```
from google.colab import drive
drive.mount('/content/drive')
```

```
Drive already mounted at /content/drive; to attempt to forcibly remount, call drive.mount("/content/drive", force_remount=True).
```

```
%ls
```

```
chrome-win64/  data_types.xlsx  kr-vs-kp.data   output.csv       processed_dataset.csv
data.csv       Index            kr-vs-kp.names  processdata.csv  StudentDropout.ipynb
```

```
%cd Predictive Analysis using Python - Assignment
```

```
/content/drive/MyDrive/Predictive Analysis using Python - Assignment
```

```
import pandas as pd

# Read the CSV file, specifying the delimiter as semicolon
df = pd.read_csv("data.csv", delimiter=";")

# Remove quotes from column headers
df.columns = df.columns.str.replace('"', '')

# Export the DataFrame to a new CSV file
df.to_csv("processed_dataset.csv", index=False)
```

```
df
```

	Marital status	Application mode	Application order	Course	Daytime/evening attendance\t	Previous qualification	qual
0	1	17	5	171	1	1	
1	1	15	1	9254	1	1	
2	1	1	5	9070	1	1	
3	1	17	2	9773	1	1	
4	2	39	1	8014	0	1	
...	...	...	...	...	...	...	
4419	1	1	6	9773	1	1	
4420	1	1	2	9773	1	1	
4421	1	1	1	9500	1	1	
4422	1	1	1	9147	1	1	
4423	1	10	1	9773	1	1	

4424 rows × 37 columns

```
# Display the first few rows of the DataFrame
print("First few rows of the DataFrame:")
print(df.head())

# Display the information about the DataFrame, including the data types and non-null values
print("\nInformation about the DataFrame:")
print(df.info())

# Display the summary statistics of the DataFrame
print("\nSummary statistics of the DataFrame:")
print(df.describe())

# Display the shape of the DataFrame (number of rows and columns)
print("\nShape of the DataFrame:", df.shape)
```

```
50%                                6.000000
75%                                7.000000
max                               23.000000

       Curricular units 2nd sem (evaluations)  \
count                            4424.000000
mean                                8.063291
std                                 3.947951
min                                 0.000000
25%                                 6.000000
50%                                 8.000000
75%                                10.000000
max                                33.000000

       Curricular units 2nd sem (approved)  Curricular units 2nd sem (grade)  \
count                          4424.000000                       4424.000000
mean                              4.435805                         10.230206
std                               3.014764                          5.210808
min                               0.000000                          0.000000
25%                               2.000000                         10.750000
50%                               5.000000                         12.200000
75%                               6.000000                         13.333333
max                              20.000000                         18.571429

       Curricular units 2nd sem (without evaluations)  Unemployment rate  \
count                                     4424.000000        4424.000000
mean                                         0.150316          11.566139
std                                          0.753774           2.663850
min                                          0.000000           7.600000
25%                                          0.000000           9.400000
50%                                          0.000000          11.100000
75%                                          0.000000          13.900000
max                                         12.000000          16.200000

       Inflation rate          GDP
count     4424.000000  4424.000000
mean         1.228029     0.001969
std          1.382711     2.269935
min         -0.800000    -4.060000
25%          0.300000    -1.700000
50%          1.400000     0.320000
75%          2.600000     1.790000
max          3.700000     3.510000

[8 rows x 36 columns]

Shape of the DataFrame: (4424, 37)
```

```
# Check for NaN values in the DataFrame
nan_df = df.isna()

# Count the number of NaN values in each column
nan_counts = nan_df.sum()

print("Missing Values:")
print(nan_counts)
```

```
Missing Values:
Marital status                                  0
Application mode                                0
Application order                               0
Course                                          0
Daytime/evening attendance\t                    0
Previous qualification                          0
Previous qualification (grade)                  0
Nacionality                                     0
Mother's qualification                          0
