

```
In [1]: # Loading the dataset and importing the pandas library
import pandas as pd
df = pd.read_csv('unicorntable PFM.csv')
print(df)
```

	No.	Company	Sector \
0	1	InMobi	Adtech - Mobile Ads
1	2	Flipkart^	E-Commerce
2	3	Mu Sigma	SaaS - Analytics
3	4	Snapdeal*	E-Commerce
4	5	PayTM^	Fintech - Payments & Wallet
..
96	97	PhysicsWallah	Edtech
97	98	Purpille	E-Commerce - Personal Care & Cosmetics
98	99	Leadsquared	SaaS - CRM
99	100	OneCard	Fintech - Credit Cards
100	101	Shiprocket	Aggregator - Logistics Services

	Entry	Valuation^^ (\$B)	Valuation (\$B)	Entry	Location \
0		1.00	1.00	Sep-11	Bangalore/Singapore
1		1.00	37.60	Feb-12	Bangalore/Singapore
2		1.00	1.50	Feb-13	Bangalore/Chicago
3		1.80	2.40	Oct-14	Delhi
4		1.70	16.00	Feb-15	Noida
..	
96		1.10	1.10	Jun-22	Noida
97		1.10	1.10	Jun-22	Mumbai
98		1.00	1.00	Jun-22	Bangalore
99		1.30	1.30	Jul-22	Pune
100		1.23	1.23	Aug-22	Delhi

	Select Investors
0	KPCB, Sherpalo Ventures, SoftBank
1	Accel, Tiger Global, Naspers, SoftBank, Tencent
2	Accel, Sequoia Capital, General Atlantic
3	Kalaari Capital, Nexus Ventures, Bessemer, Sof...
4	Saama Capital, Elevation Capital, Alibaba, Ber...
..	...
96	GSV Ventures, WestBridge
97	JSW Ventures, IvyCap Ventures, Blume Ventures,...
98	Stakeboat Capital, Gaja Capital, WestBridge
99	QED Investors, Matrix Partners India, Sequoia ...
100	Lightrock India, Info Edge, Tribe Capital, Tem...

[101 rows x 8 columns]

```
In [6]: # getting the information about the dataset
import pandas as pd
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 101 entries, 0 to 100
Data columns (total 8 columns):
#   Column                                Non-Null Count  Dtype
---  -
0   No.                                    101 non-null   int64
1   Company                              101 non-null   object
2   Sector                               101 non-null   object
3   Entry Valuation^^ ($B)               101 non-null   float64
4   Valuation ($B)                       101 non-null   float64
5   Entry                                101 non-null   object
6   Location                             101 non-null   object
7   Select Investors                     101 non-null   object
dtypes: float64(2), int64(1), object(5)
memory usage: 6.4+ KB
```

```
In [3]: # to showcase the statistical information of the dataset
df.describe()
```

```
Out[3]:
```

	No.	Entry Valuation^^ (\$B)	Valuation (\$B)
count	101.000000	101.000000	101.000000
mean	51.000000	1.547723	3.352158
std	29.300171	1.143688	4.656116
min	1.000000	1.000000	0.568000
25%	26.000000	1.000000	1.200000
50%	51.000000	1.200000	1.900000
75%	76.000000	1.600000	3.500000
max	101.000000	10.000000	37.600000

```
In [6]: # checking wheteher their is any null value
df.isnull().sum()
```

```
Out[6]: No.                                0
Company                                0
Sector                                0
Entry Valuation^^ ($B)                 0
Valuation ($B)                        0
Entry                                  0
Location                              0
Select Investors                       0
dtype: int64
```

```
In [8]: # to return the top 5 rows of the dataset
df.head(5)
```

Out[8]:

