Just IT

B2W group

Apprenticeships | Training | Recruitment

Data Technician

Name: Thandeka Mukazi

Course Date: 17th June – 20th June 2025

Table of contents

Day 2: Task 1

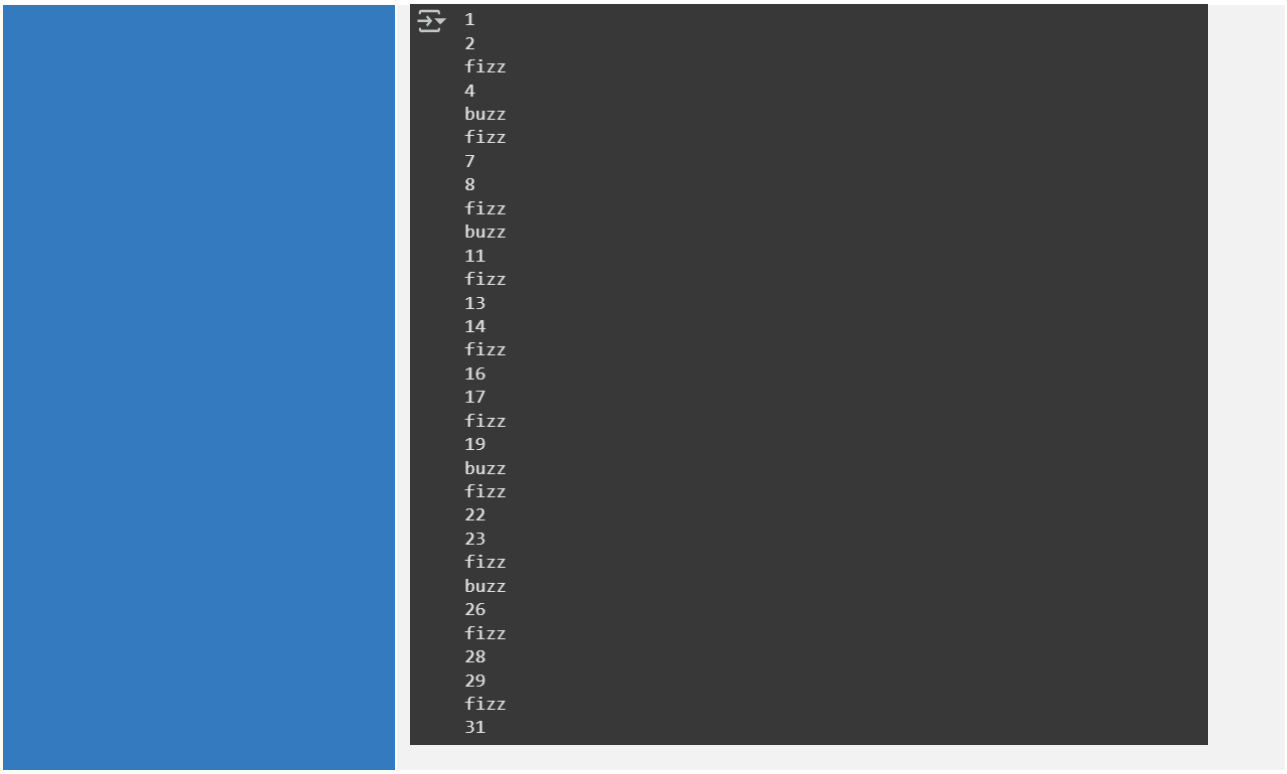
It is a common software development interview question to create the below with a certain programming language. Create the below using Python syntax, test it and past the completed syntax and output below.

FizzBuzz:

Go through the integers from 1 to 100.
If a number is divisible by 3, print "fizz."
If a number is divisible by 5, print "buzz."
If a number is both divisible by 3 and by 5, print "fizzbuzz."
Otherwise, print just the number.

Paste your completed work to the right

```
for number in range(1,101):
  if number % 3 == 0:
    print("fizz")
  elif number % 5 == 0:
      print("buzz")
  elif number % 3 == 0 and number % 5 == 0:
        print("fizzbuzz")
  else:
          print(number)
```



Day 3: Task 1

Download the 'student.csv', complete the below exercises as a group and paste your input and output. Although this is a group activity, everyone should have the below answered so it supports your portfolio:

Exercise 1: Loading and Exploring the Data

1. Question: "Write the code to read a CSV file into a Pandas DataFrame."
2. Question: "Write the code to display the first 5 rows of the DataFrame."
3. Question: "Write the code to get the information about the DataFrame."
4. Question: "Write the code to get summary statistics for the DataFrame."

