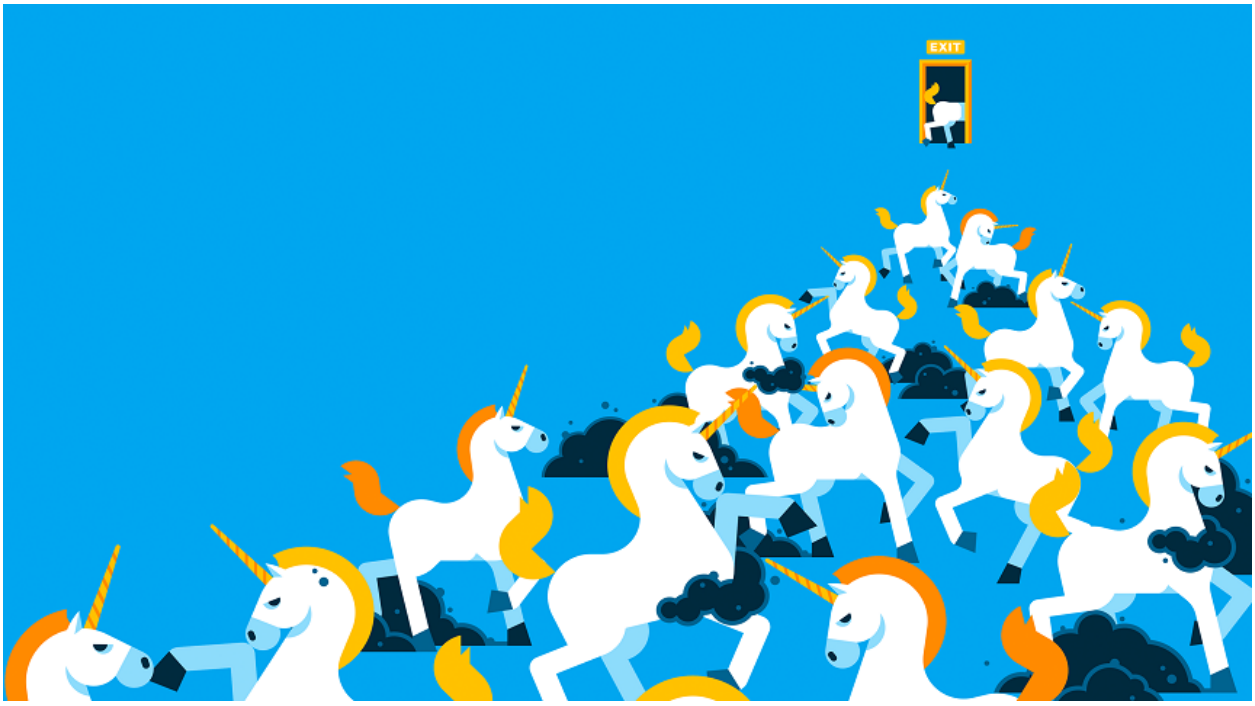
Exploratory Data Analysis Of Unicorn Companies In The World



Introduction

Exploratory data analysis is an essential step toward gaining insights from data. In this project, we will perform data cleaning and EDA on unicorn companies. The term unicorn refers to a privately held startup company with a value of over $1 billion. The report provides a comprehensive analysis of the data collected on unicorn companies, including their geographical distribution, industry focus, and funding history.

Data

The source of this dataset is kaggle. The dataset has 13 columns and 1035 rows.

Dataset URL: https://www.kaggle.com/datasets/deepcontractor/unicorn-companies-dataset

```
df.head()
```

	Company	Valuation ($B)	Date Joined	Country	City	Industry	Select Inverstors	Founded Year	Total Raised	Financial Stage	Investors Count	Deal Terms	Portfolio Exits
0	Bytedance	$140	4/7/2017	China	Beijing	Artificial intelligence	Sequoia Capital China, SIG Asia Investments, S...	2012	$7.44B	IPO	28	8	5
1	SpaceX	$100.3	12/1/2012	United States	Hawthorne	Other	Founders Fund, Draper Fisher Jurvetson, Rothen...	2002	$6.874B	None	29	12	None
2	Stripe	$95	1/23/2014	United States	San Francisco	Fintech	Khosla Ventures, LowercaseCapital, capitalG	2010	$2.901B	Asset	39	12	1
3	Klarna	$45.6	12/12/2011	Sweden	Stockholm	Fintech	Institutional Venture Partners, Sequoia Capita...	2005	$3.472B	Acquired	56	13	1
4	Epic Games	$42	10/26/2018	United States	Cary	Other	Tencent Holdings, KKR, Smash Ventures	1991	$4.377B	Acquired	25	5	2

Columns: Company, Valuation, Date Joined, City, Industry, Select Investors, Founded year, Total Raised, Financial Stage, Investors Count, Deal Terms, Portfolio Exits.

```
Company              object
Valuation ($B)       object
Date Joined          object
Country              object
City                 object
Industry             object
Select Inverstors    object
Founded Year         object
Total Raised         object
Financial Stage      object
Investors Count      object
Deal Terms           object
Portfolio Exits      object
dtype: object
```

Data Cleaning

- **Dropping Rows that have "None" values in Founded Year, Total Raised, Investors Count, and Select Investors.**

```
df=df.drop(df[df["Founded Year"]=="None"].index)

df=df.drop(df[df["Total Raised"]=="None"].index)

df=df.drop(df[df["Investors Count"]=="None"].index)

df=df.drop(df[df["Select Inverstors"]=="None"].index)

df
```

- **As in the Columns, Financial Stage and Portfolio Exits most of the data is missing, therefore dropping them will be appropriate.**

```
df=df.drop(["Financial Stage","Portfolio Exits"],axis=1)

df
```

- **Getting the actual value of the Total Raised**

```
df["Total Raised Unit"] = df["Total Raised"].str[-1]



df["Total Raised"] = df["Total Raised"].replace({"\$":"",
"B$":"", "M$":"", "None":np.nan, "K$":""}, regex = True)

df["Total Raised"] = df["Total Raised"].astype(float)



for i, row in df.iterrows():

  if row["Total Raised Unit"] == "B":

    df.loc[i , "Total Raised"] = row["Total Raised"] *
1_000_000_000
```

```
  elif row["Total Raised Unit"] == "M":
    df.loc[i, "Total Raised"] = row["Total Raised"] * 1_000_000


  elif row["Total Raised Unit"] == "K":
    df.loc[i, "Total Raised"] = row["Total Raised"] * 1_000


df = df.drop("Total Raised Unit", axis=1)


df
```

- **Replacing the wrong spellings in the data.**

```
df["Industry"]=df["Industry"].str.replace("Artificial
intelligence","Artificial Intelligence")

df["Industry"]=df["Industry"].str.replace("Finttech","Fintech")

df["Valuation ($B)"]=df["Valuation ($B)"].astype(float)

df["Valuation ($B)"]=df["Valuation ($B)"].str.replace("$"," ")
```

- **Changing Column Types.**

```
df["Valuation ($B)"]=df["Valuation ($B)"].astype(float)

df["Investors Count"]=df["Investors Count"].astype(int)

df["Date"]=df["Date"].astype(int)

df["Month"]=df["Month"].astype(int)

df["Year"]=df["Year"].astype(int)

df["Founded Year"]=df["Founded Year"].astype(int)
```

- **Converting Date Joined to date time format**

```
from datetime import datetime

df["Date Joined"]=pd.to_datetime(df["Date Joined"])
```

- **Splitting Date Joined into Date, Month, and Year will be helpful to find the years taken to become a unicorn.**

```
df[['Date','Month','Year']]=df['Date
Joined'].str.split('/',expand=True)

df
```

