PB:
```