```
!pip install kaggle
```

```
Requirement already satisfied: kaggle in
/usr/local/lib/python3.12/dist-packages (1.7.4.5)
Requirement already satisfied: bleach in
/usr/local/lib/python3.12/dist-packages (from kaggle) (6.2.0)
Requirement already satisfied: certifi>=14.05.14 in
/usr/local/lib/python3.12/dist-packages (from kaggle) (2025.8.3)
Requirement already satisfied: charset-normalizer in
/usr/local/lib/python3.12/dist-packages (from kaggle) (3.4.3)
Requirement already satisfied: idna in /usr/local/lib/python3.12/dist-
packages (from kaggle) (3.10)
Requirement already satisfied: protobuf in
/usr/local/lib/python3.12/dist-packages (from kaggle) (5.29.5)
Requirement already satisfied: python-dateutil>=2.5.3 in
/usr/local/lib/python3.12/dist-packages (from kaggle) (2.9.0.post0)
Requirement already satisfied: python-slugify in
/usr/local/lib/python3.12/dist-packages (from kaggle) (8.0.4)
Requirement already satisfied: requests in
/usr/local/lib/python3.12/dist-packages (from kaggle) (2.32.4)
Requirement already satisfied: setuptools>=21.0.0 in
/usr/local/lib/python3.12/dist-packages (from kaggle) (75.2.0)
Requirement already satisfied: six>=1.10 in
/usr/local/lib/python3.12/dist-packages (from kaggle) (1.17.0)
Requirement already satisfied: text-unidecode in
/usr/local/lib/python3.12/dist-packages (from kaggle) (1.3)
Requirement already satisfied: tqdm in /usr/local/lib/python3.12/dist-
packages (from kaggle) (4.67.1)
Requirement already satisfied: urllib3>=1.15.1 in
/usr/local/lib/python3.12/dist-packages (from kaggle) (2.5.0)
Requirement already satisfied: webencodings in
/usr/local/lib/python3.12/dist-packages (from kaggle) (0.5.1)
```

```
from google.colab import files
print("Please upload your kaggle.json file")
files.upload()
```

```
Please upload your kaggle.json file
```

```
<IPython.core.display.HTML object>
```

```
Saving kaggle.json to kaggle.json
```

```
{'kaggle.json':
b'{"username":"hiruna1","key":"9714130537f084215673f2081071682b"}'}
```

```
# Step 3: Configure Kaggle and download the dataset
!mkdir -p ~/.kaggle
!cp kaggle.json ~/.kaggle/
!chmod 600 ~/.kaggle/kaggle.json
```

```
!kaggle datasets download hopesb/student-depression-dataset

Dataset URL: https://www.kaggle.com/datasets/hopesb/student-
depression-dataset
License(s): apache-2.0
Downloading student-depression-dataset.zip to /content
  0% 0.00/454k [00:00<?, ?B/s]
100% 454k/454k [00:00<00:00, 221MB/s]

# Step 4: Unzip the dataset
!unzip student-depression-dataset.zip
print("\n Kaggle dataset downloaded and ready to use!")

Archive:  student-depression-dataset.zip
  inflating: Student Depression Dataset.csv

 Kaggle dataset downloaded and ready to use!

import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt

sns.set(style="whitegrid")

file_path = 'Student Depression Dataset.csv'
df = pd.read_csv(file_path)
```