- **Adding a new column "Years Taken to become Unicorn"**

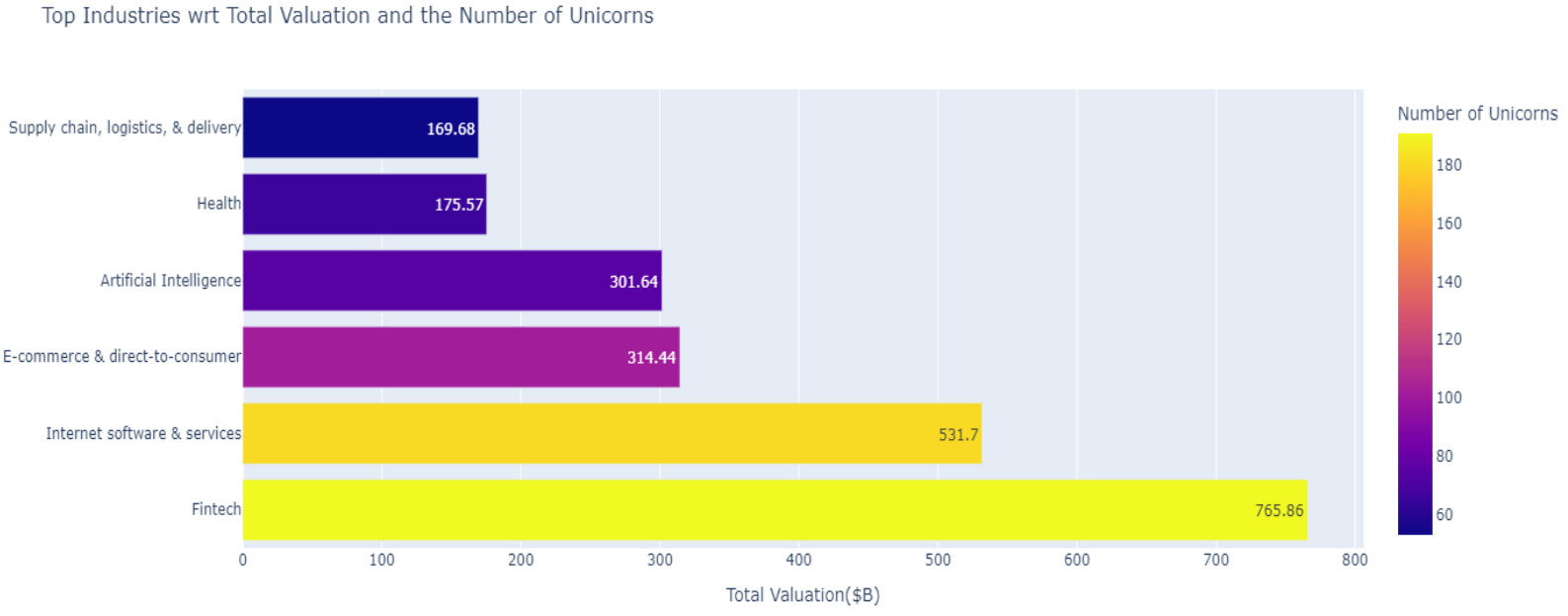
```
df['Years Taken to become Unicorn'] = df['Year'] - df['Founded
Year']

df['Years Taken to become Unicorn']=df['Years Taken to become
Unicorn'].astype(int)

df
```

EDA

Industry Wise Analysis:

- **Top Industries wrt the Highest Valuation($B) and the Number of Unicorns**

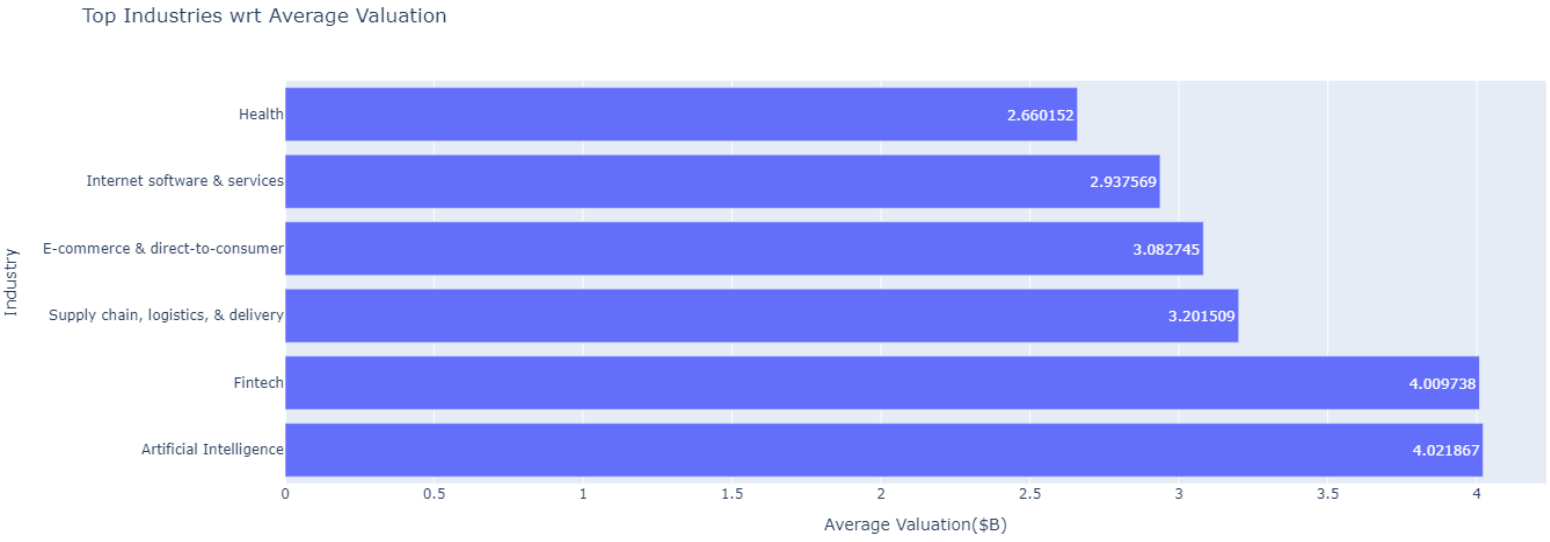
```
temp=df.groupby(by=['Industry']).agg({'Company':['count'],'Valuat
ion ($B)': ['sum','mean']})

temp=temp.sort_values([('Company', 'count')],
ascending=False)[:6]

fig=px.bar(x=temp[('Valuation
($B)','sum')],y=temp.index,title="Top Industries wrt Total
Valuation and the Number of
Unicorns",text_auto=True,color=temp[('Company',
'count')],labels={

                    "y": "Industry",

                    "x": "Total Valuation($B)",

                    "color":"Number of Unicorns"})

fig.show()
```



Fintech, Internet software & services, and E-commerce are the top 3 industries under which many companies have become successful. Artificial Intelligence is also on par with other industries.

