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

1. Load the file

```
df=pd.read_csv("/content/2022_forbes_billionaires (2).csv")
```

2. Print first five rows of data

```
df.head()
```

	Unnamed: 0	rank	name	networth	age	country	source	industry
0	0	1	Elon Musk	$219 B	50	United States	Tesla, SpaceX	Automotive
1	1	2	Jeff Bezos	$171 B	58	United States	Amazon	Technology
2	2	3	Bernard Arnault & family	$158 B	73	France	LVMH	Fashion & Retail
3	3	4	Bill Gates	$129 B	66	United States	Microsoft	Technology
4	4	5	Warren Buffett	$118 B	91	United States	Berkshire Hathaway	Finance & Investments

3. Print last five rows of data

```
df.tail()
```

	Unnamed: 0	rank	name	networth	age	country	source	industry
2595	2595	2578	Jorge Gallardo Ballart	$1 B	80	Spain	pharmaceuticals	Healthcare
2596	2596	2578	Nari Genomal	$1 B	82	Philippines	apparel	Fashion & Retail
2597	2597	2578	Ramesh Genomal	$1 B	71	Philippines	apparel	Fashion & Retail
2598	2598	2578	Sunder Genomal	$1 B	68	Philippines	garments	Fashion & Retail
2599	2599	2578	Horst-Otto Gerberding	$1 B	69	Germany	flavors and fragrances	Food & Beverage

4. Check for missing and null values and duplicate data

```
df.columns
```

```
Index(['Unnamed: 0', 'rank', 'name', 'networth', 'age', 'country', 'source',
       'industry'],
      dtype='object')
```

5. Get some info about the data

```
df.info ()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2600 entries, 0 to 2599
Data columns (total 8 columns):
 #   Column      Non-Null Count  Dtype
---  ------      --------------  -----
 0   Unnamed: 0  2600 non-null   int64
 1   rank        2600 non-null   int64
 2   name        2600 non-null   object
 3   networth    2600 non-null   object
 4   age         2600 non-null   int64
 5   country     2600 non-null   object
 6   source      2600 non-null   object
 7   industry    2600 non-null   object
dtypes: int64(3), object(5)
memory usage: 162.6+ KB
```

6. Get some describtion about data

```
df.describe()
```

	Unnamed: 0	rank	age
count	2600.000000	2600.000000	2600.000000
mean	1299.500000	1269.570769	64.271923
std	750.699674	728.146364	13.220607
min	0.000000	1.000000	19.000000
25%	649.750000	637.000000	55.000000
50%	1299.500000	1292.000000	64.000000
75%	1949.250000	1929.000000	74.000000
max	2599.000000	2578.000000	100.000000