--- **Exploratory Data Analysis** ---

First 5 Rows of the Dataset

```
display(df.head())

{"summary":"{\n  \"name\": \"display(df\",\n  \"rows\": 5,\n
\"fields\": [\n    {\n      \"column\": \"id\",\n      \"properties\":
{\n        \"dtype\": \"number\",\n        \"std\": 13,\n
\"min\": 2,\n        \"max\": 32,\n        \"num_unique_values\": 5,\n
\"samples\": [\n          8,\n          32,\n          26\n        ],\
n        \"semantic_type\": \"\",\n        \"description\": \"\"\n
}\n    },\n    {\n      \"column\": \"Gender\",\n      \"properties\":
{\n        \"dtype\": \"category\",\n        \"num_unique_values\":
2,\n        \"samples\": [\n          \"Female\",\n          \"Male\"\
n        ],\n        \"semantic_type\": \"\",\n
\"description\": \"\"\n      }\n    },\n    {\n      \"column\":
\"Age\",\n      \"properties\": {\n        \"dtype\": \"number\",\n
\"std\": 3.8340579025361627,\n        \"min\": 24.0,\n        \"max\":
33.0,\n        \"num_unique_values\": 5,\n        \"samples\": [\n
24.0,\n          25.0\n        ],\n        \"semantic_type\": \"\",\n
\"description\": \"\"\n      }\n    },\n    {\n      \"column\":
\"City\",\n      \"properties\": {\n        \"dtype\": \"string\",\n
```

```
\"num_unique_values\": 5,\n        \"samples\": [\n
\"Bangalore\",\n          \"Jaipur\"\n        ],\n
\"semantic_type\": \"\",\n        \"description\": \"\"\n      }\
n    },\n    {\n      \"column\": \"Profession\",\n
\"properties\": {\n        \"dtype\": \"category\",\n
\"num_unique_values\": 1,\n        \"samples\": [\n
\"Student\"\n        ],\n        \"semantic_type\": \"\",\n
\"description\": \"\"\n      }\n    },\n    {\n      \"column\":
\"Academic Pressure\",\n      \"properties\": {\n        \"dtype\":
\"number\",\n        \"std\": 1.140175425099138,\n        \"min\":
2.0,\n        \"max\": 5.0,\n        \"num_unique_values\": 4,\n
\"samples\": [\n          2.0\n        ],\n        \"semantic_type\":
\"\",\n        \"description\": \"\"\n      }\n    },\n    {\n
\"column\": \"Work Pressure\",\n      \"properties\": {\n
\"dtype\": \"number\",\n        \"std\": 0.0,\n        \"min\": 0.0,\n
\"max\": 0.0,\n        \"num_unique_values\": 1,\n        \"samples\":
[\n          0.0\n        ],\n        \"semantic_type\": \"\",\n
\"description\": \"\"\n      }\n    },\n    {\n      \"column\":
\"CGPA\",\n      \"properties\": {\n        \"dtype\": \"number\",\n
\"std\": 1.4387425064965589,\n        \"min\": 5.59,\n        \"max\":
8.97,\n        \"num_unique_values\": 5,\n        \"samples\": [\n
5.9\n        ],\n        \"semantic_type\": \"\",\n
\"description\": \"\"\n      }\n    },\n    {\n      \"column\":
\"Study Satisfaction\",\n      \"properties\": {\n        \"dtype\":
\"number\",\n        \"std\": 1.5165750888103102,\n        \"min\":
2.0,\n        \"max\": 5.0,\n        \"num_unique_values\": 3,\n