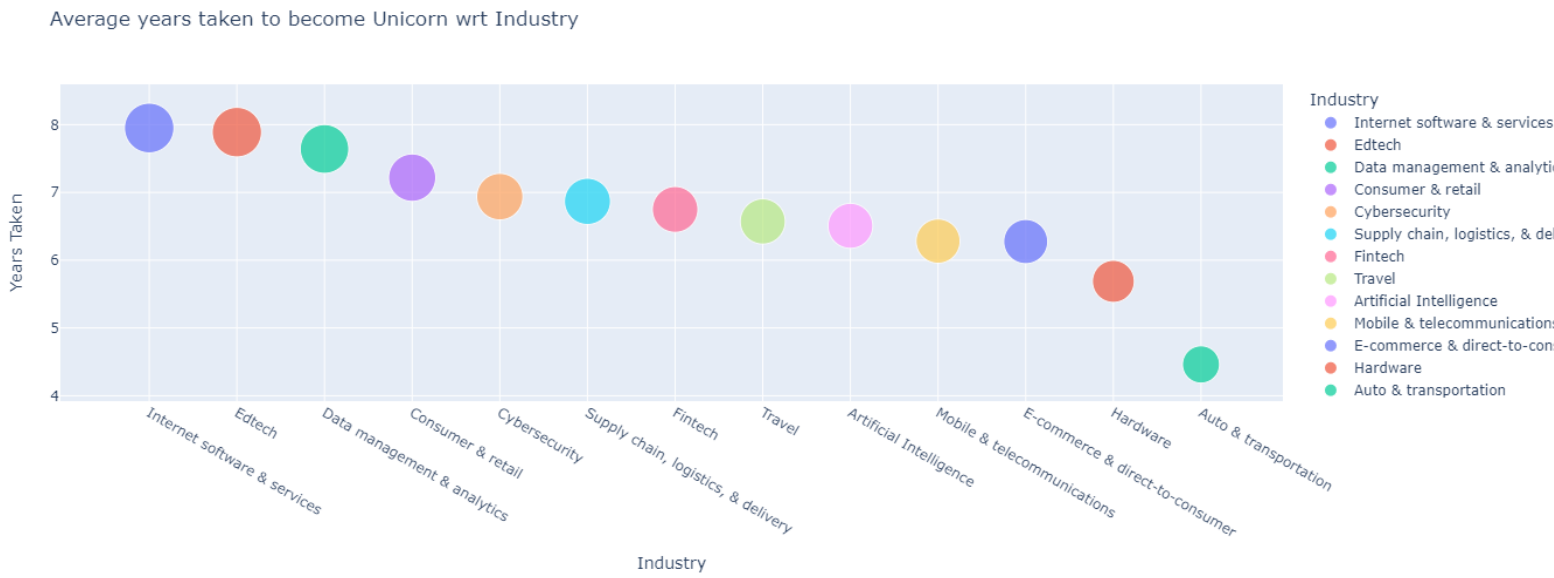
- **Top Industries wrt the Average Valuation($B)**

```
temp=temp.sort_values([('Valuation ($B)', 'mean')],
ascending=False)[:6]

fig=px.bar(x=temp[('Valuation ($B)',
'mean')],y=temp.index,title="Top Industries wrt Average
Valuation",text_auto=True,labels={

                    "y": "Industry", "x": "Average
Valuation($B)",

                  })

fig.show()
```



- **Average Years taken to become Unicorn from different Industries**

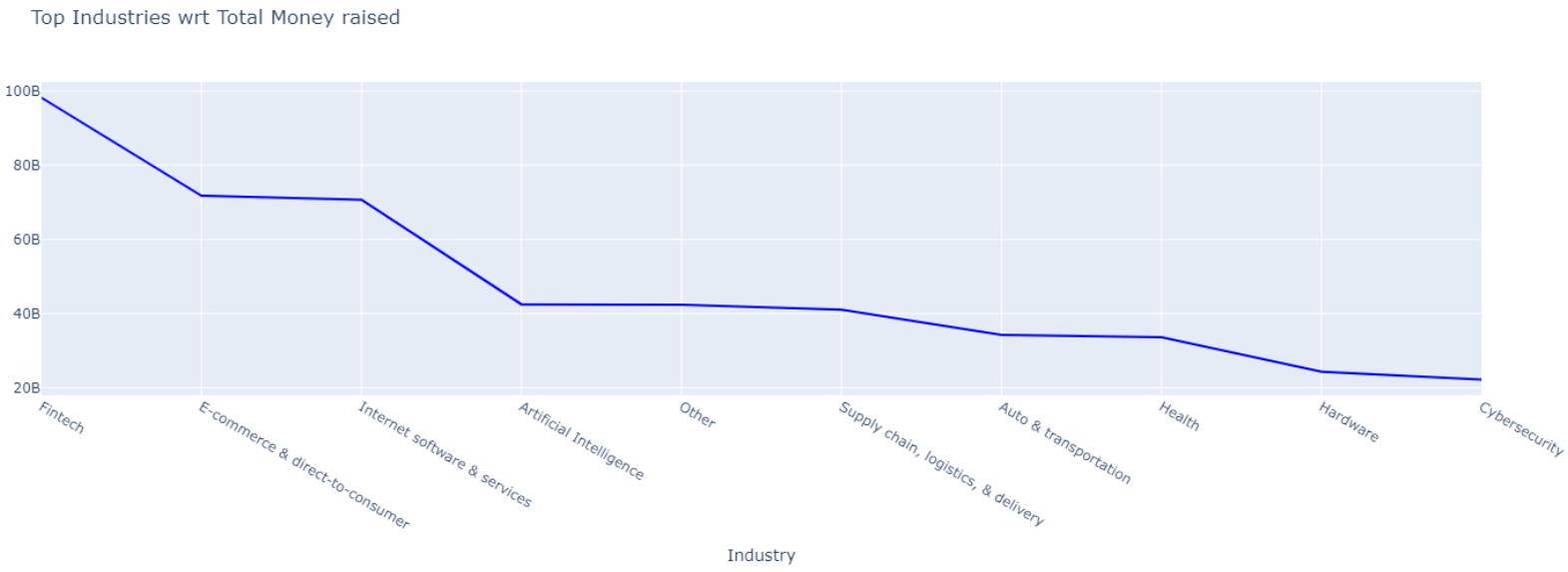
```
temp=df.groupby(by=['Industry']).agg({'Years Taken to become Unicorn':['mean']})

temp=temp.sort_values([('Years Taken to become Unicorn', 'mean')], ascending=False)[2:15]

fig=px.scatter(y=temp[('Years Taken to become Unicorn','mean')],x=temp.index,size=temp[('Years Taken to become Unicorn', 'mean')],size_max=30,color=temp.index,title="Average years taken to become Unicorn wrt Industry",labels={"y":"Years Taken","x":"Industry", "color": "Industry"})

fig.show()
```



Companies in the Auto & Transportation industry take the least number of years on average to become unicorns whereas companies in the Internet software and Edtech sector take considerably more time.

- ## Distribution of Money Raised wrt Industries

```
temp=df.groupby(by=['Industry']).agg({'Total Raised':['sum']})

temp=temp.sort_values([('Total Raised', 'sum')],
ascending=False)[:10]

fig=px.line(y=temp[('Total Raised',
'sum')],x=temp.index,title="Top Industries wrt Total Money
raised",labels={"y":"Total Raised($B)","x":"Industry"})

fig.update_traces(line_color="blue")

fig.show()
```

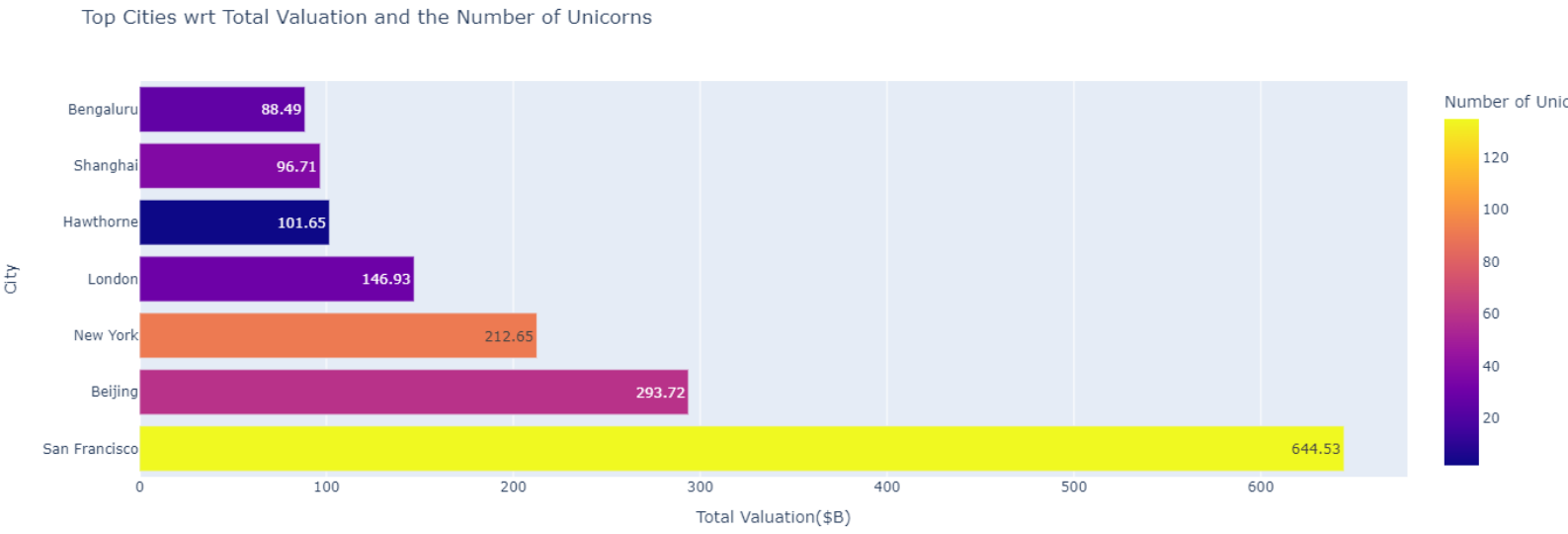


Companies in the Fintech industry have raised the highest amount of money, followed by E-commerce and Internet Software and Services.

