In [1]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

In [2]:
```
df = pd.read_csv(r"C:\Users\veera\Downloads\PRANAV B\docs\projects\Transactional
```

In [3]:
```
print("Shape of data:", df.shape)
print("\nColumns:", df.columns.tolist())
print("\nData types:\n", df.dtypes)
print("\nMissing values:\n", df.isnull().sum().sum())
```

```
Shape of data: (284807, 31)

Columns: ['Time', 'V1', 'V2', 'V3', 'V4', 'V5', 'V6', 'V7', 'V8', 'V9', 'V10', 'V
11', 'V12', 'V13', 'V14', 'V15', 'V16', 'V17', 'V18', 'V19', 'V20', 'V21', 'V22',
'V23', 'V24', 'V25', 'V26', 'V27', 'V28', 'Amount', 'Class']

Data types:
 Time      float64
V1        float64
V2        float64
V3        float64
V4        float64
V5        float64
V6        float64
V7        float64
V8        float64
V9        float64
V10       float64
V11       float64
V12       float64
V13       float64
V14       float64
V15       float64
V16       float64
V17       float64
V18       float64
V19       float64
V20       float64
V21       float64
V22       float64
V23       float64
V24       float64
V25       float64
V26       float64
V27       float64
V28       float64
Amount    float64
Class       int64
dtype: object

Missing values:
 0
```

In [4]:
```
df.head()
```

Out[4]:

	Time	V1	V2	V3	V4	V5	V6	V7	
0	0.0	-1.359807	-0.072781	2.536347	1.378155	-0.338321	0.462388	0.239599	0.0
1	0.0	1.191857	0.266151	0.166480	0.448154	0.060018	-0.082361	-0.078803	0.0
2	1.0	-1.358354	-1.340163	1.773209	0.379780	-0.503198	1.800499	0.791461	0.2
3	1.0	-0.966272	-0.185226	1.792993	-0.863291	-0.010309	1.247203	0.237609	0.3
4	2.0	-1.158233	0.877737	1.548718	0.403034	-0.407193	0.095921	0.592941	-0.2

5 rows × 31 columns

In [5]:
```
df.describe().T
#Amount – transaction values
#Class – 1 = Fraud, 0 = Legit
```

Out[5]:

	count	mean	std	min	25%	50%
Time	284807.0	9.481386e+04	47488.145955	0.000000	54201.500000	84692.000000
V1	284807.0	1.168375e-15	1.958696	-56.407510	-0.920373	0.018109
V2	284807.0	3.416908e-16	1.651309	-72.715728	-0.598550	0.065486
V3	284807.0	-1.379537e-15	1.516255	-48.325589	-0.890365	0.179846
V4	284807.0	2.074095e-15	1.415869	-5.683171	-0.848640	-0.019847
V5	284807.0	9.604066e-16	1.380247	-113.743307	-0.691597	-0.054336
V6	284807.0	1.487313e-15	1.332271	-26.160506	-0.768296	-0.274187
V7	284807.0	-5.556467e-16	1.237094	-43.557242	-0.554076	0.040103
V8	284807.0	1.213481e-16	1.194353	-73.216718	-0.208630	0.022358
V9	284807.0	-2.406331e-15	1.098632	-13.434066	-0.643098	-0.051429
V10	284807.0	2.239053e-15	1.088850	-24.588262	-0.535426	-0.092917
V11	284807.0	1.673327e-15	1.020713	-4.797473	-0.762494	-0.032757
V12	284807.0	-1.247012e-15	0.999201	-18.683715	-0.405571	0.140033
V13	284807.0	8.190001e-16	0.995274	-5.791881	-0.648539	-0.013568
V14	284807.0	1.207294e-15	0.958596	-19.214325	-0.425574	0.050601
V15	284807.0	4.887456e-15	0.915316	-4.498945	-0.582884	0.048072
V16	284807.0	1.437716e-15	0.876253	-14.129855	-0.468037	0.066413
V17	284807.0	-3.772171e-16	0.849337	-25.162799	-0.483748	-0.065676
V18	284807.0	9.564149e-16	0.838176	-9.498746	-0.498850	-0.003636
V19	284807.0	1.039917e-15	0.814041	-7.213527	-0.456299	0.003735
V20	284807.0	6.406204e-16	0.770925	-54.497720	-0.211721	-0.062481
V21	284807.0	1.654067e-16	0.734524	-34.830382	-0.228395	-0.029450
V22	284807.0	-3.568593e-16	0.725702	-10.933144	-0.542350	0.006782
V23	284807.0	2.578648e-16	0.624460	-44.807735	-0.161846	-0.011193
V24	284807.0	4.473266e-15	0.605647	-2.836627	-0.354586	0.040976
V25	284807.0	5.340915e-16	0.521278	-10.295397	-0.317145	0.016594
V26	284807.0	1.683437e-15	0.482227	-2.604551	-0.326984	-0.052139
V27	284807.0	-3.660091e-16	0.403632	-22.565679	-0.070840	0.001342