7. Get some shape of the data

```
print(df.shape)
```

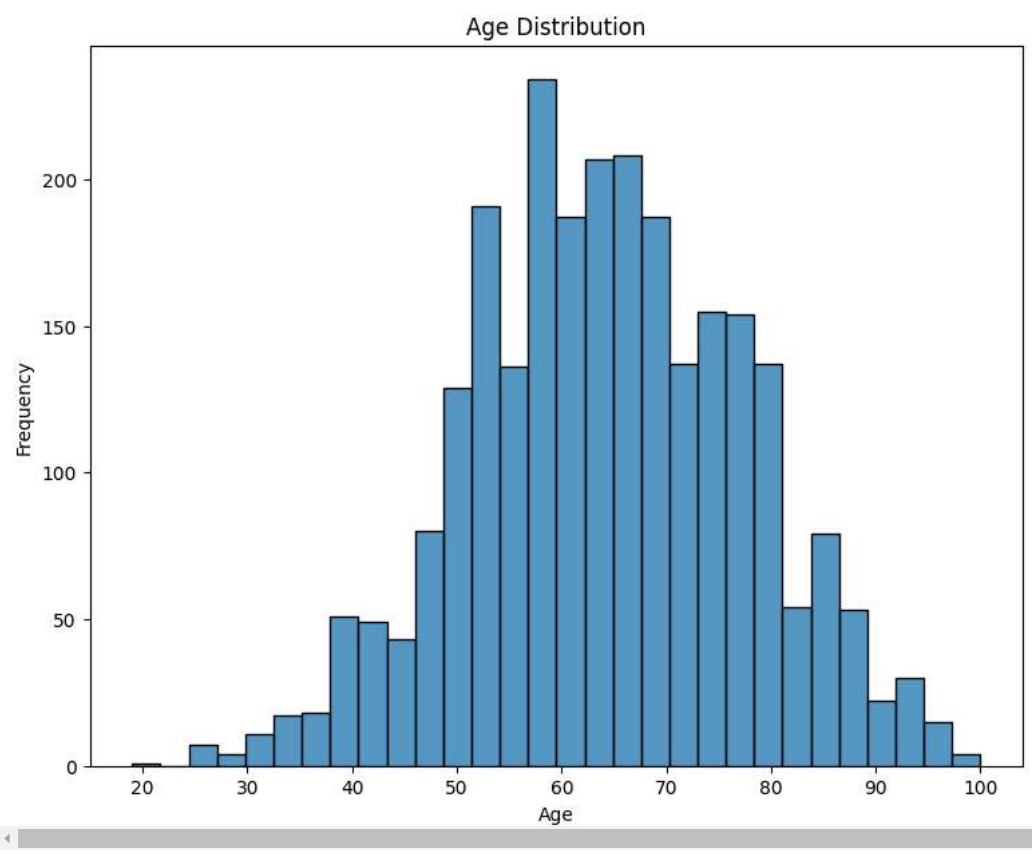
```
(2600, 8)
```

Visualization

1. Show the Age distribution among the data using bar plot

```
plt.figure(figsize=(9, 7))
sns.histplot(df['age'], bins=30, kde=False)
plt.title('Age Distribution')
plt.xlabel('Age')
plt.ylabel('Frequency')
```