\"samples\": [\n          2.0\n        ],\n        \"semantic_type\":
\"\",\n        \"description\": \"\"\n      }\n    },\n    {\n
\"column\": \"Job Satisfaction\",\n      \"properties\": {\n
\"dtype\": \"number\",\n        \"std\": 0.0,\n        \"min\": 0.0,\n
\"max\": 0.0,\n        \"num_unique_values\": 1,\n        \"samples\":
[\n          0.0\n        ],\n        \"semantic_type\": \"\",\n
\"description\": \"\"\n      }\n    },\n    {\n      \"column\":
\"Sleep Duration\",\n      \"properties\": {\n        \"dtype\":
\"string\",\n        \"num_unique_values\": 3,\n        \"samples\":
[\n          \"5-6 hours\"\n        ],\n        \"semantic_type\":
\"\",\n        \"description\": \"\"\n      }\n    },\n    {\n
\"column\": \"Dietary Habits\",\n      \"properties\": {\n
\"dtype\": \"category\",\n        \"num_unique_values\": 2,\n
\"samples\": [\n          \"Moderate\"\n        ],\n
\"semantic_type\": \"\",\n        \"description\": \"\"\n      }\
n    },\n    {\n      \"column\": \"Degree\",\n      \"properties\":
{\n        \"dtype\": \"string\",\n        \"num_unique_values\": 5,\n
\"samples\": [\n          \"BSc\"\n        ],\n
\"semantic_type\": \"\",\n        \"description\": \"\"\n      }\
n    },\n    {\n      \"column\": \"Have you ever had suicidal
thoughts ?\",\n      \"properties\": {\n        \"dtype\":
\"category\",\n        \"num_unique_values\": 2,\n        \"samples\":
[\n          \"No\"\n        ],\n        \"semantic_type\": \"\",\n
```

```
\"description\": \"\"\n      }\n    },\n    {\n      \"column\":
\"Work/Study Hours\",\n      \"properties\": {\n        \"dtype\":
\"number\",\n        \"std\": 3.0,\n        \"min\": 1.0,\n
\"max\": 9.0,\n        \"num_unique_values\": 4,\n        \"samples\":
[\n          9.0\n        ],\n        \"semantic_type\": \"\",\n
\"description\": \"\"\n      }\n    },\n    {\n      \"column\":
\"Financial Stress\",\n      \"properties\": {\n        \"dtype\":
\"number\",\n        \"std\": 1.7320508075688772,\n        \"min\":
1.0,\n        \"max\": 5.0,\n        \"num_unique_values\": 3,\n
\"samples\": [\n          1.0\n        ],\n        \"semantic_type\":
\"\",\n        \"description\": \"\"\n      }\n    },\n    {\n
\"column\": \"Family History of Mental Illness\",\n
\"properties\": {\n        \"dtype\": \"category\",\n
\"num_unique_values\": 2,\n        \"samples\": [\n          \"Yes\"\n
],\n        \"semantic_type\": \"\",\n        \"description\": \"\"\n
}\n    },\n    {\n      \"column\": \"Depression\",\n
\"properties\": {\n        \"dtype\": \"number\",\n        \"std\":
0,\n        \"min\": 0,\n        \"max\": 1,\n
\"num_unique_values\": 2,\n        \"samples\": [\n          0\n
],\n        \"semantic_type\": \"\",\n        \"description\": \"\"\n
}\n    }\n  ]\n}","type":"dataframe"}
```


