world-startups-2021

May 18, 2022

1 Top Startup Statistics 2021

1.1 Installing Required Libraries

1] Numpy
2] Pandas
3] MatplotLib
4] Seaborn
5] Jovian

Install all the Libraries Before Starting Your Project

[1]:
```
!pip install jovian --upgrade --quiet
```

[2]:
```
import jovian
```

[3]:
```
# Execute this to save new versions of the notebook
jovian.commit(project="world-startups-2021")
```

```
<IPython.core.display.Javascript object>
```

```
[jovian] Updating notebook "shrey2627/world-startups-2021" on
https://jovian.ai
[jovian] Committed successfully! https://jovian.ai/shrey2627/world-
startups-2021
```

[3]:
```
'https://jovian.ai/shrey2627/world-startups-2021'
```

[4]:
```
project_name = "World-Startups"
```

[5]:
```
import numpy as np
import pandas as pd
import matplotlib
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline
```

[6]:
```
import warnings
warnings.filterwarnings('ignore')
```

1.2 Getting Dataset

Here I am taking dataset from Kaggle
https://www.kaggle.com/datasets/khaiid/startups-by-valuation

[7]:
```
data_df = pd.read_csv("startups.csv")
```

1.3 Reading Dataset

Lets go through the entire dataset

[8]:
```
data_df
```

[8]:
```
           Company  Valuation Valuation_date                 Industry  \
0        ByteDance     140.00       April-21                 Internet
1           SpaceX     100.00     October-21                Aerospace
2           Stripe      95.00       March-21       Financial services
3           Klarna      45.60        June-21                  Fintech
4            Canva      40.00   September-21            Graphic design
..             …          …             …                        …
245         Bitso       2.20         May-21            Cryptocurrency
246       Gympass       2.20        June-21                      NaN
247         Kurly       2.20        July-21                E-commerce
248       Addepar       2.17        June-21                   Fintech
249  Eightfold.ai       2.10        June-21  Artificial intelligence

                        Country
0                         China
1                 United States
2      United States /  Ireland
3                        Sweden
4                     Australia
..                          …
245                      Mexico
246    Brazil /   United States
247                 South Korea
248               United States
249               United States

[250 rows x 5 columns]
```

[9]:
```
data_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 250 entries, 0 to 249
Data columns (total 5 columns):
 #   Column          Non-Null Count  Dtype
---  ------          --------------  -----
```

```
 0   Company         250 non-null    object
 1   Valuation       250 non-null    float64
 2   Valuation_date  250 non-null    object
 3   Industry        233 non-null    object
 4   Country         250 non-null    object
dtypes: float64(1), object(4)
memory usage: 9.9+ KB
```

[10]:
```
data_df.columns
```

[10]:
```
Index(['Company', 'Valuation', 'Valuation_date', 'Industry', 'Country'],
dtype='object')
```

[11]:
```
data_df.shape
```

[11]:
```
(250, 5)
```

2 Data Cleaning and Transforming

Here we will perform some cleaning methods to remove Null values and change data types of colu

Lets Clean Industry Column and fill the missing values.

[12]:
```
data_df['Industry'].unique()
```

[12]:
```
array(['Internet', 'Aerospace', 'Financial services', 'Fintech',
       'Graphic design', 'Retail', 'Software', 'Video games',
       'Cryptocurrency', 'Education technology', 'E-commerce',
       'Technology', 'Financial Technology', 'Finance', 'Health', nan,
       'Artificial intelligence', 'Batteries', 'Self-driving cars',
       'Marketplace', 'Venture capital', 'Internet media',
       'Computer software', 'Software as a service', 'Hospitality',
       'Marketing', 'Software development', 'Robotics',
       'Application security', 'Cloud security', 'Fantasy sports',
       'Logistics', 'Virtual events', 'NFT', 'Computer security',
       'Transportation', 'Travel', 'Robotic process automation',
       'Healthcare', 'Real estate technology', 'Workforce management',
       'Real estate', 'Cybersecurity', 'Collaborative software',
       'Mass media', 'Data infrastructure', 'Health technology',
       'Food delivery', 'Data governance', 'Ghost kitchen',
       'Electronic Cigarettes', 'Flexible electronics',
       'Defense Technology', 'Augmented Reality', 'Identity management',
       'Human resource management', 'Consumer electronics', 'Sales',
       'Blockchain', 'Data analytics', 'Software industry', 'Messaging',
       'Interior design', 'Insurance', 'Real Estate',
       'Wearable technology', 'Data storage', 'Computer storage',
       'Mobile gaming', 'Electric vehicles', 'Education Technology',
```

```
       'Educational technology', 'Quantum computing', 'B2B e-commerce',
       'Education', 'Content management system', 'Semiconductors',
       'Cloud communications', 'Home improvement', 'Trucking', 'CRM',
       'Personal finance'], dtype=object)
```

