Machine Learning 14th February 2026

Danny Idukundatwese
Reg No: 2405001032

Question 1. Pandas DataFrame

```
[11]: import pandas as pd
      import numpy as np
      import matplotlib.pyplot

      data = {
          'Student_ID': ['S1','S2','S3','S4','S5','S6','S7','S8','S9'],
          'Age': ['20','21',None,'19','22','twenty','23','24',None],
          'Score': [85,np.nan,78,90,88,92,np.nan,87,80],
          'Hours_Studied': [10,15,7,np.nan,12,9,14,11,8]
      }

      df = pd.DataFrame(data)

      print(df)
```

	Student_ID	Age	Score	Hours_Studied
0	S1	20	85.0	10.0
1	S2	21	NaN	15.0
2	S3	None	78.0	7.0
3	S4	19	90.0	NaN
4	S5	22	88.0	12.0
5	S6	twenty	92.0	9.0
6	S7	23	NaN	14.0
7	S8	24	87.0	11.0
8	S9	None	80.0	8.0

Question 2. Dataset and contents summary

```
[93]: print('Datasets Summary: \n')
      print(df.info())
```

```
Datasets Summary:

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 9 entries, 0 to 8
Data columns (total 4 columns):
 #   Column         Non-Null Count  Dtype
---  ------         --------------  -----
 0   Student_ID     9 non-null      object
 1   Age            9 non-null      object
 2   Score          7 non-null      float64
 3   Hours_Studied  8 non-null      float64
dtypes: float64(2), object(2)
memory usage: 420.0+ bytes
None
```

Question 3. Missing Data Summary

```
[94]: print(f'\nMissing Data Summary: \n{df.isnull().sum()}')
```

```
Missing Data Summary:
Student_ID       0
Age              0
Score            2
Hours_Studied    1
dtype: int64
```

Question 4. Converting Age and Score columns

```
[95]: print('Converting Age and Score Columns to numeric type: \n')
      df['Age'] = pd.to_numeric(df['Age'])
      df['Score'] = pd.to_numeric(df['Score'])
      print(df)
```

```
Converting Age and Score Columns to numeric type:
```

	Student_ID	Age	Score	Hours_Studied
0	S1	20.0	85.0	10.0
1	S2	21.0	NaN	15.0
2	S3	NaN	78.0	7.0
3	S4	19.0	90.0	NaN
4	S5	22.0	88.0	12.0
5	S6	20.0	92.0	9.0
6	S7	23.0	NaN	14.0
7	S8	24.0	87.0	11.0
8	S9	NaN	80.0	8.0

Question 5. Treating all the Missing Data

```
[99]: print('\nTreating the Missing Values:')
      df['Age'].fillna(df['Age'].mean(), inplace=True)
      df['Score'].fillna(df['Score'].mean(), inplace=True)
      df['Hours_Studied'].fillna(df['Hours_Studied'].mean(), inplace=True)
      print(df)
```

```
Treating the Missing Values:
  Student_ID        Age      Score  Hours_Studied
0         S1  20.000000  85.000000          10.00
1         S2  21.000000  85.714286          15.00
2         S3  21.285714  78.000000           7.00
3         S4  19.000000  90.000000          10.75
4         S5  22.000000  88.000000          12.00
5         S6  20.000000  92.000000           9.00
6         S7  23.000000  85.714286          14.00
7         S8  24.000000  87.000000          11.00
8         S9  21.285714  80.000000           8.00
```

```
C:\Users\LabStudent\AppData\Local\Temp\ipykernel_1888\426333949.py:2: FutureWarning: A value is trying to be set on a copy of a DataFrame or Series throu
gh chained assignment using an inplace method.
The behavior will change in pandas 3.0. This inplace method will never work because the intermediate object on which we are setting values always behaves
as a copy.

For example, when doing 'df[col].method(value, inplace=True)', try using 'df.method({col: value}, inplace=True)' or df[col] = df[col].method(value) inste
ad, to perform the operation inplace on the original object.


  df['Age'].fillna(df['Age'].mean(), inplace=True)
```

Question 6. Plot a scatter plot

```
[100]: x = df['Hours_Studied']
       y = df['Score']
       plt.scatter(x,y,color='red')
       plt.title('Hours Studied vs Score')
       plt.xlabel('Hours Studied')
       plt.ylabel('Score')
       plt.show()
```