```
import pandas as pd
import numpy as np
df = pd.read_csv("/academic_performance_dataset_V2.csv")
df.head()
```

| | ID No | Prog Code | Gender | YoG | CGPA | CGPA100 | CGPA200 | CGPA300 | CGPA400 | SGPA |
|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 42308 | ICE | Female | 2010 | 3.23 | 2.88 | 3.48 | 2.62 | 2.90 | 3.13 |
| 1 | 70978 | BCH | Female | 2010 | 3.58 | 3.25 | 4.26 | 3.37 | 3.47 | 3.02 |
| 2 | 31602 | BCH | Male | 2010 | 2.21 | 1.78 | 1.98 | 1.49 | 2.51 | 2.19 |
| 3 | 63847 | BCH | Male | 2010 | 2.70 | 2.67 | 2.44 | 2.00 | 2.35 | 3.19 |
| 4 | 30158 | BCH | Female | 2010 | 3.88 | 3.61 | 3.69 | 3.63 | 4.58 | 4.24 |

Next steps: Generate code with df | New interactive sheet

```
df.isnull().sum()
```

| | 0 |
|---|---|
| ID No | 0 |
| Prog Code | 0 |
| Gender | 0 |
| YoG | 0 |
| CGPA | 0 |
| CGPA100 | 0 |
| CGPA200 | 0 |
| CGPA300 | 0 |
| CGPA400 | 0 |
| SGPA | 0 |

dtype: int64

```
df.describe(include='all')
```

| | ID No | Prog Code | Gender | YoG | CGPA | CGPA100 | CGPA200 | CGPA300 | CGPA400 | SGPA |
|---|---|---|---|---|---|---|---|---|---|---|
| count | 3046.000000 | 3046 | 3046 | 3046.000000 | 3046.000000 | 3046.000000 | 3046.000000 | 3046.000000 | 3046.000000 | 3046.000000 |
| unique | NaN | 17 | 2 | NaN | NaN | NaN | NaN | NaN | NaN | NaN |
| top | NaN | EEE | Male | NaN | NaN | NaN | NaN | NaN | NaN | NaN |
| freq | NaN | 418 | 1953 | NaN | NaN | NaN | NaN | NaN | NaN | NaN |
| mean | 61082.817466 | NaN | NaN | 2012.183848 | 3.494813 | 3.636106 | 3.321687 | 3.418578 | 3.532508 | 3.119645 |
| std | 21256.532751 | NaN | NaN | 1.329104 | 0.691614 | 0.679256 | 0.782473 | 0.858518 | 0.802228 | 0.616151 |
| min | 23462.000000 | NaN | NaN | 2010.000000 | 1.520000 | 1.570000 | 1.170000 | 0.630000 | 0.000000 | 1.460000 |
| 25% | 42654.250000 | NaN | NaN | 2011.000000 | 3.000000 | 3.180000 | 2.760000 | 2.810000 | 3.000000 | 2.660000 |
| 50% | 61759.000000 | NaN | NaN | 2012.000000 | 3.560000 | 3.690000 | 3.340000 | 3.510000 | 3.620000 | 3.060000 |
| 75% | 79236.250000 | NaN | NaN | 2013.000000 | 4.010000 | 4.150000 | 3.920000 | 4.100000 | 4.150000 | 3.570000 |

```
df.shape
```

```
(3046, 10)
```

```
df.dtypes
```

| | 0 |
|---|---|
| **ID No** | int64 |
| **Prog Code** | object |
| **Gender** | object |
| **YoG** | int64 |
| **CGPA** | float64 |
| **CGPA100** | float64 |
| **CGPA200** | float64 |

```
# Convert object columns to categorical
for col in df.select_dtypes(include='object').columns:
    df[col] = df[col].astype('category')

df.dtypes
```

| | 0 |
|---|---|
| **ID No** | int64 |
| **Prog Code** | category |
| **Gender** | category |
| **YoG** | int64 |
| **CGPA** | float64 |
| **CGPA100** | float64 |
| **CGPA200** | float64 |
| **CGPA300** | float64 |
| **CGPA400** | float64 |
| **SGPA** | float64 |

**dtype:** object

**dtype:** object

```
for col in df.select_dtypes(include='category').columns:
    df[col + "_code"] = df[col].cat.codes

df.head()
```

| | ID No | Prog Code | Gender | YoG | CGPA | CGPA100 | CGPA200 | CGPA300 | CGPA400 | SGPA | Prog Code_code | Gender_code |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | 42308 | ICE | Female | 2010 | 3.23 | 2.88 | 3.48 | 2.62 | 2.90 | 3.13 | 8 | 0 |
| **1** | 70978 | BCH | Female | 2010 | 3.58 | 3.25 | 4.26 | 3.37 | 3.47 | 3.02 | 0 | 0 |
| **2** | 31602 | BCH | Male | 2010 | 2.21 | 1.78 | 1.98 | 1.49 | 2.51 | 2.19 | 0 | 1 |
| **3** | 63847 | BCH | Male | 2010 | 2.70 | 2.67 | 2.44 | 2.00 | 2.35 | 3.19 | 0 | 1 |
| **4** | 30158 | BCH | Female | 2010 | 3.88 | 3.61 | 3.69 | 3.63 | 4.58 | 4.24 | 0 | 0 |

Next steps: Generate code with df | New interactive sheet