[13]:
```
data_df[data_df['Industry'].isna()]
```

[13]:

	Company	Valuation	Valuation_date	Industry	Country
29	GoodLeap	12.00	October-21	NaN	United States
97	Thrasio	5.00	October-21	NaN	United States
109	Checkr	4.60	August-21	NaN	United States
134	Patreon	4.00	April-21	NaN	United States
143	VAST Data	3.70	May-21	NaN	United States
145	Shouqi	3.55	December-16	NaN	China
170	SentinelOne	3.10	November-20	NaN	United States
195	VANCL	3.00	February-14	NaN	China
198	Wefox	3.00	June-21	NaN	Germany
199	Yixia Technology	3.00	November-16	NaN	China
220	AmWINS Group	2.60	October-16	NaN	United States
223	Sourcegraph	2.60	July-21	NaN	United States
226	Infra.Market	2.50	August-21	NaN	India
235	Wemakeprice	2.33	September-15	NaN	South Korea
238	Uptake	2.30	November-17	NaN	United States
242	Zume	2.25	November-18	NaN	United States
246	Gympass	2.20	June-21	NaN	Brazil / United States

[14]:
```
data_df['Industry'].fillna('Unknown',inplace = True)
```

In Valuation_date column, There is no use of having date so lets split the column into two.

[15]:
```
data_df[['Valuation_month','Valuation_day']] = data_df['Valuation_date'].str.
↪split('-', expand=True)
```

[16]:
```
data_df['Valuation_date'].astype('string')
```

[16]:
```
0          April-21
1        October-21
2          March-21
3           June-21
4      September-21
           …
245          May-21
246         June-21
247         July-21
248         June-21
249         June-21
Name: Valuation_date, Length: 250, dtype: string
```

[17]:
```
data_df
```

[17]:

	Company	Valuation	Valuation_date	Industry	Country	Valuation_month	Valuation_day
0	ByteDance	140.00	April-21	Internet	China	April	21
1	SpaceX	100.00	October-21	Aerospace	United States	October	21
2	Stripe	95.00	March-21	Financial services	United States / Ireland	March	21
3	Klarna	45.60	June-21	Fintech	Sweden	June	21
4	Canva	40.00	September-21	Graphic design	Australia	September	21
..	…	…	…	…	…	…	…
245	Bitso	2.20	May-21	Cryptocurrency	Mexico	May	21
246	Gympass	2.20	June-21	Unknown	Brazil / United States	June	21
247	Kurly	2.20	July-21	E-commerce	South Korea	July	21
248	Addepar	2.17	June-21	Fintech	United States	June	21
249	Eightfold.ai	2.10	June-21	Artificial intelligence	United States	June	21

```
[250 rows x 7 columns]
```

Now lets drop unwanted columns such as Valuation_date and Valuation_day

[18]:
```
data_df.drop(['Valuation_date','Valuation_day'], axis=1, inplace = True)
```

[19]:
```
data_df
```

[19]:
```
           Company  Valuation                 Industry  \
0        ByteDance     140.00                 Internet
1           SpaceX     100.00                Aerospace
2           Stripe      95.00       Financial services
3           Klarna      45.60                  Fintech
4            Canva      40.00           Graphic design
..             …          …                      …
245         Bitso       2.20           Cryptocurrency
246       Gympass       2.20                  Unknown
247         Kurly       2.20               E-commerce
248       Addepar       2.17                  Fintech
249  Eightfold.ai       2.10  Artificial intelligence

                        Country Valuation_month
0                         China           April
1                 United States         October
2      United States /  Ireland           March
3                        Sweden            June
4                     Australia       September
..                          …               …
245                     Mexico             May
246    Brazil /   United States            June
247                 South Korea            July
248               United States            June
249               United States            June

[250 rows x 5 columns]
```

