Spring 22

Unicorn Companies

CIS 5270-Business Intelligence; Prof. Shilpa Balan

Mounica Durga, CIN:401991171
Mayur Patil, CIN:401349504

Table of Contents

Introduction

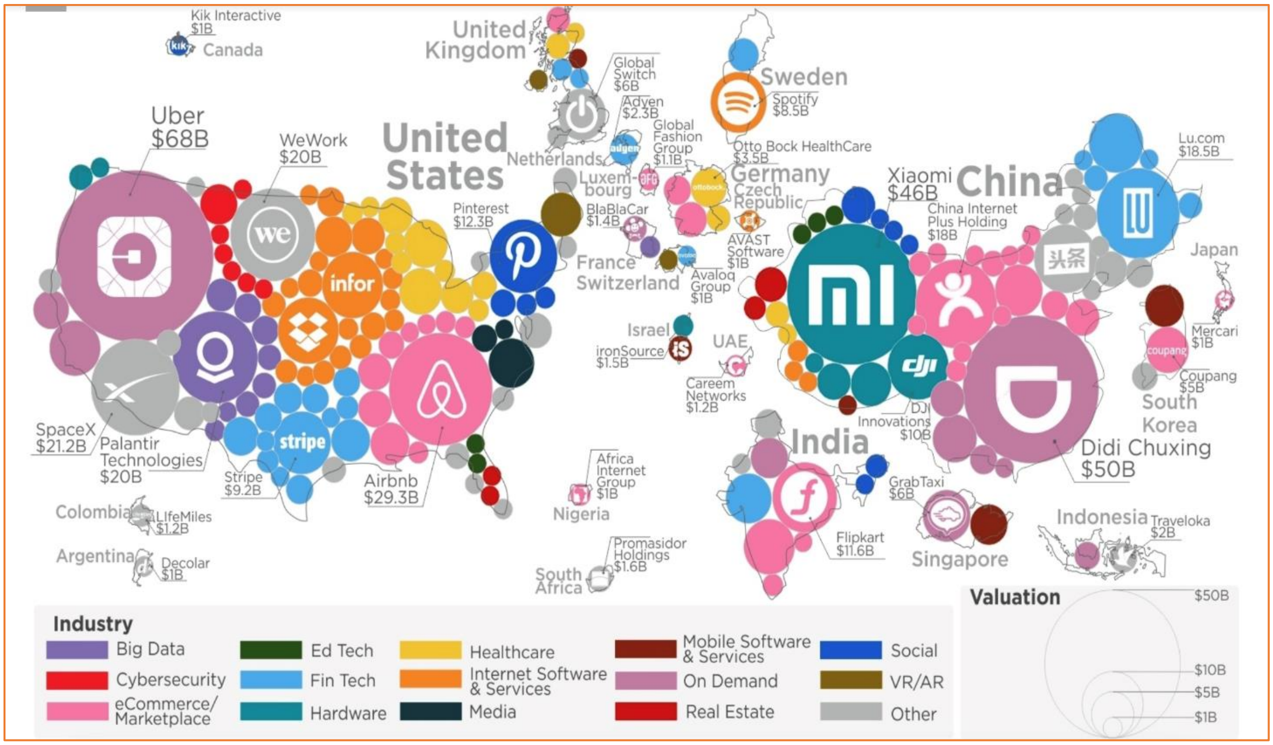
What is a Unicorn Company? A unicorn is a term used in the venture capital industry to describe a startup company with a market cap of over $1 billion. As of March 2022, there are more than 1,000 unicorn companies around the globe. Among the more notable unicorns located in the United States are home-sharing giant Airbnb, video game company Epic Games, as well as fintech company Robinhood. There are many factors that contribute to the success or failure of a startup. A few key factors to consider include:

A great ally of these organizations is social media. They use Facebook, Twitter, Instagram, etc. to communicate their message. Using segmentation, they are effectively amplifying their message and impacting on their target audience at a much lower cost than through television, for example.

The user experience is the most crucial factor. In the past, the product was all that mattered. These days, the buying experience is at least as important as the product itself.

The team is multidisciplinary and multicultural.[1] Consequently, they benefit from a remarkably diverse range of professional profiles, which is one of their greatest advantages in creating disruptive ideas. Furthermore, they are young companies that value talent and creativity.

The line between success and failure is extremely thin, and uncertainty is part of everyday life. Businesses with these characteristics are aware of that fact, so they develop a special resilience to handle tough times. For example, there was no Uber seven years ago. It was restricted to San Francisco a few years ago. Today, Uber operates in more than 65 countries. Through its expansion and warding off competition, the company has amassed an impressive war chest.[2] It has raised more than $8 billion from private investors. Upon reviewing the list of unicorns, the often-cited concentration of unicorns in certain geographical regions becomes apparent. More than three-quarters of the unicorns in existence today are in the United States and China.[3] While the number of unicorns founded in China has increased continuously since 2014, the number of unicorns founded in the United States remains stable.



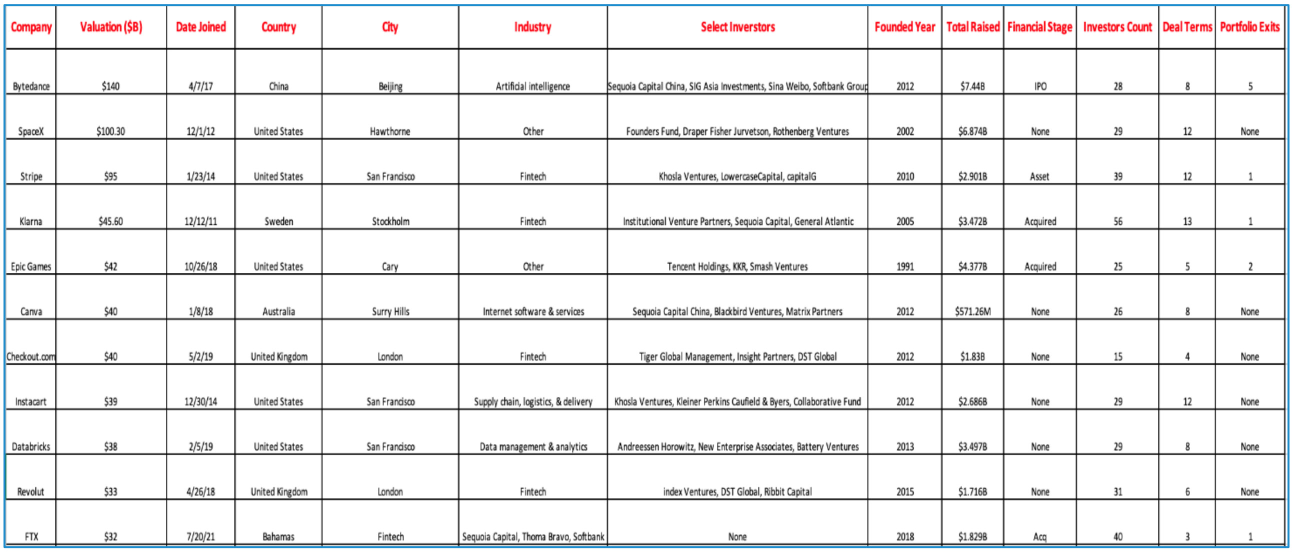
Project Goal: A dataset of unicorn companies has been found with a few key details such as valuation, investors, year of founding, country, and industry type. The purpose of our study is to analyze the influence a region, an investor, and a country might have on a company's ability to become a unicorn. Additionally, we wanted to check which countries possess most unicorns and their valuation over time.

Dataset Description

Column No.	Field Name	Description	Example Value
1	company	Name of the Unicorn Company	Space X
2	valuation_in_billions	Value of a company in billions of dollars	$140
3	date_joined	To date a company has been valued as a unicorn	4/7/17
4	country	Unicorn's country of origin	United States
5	city	Place in a country where the Unicorn exists	San Francisco
6	industry	Type of Unicorn industry	Supply Chain, Logistics, and delivery
7	select_investors	Any person/entity who commits capital in Unicorn with the expectation of receiving financial returns	Sequoia Capital China, SIG Asia Investments, Softbank Group
8	founded_year	In the year unicorn was founded	2012
9	total_raised	To date, amount raised by investors for a unicorn	$7.44B
10	financial_stage	In what stage of financial development is unicorn company now	IPO, Acquired
11	investors_count	Number of investors who have invested in Unicorn	28
12	deal_terms	Total number of deal terms agreed between Unicorns and investors	8
13	portfolio_exits	Number of investors who have exited Unicorns	1