City Wise Analysis:

- **Top Cities wrt the Highest Valuation($B) and the Number of Unicorns**

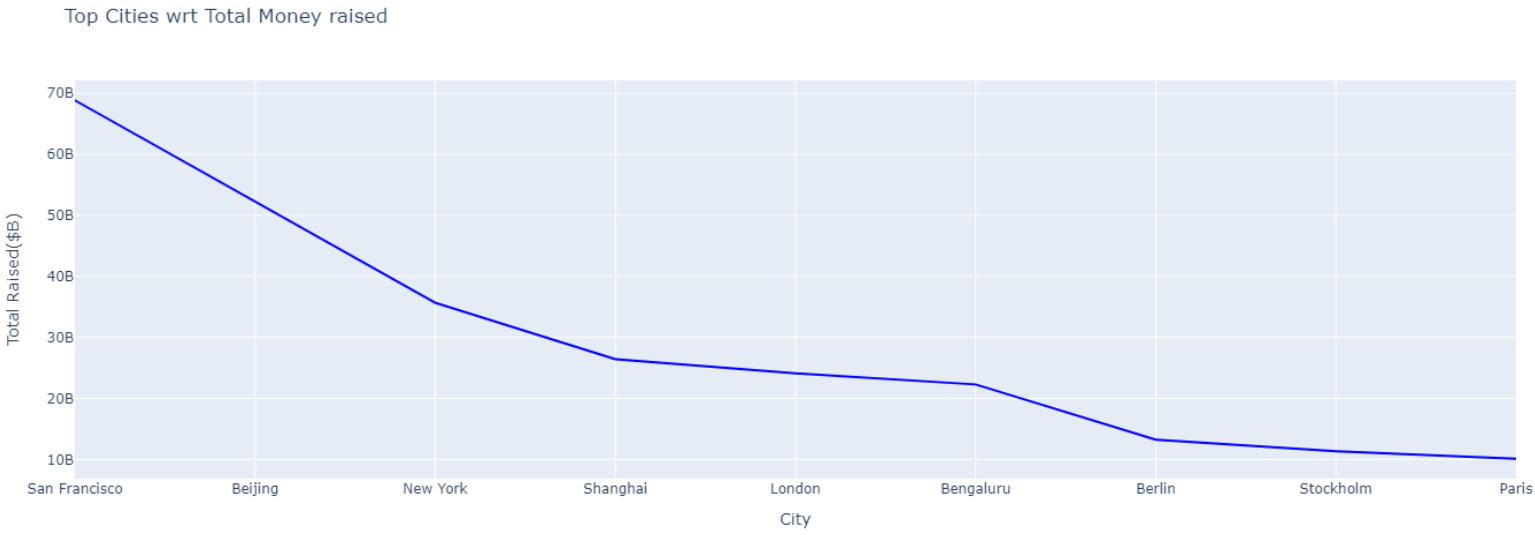
```
temp=df.groupby(by=['City']).agg({'Company':['count'],'Valuation ($B)':
['sum']})

temp=temp.sort_values([('Valuation ($B)', 'sum')], ascending=False)[:7]

fig=px.bar(x=temp[('Valuation ($B)','sum')],y=temp.index,title="Top
Cities wrt Total Valuation and the Number of
Unicorns",text_auto=True,color=temp[('Company', 'count')], labels={

                    "y": "City",

                    "x": "Total Valuation($B)",

                    "color":"Number of Unicorns"})

fig.show()
```



This is a follow-up to the previous graph. San Francisco, a city in the United States, has the highest number of unicorns and total valuation, followed by Beijing and closely by New York, cities of China, and the United States respectively. Many successful companies are also likely to be found in other cities like London, Bengaluru, and Shanghai.

- ## Distribution of Money Raised in Top Cities

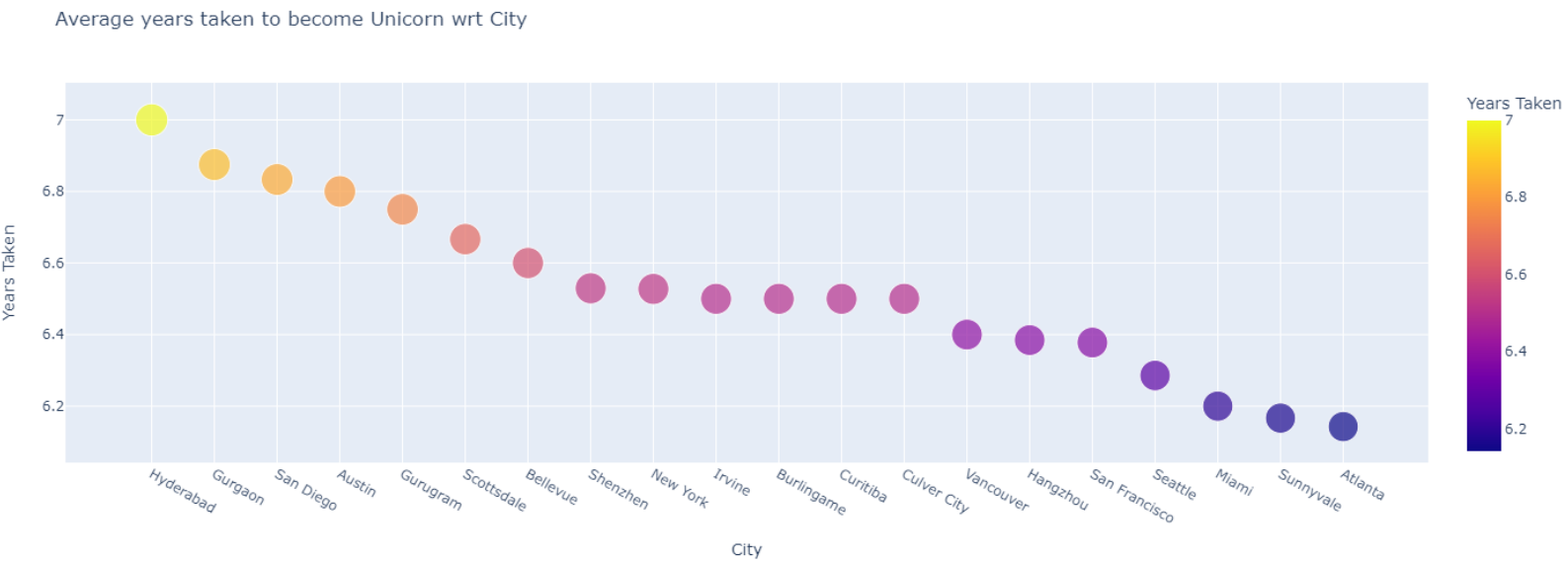
```
temp=df.groupby(by=['City']).agg({'Total Raised':['sum']})

temp=temp.sort_values([('Total Raised', 'sum')],
ascending=False)[:9]

fig=px.line(y=temp[('Total Raised',
'sum')],x=temp.index,title="Top Cities wrt Total Money
raised",labels={"y":"Total Raised($B)","x":"City"})

fig.update_traces(line_color="blue")

fig.show()
```



San Francisco, Beijing, and New York are the top 3 cities with respect to the total money raised by all the companies present in the respective cities. Cities like Bengaluru and Berlin also have a quite good share in the total money raised on average.

