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


df=pd.read_csv('/content/NEO Earth Close Approaches.csv')
df.head(),df.info()
```

Show hidden output

The dataset contains 1,573 entries and 9 columns. No missing values. The Close-Approach (CA) Date column is in string format and needs conversion to datetime. The Diameter column contains ranges (e.g., "13 m - 29 m"), requiring parsing into min/max values.

```
df.describe()
```

	Close-Approach (CA) Date	CA DistanceNominal (au)	CA DistanceMinimum (au)	V relative(km/s)	V infinity(km/s)	H(mag)	Rarity	Diameter Min (m)	Diam Max
count	1573	1573.000000	1573.000000	1573.000000	1573.000000	1573.000000	1573.000000	1573.000000	1573.00
mean	2023-11-03 04:32:41.042593792	0.021108	0.020705	10.596853	10.540846	26.136554	0.127146	23.903566	50.04
min	2023-05-03 01:15:00	0.000070	0.000070	0.440000	0.130000	18.500000	0.000000	0.910000	2.00
25%	2023-09-07 00:19:00	0.008530	0.008310	7.040000	7.000000	25.200000	0.000000	9.600000	21.00
50%	2023-10-27 11:16:00	0.019840	0.019190	9.600000	9.570000	26.200000	0.000000	15.000000	34.00
75%	2024-01-14 00:04:00	0.032290	0.031630	13.360000	13.320000	27.200000	0.000000	25.000000	55.00

```
df['Close-Approach (CA) Date'] = pd.to_datetime(
    df['Close-Approach (CA) Date'].str.split('±').str[0].str.strip(),
    format='%Y-%b-%d %H:%M',
    errors='coerce'
)

df["Diameter Min (m)"] = df["Diameter"].str.extract(r'(\d+\.?\d*) m')[0].astype(float)
df["Diameter Max (m)"] = df["Diameter"].str.extract(r'-\s*(\d+\.?\d*) m')[0].astype(float)

df.drop(columns=["Diameter"], inplace=True)

df.head()
```

	Object	Close-Approach (CA) Date	CA DistanceNominal (au)	CA DistanceMinimum (au)	V relative(km/s)	V infinity(km/s)	H(mag)	Rarity	Diameter Min (m)	Diameter Max (m)
0	(2023 JA1)	2023-05-03 01:15:00	0.02518	0.02509	5.01	4.99	26.6	0	13.0	29.0
1	(2023 HV5)	2023-05-03 03:16:00	0.00291	0.00290	8.93	8.82	27.2	0	9.8	22.0
	(2023	2023-05-03								

Next steps: View recommended plots | New interactive sheet

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

	0
Object	0
Close-Approach (CA) Date	0
CA DistanceNominal (au)	0
CA DistanceMinimum (au)	0
V relative(km/s)	0
V infinity(km/s)	0
H(mag)	0
Rarity	0
Diameter Min (m)	0
Diameter Max (m)	0

dtype: int64