Dataset URL: https://www.kaggle.com/code/al3xrods/unicorn-companies/data

Sample data



Company	Valuation ($B)	Date Joined	Country	City	Industry	Select Inverstors	Founded Year	Total Raised	Financial Stage	Investors Count	Deal Terms	Portfolio Exits
Bytedance	$140	4/7/17	China	Beijing	Artificial intelligence	Sequoia Capital China, SIG Asia Investments, Sina Weibo, Softbank Group	2012	$7.44B	IPO	28	8	5
SpaceX	$100.30	12/1/12	United States	Hawthorne	Other	Founders Fund, Draper Fisher Jurvetson, Rothenberg Ventures	2002	$6.874B	None	29	12	None
Stripe	$95	1/23/14	United States	San Francisco	Fintech	Khosla Ventures, LowercaseCapital, capitalG	2010	$2.901B	Asset	39	12	1
Klarna	$45.60	12/12/11	Sweden	Stockholm	Fintech	Institutional Venture Partners, Sequoia Capital, General Atlantic	2005	$3.472B	Acquired	56	13	1
Epic Games	$42	10/26/18	United States	Cary	Other	Tencent Holdings, KKR, Smash Ventures	1991	$4.377B	Acquired	25	5	2
Canva	$40	1/8/18	Australia	Surry Hills	Internet software & services	Sequoia Capital China, Blackbird Ventures, Matrix Partners	2012	$571.26M	None	26	8	None
Checkout.com	$40	5/2/19	United Kingdom	London	Fintech	Tiger Global Management, Insight Partners, DST Global	2012	$1.83B	None	15	4	None
Instacart	$39	12/30/14	United States	San Francisco	Supply chain, logistics, & delivery	Khosla Ventures, Kleiner Perkins Caufield & Byers, Collaborative Fund	2012	$2.686B	None	29	12	None
Databricks	$38	2/5/19	United States	San Francisco	Data management & analytics	Andreessen Horowitz, New Enterprise Associates, Battery Ventures	2013	$3.497B	None	29	8	None
Revolut	$33	4/26/18	United Kingdom	London	Fintech	index Ventures, DST Global, Ribbit Capital	2015	$1.716B	None	31	6	None
FTX	$32	7/20/21	Bahamas	Fintech	Sequoia Capital, Thoma Bravo, Softbank	None	2018	$1.829B	Acq	40	3	1

Data Cleaning

1. Changing column names to lowercase and replacing ‘ ‘ with ‘_’ for columns

Explanation

Problem Statement

In the raw data, all column names are not in one case. They also have spaces in between which makes it difficult to analyze specific column data in Python.

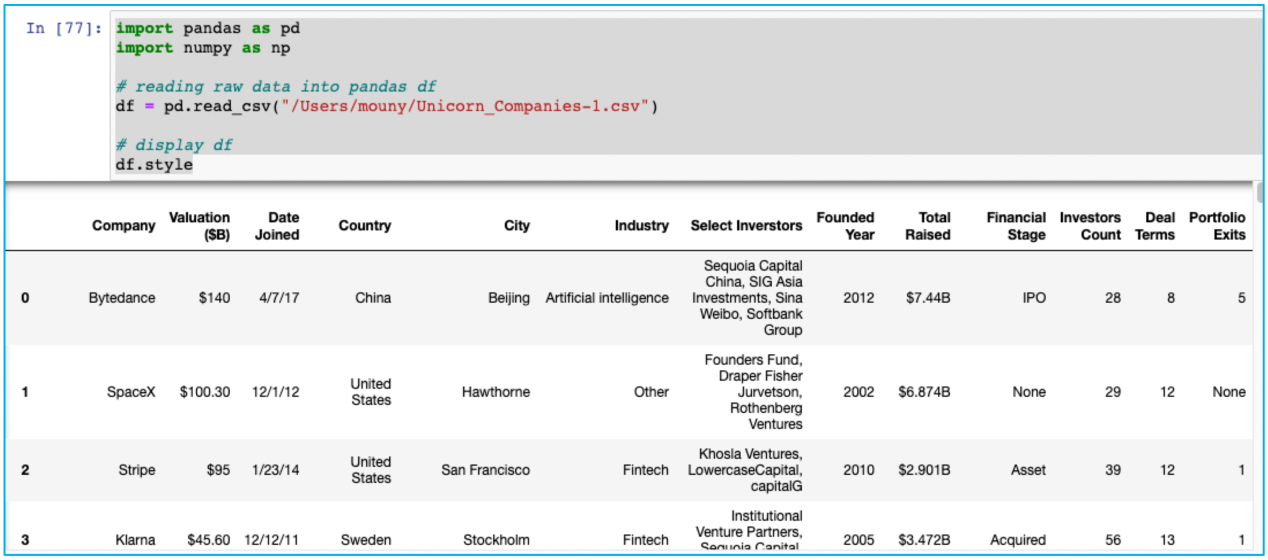
Steps Taken

1. Changed all the column names to single case- lower case
2. Replaced spaces (‘ ‘) with underscore(_)

Code

```
import pandas as pd

import numpy as np

 df = pd.read_csv("/Users/mouny/Unicorn_Companies-1.csv")

 df.style

df.columns = map(str.lower, df.columns)

 df.columns = [column.replace(" ", "_") for column in df.columns]

Df.style
```

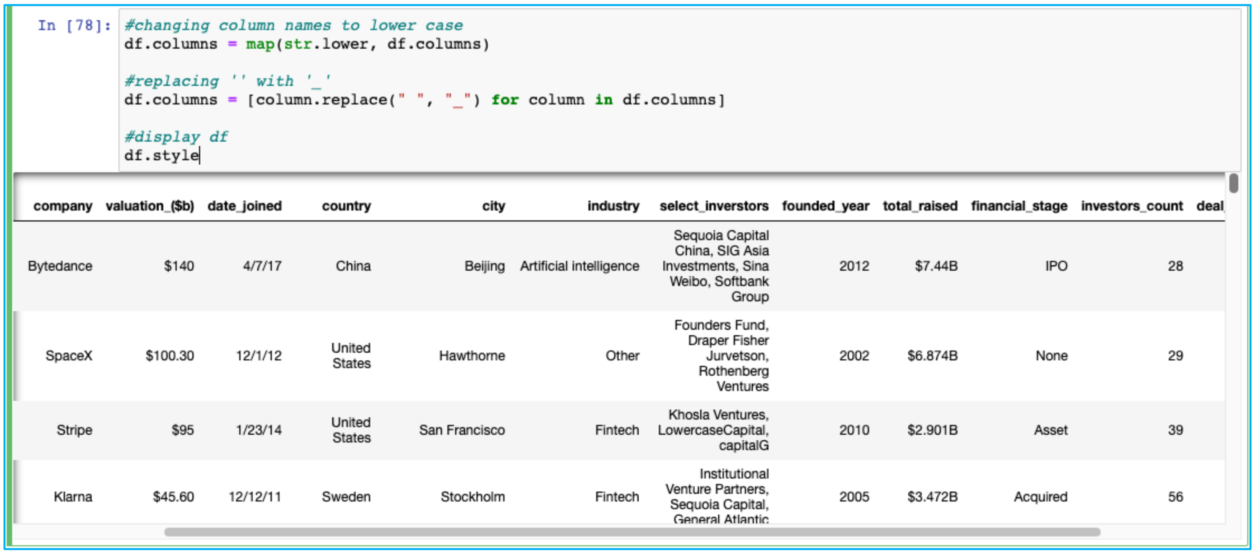
Before Cleaning



```
In [77]: import pandas as pd
         import numpy as np

         # reading raw data into pandas df
         df = pd.read_csv("/Users/mouny/Unicorn_Companies-1.csv")

         # display df
         df.style
```

	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	4/7/17	China	Beijing	Artificial intelligence	Sequoia Capital China, SIG Asia Investments, Sina Weibo, Softbank Group	2012	$7.44B	IPO	28	8	5
1	SpaceX	$100.30	12/1/12	United States	Hawthorne	Other	Founders Fund, Draper Fisher Jurvetson, Rothenberg Ventures	2002	$6.874B	None	29	12	None
2	Stripe	$95	1/23/14	United States	San Francisco	Fintech	Khosla Ventures, LowercaseCapital, capitalG	2010	$2.901B	Asset	39	12	1
3	Klarna	$45.60	12/12/11	Sweden	Stockholm	Fintech	Institutional Venture Partners, Sequoia Capital	2005	$3.472B	Acquired	56	13	1

After Cleaning



```
In [78]: #changing column names to lower case
         df.columns = map(str.lower, df.columns)

         #replacing '' with '_'
         df.columns = [column.replace(" ", "_") for column in df.columns]

         #display df
         df.style
```

company	valuation_($b)	date_joined	country	city	industry	select_inverstors	founded_year	total_raised	financial_stage	investors_count	deal
Bytedance	$140	4/7/17	China	Beijing	Artificial intelligence	Sequoia Capital China, SIG Asia Investments, Sina Weibo, Softbank Group	2012	$7.44B	IPO	28	
SpaceX	$100.30	12/1/12	United States	Hawthorne	Other	Founders Fund, Draper Fisher Jurvetson, Rothenberg Ventures	2002	$6.874B	None	29	
Stripe	$95	1/23/14	United States	San Francisco	Fintech	Khosla Ventures, LowercaseCapital, capitalG	2010	$2.901B	Asset	39	
Klarna	$45.60	12/12/11	Sweden	Stockholm	Fintech	Institutional Venture Partners, Sequoia Capital, General Atlantic	2005	$3.472B	Acquired	56	

2. Renaming column Name, replacing string value '$' with ' ', changing datatype to float

Explanation

Problem Statement

During analysis, the column valuation($B) provides information regarding the company's valuation in billions of dollars. However, this column contains special characters that make it difficult to read in python. Additionally, the data type of this column is not numeric.