[20]:
```
data_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 250 entries, 0 to 249
Data columns (total 5 columns):
 #   Column           Non-Null Count  Dtype
---  ------           --------------  -----
 0   Company          250 non-null    object
 1   Valuation        250 non-null    float64
 2   Industry         250 non-null    object
 3   Country          250 non-null    object
```

```
 4   Valuation_month  250 non-null    object
dtypes: float64(1), object(4)
memory usage: 9.9+ KB
```

[21]:
```
data_df['Valuation_month'].unique()
```

[21]:
```
array(['April', 'October', 'March', 'June', 'September', 'August', 'July',
       'January', 'November', 'May', 'February', 'December'], dtype=object)
```

[22]:
```
data_df['Country'] = data_df['Country'].astype('string')
```

[23]:
```
data_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 250 entries, 0 to 249
Data columns (total 5 columns):
 #   Column           Non-Null Count  Dtype
---  ------           --------------  -----
 0   Company          250 non-null    object
 1   Valuation        250 non-null    float64
 2   Industry         250 non-null    object
 3   Country          250 non-null    string
 4   Valuation_month  250 non-null    object
dtypes: float64(1), object(3), string(1)
memory usage: 9.9+ KB
```

[24]:
```
data_df = data_df.
↪set_index(['Company','Valuation','Industry','Valuation_month']).apply(lambda␣
↪x:x.str.split('/').explode()).reset_index()
```

[25]:
```
data_df
```

[25]:

	Company	Valuation	Industry	Valuation_month	Country
0	ByteDance	140.00	Internet	April	China
1	SpaceX	100.00	Aerospace	October	
2	Stripe	95.00	Financial services	March	
3	Stripe	95.00	Financial services	March	
4	Klarna	45.60	Fintech	June	
..	…	…	…	…	
264	Gympass	2.20	Unknown	June	
265	Gympass	2.20	Unknown	June	
266	Kurly	2.20	E-commerce	July	
267	Addepar	2.17	Fintech	June	
268	Eightfold.ai	2.10	Artificial intelligence	June	

```
1       United States
2      United States
3             Ireland
4              Sweden
..                ...
264           Brazil
265     United States
266       South Korea
267     United States
268     United States

[269 rows x 5 columns]
```

[26]:
```
data_df.Country = data_df.Country.apply(lambda x:x.replace(' ',''))
```

[38]:
```
data_df.Country = data_df.Country.str.lstrip()
```

[39]:
```
data_df
```

[39]:
```
           Company  Valuation                 Industry Valuation_month  \
0       ByteDance     140.00                 Internet           April
1          SpaceX     100.00                Aerospace         October
2          Stripe      95.00       Financial services           March
3          Stripe      95.00       Financial services           March
4          Klarna      45.60                  Fintech            June
..            ...        ...                      ...             ...
264       Gympass       2.20                  Unknown            June
265       Gympass       2.20                  Unknown            June
266         Kurly       2.20               E-commerce            July
267       Addepar       2.17                  Fintech            June
268  Eightfold.ai       2.10  Artificial intelligence            June

           Country
0            China
1     UnitedStates
2    UnitedStates
3           Ireland
4            Sweden
..              ...
264         Brazil
265    UnitedStates
266      SouthKorea
267    UnitedStates
268    UnitedStates

[269 rows x 5 columns]
```

All other column looks good, So we will move over to the Visualization part of Data analysis.