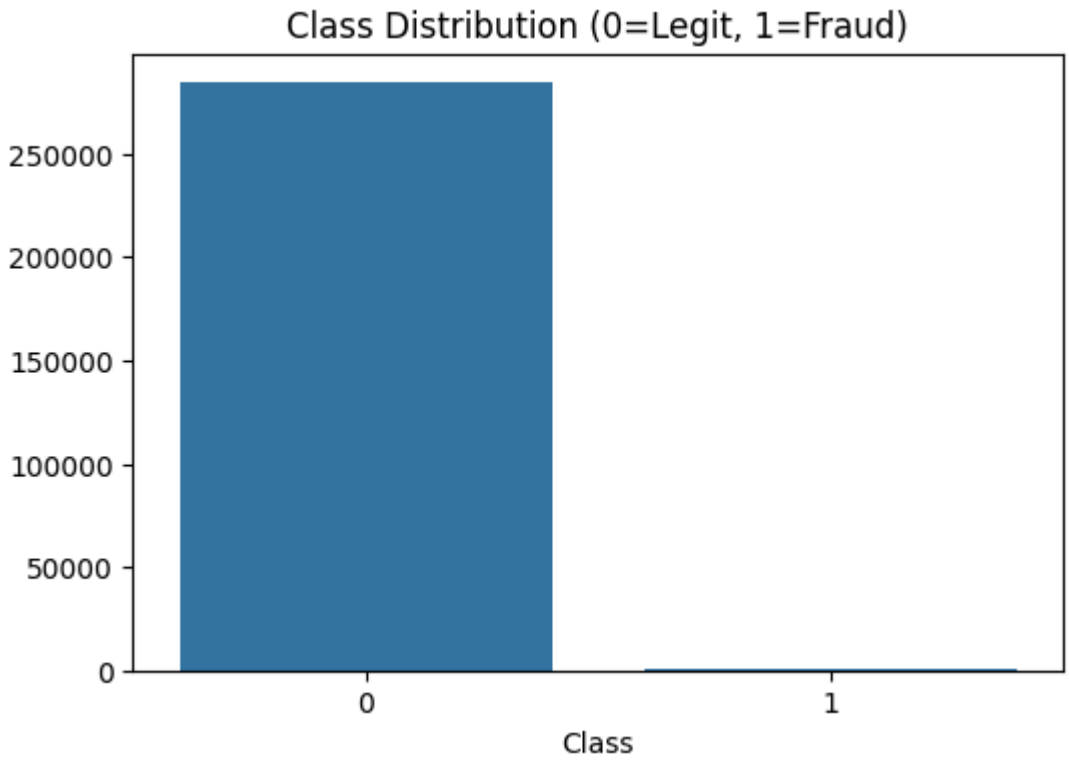
	count	mean	std	min	25%	50%
V28	284807.0	-1.227390e-16	0.330083	-15.430084	-0.052960	0.011244
Amount	284807.0	8.834962e+01	250.120109	0.000000	5.600000	22.000000
Class	284807.0	1.727486e-03	0.041527	0.000000	0.000000	0.000000

In [6]:
```
fraud_count = df['Class'].value_counts()
print(fraud_count)

plt.figure(figsize=(6,4))
sns.barplot(x=fraud_count.index, y=fraud_count.values)
plt.title("Class Distribution (0=Legit, 1=Fraud)")
plt.show()

fraud_percentage = (fraud_count[1] / fraud_count.sum()) * 100
print(f"Fraud Percentage: {fraud_percentage:.4f}%")
```

```
Class
0    284315
1       492
Name: count, dtype: int64
```

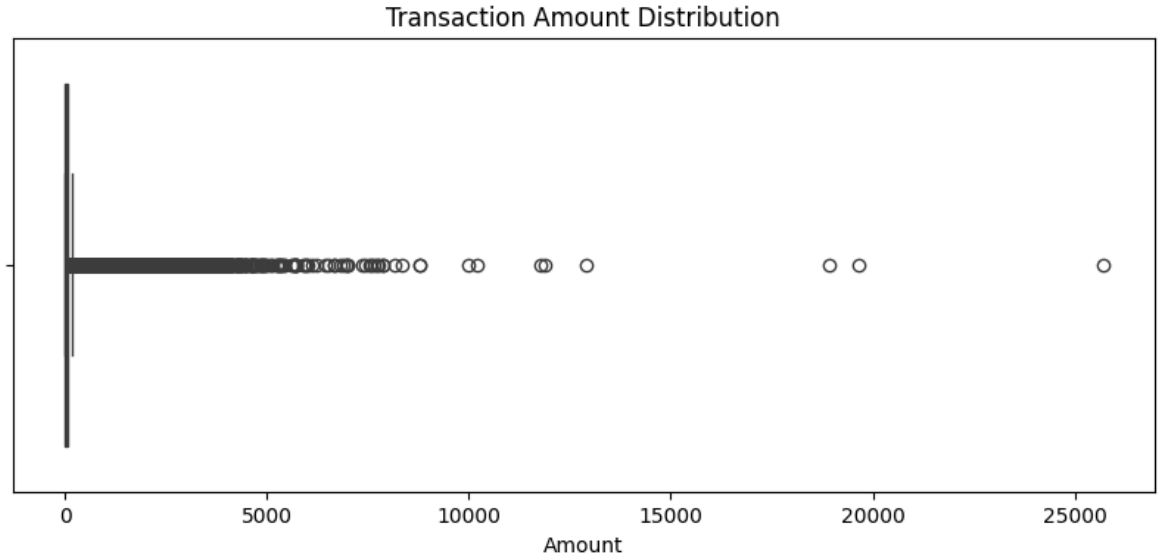



```
Fraud Percentage: 0.1727%
```

In [7]:
```
# Check for NaNs again
print(df.isnull().sum().sum())

# Quick outlier visualization for 'Amount'
plt.figure(figsize=(10,4))
sns.boxplot(x=df['Amount'])
plt.title("Transaction Amount Distribution")
plt.show()
```

```
0
```



In [8]:
```
df.to_csv(r"C:\Users\veera\Downloads\PRANAV B\docs\projects\Transactional Fraud
print("Clean data saved.")
```

```
Clean data saved.
```

In [ ]:
```
!jupyter nbconvert --to html "week1_data_exploration.ipynb" --output-dir="report
```