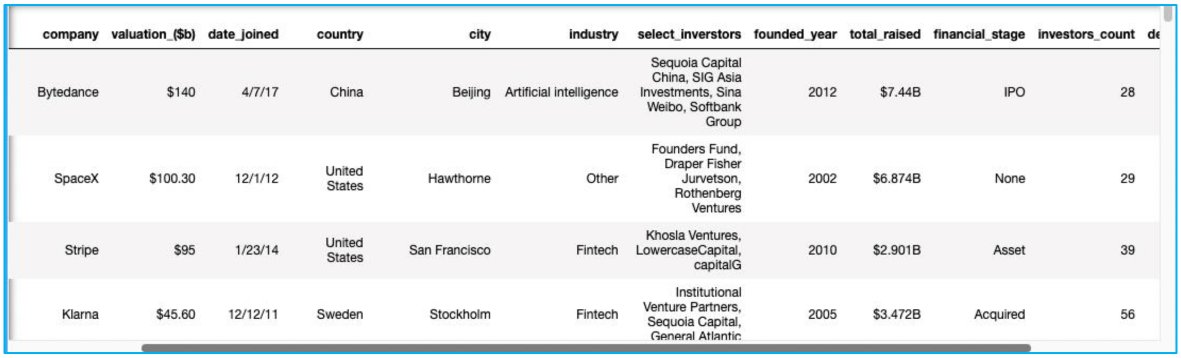
Steps Taken

1. Renamed the column name from Valuation ($B) to valuation_in_billions
2. Removed the special character '$'
3. Changed the data type to float

Code

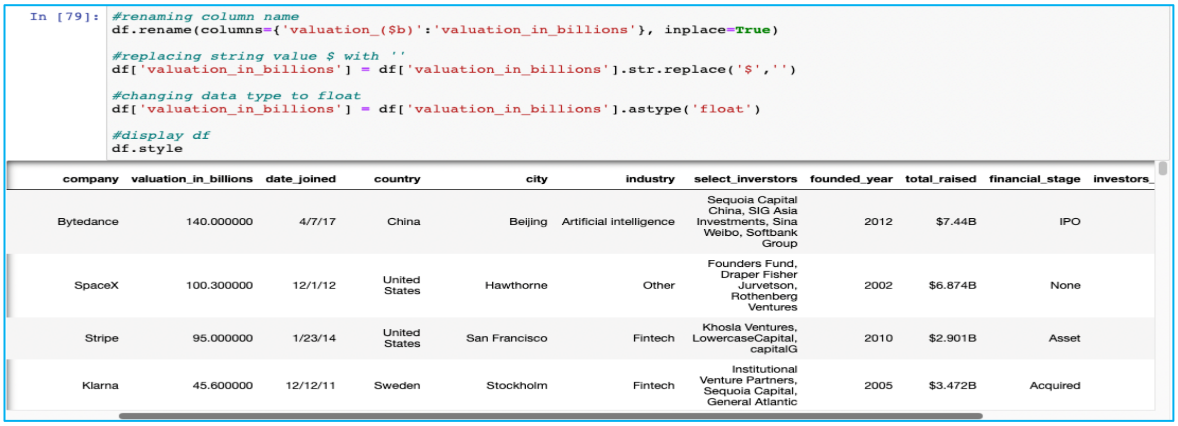
```
import pandas as pd

import numpy as np

df = pd.read_csv("/Users/mouny/Unicorn_Companies-1.csv")

df.columns = map(str.lower, df.columns)

df.columns = [column.replace(" ", "_") for column in df.columns]

df.rename(columns={'valuation_($b)':'valuation_in_billions'}, inplace=True)

df['valuation_in_billions'] = df['valuation_in_billions'].str.replace('$',")

df['valuation_in_billions'] = df['valuation_in_billions'].astype('float')

df.style
```

Before Cleaning



company	valuation_($b)	date_joined	country	city	industry	select_inverstors	founded_year	total_raised	financial_stage	investors_count	de
Bytedance	$140	4/7/17	China	Beijing	Artificial intelligence	Sequoia Capital China, SIG Asia Investments, Sina Weibo, Softbank Group	2012	$7.44B	IPO	28	
SpaceX	$100.30	12/1/12	United States	Hawthorne	Other	Founders Fund, Draper Fisher Jurvetson, Rothenberg Ventures	2002	$6.874B	None	29	
Stripe	$95	1/23/14	United States	San Francisco	Fintech	Khosla Ventures, LowercaseCapital, capitalG	2010	$2.901B	Asset	39	
Klarna	$45.60	12/12/11	Sweden	Stockholm	Fintech	Institutional Venture Partners, Sequoia Capital, General Atlantic	2005	$3.472B	Acquired	56	

After Cleaning



```
In [79]: #renaming column name
         df.rename(columns={'valuation_($b)':'valuation_in_billions'}, inplace=True)

         #replacing string value $ with ''
         df['valuation_in_billions'] = df['valuation_in_billions'].str.replace('$','')

         #changing data type to float
         df['valuation_in_billions'] = df['valuation_in_billions'].astype('float')

         #display df
         df.style
```

company	valuation_in_billions	date_joined	country	city	industry	select_inverstors	founded_year	total_raised	financial_stage	investors_
Bytedance	140.000000	4/7/17	China	Beijing	Artificial intelligence	Sequoia Capital China, SIG Asia Investments, Sina Weibo, Softbank Group	2012	$7.44B	IPO	
SpaceX	100.300000	12/1/12	United States	Hawthorne	Other	Founders Fund, Draper Fisher Jurvetson, Rothenberg Ventures	2002	$6.874B	None	
Stripe	95.000000	1/23/14	United States	San Francisco	Fintech	Khosla Ventures, LowercaseCapital, capitalG	2010	$2.901B	Asset	
Klarna	45.600000	12/12/11	Sweden	Stockholm	Fintech	Institutional Venture Partners, Sequoia Capital, General Atlantic	2005	$3.472B	Acquired	

3. Replacing None with Nan

Explanation

Problem Statement

We found columns containing values of None in the raw data. Due to the fact that it is a string called None, not a null value, having these None values would impact the analysis.

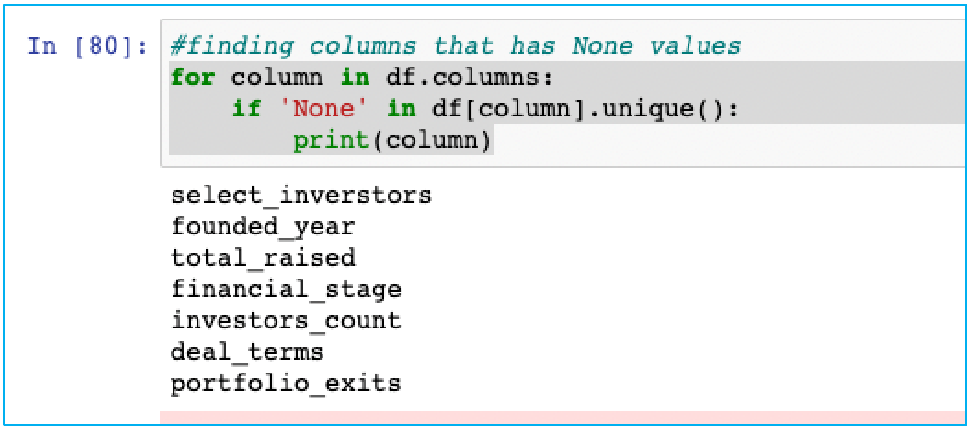
Steps Taken

1. Analyzed the columns which have None values in them
2. Replaced None values with numpy.nan

Code

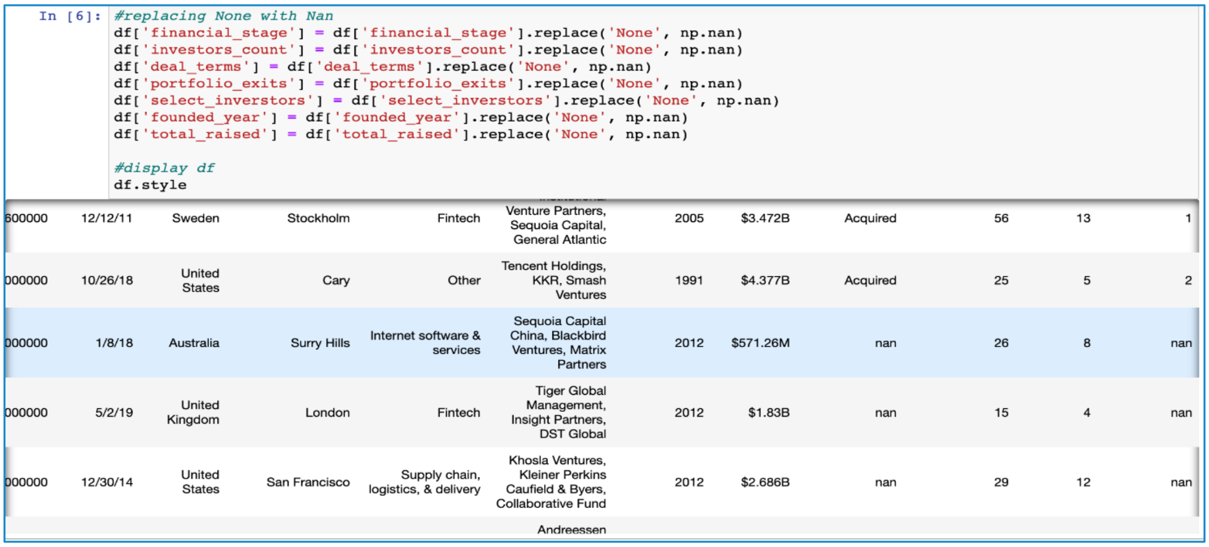
```
for column in df.columns:
 if 'None' in df[column].unique():
 print(column)
df['financial_stage'] = df['financial_stage'].replace('None', np.nan)
df['investors_count'] = df['investors_count'].replace('None', np.nan)
df['deal_terms'] = df['deal_terms'].replace('None', np.nan)
df['portfolio_exits'] = df['portfolio_exits'].replace('None', np.nan)
df['select_inverstors'] = df['select_inverstors'].replace('None', np.nan)
df['founded_year'] = df['founded_year'].replace('None', np.nan)
df['total_raised'] = df['total_raised'].replace('None', np.nan)
```