- **Average years taken to become Unicorn wrt City**

```
temp=df.groupby(by=['City']).agg({'Years Taken to become
Unicorn':['mean']})

temp=temp.sort_values([('Years Taken to become Unicorn',
'mean')], ascending=False)[115:135]

fig=px.scatter(y=temp[('Years Taken to become
Unicorn','mean')],x=temp.index,size=temp[('Years Taken to become
Unicorn', 'mean')],color=temp[('Years Taken to become Unicorn',
'mean')],title="Average years taken to become Unicorn wrt
City",labels={"y":"Years Taken","x":"City", "color": "Years
Taken"})

fig.show()
```

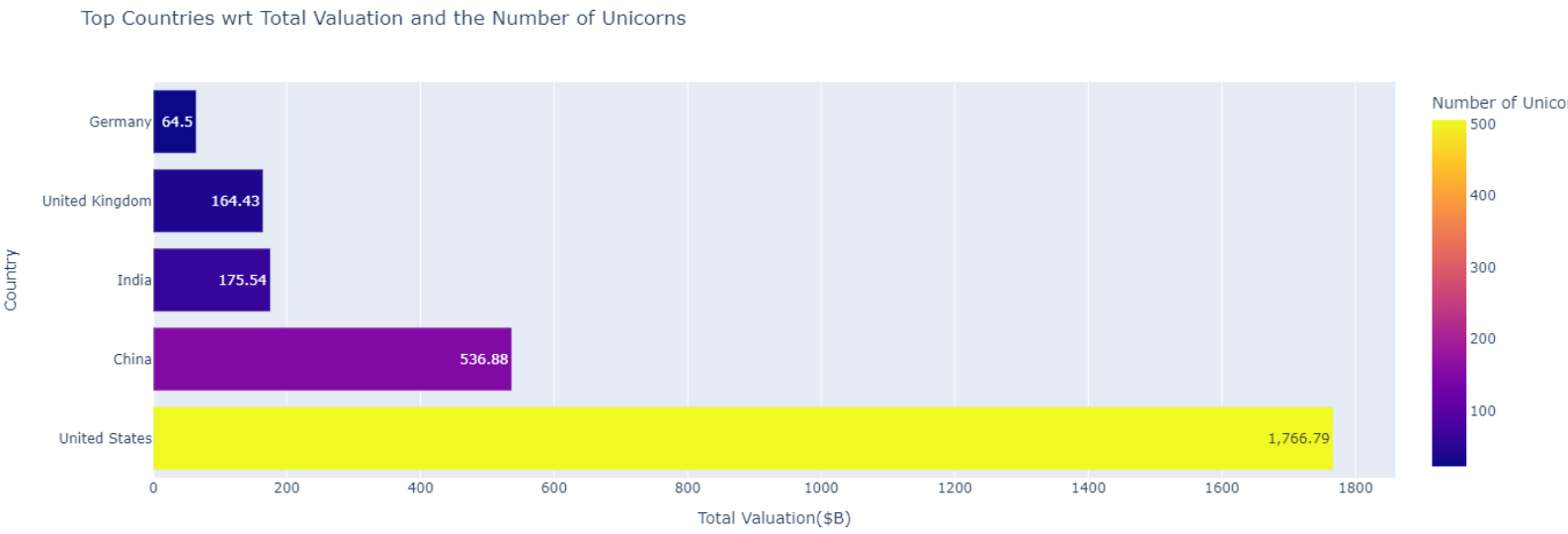


Quite easy to interpret, the above plot shows us the cities ordered with respect to the time taken to become unicorns.

Country Wise Analysis:

- **Top Countries wrt the Highest Valuation($B) and the Number of Unicorns**

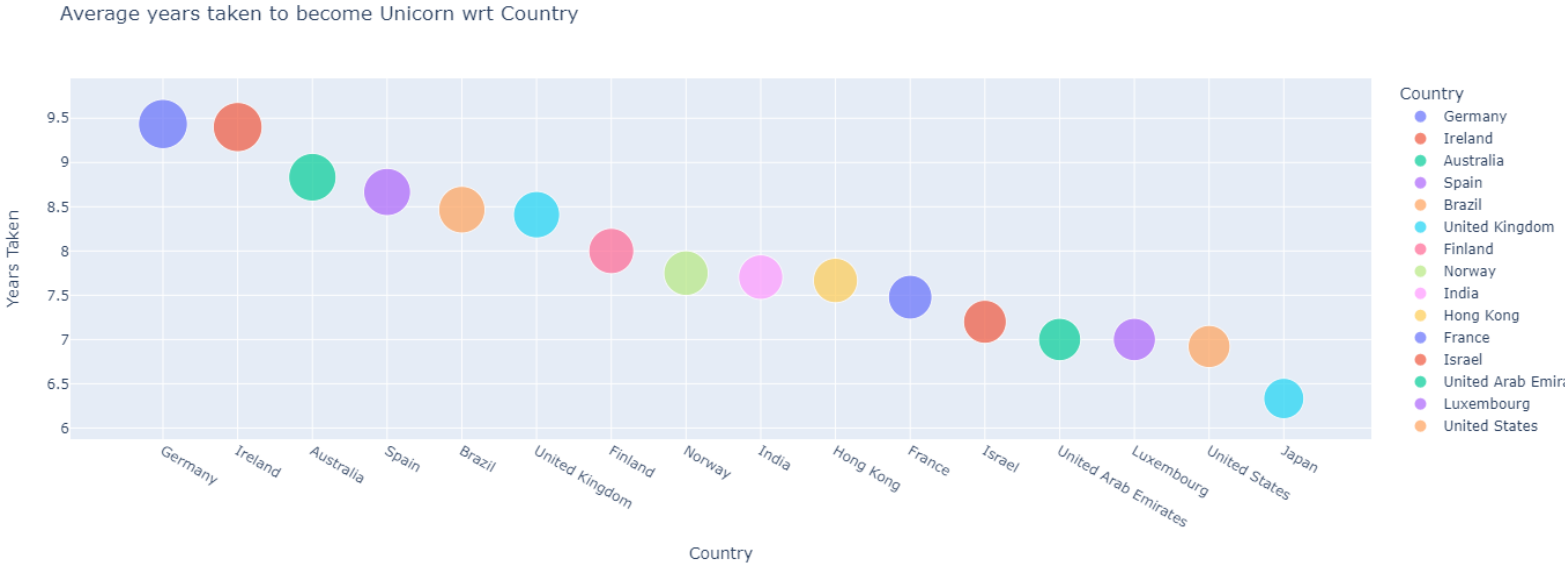
```
temp=df.groupby(by=['Country']).agg({'Company':['count'],'Valuation
($B)': ['sum']})

temp=temp.sort_values([('Company', 'count')], ascending=False)[:5]

fig=px.bar(x=temp[('Valuation ($B)','sum')],y=temp.index,title="Top
Countries wrt Total Valuation and the Number of
Unicorns",text_auto=True,color=temp[('Company', 'count')], labels={

                    "y": "Country",

                    "x": "Total Valuation($B)",

                    "color":"Number of Unicorns"})

fig.show()
```



From the above graph, we can imply that the United States has the highest number of companies that become unicorns and emerge successful with a whopping total valuation of nearly 1800 billion followed by China and India; the United Kingdom performing equally well.