Question: "Write the code to read a CSV file into a Pandas DataFrame."

```
# Read the file into a table called df_students
df_students = pd.read_csv('student.csv')
```

Question: "Write the code to display the first 5 rows of the DataFrame."

```
# Read the file into a table called df_students
df_students = pd.read_csv('student.csv')

# The first 5 rows
print(df_students.head())
```

```
   id        name  class  mark  gender
0   1    John Deo   Four    75  female
1   2    Max Ruin  Three    85    male
2   3      Arnold  Three    55    male
3   4  Krish Star   Four    60  female
4   5   John Mike   Four    60  female
```

Question: "Write the code to get the information about the DataFrame."

```
# Summary using info
df_students.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 35 entries, 0 to 34
Data columns (total 5 columns):
 #   Column  Non-Null Count  Dtype
---  ------  --------------  -----
 0   id      35 non-null     int64
 1   name    34 non-null     object
 2   class   34 non-null     object
 3   mark    35 non-null     int64
 4   gender  33 non-null     object
dtypes: int64(2), object(3)
memory usage: 1.5+ KB
```

Question: "Write the code to get summary statistics for the DataFrame."

```
# Summary statistics for marks
print(df_students['mark'].describe())
```

```
count    35.000000
mean     74.657143
std      16.401117
min      18.000000
25%      62.500000
50%      79.000000
75%      88.000000
max      96.000000
Name: mark, dtype: float64
```

Exercise 2: Indexing and Slicing

1. Question: "Write the code to select the 'name' column."
2. Question: "Write the code to select the 'name' and 'mark' columns."
3. Question: "Write the code to select the first 3 rows."
4. Question: "Write the code to select all rows where the 'class' is 'Four'."

Question: "Write the code to select the 'name' column."

```
# Select and display only the 'name' column.
df_students['name']
```

	name
0	John Deo
1	Max Ruin
2	Arnold
3	Krish Star
4	John Mike
5	Alex John
6	My John Rob
7	Asruid
8	Tes Qry
9	Big John
10	Ronald
11	Recky

Question: "Write the code to select the 'name' and 'mark' columns."

```
# Select and display 'name' and 'mark' columns.
print(df_students.loc[[2, 4, 6], ['name', 'mark']])
```

```
          name  mark
2      Arnold    55
4   John Mike    60
6  My John Rob    78
          name  class
0    John Deo   Four
1    Max Ruin  Three
2      Arnold  Three
3  Krish Star   Four
4   John Mike   Four
```

Question: "Write the code to select the first 3 rows."

```
# Select the first 3 rows
print(df_students.iloc[0:3])
```

```
   id      name  class  mark  gender
0   1  John Deo   Four    75  female
1   2  Max Ruin  Three    85    male
2   3    Arnold  Three    55    male
```

Question: "Write the code to select all rows where the 'class' is 'Four'."

```
# Select all rows where the class is 'Four'
class_four_students = df_students[df_students['class'] == 'Four']
print(class_four_students)
```

```
    id         name class  mark  gender
0    1     John Deo  Four    75  female
3    4   Krish Star  Four    60  female
4    5    John Mike  Four    60  female
5    6    Alex John  Four    55    male
9   10    Big John  Four    55  female
15  16       Gimmy  Four    88    male
20  21   Babby John  Four    69  female
30  31  Marry Toeey  Four    88    male
```

Exercise 3: Data Manipulation

1. Question: "Write the code to add a new column 'passed' that indicates whether the student passed (mark >= 60)."
2. Question: "Write the code to rename the 'mark' column to 'score'."
3. Question: "Write the code to drop the 'passed' column."

Question: "Write the code to add a new column 'passed' that indicates whether the student passed (mark >= 60)."

```
# Add a new column 'passed' that indicates whether the student passed (mark >= 60).
df_students['passed'] = df_students['mark'] >= 60
print(df_students)
```