summary of the DataFrame, including data types and non-null counts

```
df.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 27901 entries, 0 to 27900
Data columns (total 18 columns):
 #   Column                                 Non-Null Count  Dtype
---  ------                                 --------------  -----
 0   id                                     27901 non-null  int64
 1   Gender                                 27901 non-null  object
 2   Age                                    27901 non-null  float64
 3   City                                   27901 non-null  object
 4   Profession                             27901 non-null  object
 5   Academic Pressure                      27901 non-null  float64
 6   Work Pressure                          27901 non-null  float64
 7   CGPA                                   27901 non-null  float64
 8   Study Satisfaction                     27901 non-null  float64
 9   Job Satisfaction                       27901 non-null  float64
 10  Sleep Duration                         27901 non-null  object
 11  Dietary Habits                         27901 non-null  object
 12  Degree                                 27901 non-null  object
 13  Have you ever had suicidal thoughts ?  27901 non-null  object
 14  Work/Study Hours                       27901 non-null  float64
 15  Financial Stress                       27898 non-null  float64
 16  Family History of Mental Illness       27901 non-null  object
 17  Depression                             27901 non-null  int64
```

```
dtypes: float64(8), int64(2), object(8)
memory usage: 3.8+ MB
```

Descriptive Statistics

```
display(df.describe())
```


```
{"summary":"{\n  \"name\": \"display(df\",\n  \"rows\": 8,\n  \"fields\": [\n    {\n      \"column\": \"id\",\n      \"properties\": {\n        \"dtype\": \"number\",\n        \"std\": 45429.64999853807,\n        \"min\": 2.0,\n        \"max\": 140699.0,\n        \"num_unique_values\": 8,\n        \"samples\": [\n          70442.1494211677,\n          70684.0,\n          27901.0\n        ],\n        \"semantic_type\": \"\",\n        \"description\": \"\"\n      }\n    },\n    {\n      \"column\": \"Age\",\n      \"properties\": {\n        \"dtype\": \"number\",\n        \"std\": 9855.22536952496,\n        \"min\": 4.90568744892443,\n        \"max\": 27901.0,\n        \"num_unique_values\": 8,\n        \"samples\": [\n          25.82230027597577,\n          25.0,\n          27901.0\n        ],\n        \"semantic_type\": \"\",\n        \"description\": \"\"\n      }\n    },\n    {\n      \"column\": \"Academic Pressure\",\n      \"properties\": {\n        \"dtype\": \"number\",\n        \"std\": 9863.557735797185,\n        \"min\": 0.0,\n        \"max\": 27901.0,\n        \"num_unique_values\": 8,\n        \"samples\": [\n          3.1412135765743163,\n          3.0,\n          27901.0\n        ],\n        \"semantic_type\": \"\",\n        \"description\": \"\"\n      }\n    },\n    {\n      \"column\": \"Work Pressure\",\n      \"properties\": {\n        \"dtype\": \"number\",\n        \"std\": 9864.23852387096,\n        \"min\": 0.0,\n        \"max\": 27901.0,\n        \"num_unique_values\": 5,\n        \"samples\": [\n          0.00043009211139385684,\n          5.0,\n          0.043992032063926795\n        ],\n        \"semantic_type\": \"\",\n        \"description\": \"\"\n      }\n    },\n    {\n      \"column\": \"CGPA\",\n      \"properties\": {\n        \"dtype\": \"number\",\n        \"std\": 9862.367065896231,\n        \"min\": 0.0,\n        \"max\": 27901.0,\n        \"num_unique_values\": 8,\n        \"samples\": [\n          7.65610417189348,\n          7.77,\n          27901.0\n        ],\n        \"semantic_type\": \"\",\n        \"description\": \"\"\n      }\n    },\n    {\n      \"column\": \"Study Satisfaction\",\n      \"properties\": {\n        \"dtype\": \"number\",\n        \"std\": 9863.56873016315,\n        \"min\": 0.0,\n        \"max\": 27901.0,\n        \"num_unique_values\": 8,\n        \"samples\": [\n          2.943837138453819,\n          3.0,\n          27901.0\n        ],\n        \"semantic_type\": \"\",\n        \"description\": \"\"\n      }\n    },\n    {\n      \"column\": \"Job Satisfaction\",\n      \"properties\": {\n        \"dtype\": \"number\",\n        \"std\": 9864.288942729796,\n        \"min\": 0.0,\n        \"max\": 27901.0,\n        \"num_unique_values\": 5,\n        \"samples\": [\n          0.0006809791763736067,\n          4.0,\n
```

```
0.044394396218617196\n        ],\n        \"semantic_type\": \"\",\n
\"description\": \"\"\n      }\n    },\n    {\n      \"column\":
\"Work/Study Hours\",\n      \"properties\": {\n        \"dtype\":
\"number\",\n        \"std\": 9862.227879710157,\n        \"min\":
0.0,\n        \"max\": 27901.0,\n        \"num_unique_values\": 8,\n
\"samples\": [\n          7.156983620658758,\n          8.0,\n
27901.0\n        ],\n        \"semantic_type\": \"\",\n
\"description\": \"\"\n      }\n    },\n    {\n      \"column\":
\"Financial Stress\",\n      \"properties\": {\n        \"dtype\":
\"number\",\n        \"std\": 9862.443780334284,\n        \"min\":
1.0,\n        \"max\": 27898.0,\n        \"num_unique_values\": 8,\n
\"samples\": [\n          3.1398666571080365,\n          3.0,\n
27898.0\n        ],\n        \"semantic_type\": \"\",\n
\"description\": \"\"\n      }\n    },\n    {\n      \"column\":
\"Depression\",\n      \"properties\": {\n        \"dtype\":
\"number\",\n        \"std\": 9864.287182360762,\n        \"min\":
0.0,\n        \"max\": 27901.0,\n        \"num_unique_values\": 5,\n
\"samples\": [\n          0.5854987276441704,\n          1.0,\n
0.49264456369312454\n        ],\n        \"semantic_type\": \"\",\n
\"description\": \"\"\n      }\n    }\n  ]\n}","type":"dataframe"}
```


Separate into Features (X) and Target (y)

```
y = df['Depression']
X = df.drop(columns=['Depression'], errors='ignore')
```

--- Handling Missing Values ---

```
missing_values = df.isnull().sum()
print(missing_values[missing_values > 0])
```

```
Financial Stress    3
dtype: int64
```