	No.	Company	Sector	Entry Valuation^^ (\$B)	Valuation (\$B)	Entry	Location	Select Investors
0	1	InMobi	Adtech - Mobile Ads	1.0	1.0	Sep- 11	Bangalore/Singapore	KPCB, Sherpalo Ventures, SoftBank
1	2	Flipkart^	E- Commerce	1.0	37.6	Feb- 12	Bangalore/Singapore	Accel, Tiger Global, Naspers, SoftBank, Tencent
2	3	Mu Sigma	SaaS - Analytics	1.0	1.5	Feb- 13	Bangalore/Chicago	Accel, Sequoia Capital, General Atlantic
3	4	Snapdeal*	E- Commerce	1.8	2.4	Oct- 14	Delhi	Kalaari Capital, Nexus Ventures, Bessemer, Sof...
4	5	PayTM^	Fintech - Payments & Wallet	1.7	16.0	Feb- 15	Noida	Saama Capital, Elevation Capital, Alibaba, Ber...

```
In [9]: df=df.drop(['No.'], axis=1)
df()
```

Out[9]:

	Company	Sector	Entry Valuation^^ (\$B)	Valuation (\$B)	Entry	Location	Select Investors
0	InMobi	Adtech - Mobile Ads	1.00	1.00	Sep- 11	Bangalore/Singapore	KPCB, Sherpalo Ventures, SoftBank
1	Flipkart^	E- Commerce	1.00	37.60	Feb- 12	Bangalore/Singapore	Accel, Tiger Global, Naspers, SoftBank, Tencent
2	Mu Sigma	SaaS - Analytics	1.00	1.50	Feb- 13	Bangalore/Chicago	Accel, Sequoia Capital, General Atlantic
3	Snapdeal*	E- Commerce	1.80	2.40	Oct- 14	Delhi	Kalaari Capital, Nexus Ventures, Bessemer, Sof...
4	PayTM^	Fintech - Payments & Wallet	1.70	16.00	Feb- 15	Noida	Saama Capital, Elevation Capital, Alibaba, Ber...
...
96	PhysicsWallah	Edtech	1.10	1.10	Jun- 22	Noida	GSV Ventures, WestBridge
97	Purplle	E- Commerce - Personal Care & Cosmetics	1.10	1.10	Jun- 22	Mumbai	JSW Ventures, IvyCap Ventures, Blume Ventures,...
98	Leadsquared	SaaS - CRM	1.00	1.00	Jun- 22	Bangalore	Stakeboat Capital, Gaja Capital, WestBridge
99	OneCard	Fintech - Credit Cards	1.30	1.30	Jul- 22	Pune	QED Investors, Matrix Partners India, Sequoia ...
100	Shiprocket	Aggregator - Logistics Services	1.23	1.23	Aug- 22	Delhi	Lightrock India, Info Edge, Tribe Capital, Tem...

101 rows × 7 columns

```
In [10]: df.nunique()
```

```
Out[10]: Company          101  
Sector              74  
Entry Valuation^^ ($B)  25  
Valuation ($B)       45  
Entry               49  
Location            27  
Select Investors     101  
dtype: int64
```

```
In [11]: # to check the correlation between entry valuation and current valuation  
df.corr()
```

```
Out[11]:
```