	id	name	class	mark	gender	passed
0	1	John Deo	Four	75	female	True
1	2	Max Ruin	Three	85	male	True
2	3	Arnold	Three	55	male	False
3	4	Krish Star	Four	60	female	True
4	5	John Mike	Four	60	female	True
5	6	Alex John	Four	55	male	False
6	7	My John Rob	Fifth	78	male	True
7	8	Asruid	Five	85	male	True
8	9	Tes Qry	Six	78	NaN	True
9	10	Big John	Four	55	female	False
10	11	Ronald	Six	89	female	True
11	12	Recky	Six	94	female	True
12	13	Kty	Seven	88	female	True
13	14	Bigy	Seven	88	female	True
14	15	Tade Row	NaN	88	male	True
15	16	Gimmy	Four	88	male	True
16	17	Tumyu	Six	54	male	False
17	18	Honny	Five	75	male	True
18	19	Tinny	Nine	18	male	False
19	20	Jackly	Nine	65	female	True
20	21	Babby John	Four	69	female	True
21	22	Reggid	Seven	55	female	False
22	23	Herod	Eight	79	male	True
23	24	Tiddy Now	Seven	78	male	True
24	25	Giff Tow	Seven	88	male	True
25	26	Crelea	Seven	79	male	True
26	27	NaN	Three	81	NaN	True
27	28	Rojj Base	Seven	86	female	True
28	29	Tess Played	Seven	55	male	False
29	30	Reppy Red	Six	79	female	True
30	31	Marry Toeey	Four	88	male	True
31	32	Binn Rott	Seven	90	female	True
32	33	Kenn Rein	Six	96	female	True
33	34	Gain Toe	Seven	69	male	True
34	35	Rows Noump	Six	88	female	True

Question: "Write the code to rename the 'mark' column to 'score'."

```
# Rename the 'mark' column to 'score'.
df_students.rename(columns={'mark': 'score'}, inplace=True)
print(df_students)
```

	id	name	class	score	gender
0	1	John Deo	Four	75	female
1	2	Max Ruin	Three	85	male
2	3	Arnold	Three	55	male
3	4	Krish Star	Four	60	female
4	5	John Mike	Four	60	female
5	6	Alex John	Four	55	male
6	7	My John Rob	Fifth	78	male
7	8	Asruid	Five	85	male
8	9	Tes Qry	Six	78	NaN
9	10	Big John	Four	55	female
10	11	Ronald	Six	89	female
11	12	Recky	Six	94	female
12	13	Kty	Seven	88	female
13	14	Bigy	Seven	88	female
14	15	Tade Row	NaN	88	male
15	16	Gimmy	Four	88	male
16	17	Tumyu	Six	54	male
17	18	Honny	Five	75	male
18	19	Tinny	Nine	18	male
19	20	Jackly	Nine	65	female
20	21	Babby John	Four	69	female
21	22	Reggid	Seven	55	female
22	23	Herod	Eight	79	male
23	24	Tiddy Now	Seven	78	male
24	25	Giff Tow	Seven	88	male
25	26	Crelea	Seven	79	male
26	27	NaN	Three	81	NaN
27	28	Rojj Base	Seven	86	female
28	29	Tess Played	Seven	55	male
29	30	Reppy Red	Six	79	female
30	31	Marry Toeey	Four	88	male
31	32	Binn Rott	Seven	90	female
32	33	Kenn Rein	Six	96	female
33	34	Gain Toe	Seven	69	male
34	35	Rows Noump	Six	88	female

Question: "Write the code to drop the 'passed' column."

Before:

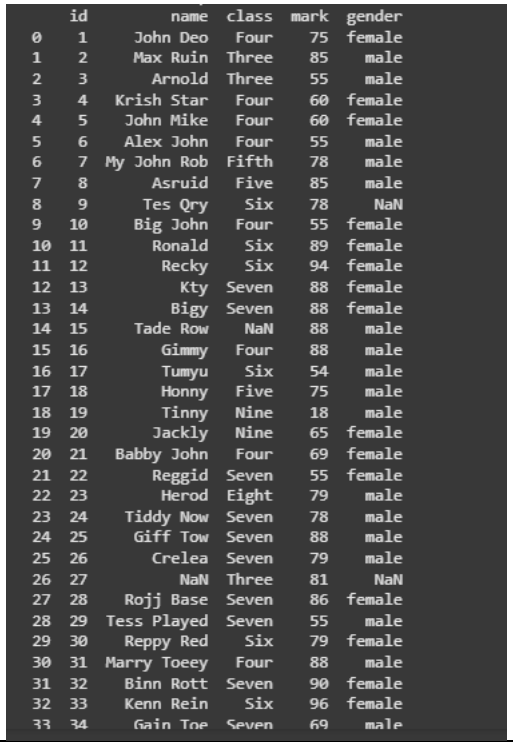
Before (running the code):

```
# Drop 'passed' column
df_students.drop(columns=['passed'], inplace=True)
print(df_students)
```

```
    id         name  class  mark  gender  passed
0    1     John Deo   Four    75  female    True
1    2    Max Ruin  Three    85    male    True
2    3      Arnold  Three    55    male   False
3    4   Krish Star   Four    60  female    True
4    5    John Mike   Four    60  female    True
5    6    Alex John   Four    55    male   False
6    7  My John Rob  Fifth    78    male    True
7    8       Asruid   Five    85    male    True
8    9      Tes Qry    Six    78     NaN    True
9   10    Big John   Four    55  female   False
10  11      Ronald    Six    89  female    True
11  12       Recky    Six    94  female    True
12  13         Kty  Seven    88  female    True
13  14        Bigy  Seven    88  female    True
14  15    Tade Row    NaN    88    male    True
15  16       Gimmy   Four    88    male    True
16  17       Tumyu    Six    54    male   False
17  18       Honny   Five    75    male    True
18  19       Tinny   Nine    18    male   False
19  20      Jackly   Nine    65  female    True
20  21   Babby John   Four    69  female    True
21  22      Reggid  Seven    55  female   False
22  23       Herod  Eight    79    male    True
23  24   Tiddy Now  Seven    78    male    True
24  25    Giff Tow  Seven    88    male    True
25  26      Crelea  Seven    79    male    True
26  27         NaN  Three    81     NaN    True
27  28    Rojj Base  Seven    86  female    True
28  29  Tess Played  Seven    55    male   False
29  30    Reppy Red    Six    79  female    True
30  31  Marry Toeey   Four    88    male    True
31  32    Binn Rott  Seven    90  female    True
32  33    Kenn Rein    Six    96  female    True
33  34     Gain Toe  Seven    69    male    True
34  35   Rows Noump    Six    88  female    True
```

After (running the code):

```python
# Drop 'passed' column
df_students.drop(columns=['passed'], inplace=True)
print(df_students)
```



	id	name	class	mark	gender
0	1	John Deo	Four	75	female
1	2	Max Ruin	Three	85	male
2	3	Arnold	Three	55	male
3	4	Krish Star	Four	60	female
4	5	John Mike	Four	60	female
5	6	Alex John	Four	55	male
6	7	My John Rob	Fifth	78	male
7	8	Asruid	Five	85	male
8	9	Tes Qry	Six	78	NaN
9	10	Big John	Four	55	female
10	11	Ronald	Six	89	female
11	12	Recky	Six	94	female
12	13	Kty	Seven	88	female
13	14	Bigy	Seven	88	female
14	15	Tade Row	NaN	88	male
15	16	Gimmy	Four	88	male
16	17	Tumyu	Six	54	male
17	18	Honny	Five	75	male
18	19	Tinny	Nine	18	male
19	20	Jackly	Nine	65	female
20	21	Babby John	Four	69	female
21	22	Reggid	Seven	55	female
22	23	Herod	Eight	79	male
23	24	Tiddy Now	Seven	78	male
24	25	Giff Tow	Seven	88	male
25	26	Crelea	Seven	79	male
26	27	NaN	Three	81	NaN
27	28	Rojj Base	Seven	86	female
28	29	Tess Played	Seven	55	male
29	30	Reppy Red	Six	79	female
30	31	Marry Toeey	Four	88	male
31	32	Binn Rott	Seven	90	female
32	33	Kenn Rein	Six	96	female
33	34	Gain Toe	Seven	69	male

Exercise 4: Aggregation and Grouping

1. Question: "Write the code to group the DataFrame by the 'class' column and calculate the mean 'mark' for each group."
2. Question: "Write the code to count the number of students in each class."
3. Question: "Write the code to calculate the average mark for each gender."

Question: "Write the code to group the DataFrame by the 'class' column and calculate the mean 'mark' for each group."

```python
# Group by 'class' and calculate the mean 'mark'
mean_marks_by_class = df_students.groupby('class')['mark'].mean()
print(mean_marks_by_class)
```

```
class
Eight    79.000000
Fifth    78.000000
Five     80.000000
Four     68.750000
Nine     41.500000
Seven    77.600000
Six      82.571429
Three    73.666667
Name: mark, dtype: float64
```

Question: "Write the code to count the number of students in each class."

```
# Count the number of students in each class
student_count_by_class = df_students['class'].value_counts()
print(student_count_by_class)
```

```
class
Seven    10
Four      8
Six       7
Three     3
Nine      2
Five      2
Fifth     1
Eight     1
Name: count, dtype: int64
```

Question: "Write the code to calculate the average mark for each gender."

```
# Calculate the average mark for each gender
average_mark_by_gender = df_students.groupby('gender')['mark'].mean()
print(average_mark_by_gender)
```

```
gender
female    77.312500
male      71.588235
Name: mark, dtype: float64
```

Exercise 5: Advanced Operations

1. Question: "Write the code to create a pivot table with 'class' as rows, 'gender' as columns, and 'mark' as values."
2. Question: "Write the code to create a new column 'grade' where marks >= 85 are 'A', 70-84 are 'B', 60-69 are 'C', and below 60 are 'D'."
3. Question: "Write the code to sort the DataFrame by 'mark' in descending order."