3 Data Visualization

Now Lets get some visuals from our Cleaned Dataset Using Python libraries like Seaborn and Mat

[28]:
```
data_df
```

[28]:
```
           Company  Valuation                 Industry Valuation_month  \
0        ByteDance     140.00                 Internet           April
1           SpaceX     100.00               Aerospace         October
2           Stripe      95.00      Financial services           March
3           Stripe      95.00      Financial services           March
4           Klarna      45.60                  Fintech            June
..             …          …                        …              …
264        Gympass       2.20                 Unknown            June
265        Gympass       2.20                 Unknown            June
266          Kurly       2.20              E-commerce            July
267        Addepar       2.17                 Fintech            June
268   Eightfold.ai       2.10  Artificial intelligence            June

              Country
0               China
1       UnitedStates
2      UnitedStates
3             Ireland
4              Sweden
..                ...
264           Brazil
265      UnitedStates
266        SouthKorea
267      UnitedStates
268      UnitedStates

[269 rows x 5 columns]
```

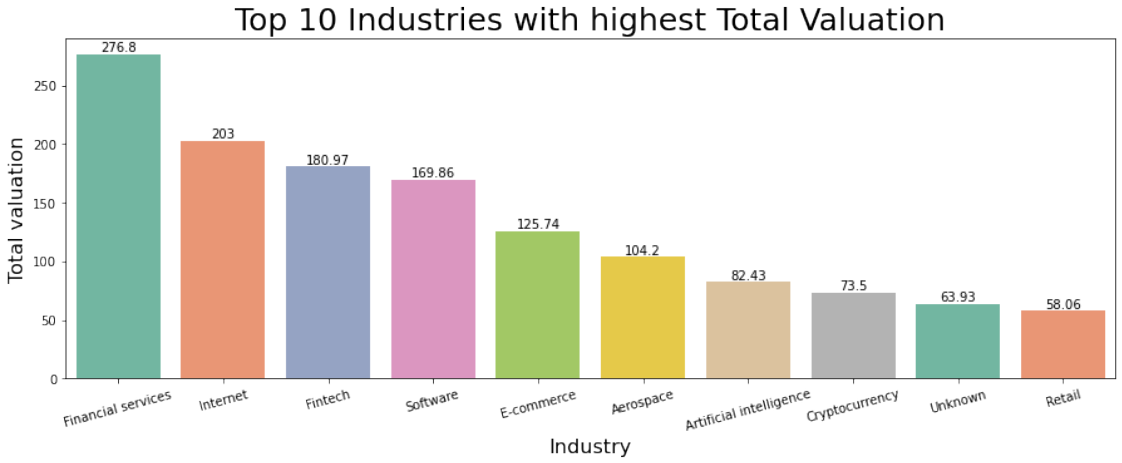
[29]:
```
top_industry = data_df.groupby('Industry').sum().sort_values(by =␣
↪'Valuation',ascending = False).head(10)
```

[30]:
```
top_industry
```

[30]:
```
                         Valuation
Industry
Financial services          276.80
Internet                    203.00
Fintech                     180.97
Software                    169.86
```

```
E-commerce                  125.74
Aerospace                   104.20
Artificial intelligence      82.43
Cryptocurrency               73.50
Unknown                      63.93
Retail                       58.06
```

[31]:
```
plt.figure(figsize=(15,5))
fig = sns.barplot(x = top_industry.index, y = top_industry.Valuation, data =␣
↪top_industry, palette ='Set2')
fig.set_xlabel('Industry',fontsize=16)
fig.set_ylabel('Total valuation',fontsize=16)
fig.set_xticklabels(fig.get_xticklabels(),rotation = 15,fontsize = 10)
fig.set_title('Top 10 Industries with highest Total Valuation',fontsize = 25)
fig.bar_label(fig.containers[0])
plt.show()
```