```
df.dtypes
```

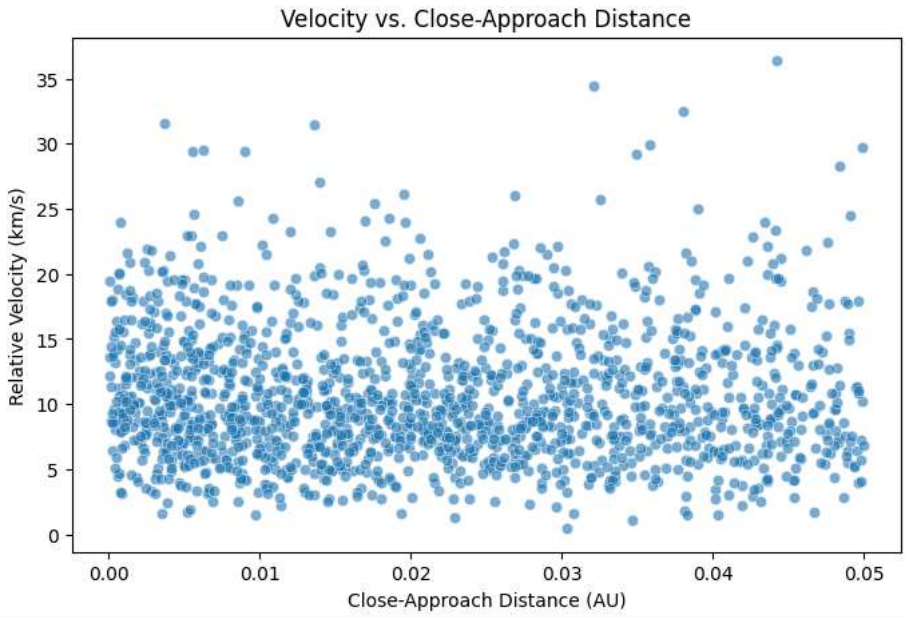
	0
Object	object
Close-Approach (CA) Date	datetime64[ns]
CA DistanceNominal (au)	float64
CA DistanceMinimum (au)	float64
V relative(km/s)	float64
V infinity(km/s)	float64
H(mag)	float64
Rarity	int64
Diameter Min (m)	float64
Diameter Max (m)	float64

dtype: object

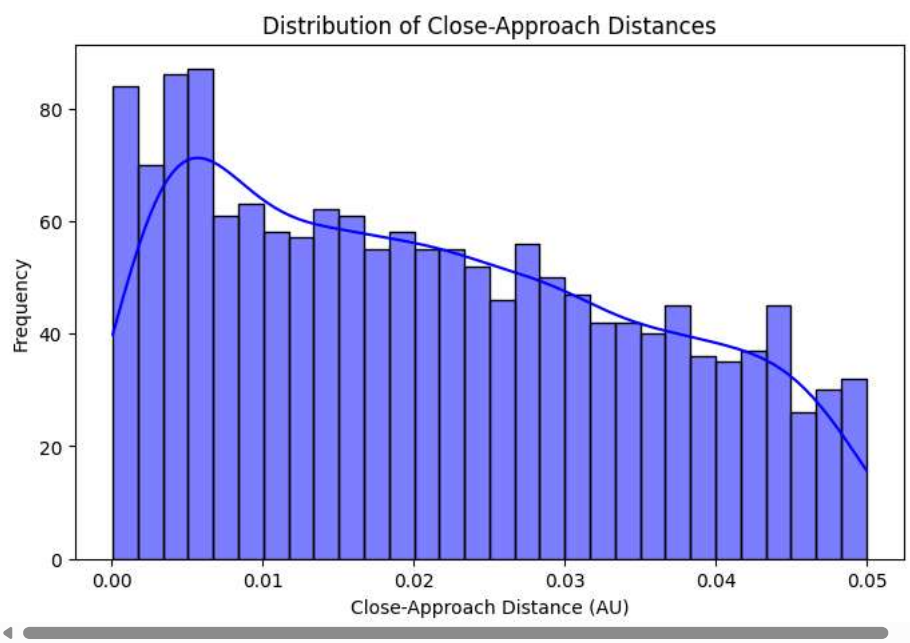
```
df["Diameter Min (m)"].fillna(df["Diameter Min (m)"].median(), inplace=True)
df["Diameter Max (m)"].fillna(df["Diameter Max (m)"].median(), inplace=True)
```

Show hidden output

```
plt.figure(figsize=(8, 5))
sns.scatterplot(x=df["CA DistanceNominal (au)"], y=df["V relative(km/s)"], alpha=0.6)
plt.xlabel("Close-Approach Distance (AU)")
plt.ylabel("Relative Velocity (km/s)")
plt.title("Velocity vs. Close-Approach Distance")
plt.show()
```

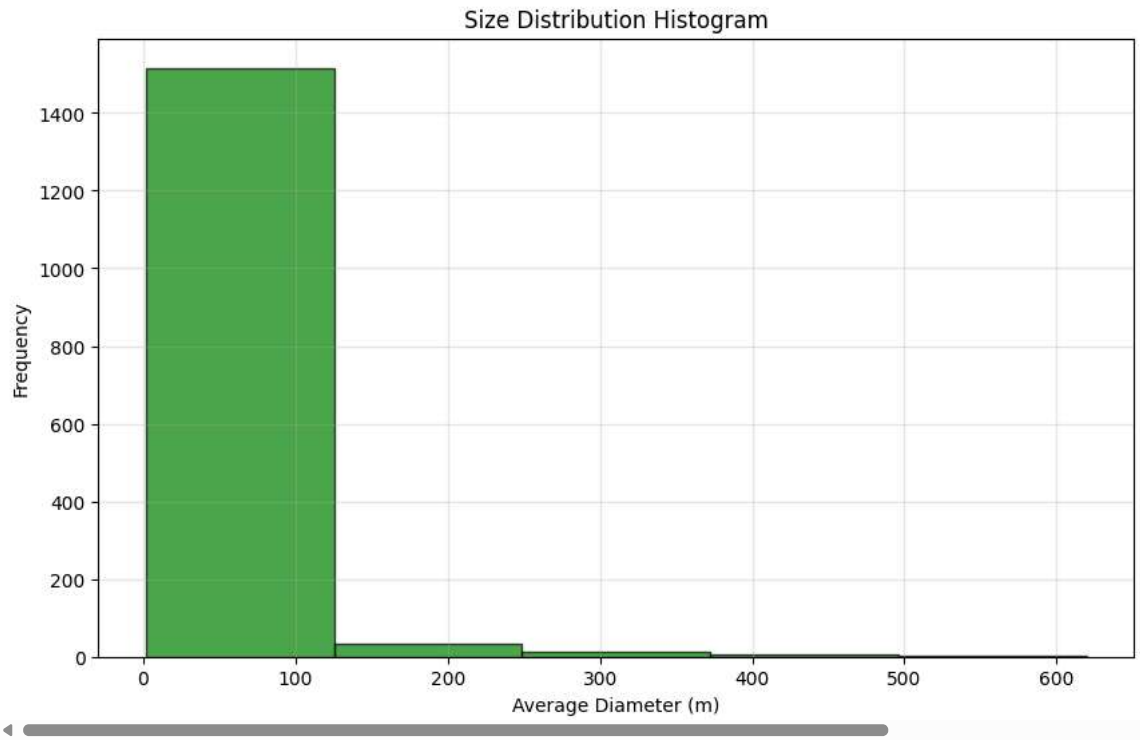