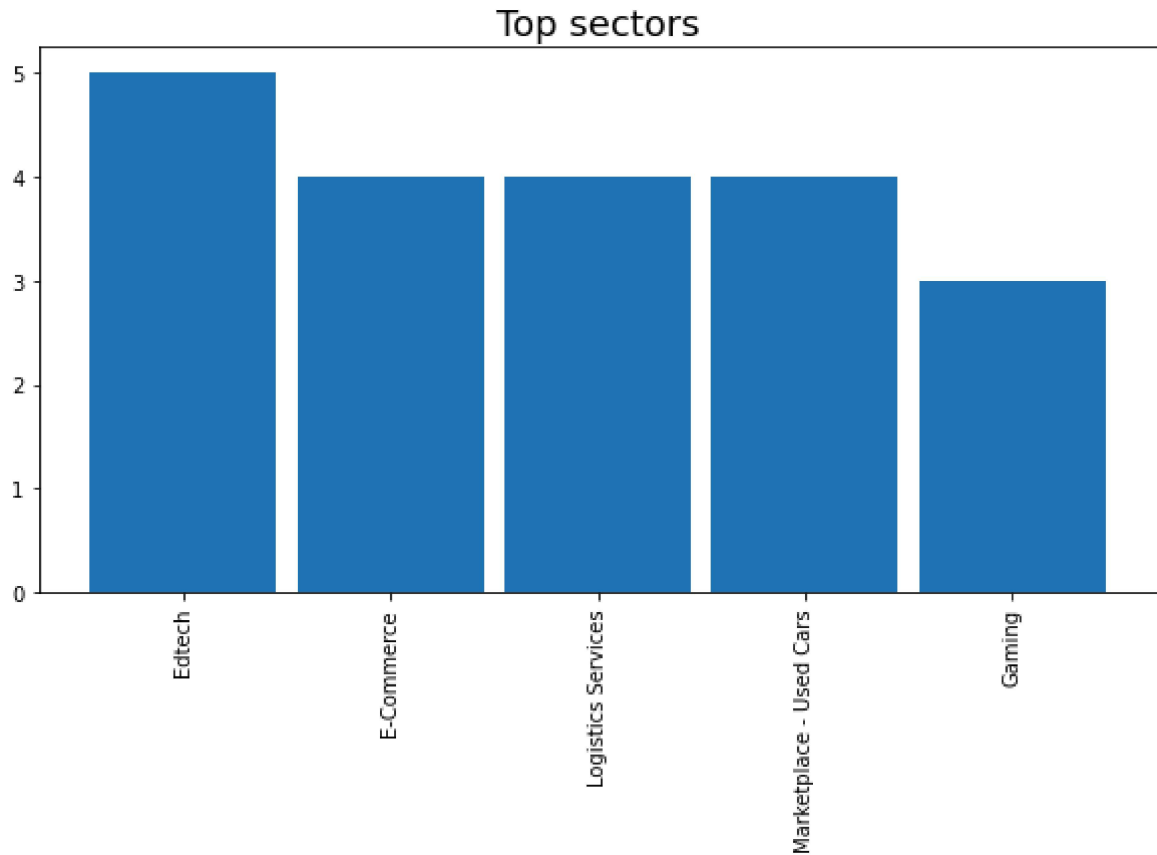
	Entry Valuation^^ (\$B)	Valuation (\$B)
Entry Valuation^^ (\$B)	1.000000	0.180287
Valuation (\$B)	0.180287	1.000000

```
In [12]: df["Sector"].unique()
```

```
Out[12]: array(['Adtech - Mobile Ads', 'E-Commerce', 'SaaS - Analytics',  
                'Fintech - Payments & Wallet', 'Mobility - Ride Aggregator',  
                'Marketplace - Classifieds', 'Foodtech',  
                'Social Media - Messaging', 'Edtech', 'Fintech - Insurance',  
                'SaaS - CRM', 'Proptech - Hotel Booking', 'B2B E-Commerce',  
                'Fintech - B2B Payments', 'Logistics Services',  
                'Logistics Services - Trucks', 'E-Commerce - Groceries', 'Gaming',  
                'SaaS - Data Management', 'SaaS - Contract Management',  
                'IT Services - Healthcare', 'Mobility - Electric',  
                'E-Commerce - Eyewear', 'Fintech - PoS Payment Solutions',  
                'E-Commerce - Personal Care & Cosmetics',  
                'SaaS - API Development & Testing', 'Fintech - Payment Gateway',  
                'Marketplace - Used Cars', 'Fintech - Payments',  
                'SaaS - Salon & Spa Management', 'Content - News',  
                'Content - Lockscreen', 'Fintech - General Insurance',  
                'SaaS - Healthcare Data Analytics',  
                'B2B E-Commerce - Construction Materials', 'NBFC',  
                'E-Commerce - Baby Care Products', 'E-Commerce - Social Commerce',  
                'Fintech - Payments & Credit Card Rewards',  
                'E-Commerce - Online Pharmacy',  
                'Fintech - Brokerage & Mutual Funds', 'Social Media',  