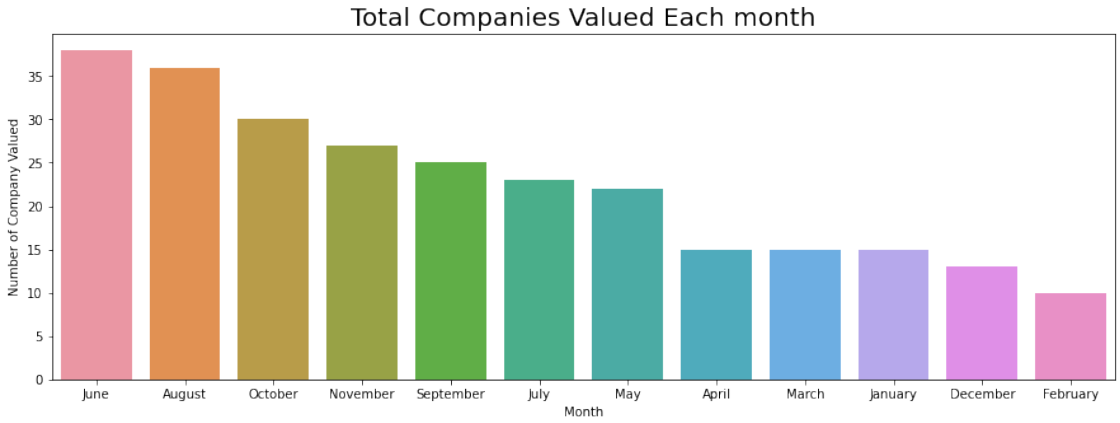
[32]:
```
month_count = data_df['Valuation_month'].value_counts()
```

[33]:
```
month_count
```

[33]:
```
June         38
August       36
October      30
November     27
September    25
July         23
May          22
April        15
March        15
January      15
```

```
December     13
February     10
Name: Valuation_month, dtype: int64
```

[34]:
```
plt.figure(figsize=(15,5))
fig1 = sns.barplot(month_count.index,month_count)
fig1.set_xlabel('Month',fontsize = 10)
fig1.set_ylabel('Number of Company Valued', fontsize = 10)
fig1.set_title('Total Companies Valued Each month', fontsize = 20)
plt.show()
```



[85]:
```
country_count = data_df['Country'].value_counts().head(15)
```

[86]:
```
country_count
```

[86]:
```
UnitedStates     128
China             27
India             27
UnitedKingdom     11
Germany            6
Canada             6
SouthKorea         6
France             5
Israel             4
China              3
Indonesia          3
Israel             3
Brazil             3
Belgium            2
Netherlands        2
Name: Country, dtype: int64
```

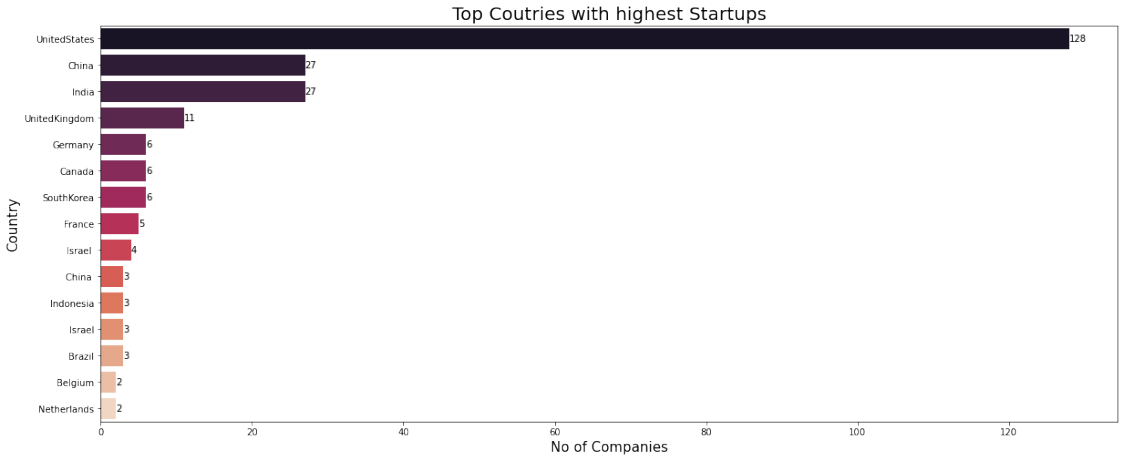
[87]:
```
data_df
```

[87]:

	Company	Valuation	Industry	Valuation_month	Country
0	ByteDance	140.00	Internet	April	China
1	SpaceX	100.00	Aerospace	October	UnitedStates
2	Stripe	95.00	Financial services	March	UnitedStates
3	Stripe	95.00	Financial services	March	Ireland
4	Klarna	45.60	Fintech	June	Sweden
..	…	…	…	…	…
264	Gympass	2.20	Unknown	June	Brazil
265	Gympass	2.20	Unknown	June	UnitedStates
266	Kurly	2.20	E-commerce	July	SouthKorea
267	Addepar	2.17	Fintech	June	UnitedStates
268	Eightfold.ai	2.10	Artificial intelligence	June	UnitedStates

[269 rows x 5 columns]

[98]:
```
plt.figure(figsize=(20,8))
fig2 = sns.barplot(country_count,country_count.index,palette = 'rocket')
fig2.set_xlabel('No of Companies',fontsize = 15)
fig2.set_ylabel('Country', fontsize = 15)
fig2.set_title('Top Coutries with highest Startups', fontsize = 20)
fig2.bar_label(fig2.containers[0])
plt.show()
```



4 Question and Answers

First Lets Create a Copy of our Dataset in order to perform operations to answer below mentioned questions. Doing this will not harm our actual dataset.

[44]:
```
data_df_copy = data_df.copy()
```

[45]:
```
data_df_copy.set_index('Company', inplace = True)
```

[46]:
```
data_df_copy
```

[46]:
```
              Valuation                 Industry Valuation_month  \
Company
ByteDance        140.00                 Internet           April
SpaceX           100.00                Aerospace         October
Stripe            95.00       Financial services           March
Stripe            95.00       Financial services           March
Klarna            45.60                  Fintech            June
…                   …                        …              …
Gympass            2.20                  Unknown            June
Gympass            2.20                  Unknown            June
Kurly              2.20               E-commerce            July
Addepar            2.17                  Fintech            June
Eightfold.ai       2.10  Artificial intelligence            June

                    Country
Company
ByteDance             China
SpaceX          UnitedStates
Stripe          UnitedStates
Stripe               Ireland
```

```
Klarna               Sweden
…                      …
Gympass              Brazil
Gympass        UnitedStates
Kurly            SouthKorea
Addepar        UnitedStates
Eightfold.ai   UnitedStates

[269 rows x 4 columns]
```

4.0.1 Question 1] Which are the top 10 Companies with highest Valuation?

[47]:
```
high = data_df_copy['Valuation'].sort_values(ascending = False).head(10)
```

[48]:
```
high
```

[48]:
```
Company
ByteDance     140.0
SpaceX        100.0
Stripe         95.0
Stripe         95.0
Klarna         45.6
Canva          40.0
Instacart      39.0
Databricks     38.0
Revolut        33.0
Nubank         30.0
Name: Valuation, dtype: float64
```

4.0.2 Question 2] In March Which Company has the highest Valuation?

[49]:
```
mar_df = data_df_copy.loc[data_df_copy['Valuation_month'] == 'March'].head(1)
```

[50]:
```
mar_df
```

[50]:
```
         Valuation            Industry Valuation_month        Country
Company
Stripe        95.0  Financial services           March  UnitedStates
```

4.0.3 Question 3] Get the list of Indian companies and check which one has the highest Valuation?

[51]:
```
indian_df = data_df_copy.loc[data_df_copy['Country'] == 'India']
```

[52]:
```
indian_df
```

[52]:

Company	Valuation	Industry	Valuation_month	Country
Byju's	21.0	Education technology	November	India
Oyo	9.6	Hospitality	August	India
Dream11	8.0	Fantasy sports	November	India
Razorpay	7.5	Fintech	December	India
Ola Cabs	7.3	Transportation	December	India
PharmEasy	5.6	Health technology	October	India
PhonePe	5.5	Fintech	December	India
Swiggy	5.5	Food delivery	July	India
Meesho	4.9	E-commerce	September	India
BrowserStack	4.0	Software	June	India
CRED	4.0	Fintech	October	India
Digit Insurance	3.5	Insurance	July	India
Unacademy	3.4	Education Technology	August	India
Eruditus	3.2	Educational technology	August	India
Udaan	3.1	B2B e-commerce	January	India
Delhivery	3.0	Logistics	May	India
Groww	3.0	Fintech	April	India
OfBusiness	3.0	B2B e-commerce	September	India
Ola Electric	3.0	Electric vehicles	September	India
Paytm Mall	3.0	E-commerce	July	India
Pine Labs	3.0	Fintech	May	India
Upstox	3.0	Fintech	November	India
BharatPe	2.8	Fintech	August	India
Urban Company	2.8	Home improvement	December	India
Infra.Market	2.5	Unknown	August	India
Lenskart	2.5	Retail	May	India
MPL	2.3	Mobile gaming	September	India