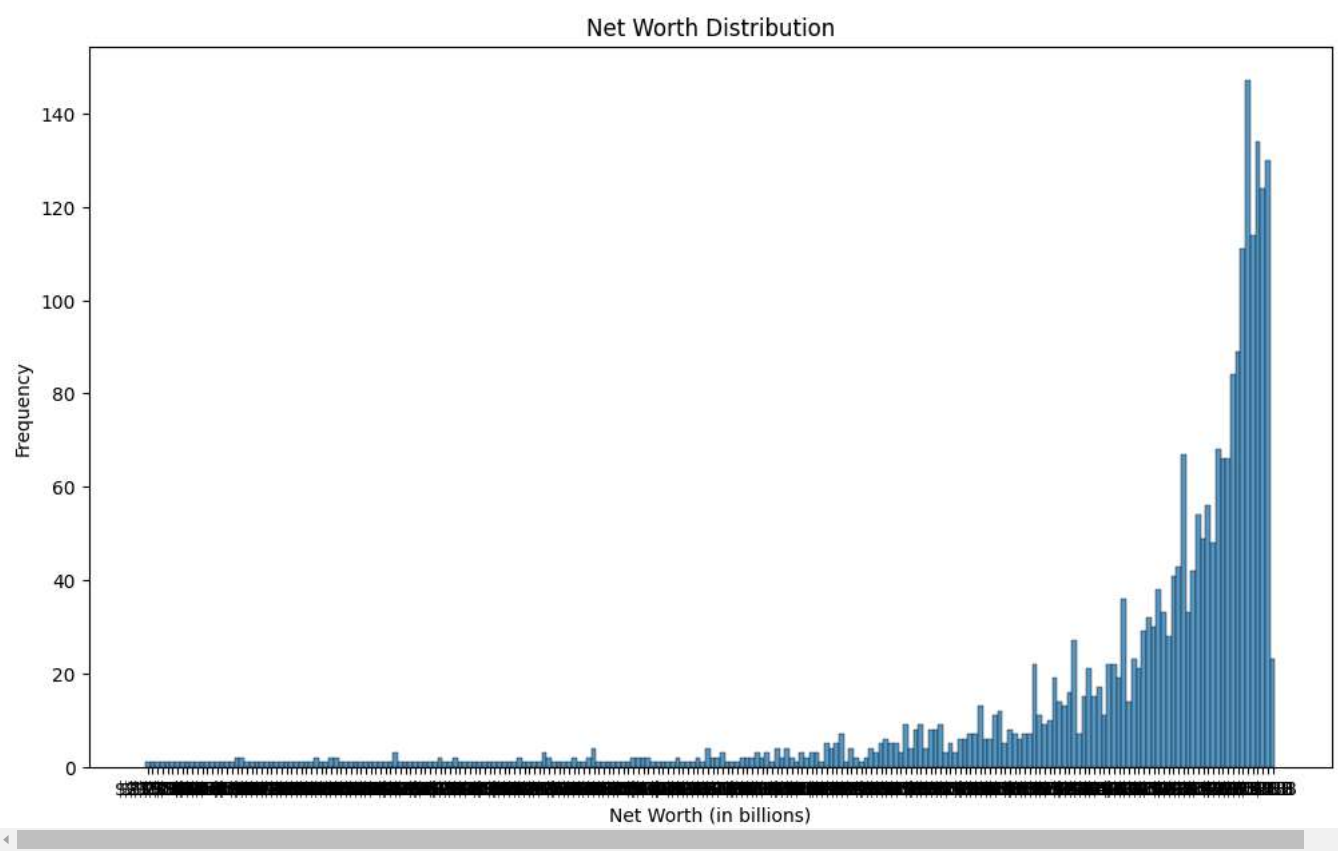
```
plt.show()
```



This plot shows the distribution of ages among the billionaires. It helps identify the most common age groups and the overall age range of the richest people.

2. Show the Net Worth vs Frequency using barplot

```
plt.figure(figsize=(12, 7))
sns.histplot(df['networth'], bins=30, kde=False)
plt.title('Net Worth Distribution')
plt.xlabel('Net Worth (in billions)')
plt.ylabel('Frequency')
plt.show()
```