Father's qualification                          0
Mother's occupation                             0
Father's occupation                             0
Admission grade                                 0
Displaced                                       0
Educational special needs                       0
Debtor                                          0
Tuition fees up to date                         0
Gender                                          0
Scholarship holder                              0
Age at enrollment                               0
International                                   0
Curricular units 1st sem (credited)             0
Curricular units 1st sem (enrolled)             0
Curricular units 1st sem (evaluations)          0
Curricular units 1st sem (approved)             0
Curricular units 1st sem (grade)                0
Curricular units 1st sem (without evaluations)  0
Curricular units 2nd sem (credited)             0
Curricular units 2nd sem (enrolled)             0
Curricular units 2nd sem (evaluations)          0
Curricular units 2nd sem (approved)             0
```

```
Curricular units 2nd sem (grade)                0
Curricular units 2nd sem (without evaluations)  0
Unemployment rate                               0
Inflation rate                                  0
GDP                                             0
Target                                          0
dtype: int64
```

```
# Assuming your DataFrame is named 'df'
column_data_types = df.dtypes
print(column_data_types)
```

```
Marital status                                    int64
Application mode                                  int64
Application order                                 int64
Course                                            int64
Daytime/evening attendance\t                      int64
Previous qualification                            int64
Previous qualification (grade)                  float64
Nacionality                                       int64
Mother's qualification                            int64
Father's qualification                            int64
Mother's occupation                               int64
Father's occupation                               int64
Admission grade                                 float64
Displaced                                         int64
Educational special needs                         int64
Debtor                                            int64
Tuition fees up to date                           int64
Gender                                            int64
Scholarship holder                                int64
Age at enrollment                                 int64
International                                     int64
Curricular units 1st sem (credited)               int64
Curricular units 1st sem (enrolled)               int64
Curricular units 1st sem (evaluations)            int64
Curricular units 1st sem (approved)               int64
Curricular units 1st sem (grade)                float64
Curricular units 1st sem (without evaluations)    int64
Curricular units 2nd sem (credited)               int64
Curricular units 2nd sem (enrolled)               int64
Curricular units 2nd sem (evaluations)            int64
Curricular units 2nd sem (approved)               int64
Curricular units 2nd sem (grade)                float64
Curricular units 2nd sem (without evaluations)    int64
Unemployment rate                               float64
Inflation rate                                  float64
GDP                                             float64
Target                                           object
dtype: object
```

```
# Define the file path for the Excel file
excel_file_path = "data_types.xlsx"

# Export the data types of the DataFrame to an Excel file
df.dtypes.to_frame().to_excel(excel_file_path, header=False)

print("Data types exported to Excel file:", excel_file_path)
```

```
Data types exported to Excel file: data_types.xlsx
```

```
# Define the mapping dictionary
target_mapping = {'Dropout': 0, 'Graduate': 1, 'Enrolled': 2}  # Update 'Other' if there are additional categories

# Apply the mapping to the target column
df['Target_encoded'] = df['Target'].map(target_mapping)

# Display the unique classes and their corresponding encoded values
print("Encoded classes:", target_mapping)
```

```
Encoded classes: {'Dropout': 0, 'Graduate': 1, 'Enrolled': 2}
```

```
df
```

	Marital status	Application mode	Application order	Course	Daytime/evening attendance\t	Previous qualification	qual
0	1	17	5	171	1	1	
1	1	15	1	9254	1	1	
2	1	1	5	9070	1	1	
3	1	17	2	9773	1	1	
4	2	39	1	8014	0	1	
...	...	...	...	...	...	...	
4419	1	1	6	9773	1	1	
4420	1	1	2	9773	1	1	
4421	1	1	1	9500	1	1	
4422	1	1	1	9147	1	1	
4423	1	10	1	9773	1	1	

4424 rows × 38 columns

```
# Check for NaN values in the DataFrame
nan_df = df.isna()

# Count the number of NaN values in each column
nan_counts = nan_df.sum()

print("Missing Values:")
print(nan_counts)
```

```
Missing Values:
Marital status                                  0
Application mode                                0
Application order                               0
Course                                          0
Daytime/evening attendance\t                    0
Previous qualification                          0
Previous qualification (grade)                  0
Nacionality                                     0
Mother's qualification                          0
Father's qualification                          0
Mother's occupation                             0
Father's occupation                             0
Admission grade                                 0
Displaced                                       0
Educational special needs                       0
Debtor                                          0
Tuition fees up to date                         0
Gender                                          0
Scholarship holder                              0
Age at enrollment                               0
International                                   0
Curricular units 1st sem (credited)             0
Curricular units 1st sem (enrolled)             0
Curricular units 1st sem (evaluations)          0
Curricular units 1st sem (approved)             0
Curricular units 1st sem (grade)                0
Curricular units 1st sem (without evaluations)  0
Curricular units 2nd sem (credited)             0
Curricular units 2nd sem (enrolled)             0
Curricular units 2nd sem (evaluations)          0
Curricular units 2nd sem (approved)             0
Curricular units 2nd sem (grade)                0
Curricular units 2nd sem (without evaluations)  0
Unemployment rate                               0
Inflation rate                                  0
GDP                                             0
Target                                          0
Target_encoded                                  0
dtype: int64
```

```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
```

```
df
```

	Marital status	Application mode	Application order	Course	Daytime/evening attendance\t	Previous qualification	qual
0	1	17	5	171	1	1	
1	1	15	1	9254	1	1	
2	1	1	5	9070	1	1	
3	1	17	2	9773	1	1	
4	2	39	1	8014	0	1	
...	...	...	...	...	...	...	
4419	1	1	6	9773	1	1	
4420	1	1	2	9773	1	1	
4421	1	1	1	9500	1	1	
4422	1	1	1	9147	1	1	
4423	1	10	1	9773	1	1	

4424 rows × 38 columns

```
# Step 2: Check the dimensions
print("Dimensions of the DataFrame:", df.shape)
```

```
Dimensions of the DataFrame: (4424, 38)
```

```
# Step 3: Preview the data
print("First few rows of the DataFrame:")
print(df.head())
```

```
First few rows of the DataFrame:
   Marital status  Application mode  Application order  Course  \
0               1                17                  5     171
1               1                15                  1    9254
2               1                 1                  5    9070
3               1                17                  2    9773
4               2                39                  1    8014

   Daytime/evening attendance\t  Previous qualification  \
0                             1                       1
1                             1                       1
2                             1                       1
3                             1                       1
4                             0                       1

   Previous qualification (grade)  Nacionality  Mother's qualification  \
0                           122.0            1                      19
1                           160.0            1                       1
2                           122.0            1                      37
3                           122.0            1                      38
4                           100.0            1                      37

   Father's qualification  ...  Curricular units 2nd sem (enrolled)  \
0                      12  ...                                    0
1                       3  ...                                    6
2                      37  ...                                    6
3                      37  ...                                    6
4                      38  ...                                    6

   Curricular units 2nd sem (evaluations)  \
0                                       0
1                                       6
2                                       0
3                                      10
4                                       6

   Curricular units 2nd sem (approved)  Curricular units 2nd sem (grade)  \
0                                    0                         0.000000
1                                    6                        13.666667
2                                    0                         0.000000
3                                    5                        12.400000
4                                    6                        13.000000

   Curricular units 2nd sem (without evaluations)  Unemployment rate  \
0                                               0               10.8
1                                               0               13.9
2                                               0               10.8
3                                               0                9.4
4                                               0               13.9
```

```
   Inflation rate   GDP    Target  Target_encoded
0             1.4  1.74   Dropout               0
1            -0.3  0.79  Graduate               1
2             1.4  1.74   Dropout               0
3            -0.8 -3.12  Graduate               1
4            -0.3  0.79  Graduate               1

[5 rows x 38 columns]
```

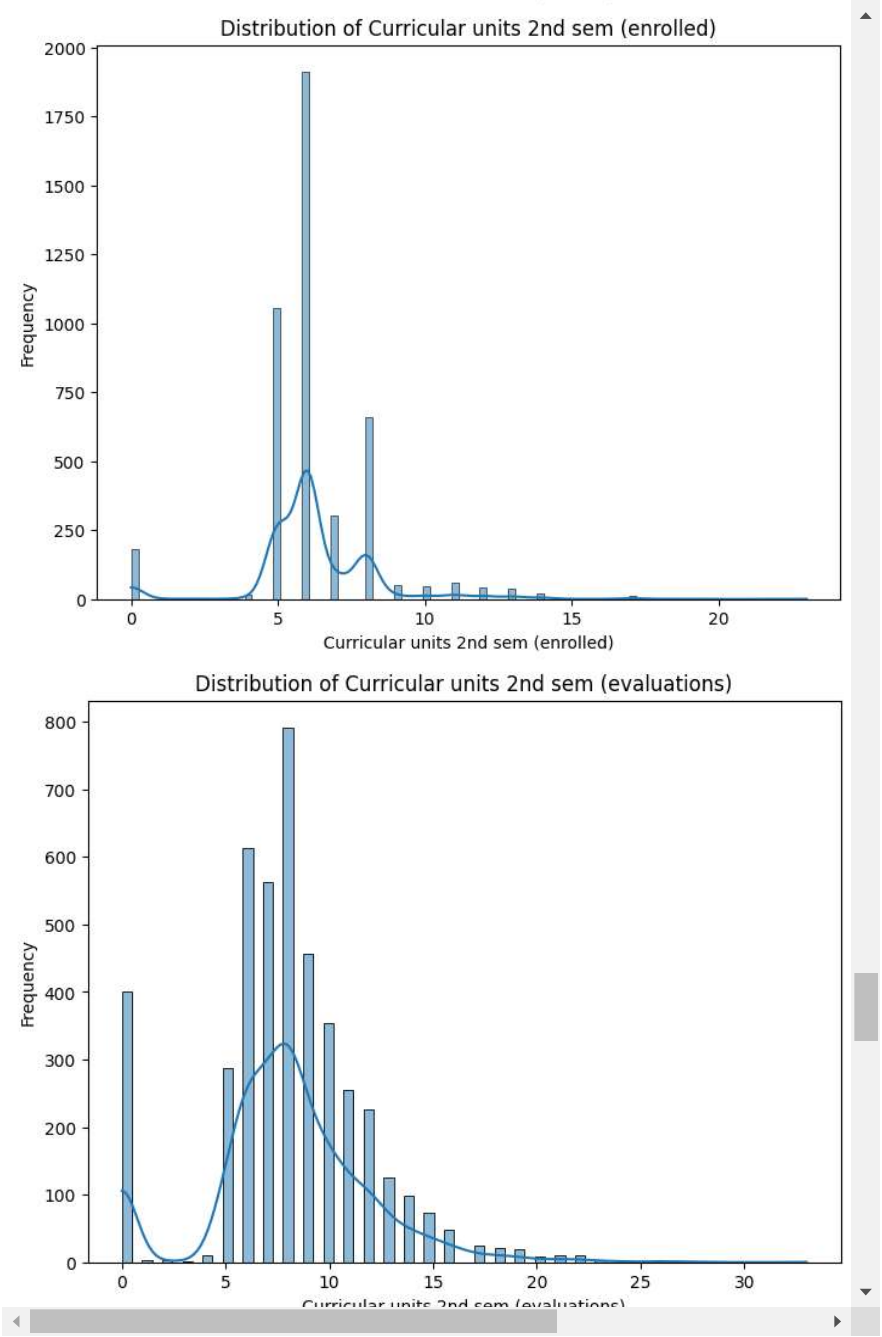
```
# Step 4: Understand the data types
print("Information about the DataFrame:")
print(df.info())
```

```
Information about the DataFrame:
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 4424 entries, 0 to 4423
Data columns (total 38 columns):
 #   Column                                          Non-Null Count  Dtype
---  ------                                          --------------  -----
 0   Marital status                                  4424 non-null   int64
 1   Application mode                                4424 non-null   int64
 2   Application order                               4424 non-null   int64
 3   Course                                          4424 non-null   int64
 4   Daytime/evening attendance                      4424 non-null   int64
 5   Previous qualification                          4424 non-null   int64
 6   Previous qualification (grade)                  4424 non-null   float64
 7   Nacionality                                     4424 non-null   int64
 8   Mother's qualification                          4424 non-null   int64
 9   Father's qualification                          4424 non-null   int64
 10  Mother's occupation                             4424 non-null   int64
 11  Father's occupation                             4424 non-null   int64
 12  Admission grade                                 4424 non-null   float64
 13  Displaced                                       4424 non-null   int64
 14  Educational special needs                       4424 non-null   int64
 15  Debtor                                          4424 non-null   int64
 16  Tuition fees up to date                         4424 non-null   int64
 17  Gender                                          4424 non-null   int64
 18  Scholarship holder                              4424 non-null   int64
 19  Age at enrollment                               4424 non-null   int64
 20  International                                   4424 non-null   int64