Before Cleaning



```
In [80]: #finding columns that has None values
         for column in df.columns:
             if 'None' in df[column].unique():
                 print(column)

         select_inverstors
         founded_year
         total_raised
         financial_stage
         investors_count
         deal_terms
         portfolio_exits
```

After Cleaning



```
In [6]: #replacing None with Nan
        df['financial_stage'] = df['financial_stage'].replace('None', np.nan)
        df['investors_count'] = df['investors_count'].replace('None', np.nan)
        df['deal_terms'] = df['deal_terms'].replace('None', np.nan)
        df['portfolio_exits'] = df['portfolio_exits'].replace('None', np.nan)
        df['select_inverstors'] = df['select_inverstors'].replace('None', np.nan)
        df['founded_year'] = df['founded_year'].replace('None', np.nan)
        df['total_raised'] = df['total_raised'].replace('None', np.nan)

        #display df
        df.style
```

600000	12/12/11	Sweden	Stockholm	Fintech	Venture Partners, Sequoia Capital, General Atlantic	2005	$3.472B	Acquired	56	13	1
000000	10/26/18	United States	Cary	Other	Tencent Holdings, KKR, Smash Ventures	1991	$4.377B	Acquired	25	5	2
000000	1/8/18	Australia	Surry Hills	Internet software & services	Sequoia Capital China, Blackbird Ventures, Matrix Partners	2012	$571.26M	nan	26	8	nan
000000	5/2/19	United Kingdom	London	Fintech	Tiger Global Management, Insight Partners, DST Global	2012	$1.83B	nan	15	4	nan
000000	12/30/14	United States	San Francisco	Supply chain, logistics, & delivery	Khosla Ventures, Kleiner Perkins Caufield & Byers, Collaborative Fund	2012	$2.686B	nan	29	12	nan
					Andreessen						

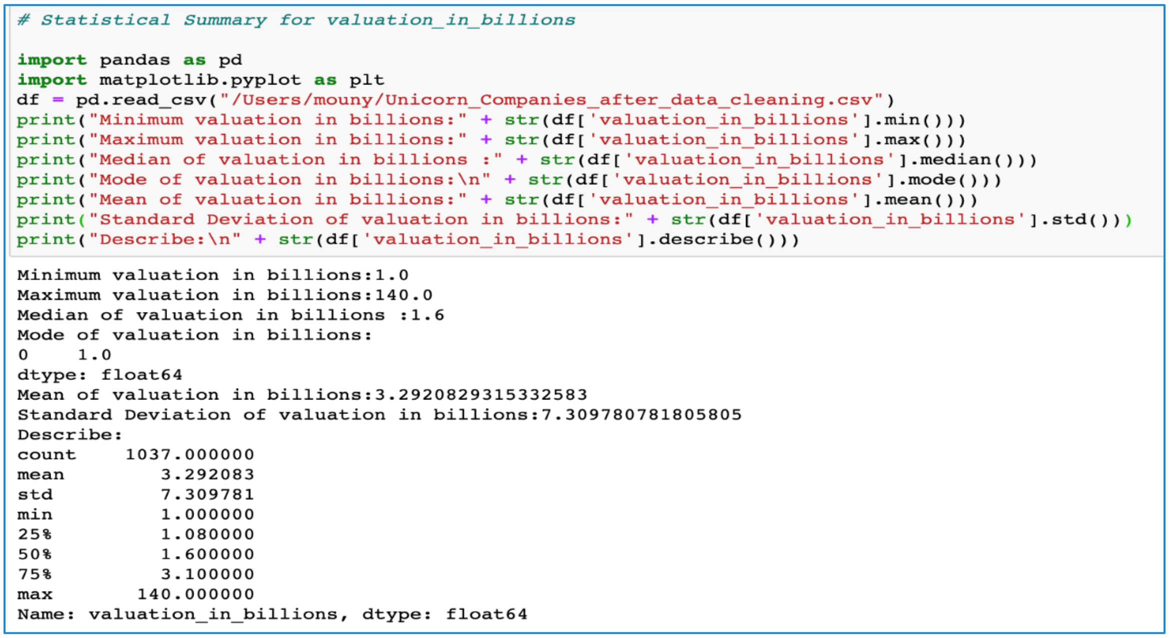
Statistical Functions

1. Summary Statistics for Valuation_in_billions

Code

```
# Statistical Summary for valuation_in_billions

import pandas as pd

import matplotlib.pyplot as plt

df = pd.read_csv("/Users/mouny/Unicorn_Companies_after_data_cleaning.csv")

print("Minimum valuation in billions:" + str(df['valuation_in_billions'].min()))

print("Maximum valuation in billions:" + str(df['valuation_in_billions'].max()))

print("Median of valuation in billions :" + str(df['valuation_in_billions'].median()))

print("Mode of valuation in billions:\n" + str(df['valuation_in_billions'].mode()))

print("Mean of valuation in billions:" + str(df['valuation_in_billions'].mean()))

print("Standard Deviation of valuation in billions:" + str(df['valuation_in_billions'].std()))

print("Describe:\n" + str(df['valuation_in_billions'].describe()))
```

Output



```
# Statistical Summary for valuation_in_billions

import pandas as pd
import matplotlib.pyplot as plt
df = pd.read_csv("/Users/mouny/Unicorn_Companies_after_data_cleaning.csv")
print("Minimum valuation in billions:" + str(df['valuation_in_billions'].min()))
print("Maximum valuation in billions:" + str(df['valuation_in_billions'].max()))
print("Median of valuation in billions :" + str(df['valuation_in_billions'].median()))
print("Mode of valuation in billions:\n" + str(df['valuation_in_billions'].mode()))
print("Mean of valuation in billions:" + str(df['valuation_in_billions'].mean()))
print("Standard Deviation of valuation in billions:" + str(df['valuation_in_billions'].std()))
print("Describe:\n" + str(df['valuation_in_billions'].describe()))
```

```
Minimum valuation in billions:1.0
Maximum valuation in billions:140.0
Median of valuation in billions :1.6
Mode of valuation in billions:
0    1.0
dtype: float64
Mean of valuation in billions:3.2920829315332583
Standard Deviation of valuation in billions:7.309780781805805
Describe:
count    1037.000000
mean        3.292083
std         7.309781
min         1.000000
25%         1.080000
50%         1.600000
75%         3.100000
max       140.000000
Name: valuation_in_billions, dtype: float64
```