[53]:
```
indian_df.head(1)
```

[53]:

Company	Valuation	Industry	Valuation_month	Country
Byju's	21.0	Education technology	November	India

4.0.4 Question 4] In China how many startups are from E-commerce?

[55]:
```
china_df = data_df_copy.loc[data_df_copy['Country']=='China']
```

[56]:
```
china_df
```

[56]:

Company	Valuation	Industry	Valuation_month	\
ByteDance	140.00	Internet	April	
Xiaohongshu	20.00	E-commerce	October	
Yuanfudao	15.50	Education technology	October	

DJI	15.00	Technology	September
Shein	15.00	E-commerce	February
Bitmain	12.00	Cryptocurrency	June
ZongMu Technology	11.00	Self-driving cars	June
Chehaoduo	10.00	Marketplace	July
Lalamove	10.00	Technology	January
WeDoctor	6.80	Healthcare	February
Ziroom	6.60	Real estate technology	March
Lianjia (Homelink)	6.04	Real estate	April
Xingsheng Youxuan	6.00	Retail	January
Hello TransTech	5.00	Transportation	March
UBtech Robotics	5.00	Robotics	May
United Imaging Healthcare	5.00	Healthcare	September
Meizu	4.40	Consumer electronics	October
Megvii	4.00	Technology	May
Shouqi	3.55	Unknown	December
WeRide	3.30	Self-driving cars	June
Youxia Motors	3.30	Electric vehicles	October
Horizon Robotics	3.00	Semiconductors	February
Souche	3.00	Marketplace	September
VANCL	3.00	Unknown	February
VIPKID	3.00	Education	June
Yixia Technology	3.00	Unknown	November
UnionPay	2.80	Finance	October

Company	Country
ByteDance	China
Xiaohongshu	China
Yuanfudao	China
DJI	China
Shein	China
Bitmain	China
ZongMu Technology	China
Chehaoduo	China
Lalamove	China
WeDoctor	China
Ziroom	China
Lianjia (Homelink)	China
Xingsheng Youxuan	China
Hello TransTech	China
UBtech Robotics	China
United Imaging Healthcare	China
Meizu	China
Megvii	China
Shouqi	China
WeRide	China

```
Youxia Motors                China
Horizon Robotics             China
Souche                       China
VANCL                        China
VIPKID                       China
Yixia Technology             China
UnionPay                     China
```

[57]:
```
china_ecom = china_df.loc[china_df['Industry']=='E-commerce']
```

[58]:
```
china_ecom
```

[58]:
```
             Valuation    Industry Valuation_month Country
Company
Xiaohongshu       20.0  E-commerce         October   China
Shein             15.0  E-commerce        February   China
```

This is how we can answer lots of questions from any Dataset once it is cleaned and transformed.

[99]:
```
!pip install jovian --upgrade --quiet
```

[100]:
```
import jovian
```

[101]:
```
jovian.commit(outputs=['startups.csv'])
```

```
<IPython.core.display.Javascript object>
```

```
[jovian] Updating notebook "shrey2627/world-startups" on https://jovian.ai
[jovian] Uploading additional outputs…
[jovian] Committed successfully! https://jovian.ai/shrey2627/world-startups
```

[101]:
```
'https://jovian.ai/shrey2627/world-startups'
```

[102]:
```
jovian.commit(project=project_name)
```

```
<IPython.core.display.Javascript object>
```

```
[jovian] Updating notebook "shrey2627/world-startups" on https://jovian.ai
[jovian] Committed successfully! https://jovian.ai/shrey2627/world-startups
```

[102]:
```
'https://jovian.ai/shrey2627/world-startups'
```

[ ]:
```

```