 21  Curricular units 1st sem (credited)             4424 non-null   int64
 22  Curricular units 1st sem (enrolled)             4424 non-null   int64
 23  Curricular units 1st sem (evaluations)          4424 non-null   int64
 24  Curricular units 1st sem (approved)             4424 non-null   int64
 25  Curricular units 1st sem (grade)                4424 non-null   float64
 26  Curricular units 1st sem (without evaluations)  4424 non-null   int64
 27  Curricular units 2nd sem (credited)             4424 non-null   int64
 28  Curricular units 2nd sem (enrolled)             4424 non-null   int64
 29  Curricular units 2nd sem (evaluations)          4424 non-null   int64
 30  Curricular units 2nd sem (approved)             4424 non-null   int64
 31  Curricular units 2nd sem (grade)                4424 non-null   float64
 32  Curricular units 2nd sem (without evaluations)  4424 non-null   int64
 33  Unemployment rate                               4424 non-null   float64
 34  Inflation rate                                  4424 non-null   float64
 35  GDP                                             4424 non-null   float64
 36  Target                                          4424 non-null   object
 37  Target_encoded                                  4424 non-null   int64
dtypes: float64(7), int64(30), object(1)
memory usage: 1.3+ MB
None
```

```
# Step 5: Summary statistics
print("Summary statistics of the DataFrame:")
print(df.describe())
```

```
25%                                 6.000000
50%                                 8.000000
75%                                10.000000
max                                33.000000

       Curricular units 2nd sem (approved)  Curricular units 2nd sem (grade)  \
count                          4424.000000                       4424.000000
mean                              4.435805                         10.230206
std                               3.014764                          5.210808
min                               0.000000                          0.000000
25%                               2.000000                         10.750000
50%                               5.000000                         12.200000
75%                               6.000000                         13.333333
max                              20.000000                         18.571429

       Curricular units 2nd sem (without evaluations)  Unemployment rate  \
count                                     4424.000000        4424.000000
mean                                         0.150316          11.566139
std                                          0.753774           2.663850
min                                          0.000000           7.600000
25%                                          0.000000           9.400000
50%                                          0.000000          11.100000
75%                                          0.000000          13.900000
max                                         12.000000          16.200000

       Inflation rate          GDP  Target_encoded
count     4424.000000  4424.000000     4424.000000
mean         1.228029     0.001969        0.858273
std          1.382711     2.269935        0.693326
min         -0.800000    -4.060000        0.000000
25%          0.300000    -1.700000        0.000000
50%          1.400000     0.320000        1.000000
75%          2.600000     1.790000        1.000000
max          3.700000     3.510000        2.000000

[8 rows x 37 columns]
```

```
# Step 6: Visualize distributions (for numerical columns)
numerical_columns = df.select_dtypes(include=['int64', 'float64']).columns
for column in numerical_columns:
    plt.figure(figsize=(8, 6))
    sns.histplot(df[column], kde=True)
    plt.title(f"Distribution of {column}")
    plt.xlabel(column)
    plt.ylabel("Frequency")
    plt.show()
```


Distribution of Curricular units 2nd sem (enrolled)
Frequency
Curricular units 2nd sem (enrolled)
Distribution of Curricular units 2nd sem (evaluations)
Frequency