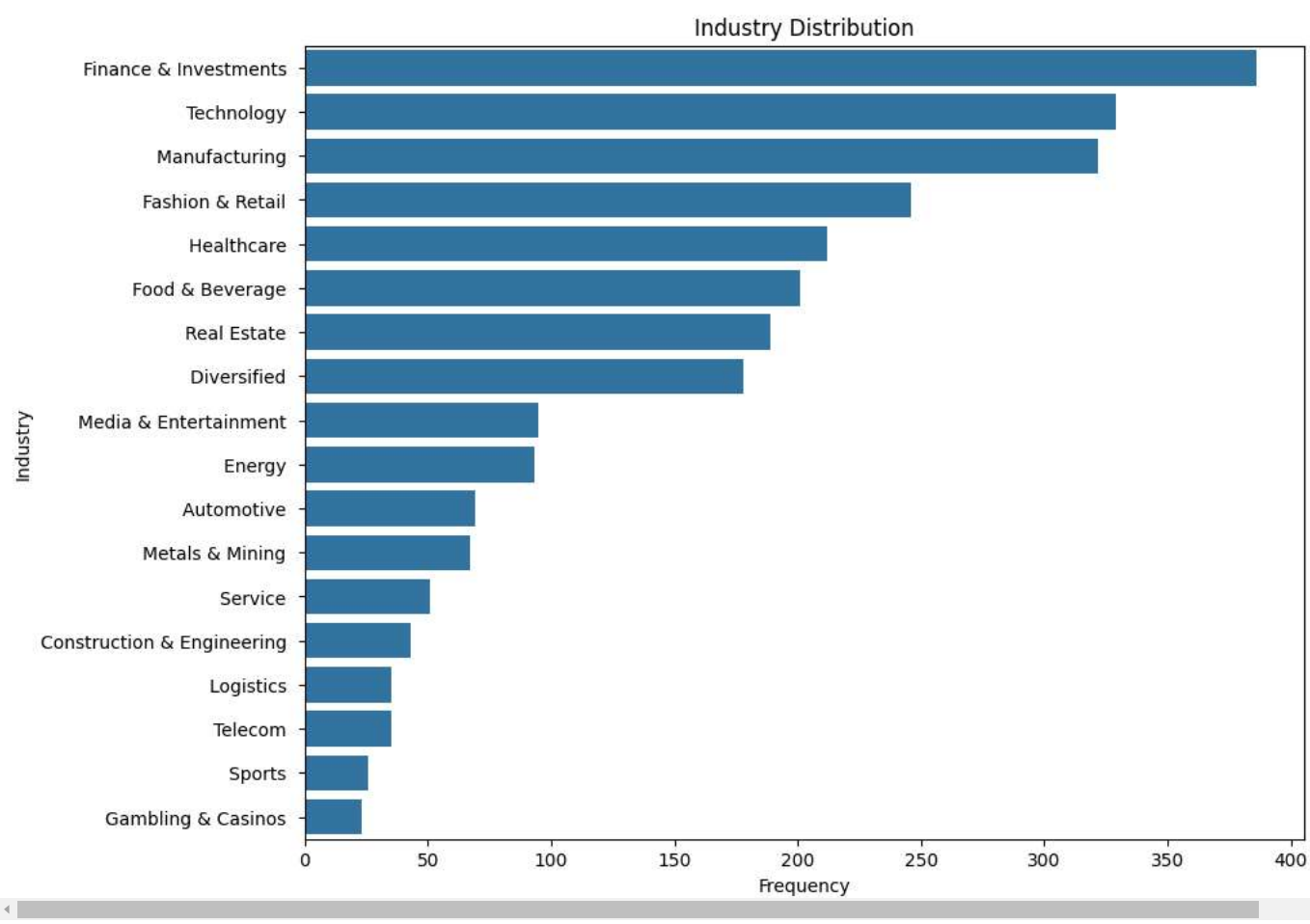


This plot illustrates the distribution of net worth among the billionaires. It highlights the most common net worth ranges and can show if there are any outliers with exceptionally high net worth.

3. Show Industry vs Frequency using bar plot

Double-click (or enter) to edit

```
plt.figure(figsize=(10, 8))
sns.countplot(y='industry', data=df, order=df['industry'].value_counts().index)
plt.title('Industry Distribution')
plt.xlabel('Frequency')
plt.ylabel('Industry')
plt.show()
```