Question: "Write the code to create a pivot table with 'class' as rows, 'gender' as columns, and 'mark' as values."

```
# Pivot Table with 'class' as rows, 'gender' as columns and 'mark' as values
pivot_table = df_students.pivot_table(values='mark', index='class', columns='gender')
print(pivot_table)
```

```
class
Eight       NaN  79.0
Fifth       NaN  78.0
Five        NaN  80.0
Four       63.8  77.0
Nine       65.0  18.0
Seven      81.4  73.8
Six        89.2  54.0
Three       NaN  70.0
```

Question: "Write the code to create a new column 'grade' where marks >= 85 are 'A', 70-84 are 'B', 60-69 are 'C', and below 60 are 'D'."

```
# Create a new column 'grade' based on mark
df_students['grade'] = pd.cut(df_students['mark'], bins=[0, 60, 70, 80, 90, 100], labels=['F', 'D', 'C', 'B', 'A'])
print(df_students)
```

```
    id         name  class  mark  gender grade
0    1     John Deo   Four    75  female     C
1    2     Max Ruin  Three    85    male     B
2    3       Arnold  Three    55    male     F
3    4   Krish Star   Four    60  female     F
4    5     John Mike   Four    60  female     F
5    6     Alex John   Four    55    male     F
6    7  My John Rob  Fifth    78    male     C
7    8       Asruid   Five    85    male     B
8    9      Tes Qry    Six    78     NaN     C
9   10     Big John   Four    55  female     F
10  11       Ronald    Six    89  female     B
11  12        Recky    Six    94  female     A
12  13          Kty  Seven    88  female     B
13  14        Bigy   Seven    88  female     B
14  15    Tade Row     NaN    88    male     B
15  16       Gimmy   Four    88    male     B
16  17       Tumyu    Six    54    male     F
17  18       Honny   Five    75    male     C
18  19       Tinny   Nine    18    male     F
19  20      Jackly   Nine    65  female     D
20  21  Babby John   Four    69  female     D
21  22      Reggid  Seven    55  female     F
22  23       Herod  Eight    79    male     C
23  24   Tiddy Now  Seven    78    male     C
24  25    Giff Tow  Seven    88    male     B
25  26      Crelea  Seven    79    male     C
26  27         NaN  Three    81     NaN     B
27  28    Rojj Base  Seven    86  female     B
28  29  Tess Played  Seven    55    male     F
29  30    Reppy Red    Six    79  female     C
30  31  Marry Toeey   Four    88    male     B
31  32    Binn Rott  Seven    90  female     B
32  33    Kenn Rein    Six    96  female     A
33  34     Gain Toe  Seven    69    male     D
34  35  Rows Noump    Six    88  female     B
```

Question: "Write the code to sort the DataFrame by 'mark' in descending order."

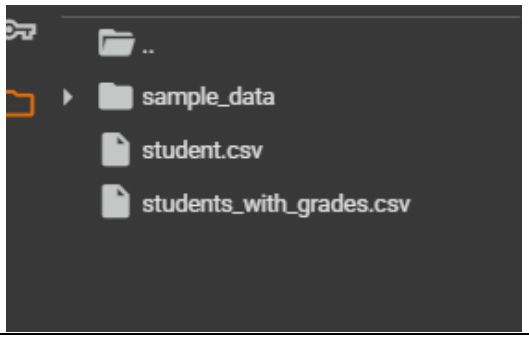
```
# Sort the DataFrame by mark in descending order
df_students_sorted = df_students.sort_values(by='mark', ascending=False)
print(df_students_sorted)
```

	id	name	class	mark	gender	grade
32	33	Kenn Rein	Six	96	female	A
11	12	Recky	Six	94	female	A
31	32	Binn Rott	Seven	90	female	B
10	11	Ronald	Six	89	female	B
30	31	Marry Toeey	Four	88	male	B
34	35	Rows Noump	Six	88	female	B
24	25	Giff Tow	Seven	88	male	B
14	15	Tade Row	NaN	88	male	B
15	16	Gimmy	Four	88	male	B
12	13	Kty	Seven	88	female	B
13	14	Bigy	Seven	88	female	B
27	28	Rojj Base	Seven	86	female	B
7	8	Asruid	Five	85	male	B
1	2	Max Ruin	Three	85	male	B
26	27	NaN	Three	81	NaN	B
29	30	Reppy Red	Six	79	female	C
25	26	Crelea	Seven	79	male	C
22	23	Herod	Eight	79	male	C
6	7	My John Rob	Fifth	78	male	C
23	24	Tiddy Now	Seven	78	male	C
8	9	Tes Qry	Six	78	NaN	C
17	18	Honny	Five	75	male	C
0	1	John Deo	Four	75	female	C
33	34	Gain Toe	Seven	69	male	D
20	21	Babby John	Four	69	female	D
19	20	Jackly	Nine	65	female	D
3	4	Krish Star	Four	60	female	F
4	5	John Mike	Four	60	female	F
2	3	Arnold	Three	55	male	F
5	6	Alex John	Four	55	male	F
9	10	Big John	Four	55	female	F
21	22	Reggid	Seven	55	female	F
28	29	Tess Played	Seven	55	male	F
16	17	Tumyu	Six	54	male	F
18	19	Tinny	Nine	18	male	F

Exercise 6: Exporting Data

1. Question: "Write the code to save the DataFrame with the new 'grade' column to a new CSV file."

```
# Save the DataFrame with the new 'grade' column to a new CSV file
df_students.to_csv('students_with_grades.csv', index=False)
```



students_with_grades.csv

1 to 10 of 35 entries Filter

id	name	class	mark	gender	grade
1	John Deo	Four	75	female	C
2	Max Ruin	Three	85	male	B
3	Arnold	Three	55	male	F
4	Krish Star	Four	60	female	F
5	John Mike	Four	60	female	F
6	Alex John	Four	55	male	F
7	My John Rob	Fifth	78	male	C
8	Asruid	Five	85	male	B
9	Tes Qry	Six	78		C
10	Big John	Four	55	female	F

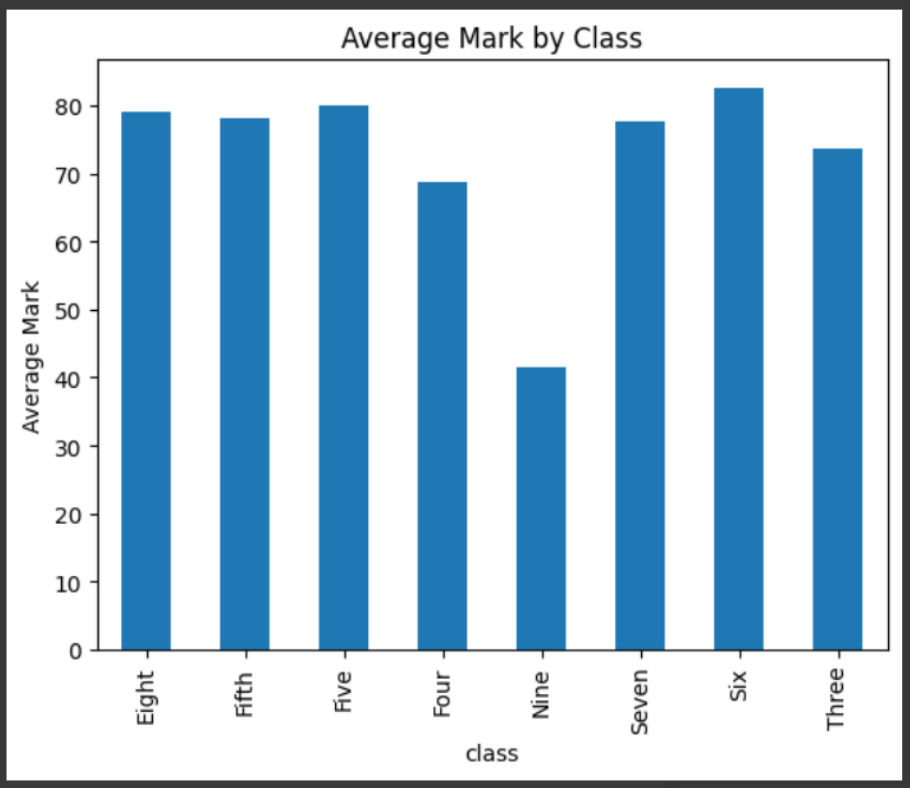
Show 10 per page 1 2 3 4

Exercise 7: If finished early try visualising the results