                'Conversational Messaging', 'SaaS - Subscription Billing Solution',  
                'Marketplace - Handyman Services',  
                'B2B E-Commerce - Industrial Equipment',  
                'Fintech - API - Banking Products', 'SaaS - Software Testing',  
                'NBFC - SME Loans', 'SaaS - HR - Training',  
                'Edtech - Higher Studies', 'Cryptocurrency Exchange',  
                'Edtech - Executive Education',  
                'Marketplace - Manufacturing Services', 'Marketplace - Jobs',  
                'D2C - Meat', 'Foodtech - Cloud Kitchen', 'D2C - Personal Care',  
                'Healthtech - Wellness', 'Aggregator - Consumer Brands',  
                'Proptech - Classifieds', 'Fintech - Brokerage',  
                'Fintech - Credit Cards', 'Healthtech - Elective Surgery Services',  
                'SaaS - HR', 'Web3 Infrastructure - Dapps',  
                'Interior Design - Modular Kitchens & Home Products',  
                'SaaS - Conversational Service Automation',  
                'SaaS - Programming Tools', 'Marketplace - Lending',  
                'SaaS - Local Ads Targeting',  
                'Fintech - Marketplace - SME Lending', 'Fintech - Neo Bank',  
                'Aggregator - Logistics Services'], dtype=object)
```

```
In [14]: # knowing the top sectors in the unicorn industries
from matplotlib import pyplot as plt
plt.rcParams["figure.figsize"] = (10, 5)
plt.title('Top sectors', fontsize=18)
df1=df["Sector"].value_counts()
df1.iloc[:5].plot(kind='bar',width = 0.9)
```

```
Out[14]: <AxesSubplot:title={'center':'Top sectors'}>
```




```
In [16]: # knowing that which loocations are best for the unicorns
```

```
df3=df["Location"].value_counts()
```