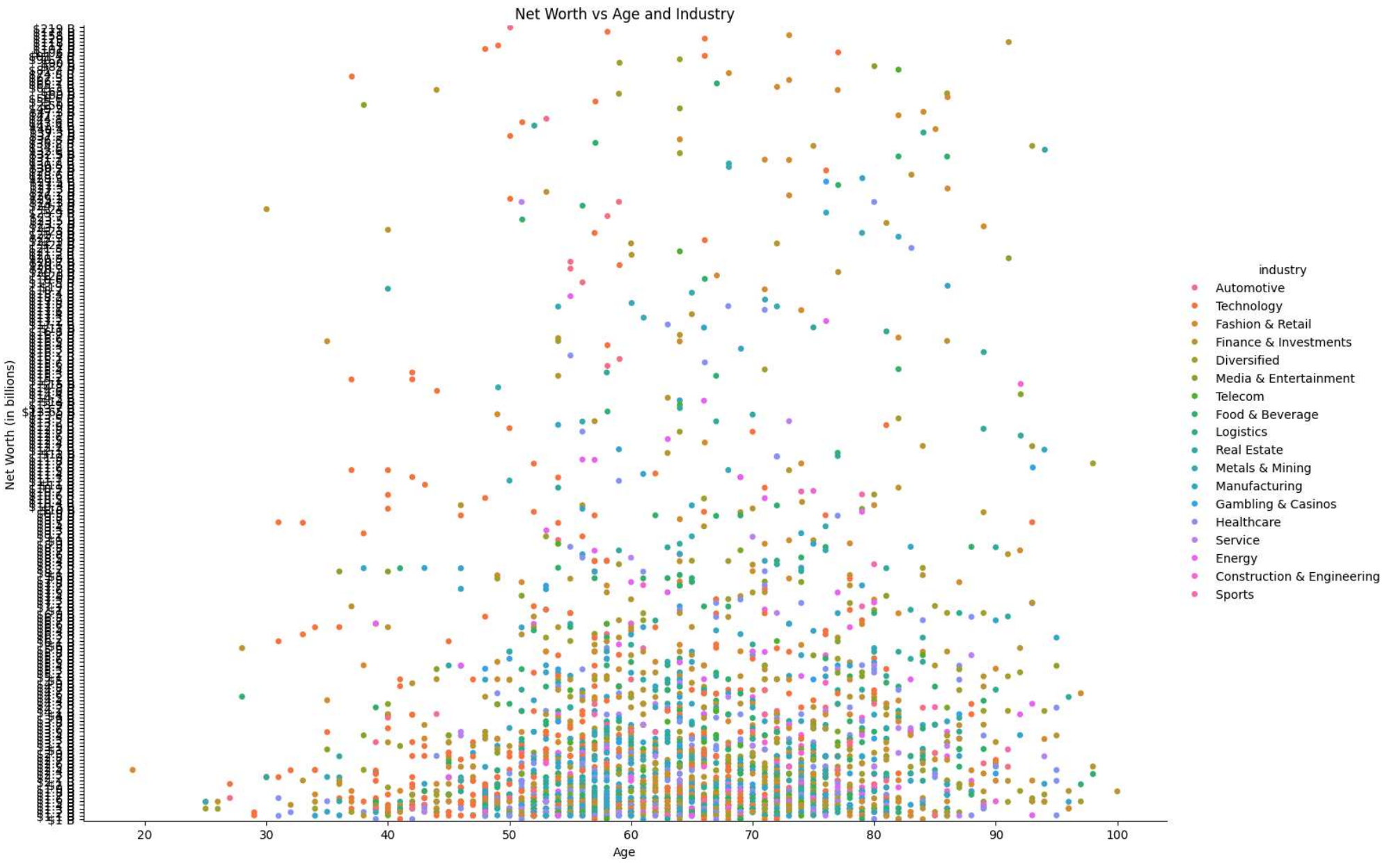


This bar plot shows the number of billionaires in each industry. It helps identify which industries are most represented among the richest people.

4. Show the how does Net worth Change with age and industry using cat plot

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

# Assuming 'data' is your DataFrame
sns.catplot(x='age', y='networth', hue='industry', data=df, kind='strip', height=10, aspect=1.4)
plt.title('Net Worth vs Age and Industry')
plt.xlabel('Age')
plt.ylabel('Net Worth (in billions)')
plt.show()
```



This cat plot shows the relationship between age, net worth, and industry. It helps visualize how net worth varies with age across different industries.

Double-click (or enter) to edit

5. Show the top 10 richest people vs net worth

Double-click (or enter) to edit