Basic Visualisations: Bar Chart, Line Graph and Scatterplot

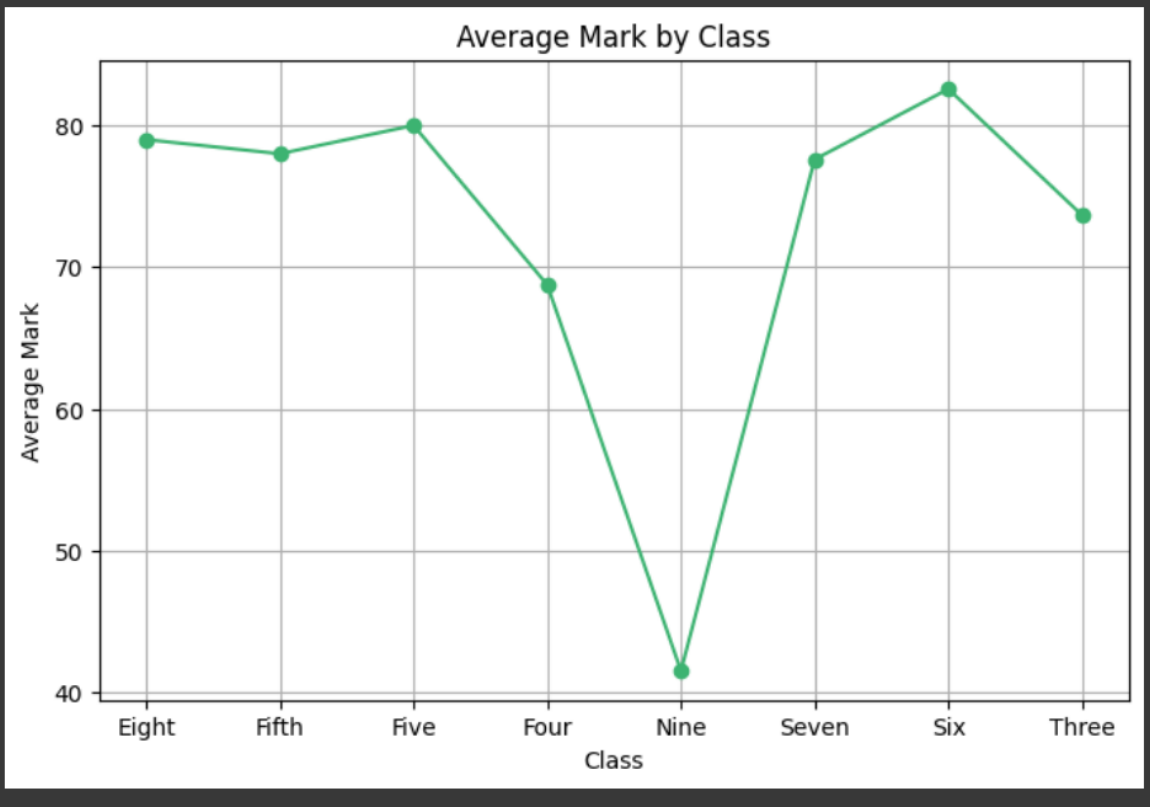
```python
# Import Matpoltlib.pyplot as plt
import matplotlib.pyplot as plt

# Bar chart showing Average Mark by Class
df_students.groupby('class')['mark'].mean().plot(kind='bar', title='Average Mark by Class')
plt.ylabel('Average Mark')
plt.show()
```



```
# Group by 'class' and calculate average marks
avg_marks = df_students.groupby('class')['mark'].mean()

# Average Mark by Class line graph
plt.figure(figsize=(8, 5))
avg_marks.plot(kind='line', marker='o', linestyle='-', color='mediumseagreen')
plt.title('Average Mark by Class')
plt.xlabel('Class')
plt.ylabel('Average Mark')
plt.grid(True)
plt.show()
```

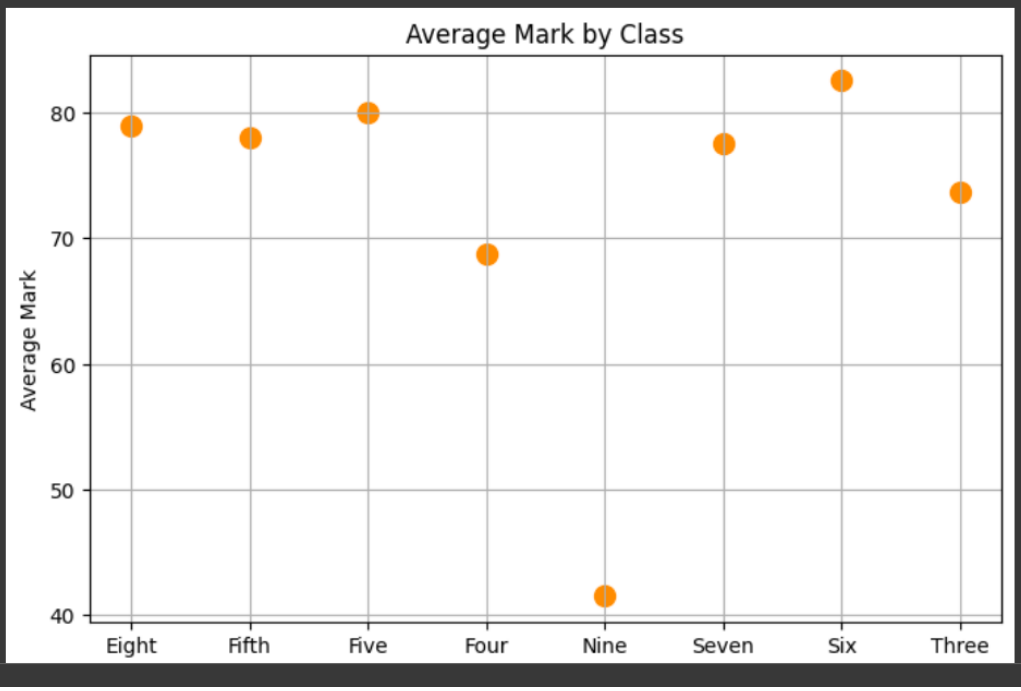


```python
# import seaborn as sns
import seaborn as sns

# Step 1: Calculate average mark per class
avg_marks_by_class = df_students.groupby('class')['mark'].mean()

# Step 2: Create scatter plot
plt.figure(figsize=(8, 5))
plt.scatter(avg_marks_by_class.index, avg_marks_by_class.values, color='darkorange', s=100)

# Step 3: Customize the plot - Average Mark by Class
plt.title('Average Mark by Class')
plt.xlabel('Class')
plt.ylabel('Average Mark')
plt.grid(True)
plt.show()
```



Day 4: Task 1

Using the 'GDP (nominal) per Capita.csv' which can be downloaded from the shared Folder, complete the below exercises and paste your input and output. Work individually, but we will work and support each other in the room.

- Read and save the 'GDP (nominal) per Capita' data to a data frame called "df" in Jyputer notebook
- Print the first 10 rows
- Print the last 5 rows
- Print 'Country/Territory' and 'UN_Region' columns

```python
# Import pandas
import pandas as pd

# Read and save the GDP data to data frame called df
df = pd.read_csv('GDP_per Capita.csv')

# Print the first 10 rows
print(df.head(10))

# Print the last 5 rows
print(df.tail(5))

# Print 'Country/Territory' and 'UN_Region' columns
print(df[['Country/Territory','UN_Region']])
```

```
   Unnamed: 0 Country/Territory UN_Region  IMF_Estimate  IMF_Year  \
0           1             Monaco    Europe             0         0
1           2      Liechtenstein    Europe             0         0
2           3         Luxembourg    Europe        132372      2023
3           4            Ireland    Europe        114581      2023
4           5            Bermuda  Americas             0         0
5           6             Norway    Europe        101103      2023
6           7        Switzerland    Europe         98767      2023
7           8          Singapore      Asia         91100      2023
8           9       Isle of Man    Europe             0         0
9          10     Cayman Islands  Americas             0         0
```

```
     Unnamed: 0 Country/Territory UN_Region  IMF_Estimate  IMF_Year  \
218         219            Malawi    Africa           496      2023
219         220       South Sudan    Africa           467      2023
220         221      Sierra Leone    Africa           415      2023
221         222       Afghanistan      Asia           611      2020
222         223           Burundi    Africa           249      2023
```

```
    Country/Territory UN_Region
0              Monaco    Europe
1       Liechtenstein    Europe
2          Luxembourg    Europe
3             Ireland    Europe
4             Bermuda  Americas
..                ...       ...
218           Malawi    Africa
219       South Sudan    Africa
220      Sierra Leone    Africa
221       Afghanistan      Asia
222           Burundi    Africa

[223 rows x 2 columns]
```

Day 4: Task 2

Back with 'GDP (nominal) per Capita'. As a group, import and work your way through the Day_4_Python_Activity.ipynb notebook which can be found on the shared Folder. There are questions to answer, but also opportunities to have fun with the data – paste your input and output below.

Once complete, and again as a group, work with some more data and have some fun – there is no set agenda for this section, other than to embed the skills developed this week. Paste your input and output below and upon return we'll discuss progress made.

Additional data found here.

Link to NoteBook:

https://colab.research.google.com/drive/1i-nL87SIX-Qii0n7YN84aeNsgZYirH7C

Course Notes

It is recommended to take notes from the course, use the space below to do so, or use the revision guide shared with the class:

https://www.youtube.com/watch?v=-E7nMqPVmyQ

We have included a range of additional links to further resources and information that you may find useful, these can be found within your revision guide.

END OF WORKBOOK

Please check through your work thoroughly before submitting and update the table of contents if required.

Please send your completed work booklet to your trainer.