Explanation

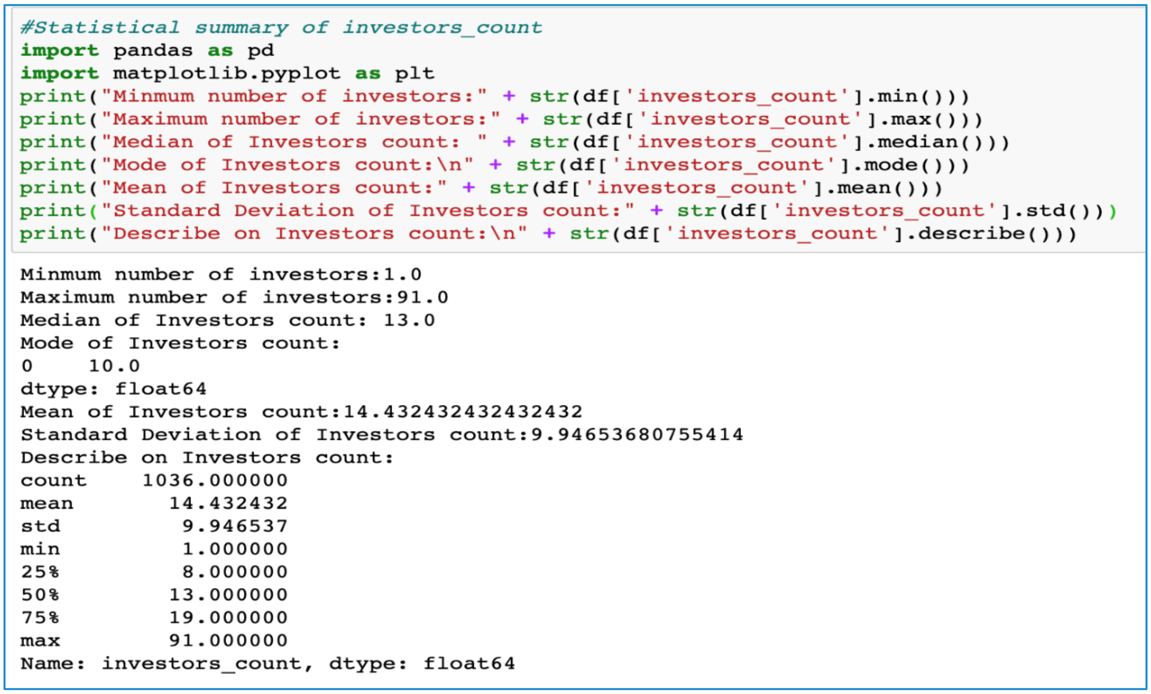
Based on summary statistics, the maximum valuation is $140B, which means unicorns are in a strong position. It is clear that the minimum is $1B, as we call a company a unicorn if its valuation is over $1B. The 25% percentile value is $3.1B. Furthermore, most companies have already crossed the minimum valuation, which is almost equal to the mean value of $3.2B. In addition, the standard deviation is $7.3B, twice the mean, which indicates that the valuations of unicorn companies are spread out between $1B and $140B, rather than clustered at the minimum valuation of $1B.

2. Summary Statistics for investors_count

Code

```
#Statistical summary of investors_count

import pandas as pd

import matplotlib.pyplot as plt

print("Minimum number of investors:" + str(df['investors_count'].min()))

print("Maximum number of investors:" + str(df['investors_count'].max()))

print("Median of Investors count: " + str(df['investors_count'].median()))

print("Mode of Investors count:\n" + str(df['investors_count'].mode()))

print("Mean of Investors count:" + str(df['investors_count'].mean()))

print("Standard Deviation of Investors count:" + str(df['investors_count'].std()))

print("Describe on Investors count:\n" + str(df['investors_count'].describe()))
```

Output



```
#Statistical summary of investors_count
import pandas as pd
import matplotlib.pyplot as plt
print("Minmum number of investors:" + str(df['investors_count'].min()))
print("Maximum number of investors:" + str(df['investors_count'].max()))
print("Median of Investors count: " + str(df['investors_count'].median()))
print("Mode of Investors count:\n" + str(df['investors_count'].mode()))
print("Mean of Investors count:" + str(df['investors_count'].mean()))
print("Standard Deviation of Investors count:" + str(df['investors_count'].std()))
print("Describe on Investors count:\n" + str(df['investors_count'].describe()))
```

```
Minmum number of investors:1.0
Maximum number of investors:91.0
Median of Investors count: 13.0
Mode of Investors count:
0    10.0
dtype: float64
Mean of Investors count:14.432432432432432
Standard Deviation of Investors count:9.94653680755414
Describe on Investors count:
count    1036.000000
mean       14.432432
std         9.946537
min         1.000000
25%         8.000000
50%        13.000000
75%        19.000000
max        91.000000
Name: investors_count, dtype: float64
```

Explanation

Based on summary statistics, the maximum number of investors is 91, while the minimum is 1. The combination of interest rates, the growth in technology, and recent regulatory perks have contributed to a rise in investment in unicorns in recent years. 75% value is 19, which indicates that most unicorns have more than ten investors. It is evident from the mean that most of the companies received more than 10 investors. The standard deviation is 9.

3. Summary Statistics for deal_terms

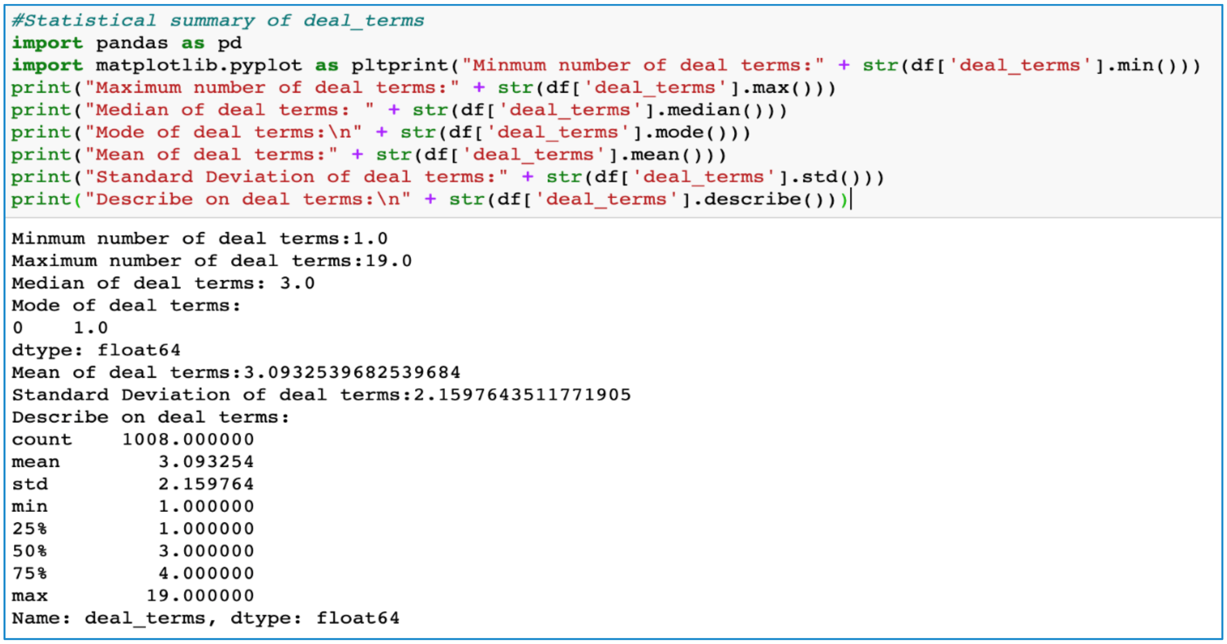
Code

#Statistical summary of deal_terms

import pandas as pd

```
import matplotlib.pyplot as pltprint("Minimum number of deal terms:" + str(df['deal_terms'].min()))

print("Maximum number of deal terms:" + str(df['deal_terms'].max()))

print("Median of deal terms: " + str(df['deal_terms'].median()))

print("Mode of deal terms:\n" + str(df['deal_terms'].mode()))

print("Mean of deal terms:" + str(df['deal_terms'].mean()))

print("Standard Deviation of deal terms:" + str(df['deal_terms'].std()))

print("Describe on deal terms:\n" + str(df['deal_terms'].describe()))
```

Output



```
#Statistical summary of deal_terms
import pandas as pd
import matplotlib.pyplot as pltprint("Minmum number of deal terms:" + str(df['deal_terms'].min()))
print("Maximum number of deal terms:" + str(df['deal_terms'].max()))
print("Median of deal terms: " + str(df['deal_terms'].median()))
print("Mode of deal terms:\n" + str(df['deal_terms'].mode()))
print("Mean of deal terms:" + str(df['deal_terms'].mean()))
print("Standard Deviation of deal terms:" + str(df['deal_terms'].std()))
print("Describe on deal terms:\n" + str(df['deal_terms'].describe()))
```

```
Minmum number of deal terms:1.0
Maximum number of deal terms:19.0
Median of deal terms: 3.0
Mode of deal terms:
0    1.0
dtype: float64
Mean of deal terms:3.0932539682539684
Standard Deviation of deal terms:2.1597643511771905
Describe on deal terms:
count    1008.000000
mean        3.093254
std         2.159764
min         1.000000
25%         1.000000
50%         3.000000
75%         4.000000
max        19.000000
Name: deal_terms, dtype: float64
```

Explanation

As summarized in the summary statistics, the maximum number of deal conditions is 19. Companies with more deal conditions must work extremely hard to obtain financing from investors. As seen in our summary statistics, 75% is 4 while the mean value is 3, which indicates that most investors invested in those unicorns who had below 4 deal terms and that most unicorns might have the same or similar deal terms. The standard deviation is 2.15, which is close to the mean of 3 and supports the statement above.

Analysis and Visualization

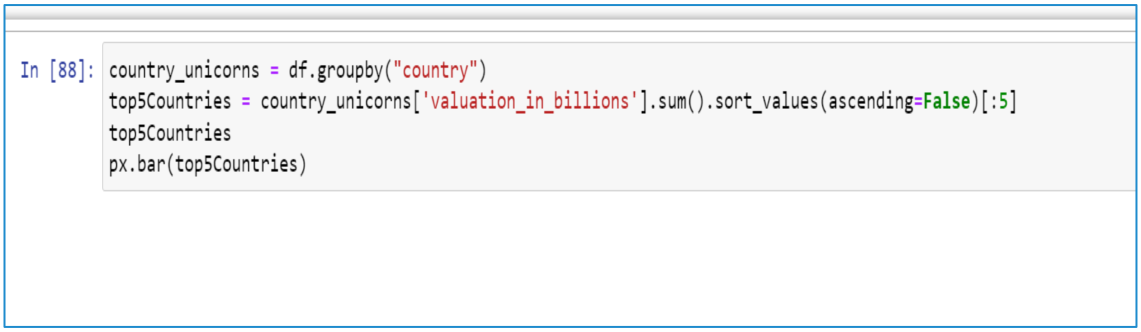
1. Which are the top countries in terms of the valuation of unicorn companies?