```
# Visualization - Histogram of Close-Approach Distance (AU)
plt.figure(figsize=(8, 5))
sns.histplot(df["CA DistanceNominal (au)"], bins=30, kde=True, color='blue')
plt.xlabel("Close-Approach Distance (AU)")
plt.ylabel("Frequency")
plt.title("Distribution of Close-Approach Distances")
plt.show()
```

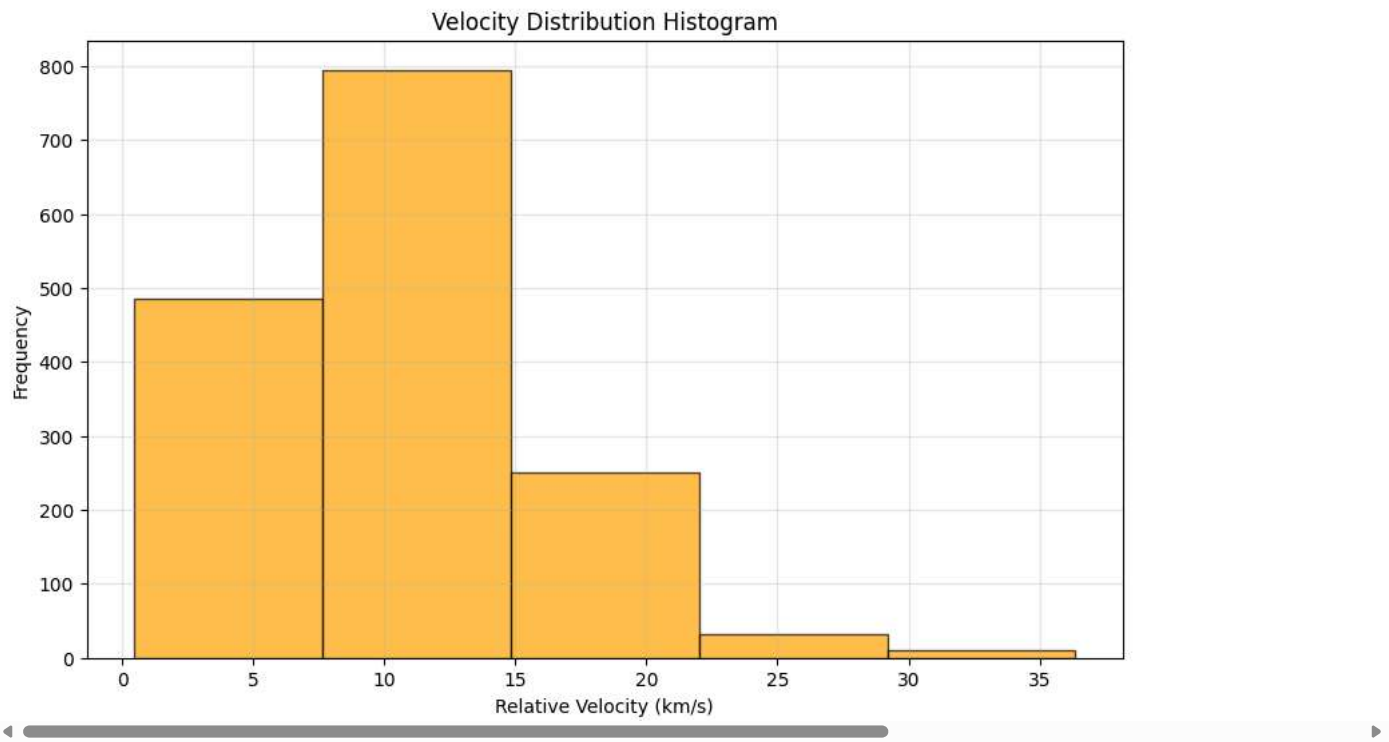