```
df[['name', 'networth']].head(10)
```

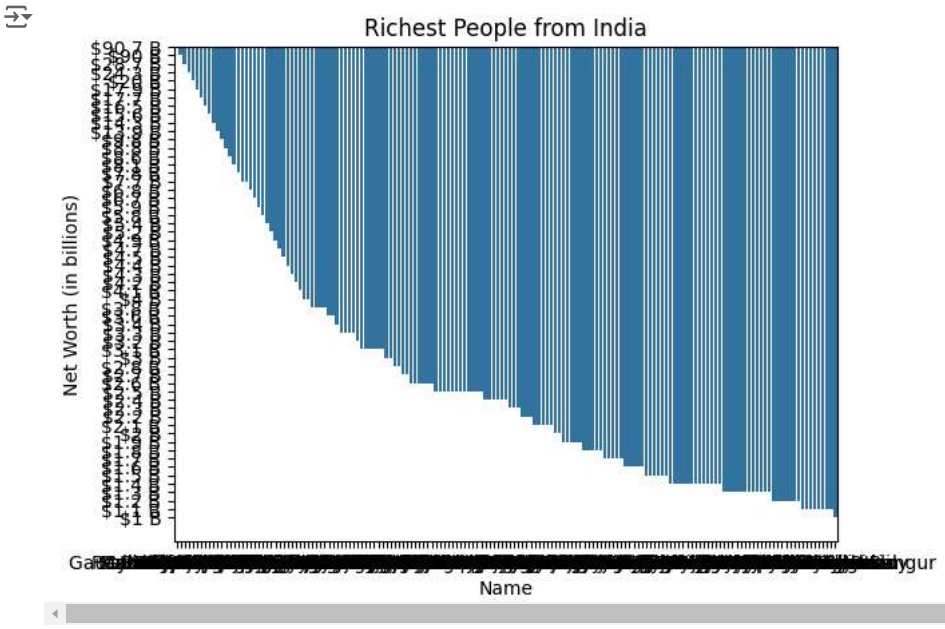
	name	networth
0	Elon Musk	$219 B
1	Jeff Bezos	$171 B
2	Bernard Arnault & family	$158 B
3	Bill Gates	$129 B
4	Warren Buffett	$118 B
5	Larry Page	$111 B
6	Sergey Brin	$107 B
7	Larry Ellison	$106 B
8	Steve Ballmer	$91.4 B
9	Mukesh Ambani	$90.7 B

This bar plot highlights the top 10 richest people and their net worth. It provides a clear comparison of the wealthiest individuals.

Double-click (or enter) to edit

6. Show the richest people from india with the names in any plot

```
india_richestpeople = df[df['country'] == 'India']
india_richestpeople[['name', 'networth']]
sns.barplot(x='name', y='networth', data=india_richestpeople)
plt.title('Richest People from India')
plt.xlabel('Name')
plt.ylabel('Net Worth (in billions)')
plt.show()
```



This plot shows the richest people from India and their net worth. It helps identify the top Indian billionaires.

7. Show the minimum age billionare <=50 with name name and industry

```
young_billionaires = df[df['age'] <= 50]
young_billionaires[['name', 'age', 'industry']]
```

	name	age	industry
0	Elon Musk	50	Automotive
5	Larry Page	49	Technology
6	Sergey Brin	48	Technology
14	Mark Zuckerberg	37	Technology
19	Changpeng Zhao	44	Finance & Investments
...	...	...	...
2567	Vlad Yatsenko	38	Finance & Investments
2570	Yu Rong	50	Healthcare
2579	Johanna Braun	42	Healthcare
2580	Karl Friedrich Braun	39	Healthcare
2587	Doug Clarke	41	Technology

359 rows × 3 columns

This code snippet identifies the youngest billionaire aged 50 or below, along with their name and industry. It highlights the youngest successful individuals.

8. Show in which industry billionare are more:-

```
top_industry = df['industry'].value_counts().idxmax()
print(f'Industry with the most billionaires: {top_industry}')
```

```
Industry with the most billionaires: Finance & Investments
```

This bar plot shows the number of billionaires in each industry, helping to identify which industries have the highest concentration of billionaires.