FUTURESKILLS AI BOOTCAMP ASSIGNMENT 1

Problem Statement:

Problem Statement:
Clean and preprocess a student dataset by handling missing values, encoding categorical data, correcting errors, and standardizing numerical columns.

This block will import essential libraries for data handling and processing.

```
import pandas as pd
import numpy as np
from sklearn.preprocessing import LabelEncoder, StandardScaler
from google.colab import files
```

We will then proceed to load the file in a pandas dataframe.

```
df = pd.read_csv('/content/Students.csv')  # Update path if required
df.head()  # Display the first few rows
```

	StudentID	Name	Gender	Age	MathScore	EnglishScore	TotalScore
0	1	Alice	Female	15	85.0	90.0	170
1	2	Bob	Male	16	NaN	95.0	95
2	3	Charlie	Male	15	80.0	NaN	80
3	4	Diana	Female	16	90.0	85.0	170
4	5	Eve	Female	15	75.0	70.0	140

Next steps: Generate code with df View recommended plots New interactive sheet

The null values in the EnglishScore and MathScore columns have to be filled by an appropriate value, in this case the arithmetic mean of their respective columns.

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

```
<ipython-input-4-c8b5a678fbb4>:1: FutureWarning: A value is trying to be set on a copy of
The behavior will change in pandas 3.0. This inplace method will never work because the i

For example, when doing 'df[col].method(value, inplace=True)', try using 'df.method({col:


  df['MathScore'].fillna(df['MathScore'].mean(), inplace=True)
<ipython-input-4-c8b5a678fbb4>:2: FutureWarning: A value is trying to be set on a copy
The behavior will change in pandas 3.0. This inplace method will never work because the
```

Stu
1
2
3
4
5
Sh
10
pa

```
  For example, when doing 'df[col].method(value, inplace=True)', try using 'df.method({col:

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

The categorical data of the Gender column has to be converted to a binary form

```
encoder = LabelEncoder()
df['Gender'] = encoder.fit_transform(df['Gender'])
```

The TotalScore Column should equate to the summation of the MathScore and EnglishScore Columns

```
df['TotalScore'] = df['MathScore'] + df['EnglishScore']
```

Now we can finally standardise the dataset using IQR [INTERQUARTILE RANGE]

```
def iqr_standardize(column):
    Q1 = column.quantile(0.25)
    Q3 = column.quantile(0.75)
    IQR = Q3 - Q1
    return (column - Q1) / IQR  # Scales values using IQR

df[['MathScore', 'EnglishScore', 'TotalScore']] = df[['MathScore', 'EnglishScore', 'TotalScore'
```

The dataset below is the resultant that we will end up with after all of these operations.

```
df.to_csv('/content/Cleaned_Students.csv', index=False)
print("Cleaned dataset saved successfully!")

files.download('/content/Cleaned_Students.csv')
```

```
Cleaned dataset saved successfully!
```