** Create the bar chart to show the missing values**

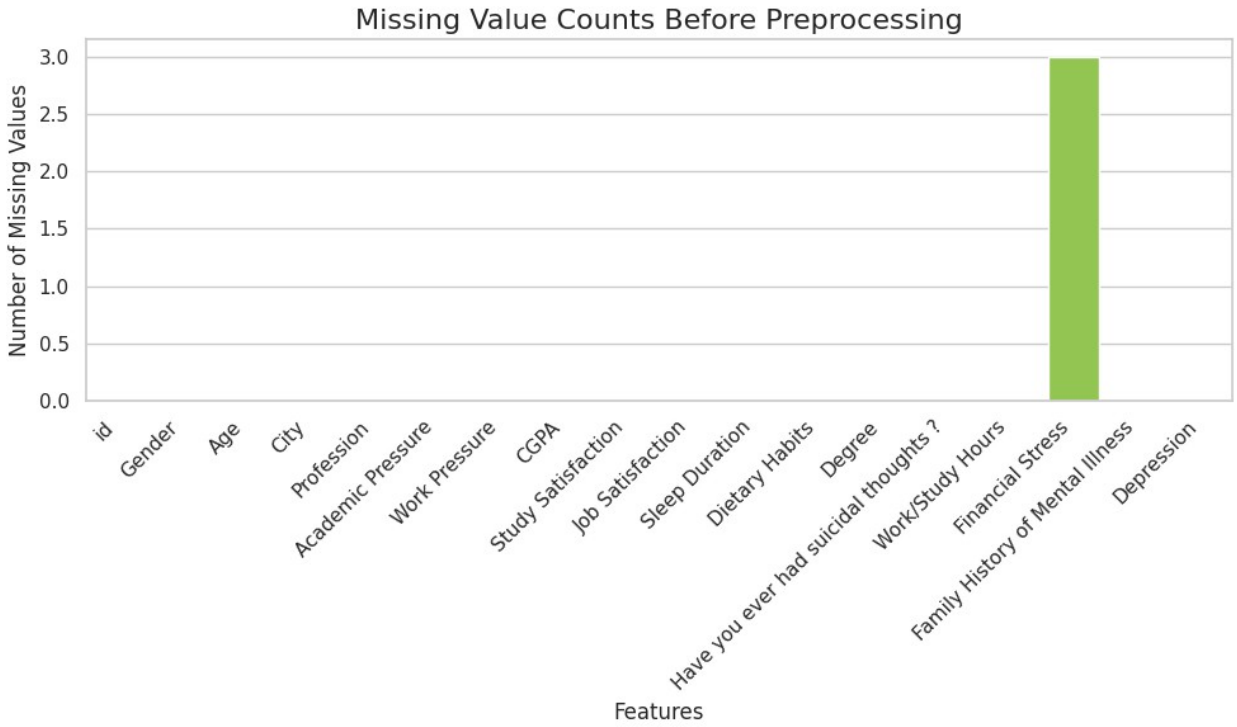
```
plt.figure(figsize=(10, 6))
sns.barplot(x=missing_values.index, y=missing_values.values,
palette="viridis")
plt.xticks(rotation=45, ha='right')
plt.title('Missing Value Counts Before Preprocessing', fontsize=16)
plt.xlabel('Features', fontsize=12)
plt.ylabel('Number of Missing Values', fontsize=12)
plt.tight_layout() # Adjust layout to make room for labels
plt.savefig('missing_values_before.png')
print("✅ 'missing_values_before.png' has been saved.")
plt.show()
```

```
/tmp/ipython-input-710432138.py:2: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be
removed in v0.14.0. Assign the `x` variable to `hue` and set
`legend=False` for the same effect.

  sns.barplot(x=missing_values.index, y=missing_values.values,
palette="viridis")

 'missing_values_before.png' has been saved.
```



Fill missing numerical values with the MEDIAN

```
numerical_cols = X.select_dtypes(include=np.number).columns

for col in numerical_cols:
    if X[col].isnull().any():
        median_val = X[col].median()
        X[col].fillna(median_val, inplace=True)
        print(f"Filled missing values in '{col}' with median:
{median_val}")

Filled missing values in 'Financial Stress' with median: 3.0

/tmp/ipython-input-1308849357.py:4: FutureWarning: A value is trying
to be set on a copy of a DataFrame or Series through 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) instead, to perform the operation inplace on the
original object.


  X[col].fillna(median_val, inplace=True)
```

Verify that all missing values have been handled

```
print(f"\nTotal missing values remaining in X:
{X.isnull().sum().sum()}")


Total missing values remaining in X: 0
```

Combine X and Y back into one DataFrame

```
final_cleaned_df = pd.concat([X, y], axis=1)
```