Explanation

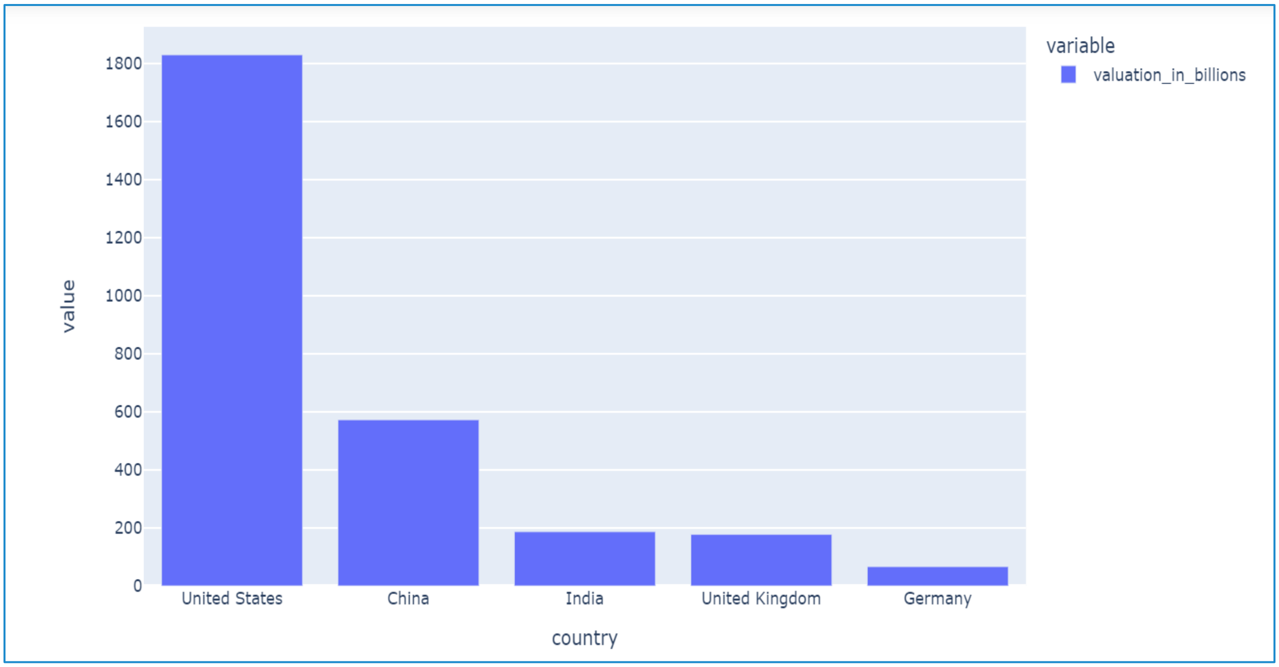
From the analysis, we found that the concentration of the number of unicorn companies and the valuation is in certain countries. We found the top 5 countries which are the United States, China, India, the United Kingdom, and Germany. Among these companies we can see from the visualization below that United States has more valuation than all other 4 countries combined. Here we have used the groupby() function. It allows you to split your data into separate groups to perform computations for better analysis. The number of unicorn companies founded in China in 2021 was bigger than those of 11 other countries, including Germany, India, and the UK, combined. The country with the most unicorns, or privately-owned companies with a valuation of one billion U.S. dollars or more, was however none of these countries, but the United States. SpaceX, Stripe, and Instacart were the biggest of these American companies in terms of market valuation.

Code

```
country_unicorns = df.groupby("country")

top5Countries =

country_unicorns['valuation_in_billions'].sum().sort_values(ascending=False)[:5]

top5Countries

px.bar(top5Countries)
```



```
In [88]: country_unicorns = df.groupby("country")
         top5Countries = country_unicorns['valuation_in_billions'].sum().sort_values(ascending=False)[:5]
         top5Countries
         px.bar(top5Countries)
```



2. Which are the top Industries that has the most valuation?

Explanation

We wanted to figure out from the data if there was any relation between the industry of the companies and the valuation. For that, we used the groupby() function to group the type of industry and calculated the valuation based on the industry. After plotting the visualization, we can see that the Fintech industry has the most valuation. The term “fintech” describes any business that uses technology to modify, enhance, or automate financial services for businesses or consumers. In recent years Fintech startups have been booming, that is why we can see the highest valuations in this industry. According to Market Data Forecast, the global FinTech market is projected to reach $324 billion by 2026, growing at a CAGR of 23.41%.

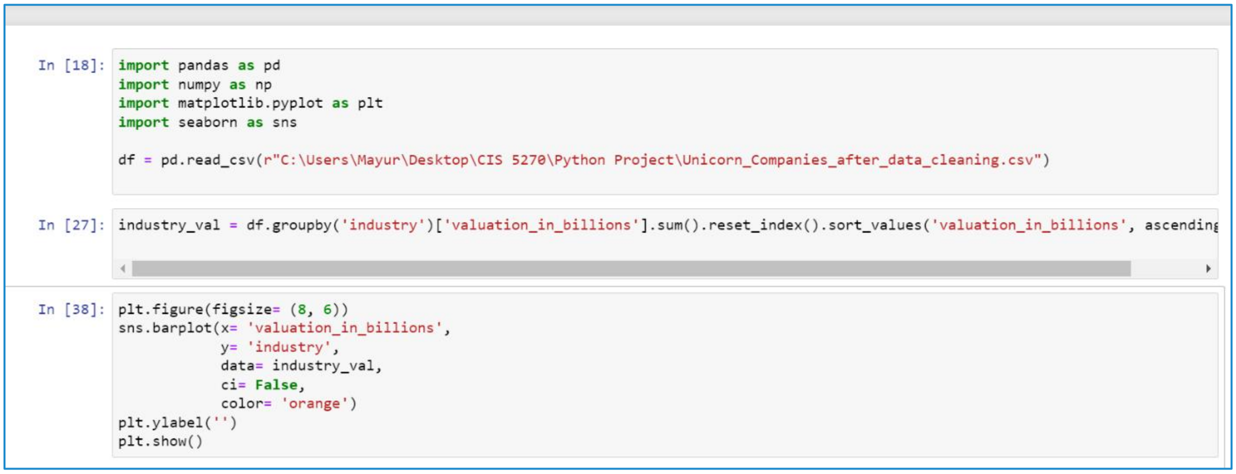
It seems that embracing fintech is the best route to traditional banking's survival. According to PwC, as of 2019, 48% of financial services organizations have embedded fintech fully into their strategic operating model and 37% have incorporated emerging technologies into the products and services they sell.

Code

```
industry_val = df.groupby('industry')['valuation_in_billions'].sum().reset_index().sort_values('valuation_in_billions', ascending= False)
plt.figure(figsize= (8, 6))
sns.barplot(x= 'valuation_in_billions', y= 'industry',
            data= industry_val,
            ci= False,
            color= 'orange')
```

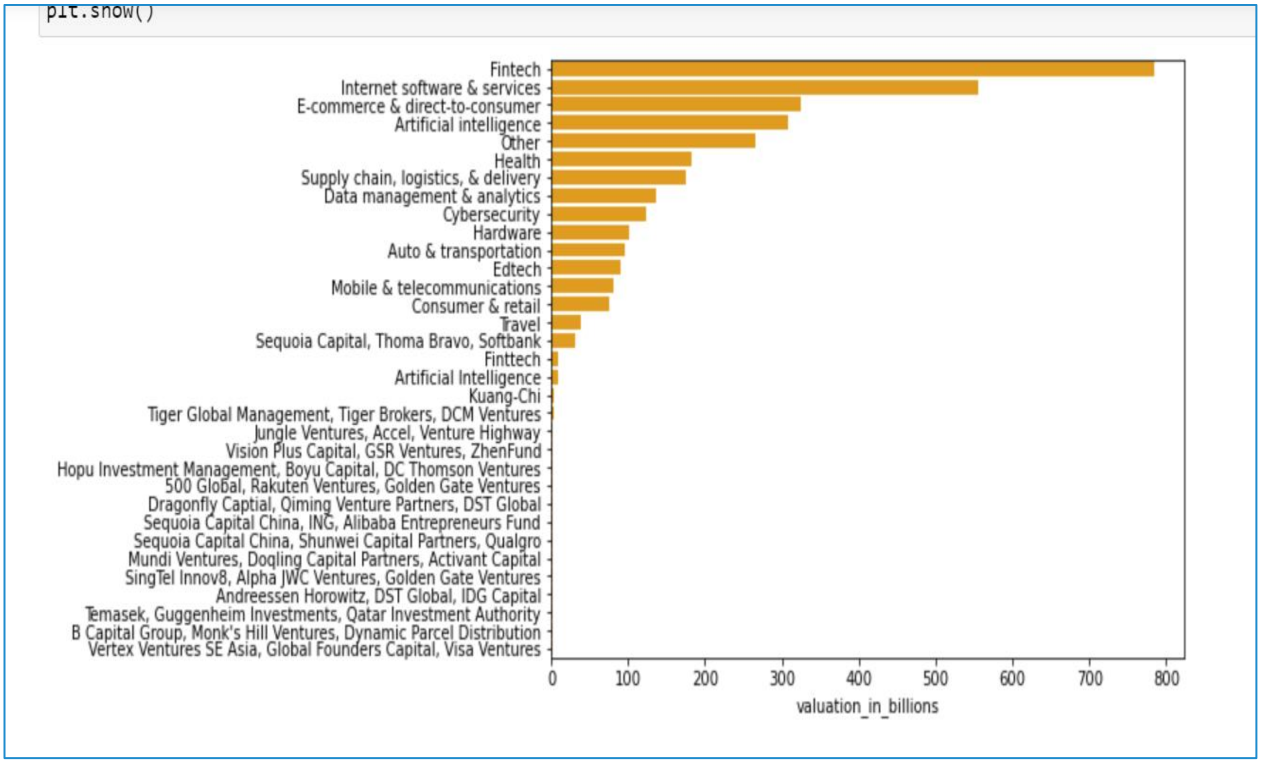
plt.ylabel('')

