In [ ]:
```
#TASK 1: Understanding Dataset & Data Types

Dataset Used: Student Performance Dataset
```

In [ ]:
```
🔹 Step 1: Load the Dataset & Inspect Rows
```

In [5]:
```
import pandas as pd

df = pd.read_csv("student-por.csv")
df.head()
df.tail()
```

Out[5]:

	school	sex	age	address	famsize	Pstatus	Medu	Fedu	Mjob	Fjob	...	fa
644	MS	F	19	R	GT3	T	2	3	services	other	...	
645	MS	F	18	U	LE3	T	3	1	teacher	services	...	
646	MS	F	18	U	GT3	T	1	1	other	other	...	
647	MS	M	17	U	LE3	T	3	1	services	services	...	
648	MS	M	18	R	LE3	T	3	2	services	other	...	

5 rows × 33 columns

In [ ]:
```
🔹 Step 2: Dataset Structure & Data Types
```

In [9]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 649 entries, 0 to 648
Data columns (total 33 columns):
 #   Column      Non-Null Count  Dtype 
---  ------      --------------  ----- 
 0   school      649 non-null    object
 1   sex         649 non-null    object
 2   age         649 non-null    int64 
 3   address     649 non-null    object
 4   famsize     649 non-null    object
 5   Pstatus     649 non-null    object
 6   Medu        649 non-null    int64 
 7   Fedu        649 non-null    int64 
 8   Mjob        649 non-null    object
 9   Fjob        649 non-null    object
 10  reason      649 non-null    object
 11  guardian    649 non-null    object
 12  traveltime  649 non-null    int64 
 13  studytime   649 non-null    int64 
 14  failures    649 non-null    int64 
 15  schoolsup   649 non-null    object
 16  famsup      649 non-null    object
 17  paid        649 non-null    object
 18  activities  649 non-null    object
 19  nursery     649 non-null    object
 20  higher      649 non-null    object
 21  internet    649 non-null    object
 22  romantic    649 non-null    object
 23  famrel      649 non-null    int64 
 24  freetime    649 non-null    int64 
 25  goout       649 non-null    int64 
 26  Dalc        649 non-null    int64 
 27  Walc        649 non-null    int64 
 28  health      649 non-null    int64 
 29  absences    649 non-null    int64 
 30  G1          649 non-null    int64 
 31  G2          649 non-null    int64 
 32  G3          649 non-null    int64 
dtypes: int64(16), object(17)
memory usage: 167.4+ KB
```

In [ ]:
```
Dataset Overview

Total rows: 649 students

Total columns: 33 features

Memory usage: ~167 KB
```

In [ ]:
```
🔹 STEP 4: Statistical Summary
```

In [11]:
```
df.describe()
```

Out[11]:

	age	Medu	Fedu	traveltime	studytime	failures	fam
count	649.000000	649.000000	649.000000	649.000000	649.000000	649.000000	649.0000
mean	16.744222	2.514638	2.306626	1.568567	1.930663	0.221880	3.9306
std	1.218138	1.134552	1.099931	0.748660	0.829510	0.593235	0.9557
min	15.000000	0.000000	0.000000	1.000000	1.000000	0.000000	1.0000
25%	16.000000	2.000000	1.000000	1.000000	1.000000	0.000000	4.0000
50%	17.000000	2.000000	2.000000	1.000000	2.000000	0.000000	4.0000
75%	18.000000	4.000000	3.000000	2.000000	2.000000	0.000000	5.0000
max	22.000000	4.000000	4.000000	4.000000	4.000000	3.000000	5.0000

In [ ]:
```
🔹 STEP 5: Check Categorical Distributions
```

In [13]:
```
df['sex'].value_counts()
df['school'].value_counts()
df['higher'].value_counts()
```

Out[13]:
```
higher
yes    580
no      69
Name: count, dtype: int64
```

In [ ]:
```
Observation

Majority of students (580 out of 649) want to pursue higher education

Only 69 students do not plan to continue further studies

This shows a clear imbalance in the higher feature.
```

In [ ]:
```
🔹 STEP 6: Dataset Size & ML Suitability
```

In [17]:
```
df.shape
```

Out[17]:
```
(649, 33)
```

In [ ]:
```
Observation:
Dataset has 649 students × 33 features

Suitable for regression models

Enough data for training & testing

This dataset is ML-ready after encoding.
```

In [21]:
```
#Check Unique Values in Categorical Columns
categorical_cols = df.select_dtypes(include='object').columns

for col in categorical_cols:
    print(f"\n{col}:\n", df[col].value_counts())
```

```
school:
 school
GP    423
MS    226
Name: count, dtype: int64

sex:
 sex
F    383
M    266
Name: count, dtype: int64

address:
 address
U    452
R    197
Name: count, dtype: int64

famsize:
 famsize
GT3    457
LE3    192
Name: count, dtype: int64

Pstatus:
 Pstatus
T    569
A     80
Name: count, dtype: int64

Mjob:
 Mjob
other       258
services    136
at_home     135
teacher      72
health       48
Name: count, dtype: int64

Fjob:
 Fjob
other       367
services    181
at_home      42
teacher      36
health       23
Name: count, dtype: int64

reason:
 reason
course        285
home          149
reputation    143
other          72
Name: count, dtype: int64

guardian:
 guardian
mother    455
father    153
```

```
other      41
Name: count, dtype: int64

schoolsup:
 schoolsup
no     581
yes     68
Name: count, dtype: int64

famsup:
 famsup
yes    398
no     251
Name: count, dtype: int64

paid:
 paid
no     610
yes     39
Name: count, dtype: int64

activities:
 activities
no     334
yes    315
Name: count, dtype: int64

nursery:
 nursery
yes    521
no     128
Name: count, dtype: int64

higher:
 higher
yes    580
no      69
Name: count, dtype: int64

internet:
 internet
yes    498
no     151
Name: count, dtype: int64

romantic:
 romantic
no     410
yes    239
Name: count, dtype: int64
```

In [23]:
```
#Observation
Description

Unique value analysis of categorical columns helps understand the distribution c
```

In [27]:
```
#Identify Target Variable & Input Features
#target variable
target = 'G3'
```

In [29]:
```
#Input Features
features = df.drop(columns=['G3'])
```

In [ ]:
```
#observation
The target variable is G3, which represents the final grade of students. All oth
```

In [31]:
```
df.shape
```

Out[31]:
```
(649, 33)
```

In [33]:
```
#Data Quality Issues & Observations
df.isnull().sum()
```

Out[33]:
```
school        0
sex           0
age           0
address       0
famsize       0
Pstatus       0
Medu          0
Fedu          0
Mjob          0
Fjob          0
reason        0
guardian      0
traveltime    0
studytime     0
failures      0
schoolsup     0
famsup        0
paid          0
activities    0
nursery       0
higher        0
internet      0
romantic      0
famrel        0
freetime      0
goout         0
Dalc          0
Walc          0
health        0
absences      0
G1            0
G2            0
G3            0
dtype: int64
```

In [ ]:
```
#Observation
The dataset contains no missing values across all features, indicating high data
```