- **Average Years taken to become a Unicorn from different Countries**

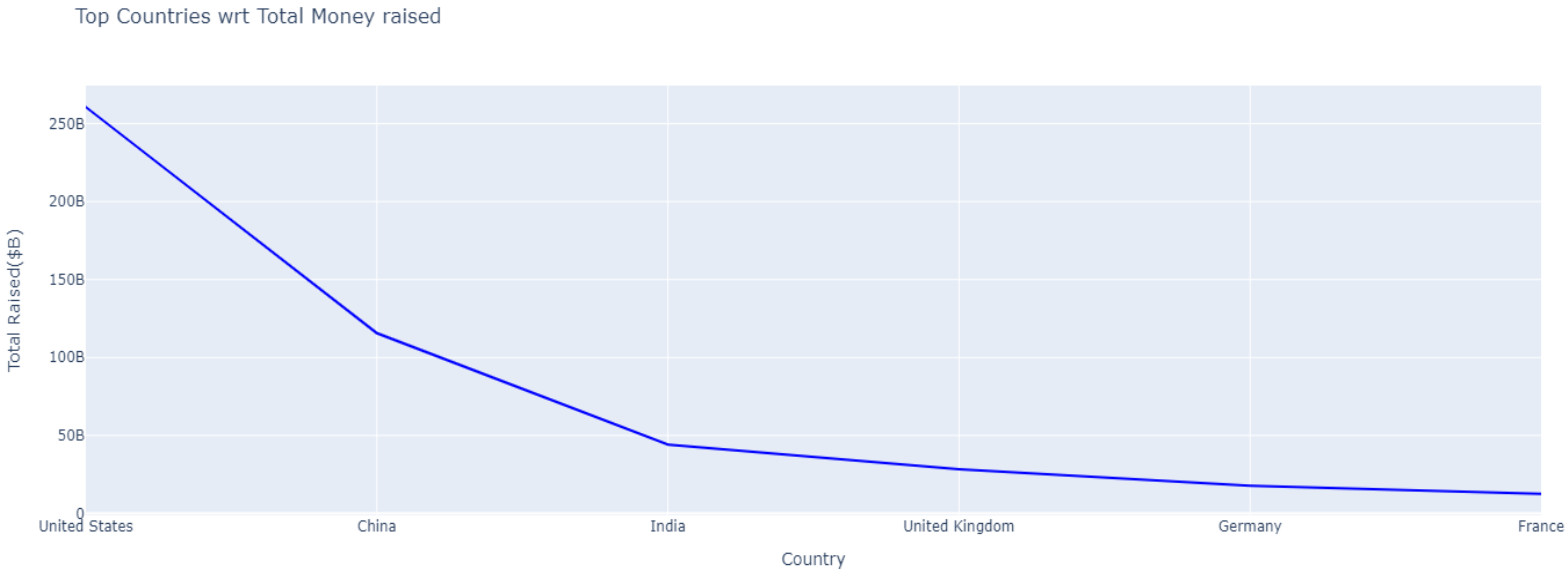
```
temp=df.groupby(by=['Country']).agg({'Years Taken to become Unicorn':['mean']})

temp=temp.sort_values([('Years Taken to become Unicorn', 'mean')], ascending=False)[9:25]

fig=px.scatter(y=temp[('Years Taken to become Unicorn','mean')],x=temp.index,size=temp[('Years Taken to become Unicorn', 'mean')],size_max=30,color=temp.index,title="Average years taken to become Unicorn wrt Country",labels={"y":"Years Taken","x":"Country", "color": "Country"})

fig.show()
```



Companies in Germany take more years to become unicorns while in countries like Japan, and the US provides a better environment for startups as their startups are becoming successful in a shorter duration.

- **Distribution of Money Raised in Top Countries**

```
temp=df.groupby(by=['Country']).agg({'Total Raised':['sum']})

temp=temp.sort_values([('Total Raised', 'sum')],
ascending=False)[:6]

fig=px.line(y=temp[('Total Raised',
'sum')],x=temp.index,title="Top Countries wrt Total Money
raised",labels={"y":"Total Raised($B)","x":"Country"})

fig.update_traces(line_color="blue")

fig.show()
```

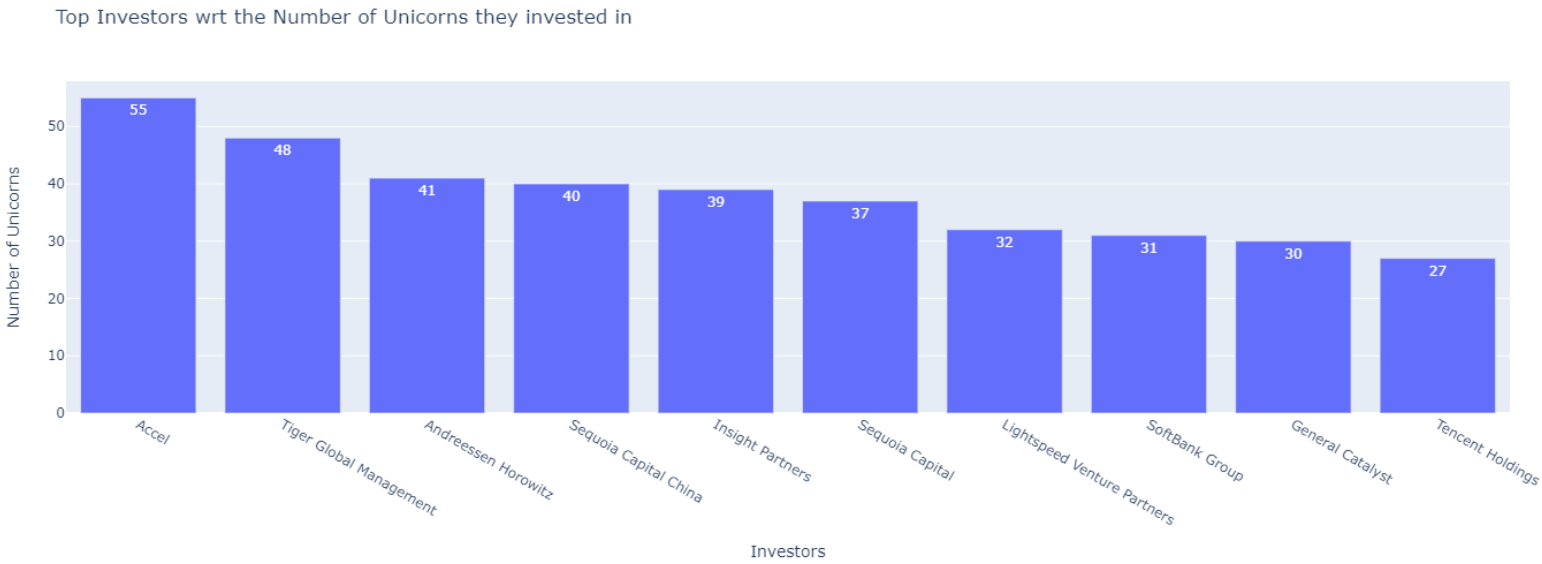


Companies in the US have raised the highest amount followed by China, India, UK, Germany, and France.

Investors and Time Wise Analysis:

- **Top Investors in terms of the Number of Unicorns they invested in**

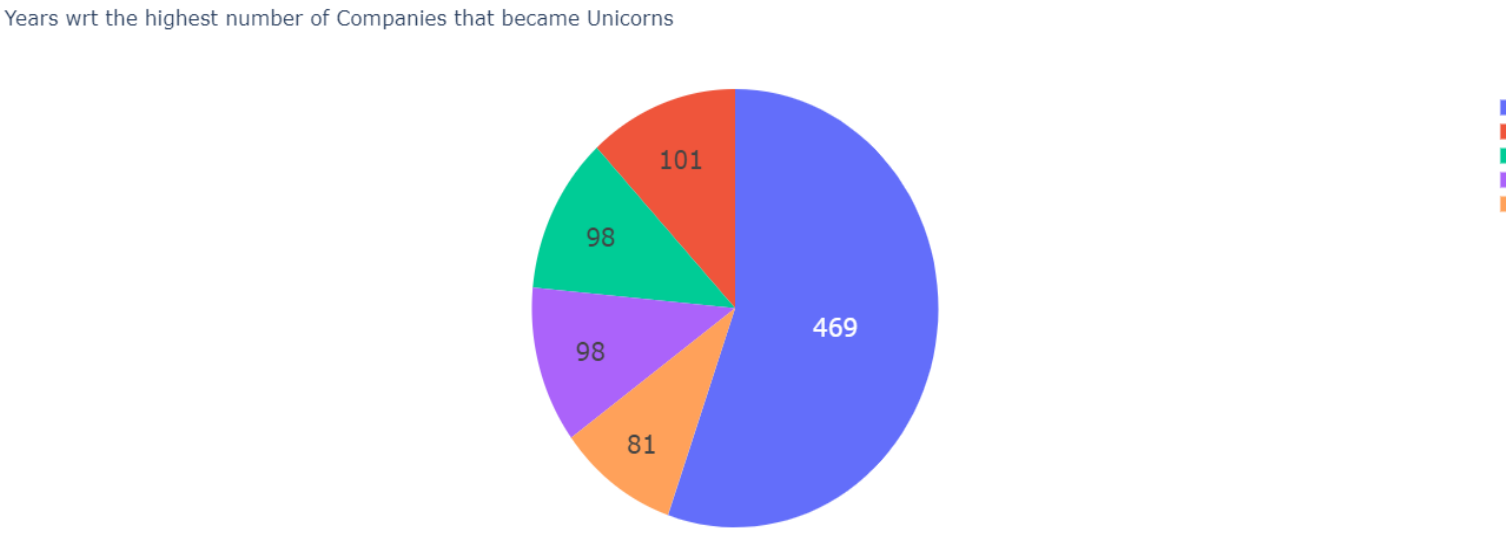
```
investors = []

for i, row in df.iterrows():  investors += row["Select
Inverstors"].split(', ')

investors = pd.Series(investors).value_counts()[:10]

investors.sort_values(ascending=False, inplace=True)

px.bar (investors.index,investors.values)

fig=px.bar(y=investors.values,x=investors.index,title="Top Investors
wrt the Number of Unicorns they invested
in",text_auto=True,labels={"x":"Investors","y":"Number of Unicorns"})

fig.show()
```



The bar graph indicates that Accel is the top investor in the world and has invested in 55 companies that became unicorns, followed by Tiger Global Management and Andreessen Horowitz.

- **Years in terms of the Highest Number of Companies that became Unicorns**

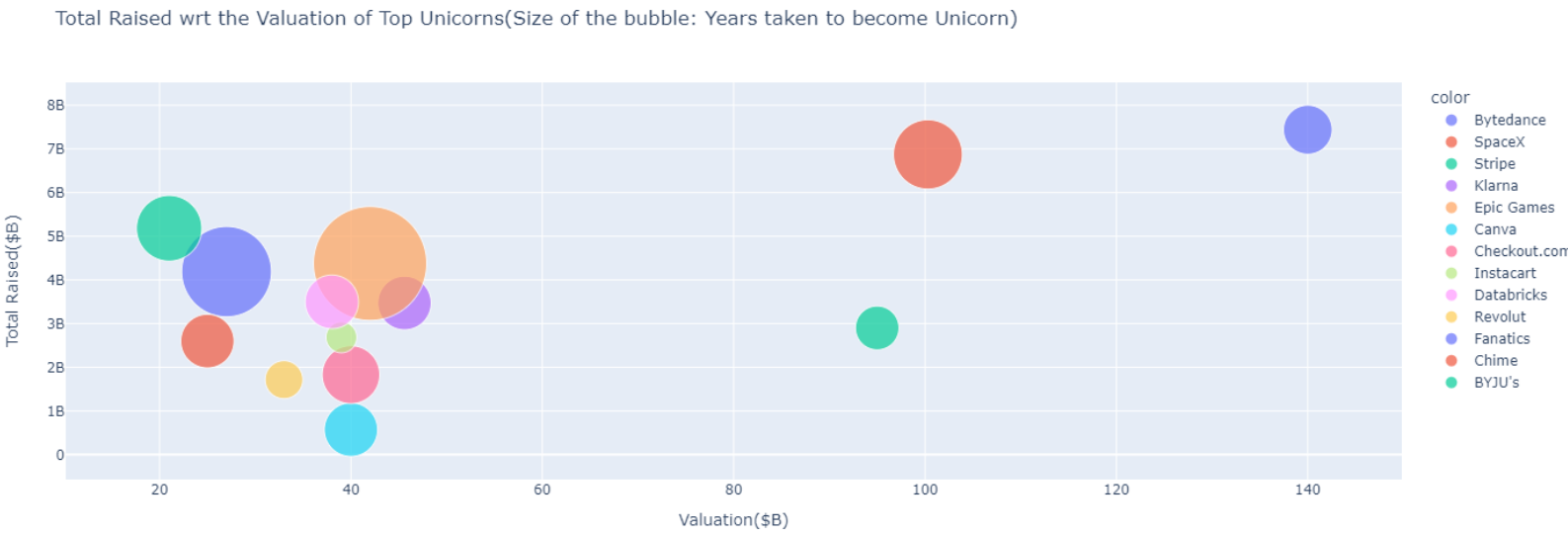
```
temp=df.groupby(by=['Year']).agg({'Company':['count']})

temp=temp.sort_values([('Company', 'count')],
ascending=False)[:5]

fig=px.pie(values=temp[('Company',
'count')],names=temp.index,title="Years wrt the highest number of
Companies that became Unicorns")

fig.update_traces(textinfo='value',textfont_size=20)

fig.show()
```



In 2021 many companies become successful, followed by the years 2020, 2019, and 2018; each year having an equal proportion of companies that became Unicorns.

- **Total Raised($B) in terms of the Valuation($B) of the Unicorns**

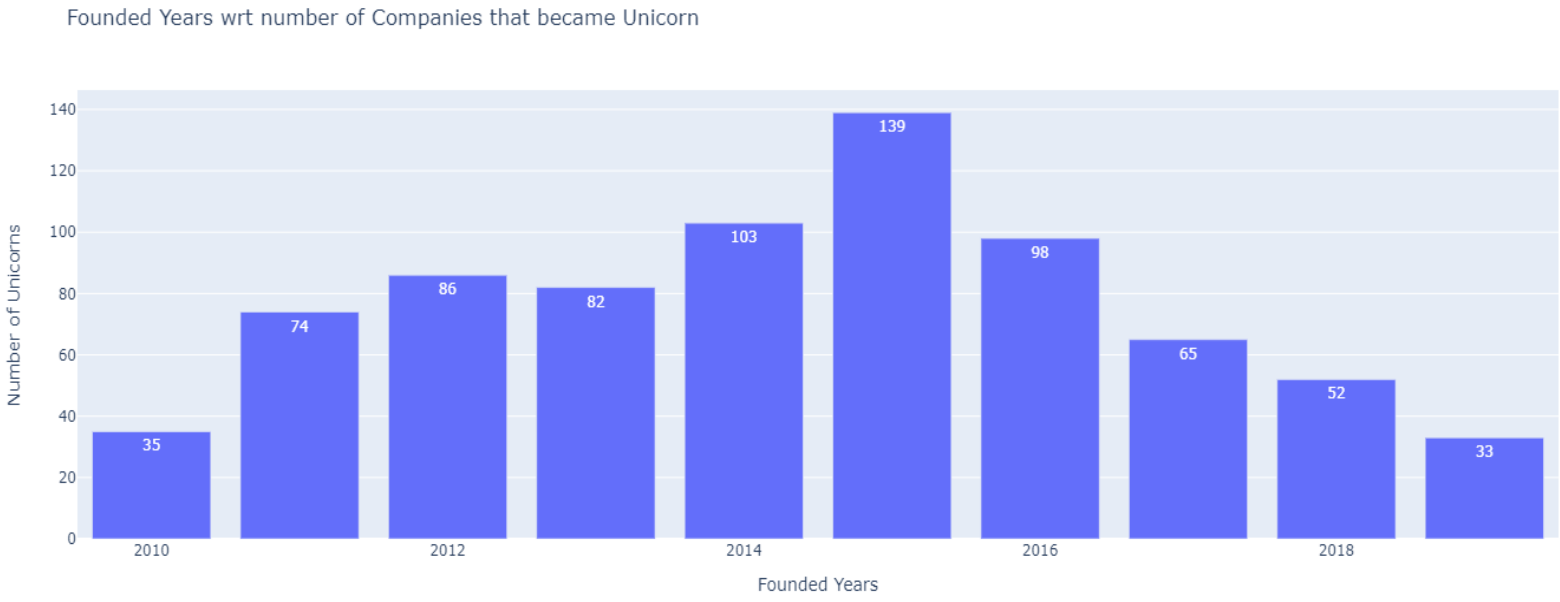
```
temp=df.sort_values("Valuation ($B)", ascending=False)[:13]

fig=px.scatter(x=temp['Valuation ($B)'], y=temp['Total
Raised'],size=temp['Years Taken to become Unicorn']
,hover_name=temp['Company'],color=temp['Company'],size_max=70,tit
le="Total Raised wrt the Valuation of Top Unicorns(Size of the
bubble: Years taken to become
Unicorn)",labels={"x":"Valuation($B)","y":"Total Raised($B)"})

fig.show()
```



This plot shows us the Total Raised in terms of the Valuation of the top Unicorns in the world, there's a correlation of 0.62 between both. The size of the dot is proportional to the number of years that were taken by the company to become a Unicorn.

- **Founded Years wrt the Number of Companies that became Unicorns**

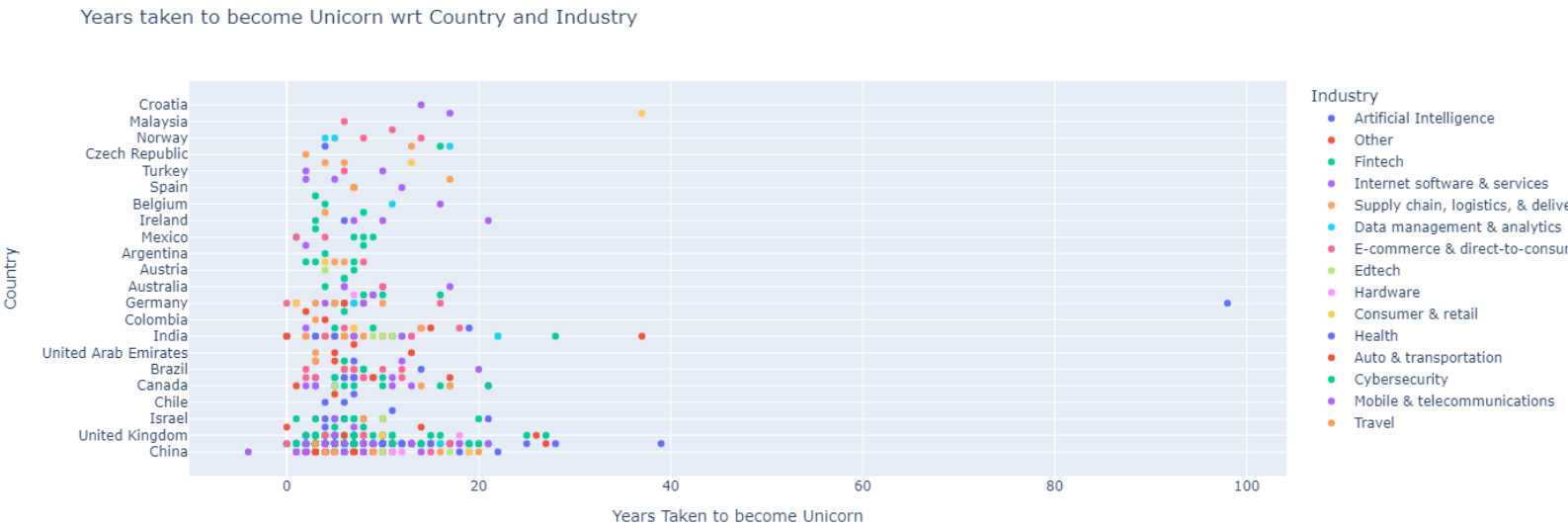
```
temp=df.groupby(by=['Founded Year']).agg({'Company':['count']})

temp=temp.sort_values([('Company','count')],
ascending=False)[:10]

fig=px.bar(y=temp[('Company','count')],x=temp.index,title="Founde
d Years wrt number of Companies that became
Unicorn",text_auto=True,labels={"x":"Founded Years","y":"Number
of Unicorns"})

fig.show()
```



From the graph we can see that most of the companies founded in the year 2015 became Unicorns in the future.

- **Years Taken to become Unicorn wrt Country and Industry**

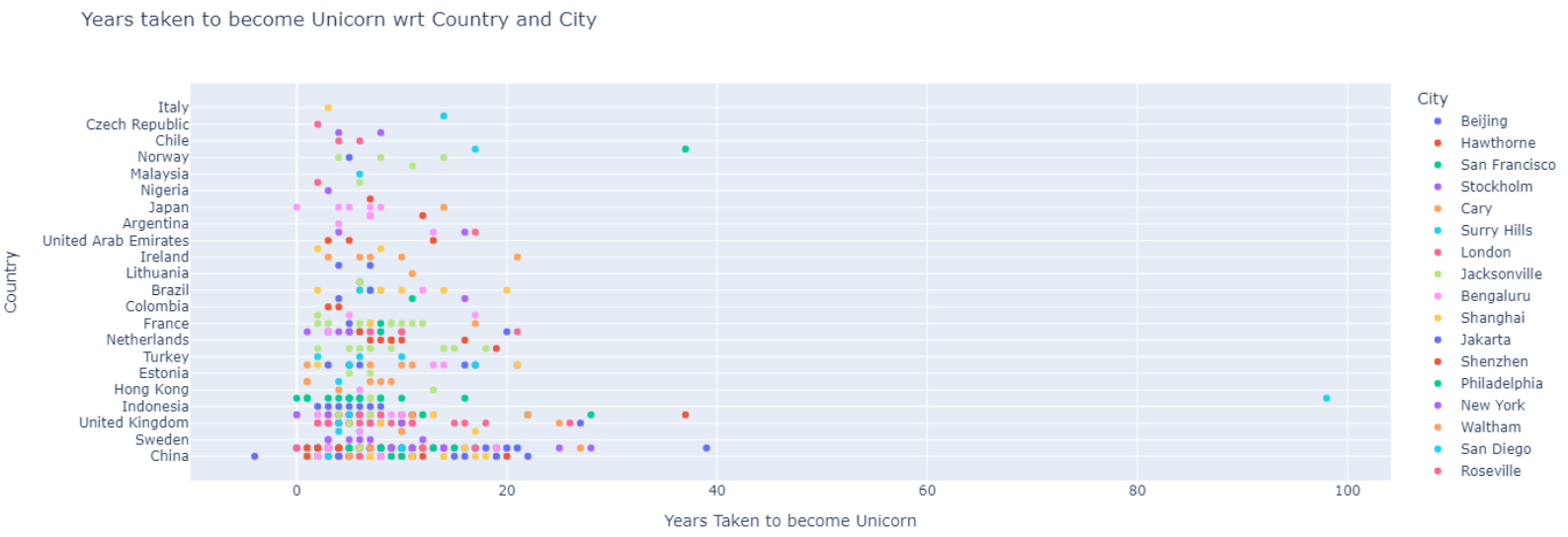
```
fig=px.scatter(df,x='Years Taken to become
Unicorn',y='Country',hover_data=['Industry'],color='Industry',tit
le="Years taken to become Unicorn wrt Country and Industry")

fig.show()
```



This plot depicts how many years are taken by companies from each industry in a country to become successful unicorns.

- **Years Taken to Become Unicorn wrt Country and City**

```
fig=px.scatter(df,x='Years Taken to become Unicorn',y='Country',hover_data=['City'],color='City',title="Years taken to become Unicorn wrt Country and City")

fig.show()
```



The above scatter plot shows the number of years required by companies to become unicorns with respect to country and city.