```
plt.figure(figsize=(10, 6))
plt.hist(
    (df["Diameter Min (m)"] + df["Diameter Max (m)"]) / 2,
    bins=5,
    color='green',
    edgecolor='black',
    alpha=0.7
)
plt.xlabel("Average Diameter (m)")
plt.ylabel("Frequency")
plt.title("Size Distribution Histogram")
plt.grid(True, alpha=0.3)
```

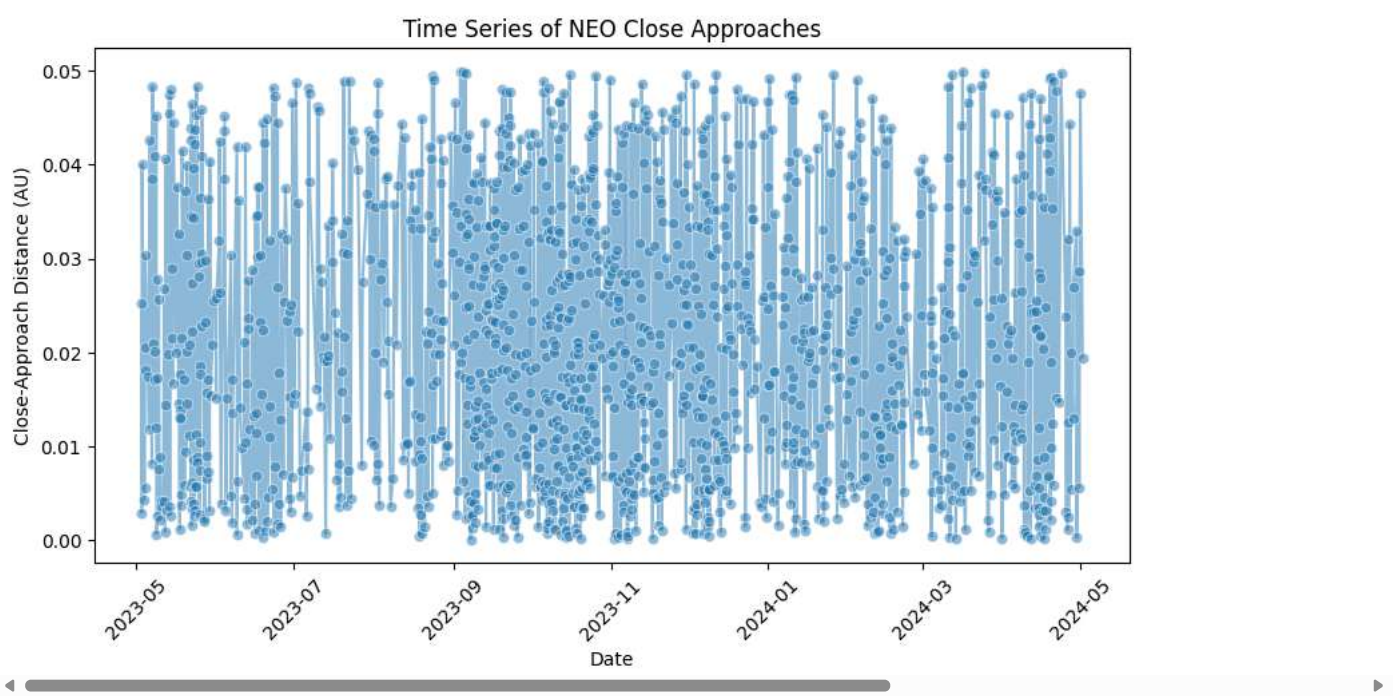
```
plt.show()
```




```
plt.figure(figsize=(10, 6))
plt.hist(
    df["V relative(km/s)"],
    bins=5,
    color='orange',
    edgecolor='black',
    alpha=0.7
)
plt.xlabel("Relative Velocity (km/s)")
plt.ylabel("Frequency")
plt.title("Velocity Distribution Histogram")
plt.grid(True, alpha=0.3)
plt.show()
```



```
#  Time series analysis of close approaches
plt.figure(figsize=(10, 5))
df_sorted = df.sort_values("Close-Approach (CA) Date")
sns.lineplot(x=df_sorted["Close-Approach (CA) Date"], y=df_sorted["CA DistanceNominal (au)"], marker="o", alpha=0.5)
plt.xlabel("Date")
plt.ylabel("Close-Approach Distance (AU)")
plt.title("Time Series of NEO Close Approaches")
plt.xticks(rotation=45)
plt.show()
```




```
# Correlation heatmap

numeric_df = df.select_dtypes(include=['float64', 'int64'])
plt.figure(figsize=(8, 5))
sns.heatmap(numeric_df.corr(), annot=True, cmap="coolwarm", fmt=".2f")
plt.title("Correlation Heatmap of NEO Data")
plt.show()
```