Create a plot to visually confirm that no missing values are left

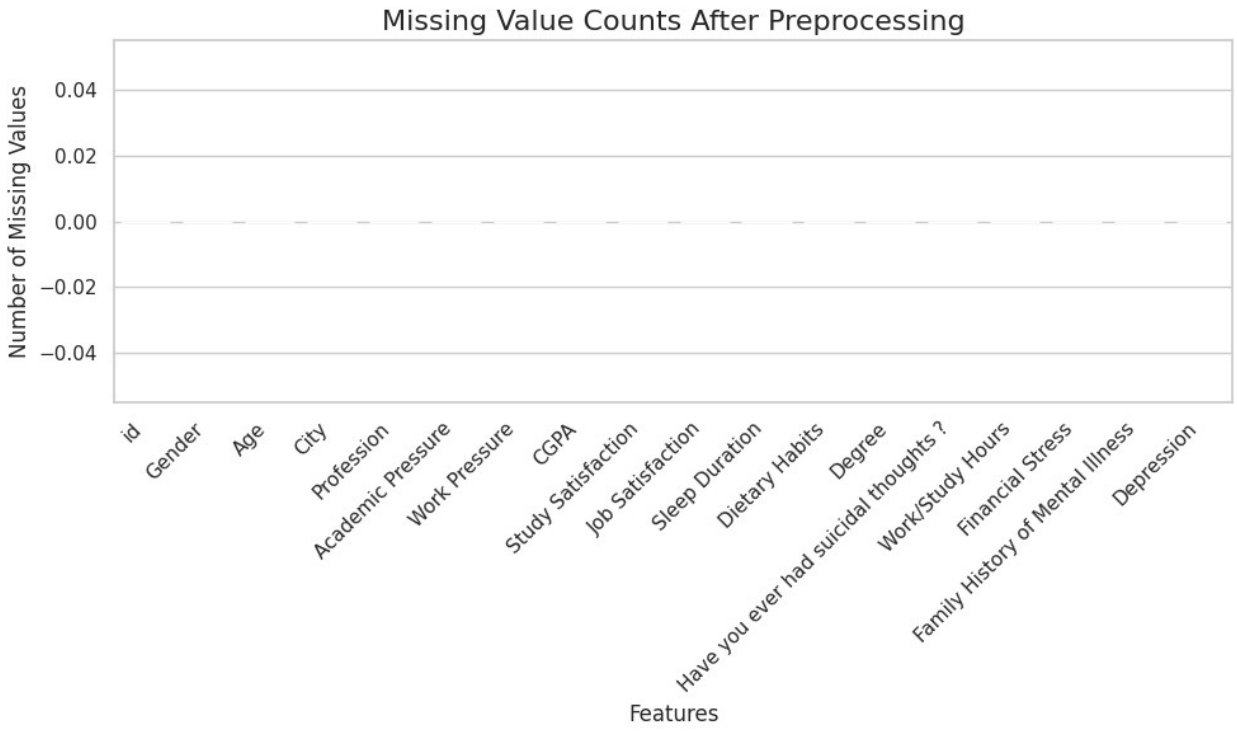
```
plt.figure(figsize=(10, 6))
missing_after = final_cleaned_df.isnull().sum()
sns.barplot(x=missing_after.index, y=missing_after.values,
palette="plasma")
plt.xticks(rotation=45, ha='right')
plt.title('Missing Value Counts After Preprocessing', fontsize=16)
plt.xlabel('Features', fontsize=12)
plt.ylabel('Number of Missing Values', fontsize=12)
plt.tight_layout()
plt.savefig('missing_values_after.png')
print("□ 'missing_values_after.png' has been saved.")
plt.show()

/tmp/ipython-input-1045203474.py:3: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be
removed in v0.14.0. Assign the `x` variable to `hue` and set
`legend=False` for the same effect.

  sns.barplot(x=missing_after.index, y=missing_after.values,
palette="plasma")

□ 'missing_values_after.png' has been saved.
```



Save the combined, clean data to a new file

```
print(f"\nTotal missing values remaining:
{final_cleaned_df.isnull().sum().sum()}")
```

```
Total missing values remaining: 0
```

```
final_cleaned_df.to_csv('data_no_missing.csv', index=False)
```