plt.show()



```
In [18]: import pandas as pd
         import numpy as np
         import matplotlib.pyplot as plt
         import seaborn as sns

         df = pd.read_csv(r"C:\Users\Mayur\Desktop\CIS 5270\Python Project\Unicorn_Companies_after_data_cleaning.csv")

In [27]: industry_val = df.groupby('industry')['valuation_in_billions'].sum().reset_index().sort_values('valuation_in_billions', ascending

In [38]: plt.figure(figsize= (8, 6))
         sns.barplot(x= 'valuation_in_billions',
                     y= 'industry',
                     data= industry_val,
                     ci= False,
                     color= 'orange')
         plt.ylabel('')
         plt.show()
```

Visualization 1

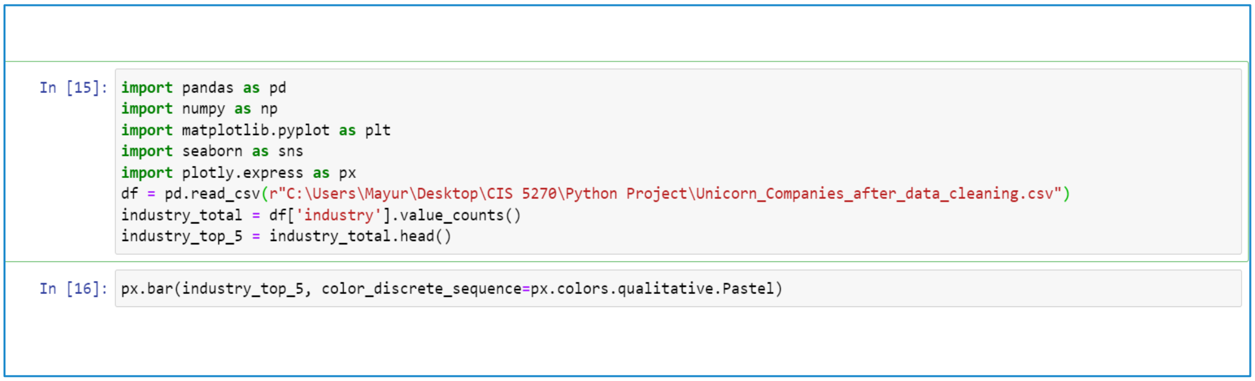


```
plt.show()
```

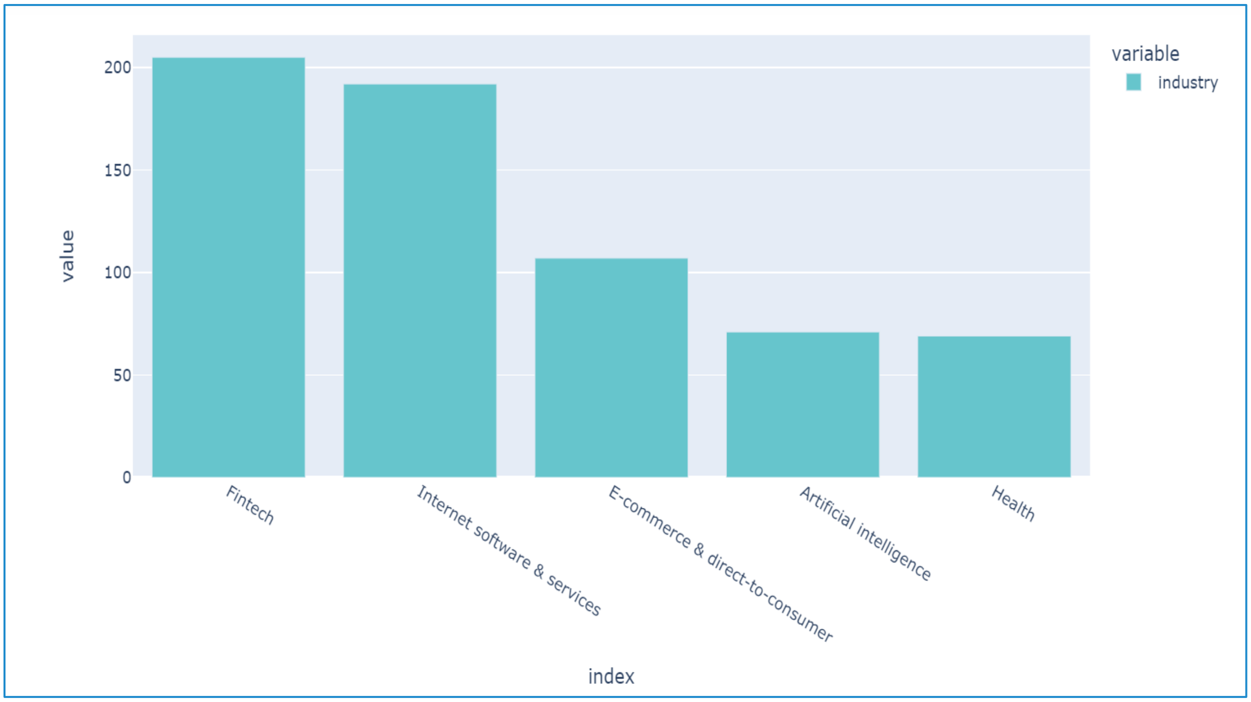
Visualization2 Top 5 industries

Code

```
import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

import seaborn as sns

import plotly.express as px

df = pd.read_csv(r"C:\Users\Mayur\Desktop\CIS 5270\Python Project\Unicorn_Companies_after_data_cleaning.csv")

industry_total = df['industry'].value_counts()

industry_top_5 = industry_total.head()

px.bar(industry_top_5, color_discrete_sequence=px.colors.qualitative.Pastel)
```



```
In [15]: import pandas as pd
         import numpy as np
         import matplotlib.pyplot as plt
         import seaborn as sns
         import plotly.express as px
         df = pd.read_csv(r"C:\Users\Mayur\Desktop\CIS 5270\Python Project\Unicorn_Companies_after_data_cleaning.csv")
         industry_total = df['industry'].value_counts()
         industry_top_5 = industry_total.head()

In [16]: px.bar(industry_top_5, color_discrete_sequence=px.colors.qualitative.Pastel)
```



3. What is the relation between the age of the company and its valuation?

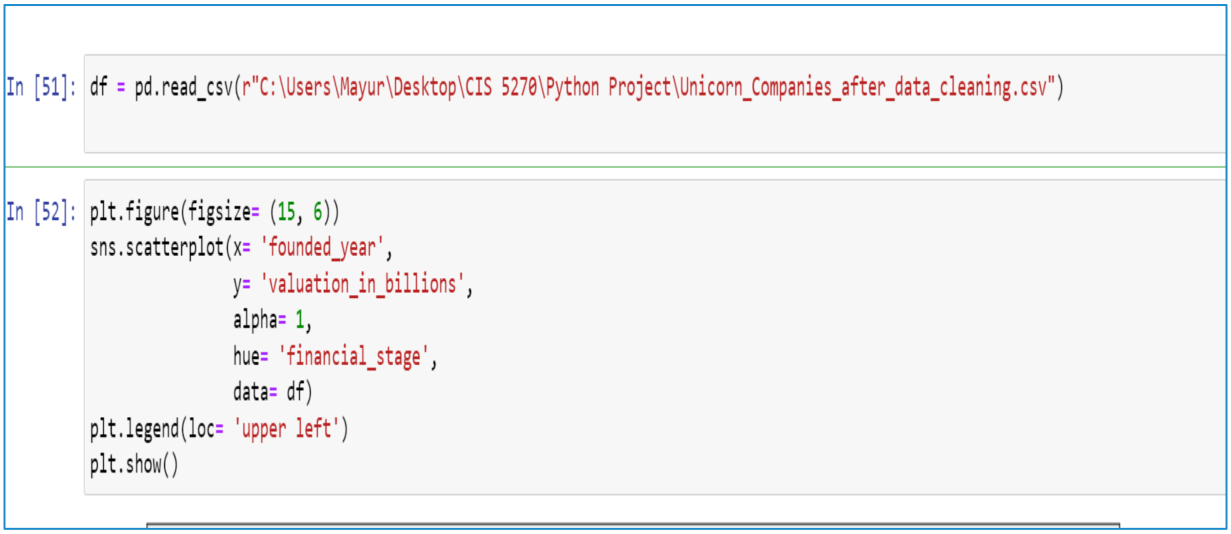
Explanation

In this interconnected world of the internet, most of these unicorn companies are related to computer software and IT. If we consider this dataset, we find that there are no such relations between the age of the company and its valuation. We can see a lot of companies founded in the 21st century have more valuations than the companies founded in the previous century.

Most of the companies in this dataset are based on computer software. The software industry expanded in the early 1960s, almost immediately after computers were first sold in mass-produced quantities. Universities, government, and business customers created a demand for software. In the early years of the 21st century, another successful business model has arisen for hosted software, called Software-as-a-service (SaaS). This rise in importance and the need for software made the value of the companies working in this domain much greater than the companies that existed before these companies in other industries.

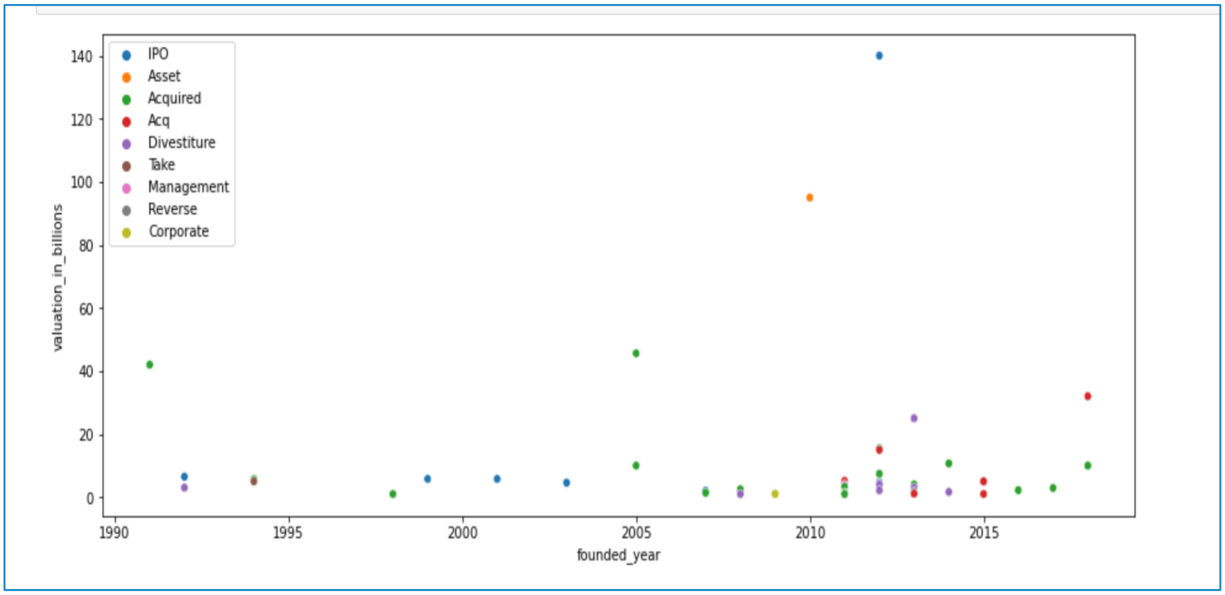
Code

```
plt.figure(figsize= (15, 6))
sns.scatterplot(x= 'founded_year',
            y= 'valuation_in_billions',
            alpha= 1,
            hue= 'financial_stage',
            data= df)
plt.legend(loc= 'upper left')
plt.show()
```



```
In [51]: df = pd.read_csv(r"C:\Users\Mayur\Desktop\CIS 5270\Python Project\Unicorn_Companies_after_data_cleaning.csv")
```

```
In [52]: plt.figure(figsize= (15, 6))
         sns.scatterplot(x= 'founded_year',
                         y= 'valuation_in_billions',
                         alpha= 1,
                         hue= 'financial_stage',
                         data= df)
         plt.legend(loc= 'upper left')
         plt.show()
```



4. How many countries have reached the unicorn status in a particular year by country?

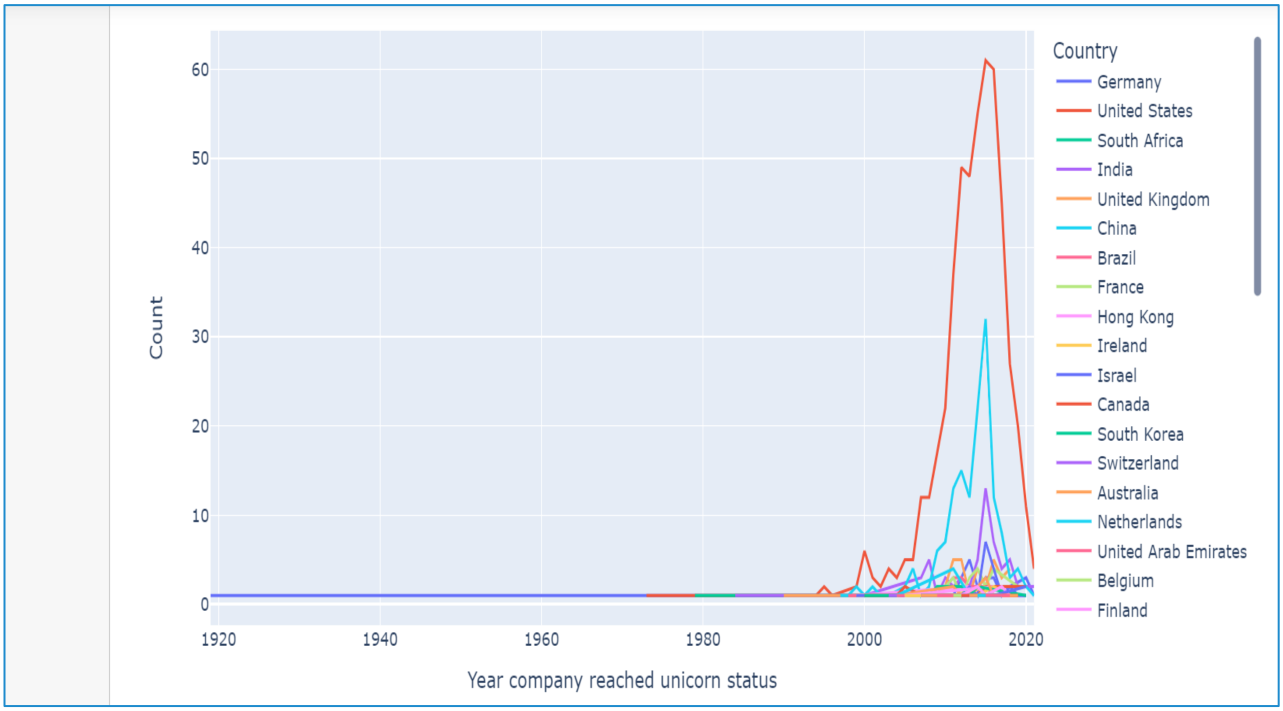
Explanation

This visualization shows us the number of companies in a particular country reaching unicorn status in a particular year. We have used the groupby() and count() functions. The maximum number of companies reached unicorn status in 2015 in all the top unicorn countries. In the United States alone, 61 companies reached the unicorn status in 2015. In China and India, 32 and 13 companies reached the unicorn status respectively. Israel had 7 companies reaching unicorn status in the year 2015. Most of the companies started reaching unicorn status after 2005. We can figure out that it must have been because of the widespread acceptance of the internet and huge funding by venture capitalists. But the biggest change was in the years from 2010 to 2021. Although the year 2015 was the peak for all countries to have the maximum number of companies reaching unicorn status, there has been a steep decline in the number of companies reaching unicorn status for every country. For example, the United States had only 20 companies reaching the unicorn status in the year 2019 and this is the same situation for almost all the countries.

Code

```
df_founded_year= df.groupby(['founded_year', 'country']).count()
fig = px.line(
df_founded_year,
x=df_founded_year.index.get_level_values(0),
y="company",
color=df_founded_year.index.get_level_values(1),
labels={
"company": "Count",
"x": "The Year company reached unicorn status",
"color": "Country"
}
)
fig.show()
```

```
df_founded_year= df.groupby(['founded_year', 'country']).count()
fig = px.line(
    df_founded_year,
    x=df_founded_year.index.get_level_values(0),
    y="company",
    color=df_founded_year.index.get_level_values(1),
    labels={
        "company": "Count",
        "x": "The Year company reached unicorn status",
        "color": "Country"
      }
)
fig.show()
```



References

[1] https://www.investopedia.com/terms/u/unicorn.asp

[2] https://www.iberdrola.com/innovation/unicorn-companies-technology-startups

[3] https://link.springer.com/article/10.1007/s40685-020-00120-2