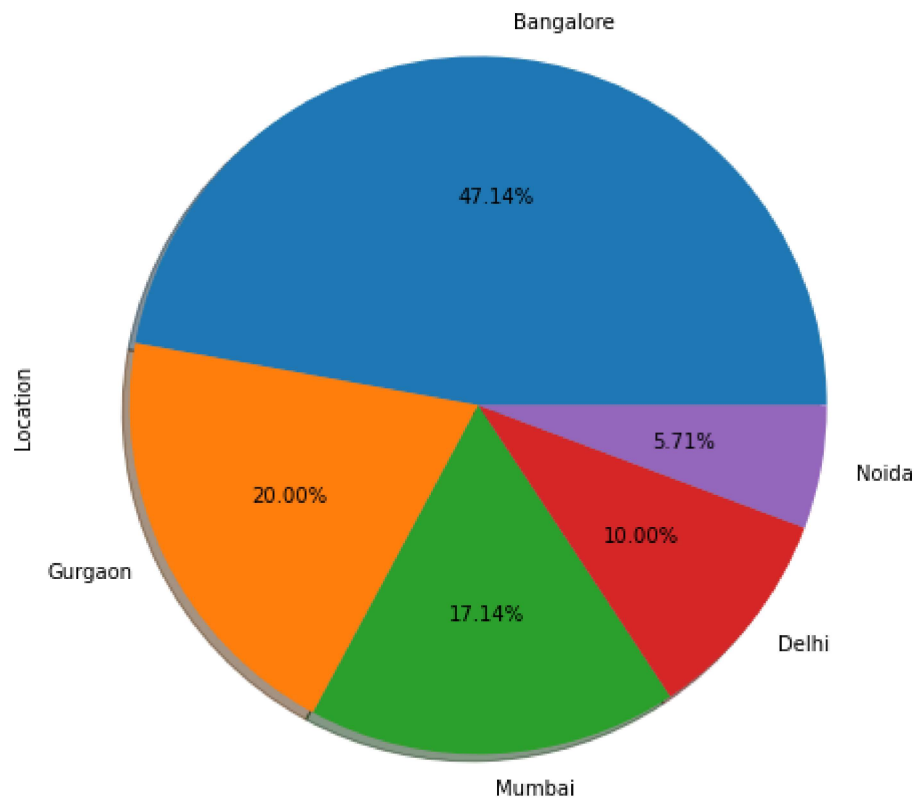
```
print(df3)
```

Bangalore	33
Gurgaon	14
Mumbai	12
Delhi	7
Noida	4
Pune	4
Bangalore/San Francisco	3
Bangalore/Singapore	3
Jaipur	2
Chennai	2
Bangalore/Chicago	1
Noida/San Francisco	1
Hyderabad/Bellevue	1
Mumbai/New York	1
Chennai/Palo Alto	1
Mumbai/Dublin	1
Pune/Sunnyvale	1
Mumbai/San Francisco	1
Pune/Bellevue	1
Thane	1
Mumbai/Princeton	1
Mumbai/Singapore	1
Hyderabad/Singapore	1
Chennai/San Mateo	1
Chennai/San Francisco	1
Karnataka	1
Noida/Singapore	1

Name: Location, dtype: int64

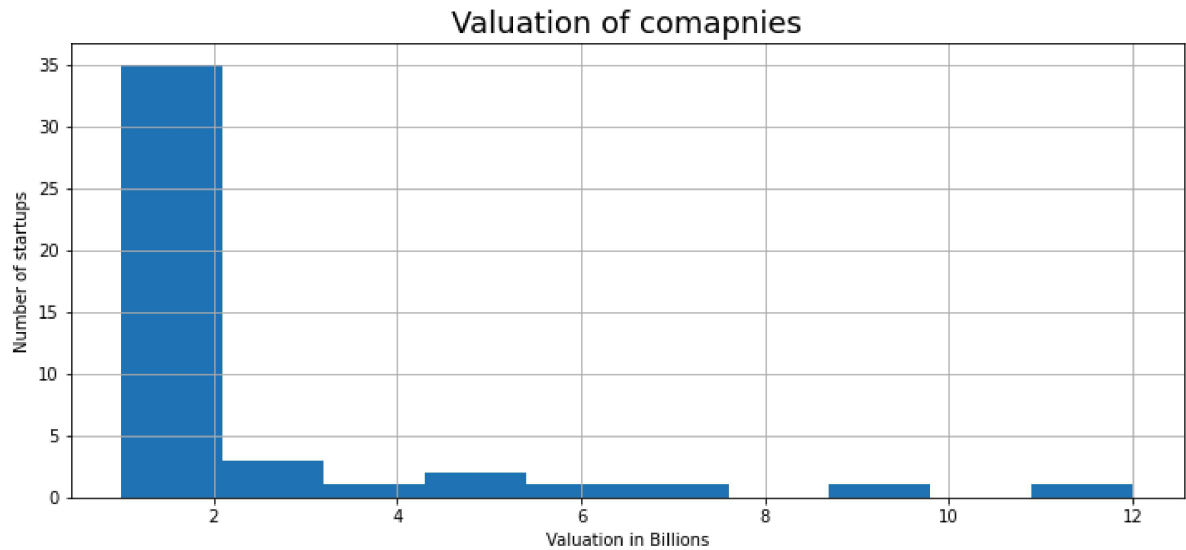
```
In [17]: # top 5 locations in percentage
plt.rcParams["figure.figsize"] = (8, 8)
df3.iloc[:5].plot(kind='pie', autopct='%0.2f%%', shadow=True)
```

```
Out[17]: <AxesSubplot:ylabel='Location'>
```



```
In [18]: # to know the number of unicorns as respect to their valuation
plt.rcParams["figure.figsize"] = (12, 5)
plt.title('Valuation of comapnies', fontsize=18)
df2=df["Valuation ($B)"].value_counts(normalize=False, sort=False, ascending=True)
plt.xlabel("Valuation in Billions")
plt.ylabel("Number of startups")
df2.hist()
```

```
Out[18]: <AxesSubplot:title={'center':'Valuation of comapnies'}, xlabel='Valuation in Billions', ylabel='Number of startups'>
```



```
In [3]: pip install plotly_express==0.4.0
```

```
Requirement already satisfied: plotly_express==0.4.0 in c:\users\vsara\anaconda3\lib\site-packages (0.4.0)
Requirement already satisfied: pandas>=0.20.0 in c:\users\vsara\anaconda3\lib\site-packages (from plotly_express==0.4.0) (1.2.4)
Requirement already satisfied: statsmodels>=0.9.0 in c:\users\vsara\anaconda3\lib\site-packages (from plotly_express==0.4.0) (0.12.2)
Requirement already satisfied: plotly>=4.0.0 in c:\users\vsara\anaconda3\lib\site-packages (from plotly_express==0.4.0) (5.18.0)
Requirement already satisfied: scipy>=0.18 in c:\users\vsara\anaconda3\lib\site-packages (from plotly_express==0.4.0) (1.6.2)
Requirement already satisfied: numpy>=1.11 in c:\users\vsara\anaconda3\lib\site-packages (from plotly_express==0.4.0) (1.20.1)
Requirement already satisfied: patsy>=0.5 in c:\users\vsara\anaconda3\lib\site-packages (from plotly_express==0.4.0) (0.5.1)
Requirement already satisfied: python-dateutil>=2.7.3 in c:\users\vsara\anaconda3\lib\site-packages (from pandas>=0.20.0->plotly_express==0.4.0) (2.8.1)
Requirement already satisfied: pytz>=2017.3 in c:\users\vsara\anaconda3\lib\site-packages (from pandas>=0.20.0->plotly_express==0.4.0) (2021.1)
Requirement already satisfied: six in c:\users\vsara\anaconda3\lib\site-packages (from patsy>=0.5->plotly_express==0.4.0) (1.15.0)
Requirement already satisfied: packaging in c:\users\vsara\anaconda3\lib\site-packages (from plotly>=4.0.0->plotly_express==0.4.0) (20.9)
Requirement already satisfied: tenacity>=6.2.0 in c:\users\vsara\anaconda3\lib\site-packages (from plotly>=4.0.0->plotly_express==0.4.0) (8.2.3)
Requirement already satisfied: pyparsing>=2.0.2 in c:\users\vsara\anaconda3\lib\site-packages (from packaging->plotly>=4.0.0->plotly_express==0.4.0) (2.4.7)
```

Note: you may need to restart the kernel to use updated packages.

```
In [2]: # checking the current valuation of the companies
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
import plotly.express as px
import plotly.graph_objects as go
from plotly.offline import init_notebook_mode, iplot
init_notebook_mode(connected=True)
df = pd.read_csv('unicorntable PFM.csv')
fig=px.scatter(df, x='Company',y='Valuation ($B)', hover_name='Company')
fig.show()
```

C:\Users\vsara\anaconda3\lib\site-packages\scipy__init__.py:138: UserWarning: A NumPy version >=1.16.5 and <1.23.0 is required for this version of SciPy (detected version 1.24.4)

warnings.warn(f"A NumPy version >={np_minversion} and <{np_maxversion} is required for this version of ")

```
In [3]: # knowing the companies entry date in respect with its current valuation
import plotly.express as px
fig1 = px.histogram(df, x="Valuation ($B)", y="Entry", color="Company", hover_
df = pd.read_csv('unicorntable PFM.csv')
fig1.show()
```

In []: