Case Study 2

In [1]:
```
import pandas as pd
import matplotlib.pyplot as plt
```

In [5]:
```
#Task 1 solution-
df_math_score = pd.read_csv("MathScoreTerm1.csv")
df_physics_score = pd.read_csv("PhysicsScoreTerm1.csv")
df_ds_score = pd.read_csv("DSScoreTerm1.csv")

print(df_math_score.head())
print(df_physics_score.head())
print(df_ds_score.head())
```

```
              Name  Score  Age         Ethinicity Subject Sex  ID
0  AI-KYUNG CHUNG   88.0   18     White American   Maths   M   1
1     ALAN HARVEY   85.0   19  European American   Maths   M   2
2    ALAN REYNAUD   45.0   19  European American   Maths   M   3
3  ALBERT CENDANA   82.0   18     White American   Maths   M   4
4  ALBERT HOLT JR   82.0   18     White American   Maths   F   5
              Name  Score  Age         Ethinicity  Subject Sex  ID
0  AI-KYUNG CHUNG   84.0   18     White American  Physics   M   1
1     ALAN HARVEY   81.0   19  European American  Physics   M   2
2    ALAN REYNAUD   41.0   19  European American  Physics   M   3
3  ALBERT CENDANA   78.0   18     White American  Physics   M   4
4  ALBERT HOLT JR   78.0   18     White American  Physics   F   5
              Name  Score  Age         Ethinicity        Subject Sex  ID
0  AI-KYUNG CHUNG   82.0   18     White American  Data Structue   M   1
1     ALAN HARVEY   79.0   19  European American  Data Structue   M   2
2    ALAN REYNAUD   39.0   19  European American  Data Structue   M   3
3  ALBERT CENDANA   76.0   18     White American  Data Structue   M   4
4  ALBERT HOLT JR   76.0   18     White American  Data Structue   F   5
```

In [6]:
```
#TASK 2 solution- remove name and ethinicity to ensure confidentiality
del df_math_score["Name"]
del df_math_score["Ethinicity"]

del df_ds_score["Name"]
del df_ds_score["Ethinicity"]

del df_physics_score["Name"]
del df_physics_score["Ethinicity"]
```

In [9]:
```
#TASK 3 solution- fill the missing values with zero
df_math_score.fillna(0)
df_ds_score.fillna(0)
df_physics_score.fillna(0)

print(df_math_score.head())
print(df_ds_score.head())
print(df_physics_score.head())
```

```
   Score  Age Subject Sex  ID
0   88.0   18   Maths   M   1
1   85.0   19   Maths   M   2
2   45.0   19   Maths   M   3
3   82.0   18   Maths   M   4
4   82.0   18   Maths   F   5
   Score  Age        Subject Sex  ID
0   82.0   18  Data Structue   M   1
1   79.0   19  Data Structue   M   2
2   39.0   19  Data Structue   M   3
3   76.0   18  Data Structue   M   4
4   76.0   18  Data Structue   F   5
   Score  Age  Subject Sex  ID
0   84.0   18  Physics   M   1
1   81.0   19  Physics   M   2
2   41.0   19  Physics   M   3
3   78.0   18  Physics   M   4
4   78.0   18  Physics   F   5
```

In [10]:
```
#TASK 4 solution- merging three files
merged_df = df_math_score.merge(df_ds_score, on="ID", suffixes=('_math', '_ds'
)).merge(df_physics_score, on= "ID", suffixes=('_ds', '_physics'))
merged_df
```

Out[10]:

	Score_math	Age_math	Subject_math	Sex_math	ID	Score_ds	Age_ds	Subject_ds	Sex
0	88.0	18	Maths	M	1	82.0	18	Data Structue	
1	85.0	19	Maths	M	2	79.0	19	Data Structue	
2	45.0	19	Maths	M	3	39.0	19	Data Structue	
3	82.0	18	Maths	M	4	76.0	18	Data Structue	
4	82.0	18	Maths	F	5	76.0	18	Data Structue	
...	...	...	...	...	...	...	...	...	
594	45.0	19	Maths	F	595	39.0	19	Data Structue	
595	75.0	18	Maths	M	596	69.0	18	Data Structue	
596	53.0	20	Maths	M	597	47.0	20	Data Structue	
597	75.0	20	Maths	M	598	69.0	20	Data Structue	
598	88.0	19	Maths	M	599	NaN	19	Data Structue	

599 rows × 13 columns

In [24]:
```
merged_df_filter_cols = merged_df.filter(["ID", "Score_math", "Score_ds", "Sco
re", "Age_math", "Sex_math"]).rename(columns={'Score':'Score_physics', 'Age_ma
th':'Age', 'Sex_math': 'Sex'})
print(merged_df_filter_cols)
```

```
      ID  Score_math  Score_ds  Score_physics  Age Sex
0      1        88.0      82.0           84.0   18   M
1      2        85.0      79.0           81.0   19   M
2      3        45.0      39.0           41.0   19   M
3      4        82.0      76.0           78.0   18   M
4      5        82.0      76.0           78.0   18   F
..   ...         ...       ...            ...  ...  ..
594  595        45.0      39.0           41.0   19   F
595  596        75.0      69.0           71.0   18   M
596  597        53.0      47.0           49.0   20   M
597  598        75.0      69.0           71.0   20   M
598  599        88.0       NaN           69.0   19   M

[599 rows x 6 columns]
```

In [29]:
```
#TASK 5 solution- change sex column
merged_df_filter_cols["Sex"] = [1 if sex == "M" else 2 for sex in merged_df_fi
lter_cols["Sex"]]
merged_df_filter_cols
```

Out[29]:

	ID	Score_math	Score_ds	Score_physics	Age	Sex
0	1	88.0	82.0	84.0	18	1
1	2	85.0	79.0	81.0	19	1
2	3	45.0	39.0	41.0	19	1
3	4	82.0	76.0	78.0	18	1
4	5	82.0	76.0	78.0	18	2
...	...	...	...	...	...	...
594	595	45.0	39.0	41.0	19	2
595	596	75.0	69.0	71.0	18	1
596	597	53.0	47.0	49.0	20	1
597	598	75.0	69.0	71.0	20	1
598	599	88.0	NaN	69.0	19	1

599 rows × 6 columns

In [33]:
```
#TASK 6 solution- Store data in new file
merged_df_filter_cols.to_csv("ScoreFinal.csv")
```

In [34]:
```
df2 = pd.read_csv("ScoreFinal.csv")
df2
```

Out[34]:

	Unnamed: 0	ID	Score_math	Score_ds	Score_physics	Age	Sex
0	0	1	88.0	82.0	84.0	18	1
1	1	2	85.0	79.0	81.0	19	1
2	2	3	45.0	39.0	41.0	19	1
3	3	4	82.0	76.0	78.0	18	1
4	4	5	82.0	76.0	78.0	18	2
...	...	...	...	...	...	...	...
594	594	595	45.0	39.0	41.0	19	2
595	595	596	75.0	69.0	71.0	18	1
596	596	597	53.0	47.0	49.0	20	1
597	597	598	75.0	69.0	71.0	20	1
598	598	599	88.0	NaN	69.0	19	1

599 rows × 7 columns

In [ ]: