Investment Analysis

Project Brief

You work for Spark Funds, an asset management company. Spark Funds wants to make investments in a few companies. The CEO of Spark Funds wants to understand the global trends in investments so that she can take the investment decisions effectively.

- **Business and Data Understanding**

Spark Funds has two minor constraints for investments:

- It wants to invest between 5 to 15 million USD per round of investment
- It wants to invest only in English-speaking countries because of the ease of communication with the companies it would invest in

(consider a country to be English speaking only if English is one of the official languages in that country. You may use this list: Click here for a list of countries where English is an official language.)

These conditions will give you sufficient information for your initial analysis. Before getting to specific questions, let's understand the problem and the data first.

- 1. What is the strategy?

Spark Funds wants to invest where most other investors are investing. This pattern is often observed among early stage startup investors.

- 1. Where did we get the data from?

We have taken real investment data from crunchbase.com, so the insights you get may be incredibly useful. For this assignment, we have divided the data into the following files:

You have to use three main data tables for the entire analysis (available for download on the next page):

1. What is Spark Funds' business objective?

The business objectives and goals of data analysis are pretty straightforward.

Business objective: The objective is to identify the best sectors, countries, and a suitable investment type for making investments. The overall strategy is to invest where others are investing, implying that the 'best' sectors and countries are the ones 'where most investors are investing'.

Goals of data analysis: Your goals are divided into three sub-goals:

- 1.Investment type analysis: Comparing the typical investment amounts in the venture, seed, angel, private equity etc. so that Spark Funds can choose the type that is best suited for their strategy.
- 2.Country analysis: Identifying the countries which have been the most heavily invested in the past. These will be Spark Funds' favourites as well.
- 3.Sector analysis: Understanding the distribution of investments across the eight main sectors. (Note that we are interested in the eight 'main sectors' provided in the mapping file. The two files — companies and

rounds2 — have numerous sub-sector names; hence, you will need to map each sub-sector to its main sector.)

In [1]:
```
import pandas as pd
```

In [68]:
```
# reading data files
# using encoding = "ISO-8859-1" to avoid pandas encoding error
companies = pd.read_csv(r"C:\Users\ibeme\Downloads\companies.csv",encoding = "ISO-8859-1")
```

In [69]:
```
rounds2 = pd.read_csv(r"C:\Users\ibeme\Downloads\rounds2.csv",encoding="ISO-8859-1")
```

In [70]:
```
companies.head()
```

Out[70]:

	permalink	name	homepage_url	category_list	status	country_code	state_code
0	/Organization/-Fame	#fame	http://livfame.com	Media	operating	IND	16
1	/Organization/-Qounter	:Qounter	http://www.qounter.com	Application Platforms|Real Time|Social Network...	operating	USA	DE
2	/Organization/-The-One-Of-Them-Inc-	(THE) ONE of THEM,Inc.	http://oneofthem.jp	Apps|Games|Mobile	operating	NaN	NaN
3	/Organization/0-6-Com	0-6.com	http://www.0-6.com	Curated Web	operating	CHN	22
4	/Organization/004-Technologies	004 Technologies	http://004gmbh.de/en/004-interact	Software	operating	USA	IL

In [71]:
```
rounds2.head()
```

Out[71]:

	company_permalink	funding_round_permalink	funding_round_type	funding_round_code	funded_a
0	/organization/-fame	/funding-round/9a01d05418af9f794eebff7ace91f638	venture	B	05-01 201
1	/ORGANIZATION/-QOUNTER	/funding-round/22dacff496eb7acb2b901dec1dfe5633	venture	A	14-10 201
2	/organization/-qounter	/funding-round/b44fbb94153f6cdef13083530bb48030	seed	NaN	01-03 201
3	/ORGANIZATION/-THE-ONE-OF-THEM-INC-	/funding-round/650b8f704416801069bb178a1418776b	venture	B	30-01 201
4	/organization/0-6-com	/funding-round/5727accaeaa57461bd22a9bdd945382d	venture	A	19-03 200

In [72]:
```
companies.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 66368 entries, 0 to 66367
```

```
Data columns (total 10 columns):
 #   Column         Non-Null Count  Dtype
---  ------         --------------  -----
 0   permalink      66368 non-null  object
 1   name           66367 non-null  object
 2   homepage_url   61310 non-null  object
 3   category_list  63220 non-null  object
 4   status         66368 non-null  object
 5   country_code   59410 non-null  object
 6   state_code     57821 non-null  object
 7   region         58338 non-null  object
 8   city           58340 non-null  object
 9   founded_at     51147 non-null  object
dtypes: object(10)
memory usage: 5.1+ MB
```

In [73]:
```
rounds2.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 114949 entries, 0 to 114948
Data columns (total 6 columns):
 #   Column                   Non-Null Count   Dtype
---  ------                   --------------   -----
 0   company_permalink        114949 non-null  object
 1   funding_round_permalink  114949 non-null  object
 2   funding_round_type       114949 non-null  object
 3   funding_round_code       31140 non-null   object
 4   funded_at                114949 non-null  object
 5   raised_amount_usd        94959 non-null   float64
dtypes: float64(1), object(5)
memory usage: 5.3+ MB
```

In [74]:
```
companies['permalink'].nunique()
```

Out[74]:
```
66368
```

In [75]:
```
companies.shape
```

Out[75]:
```
(66368, 10)
```

In [76]:
```
companies['permalink']=companies['permalink'].str.lower()
```

In [77]:
```
rounds2.shape
```

Out[77]:
```
(114949, 6)
```

In [78]:
```
rounds2['company_permalink']=rounds2['company_permalink'].str.lower()
```

In [79]:
```
rounds2['company_permalink'].nunique()
```

Out[79]:
```
66370
```

In [80]:
```
# Companies present in rounds but not in companies list
rounds2[~(rounds2['company_permalink'].isin(companies['permalink']))]
```

Out[80]:

	company_permalink	funding_round_permalink	funding_round_type	funding_round_code
29597	/organization/e-cã□bica	/funding-round/8491f74869e4fe8ba9c378394f8fbdea	seed	NaN
31863	/organization/energystone-games-ç□µç□³æ„æ□□	/funding-round/b89553f3d2279c5683ae93f45a21cfe0	seed	NaN
45176	/organization/huizuche-com-æ□ ç§ÿè½¦	/funding-round/8f8a32dbeeb0f831a78702f83af78a36	seed	NaN
58473	/organization/magnet-tech-ç£□ç□³ç§□æ□□	/funding-round/8fc91fbb32bc95e97f151dd0cb4166bf	seed	NaN
101036	/organization/tipcat-interactive-æ²□è□ÿä¿¡æ□¯ç...	/funding-round/41005928a1439cb2d706a43cb661f60f	seed	NaN
109969	/organization/weiche-tech-å□□è½¦ç§□æ□□	/funding-round/f74e457f838b81fa0b29649740f186d8	venture	A
113839	/organization/zengame-ç¦□æ„ç§□æ□□	/funding-round/6ba28fb4f3eadf5a9c6c81bc5dde6cdf	seed	NaN

The company weird characters appear when you import the data file. To confirm whether these characters are actually present in the given data or whether python has introduced them while importing into pandas, let's have a look at the original CSV file in Excel.

Seems there is some encoding issue First, let's try to figure out the encoding type of this file. Then we can try specifying the encoding type at the time of reading the file. The `chardet` library shows the encoding type of a file.

Data Cleaning - I

In [81]:
```
import chardet
```

In [16]:
```
rawdata = open(r'C:\Users\ibeme\Downloads\rounds2.csv', 'rb').read()
result = chardet.detect(rawdata)
charenc = result['encoding']
print(charenc)
```

```
Windows-1254
```

In [417…
```
# trying different encodings
# encoding="cp1254" throws an error
# rounds_original = pd.read_csv(r'C:\Users\ibeme\Downloads\rounds2.csv', encoding="cp1254'
# rounds_original.iloc[[29597, 31863, 45176], :]
```

In [83]:
```
# Companies which are in
rounds2['company_permalink'] = rounds2.company_permalink.str.encode('utf-8').str.decode('a
rounds2.loc[~rounds2['company_permalink'].isin(companies['permalink']), :]
```

Out[83]:

	company_permalink	funding_round_permalink	funding_round_type	funding_round_code
77	/organization/10north	/funding-round/b41ff7de932f8b6e5bbeed3966c0ed6a	equity_crowdfunding	NaN

	company_permalink	funding_round_permalink	funding_round_type	funding_round_code
729	/organization/51wofang-	/funding-round/346b9180d276a74e0fbb2825e66c6f5b	venture	A
2670	/organization/adslinked	/funding-round/449ae54bb63c768c232955ca6911dee4	seed	NaN
3166	/organization/aesthetic-everything-social-network	/funding-round/62593455f1a69857ed05d5734cc04132	equity_crowdfunding	NaN
3291	/organization/affluent-attach-club-2	/funding-round/626678bdf1654bc4df9b1b34647a4df1	seed	NaN
...	...	...	...	...
110545	/organization/whodats-spaces	/funding-round/d5d6db3d1e6c54d71a63b3aa0c9278e6	seed	NaN
113839	/organization/zengame-	/funding-round/6ba28fb4f3eadf5a9c6c81bc5dde6cdf	seed	NaN
114946	/organization/eron	/funding-round/59f4dce44723b794f21ded3daed6e4fe	venture	A
114947	/organization/asys-2	/funding-round/35f09d0794651719b02bbfd859ba9ff5	seed	NaN
114948	/organization/novatiff-reklam-ve-tantm-hizmetl...	/funding-round/af942869878d2cd788ef5189b435ebc4	grant	NaN

74 rows × 6 columns

In [84]:
```
# Look at unique values again
len(rounds2.company_permalink.unique())
```

Out[84]:
```
66368
```

In [85]:
```
rounds2.shape
```

Out[85]:
```
(114949, 6)
```

Now it makes sense - there are 66368 unique companies in both the `rounds` and `companies` dataframes.

It is possible that a similar encoding problems are present in the companies file as well. Let's look at the companies which are present in the companies file but not in the rounds file - if these have special characters, then it is most likely because the `companies` file is encoded (while rounds is not).

In [86]:
```
# companies present in companies df but not in rounds df
companies.loc[~companies['permalink'].isin(rounds2['company_permalink'])]
```

Out[86]:

	permalink	name	homepage_url	category_list	status	cour
43	/organization/10â°north	10Â°North	NaN	Fashion	operating	
426	/organization/51wofang-æ□ å¿§æ□□æ□¿	51wofang æ□ å¿§æ□□æ□¿	http://www.51wofang.com	NaN	closed	
1506	/organization/adslinkedâ□¢	AdsLinkedâ□¢	http://www.adslinked.com	Advertising|Internet	operating	

	permalink	name	homepage_url	category_list	status	cour
1775	/organization/aesthetic-everythingâ®-social-ne...	Aesthetic EverythingÂ® Social Network	http://aestheticeverything.com/	Public Relations	operating	
1834	/organization/affluent-attachã©-club-2	Affluent AttachÃ© Club	http://www.affluentattache.com/	Hospitality	operating	
...	...	...	...	...	...	
63833	/organization/whodatâ□□s-spaces	Whodatâ□□s Spaces	NaN	Apps	operating	
65778	/organization/zengame-ç¦□æ„ç§□æ□□	ZenGame ç¦□æ„ç§□æ□□	http://www.zen-game.com	Internet|Mobile Games|Online Gaming	closed	
66365	/organization/ã□eron	Ã□ERON	http://www.aeron.hu/	NaN	operating	
66366	/organization/ã□asys-2	Ã□asys	http://www.oasys.io/	Consumer Electronics|Internet of Things|Teleco...	operating	
66367	/organization/ä°novatiff-reklam-ve-tanä±tä±m-h...	Ä°novatiff Reklam ve TanÄ±tÄ±m Hizmetleri Tic	http://inovatiff.com	Consumer Goods|E-Commerce|Internet	operating	

68 rows × 10 columns

In [87]:
```
# remove encoding from companies df
companies['permalink'] = companies.permalink.str.encode('utf-8').str.decode('ascii', 'ignc
```

In [88]:
```
# companies present in companies df but not in rounds df
companies.loc[~companies['permalink'].isin(rounds2['company_permalink'])]
```

Out[88]:

permalink	name	homepage_url	category_list	status	country_code	state_code	region	city	founded_at

In [60]:
```
len(companies['permalink'].unique())
```

Out[60]:
```
66368
```

In [61]:
```
len(rounds2['company_permalink'].unique())
```

Out[61]:
```
66368
```

In [89]:
```
# write rounds file
rounds2.to_csv("rounds_clean.csv", index=False)

# write companies file
companies.to_csv("companies_clean.csv", index=False)
```

DataCleaning - II

In [90]:
```
companies = pd.read_csv("companies_clean.csv")
```

```
rounds = pd.read_csv("rounds_clean.csv")
```

In [92]:
```
len(companies.permalink.unique())
```

Out[92]:
```
66368
```

In [93]:
```
len(rounds.company_permalink.unique())
```

Out[93]:
```
66368
```

Missing Value Treatment

In [94]:
```
companies.isnull().sum()
```

Out[94]:
```
permalink            0
name                 1
homepage_url      5058
category_list     3148
status               0
country_code      6958
state_code        8547
region            8030
city              8028
founded_at       15221
dtype: int64
```

In [96]:
```
rounds.isnull().sum()
```

Out[96]:
```
company_permalink              0
funding_round_permalink        0
funding_round_type             0
funding_round_code         83809
funded_at                      0
raised_amount_usd          19990
dtype: int64
```

In [98]:
```
master_df = pd.merge(companies,rounds,how = "inner",left_on="permalink",right_on='company_
```

In [100…
```
master_df.shape
```

Out[100…
```
(114949, 16)
```

In [101…
```
master_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 114949 entries, 0 to 114948
Data columns (total 16 columns):
 #   Column                   Non-Null Count   Dtype
---  ------                   --------------   -----
 0   permalink                114949 non-null  object
 1   name                     114948 non-null  object
 2   homepage_url             108815 non-null  object
 3   category_list            111539 non-null  object
 4   status                   114949 non-null  object
 5   country_code             106271 non-null  object
 6   state_code               104003 non-null  object
```

```
 7   region                   104782 non-null  object
 8   city                     104785 non-null  object
 9   founded_at               94428 non-null   object
 10  company_permalink        114949 non-null  object
 11  funding_round_permalink  114949 non-null  object
 12  funding_round_type       114949 non-null  object
 13  funding_round_code       31140 non-null   object
 14  funded_at                114949 non-null  object
 15  raised_amount_usd        94959 non-null   float64
dtypes: float64(1), object(15)
memory usage: 14.9+ MB
```

In [104…]
```
round(100*(master_df.isnull().sum()/len(master_df.index)),2)
```

Out[104…]
```
permalink                  0.00
name                       0.00
homepage_url               5.34
category_list              2.97
status                     0.00
country_code               7.55
state_code                 9.52
region                     8.84
city                       8.84
founded_at                17.85
company_permalink          0.00
funding_round_permalink    0.00
funding_round_type         0.00
funding_round_code        72.91
funded_at                  0.00
raised_amount_usd         17.39
dtype: float64
```

We can see that funding_round_code col is not useful as 73% of that column is of NAs.As per the business objective, we can remove homepage_url,founded_at,state_code,region and city

In [107…]
```
master_df = master_df.drop(['funding_round_code','homepage_url','founded_at','state_code',
```

In [108…]
```
master_df.head()
```

Out[108…]

	permalink	name	category_list	status	country_code	company_permalink	fund
0	/organization/-fame	#fame	Media	operating	IND	/organization/-fame	round/9a01d05418a
1	/organization/-qounter	:Qounter	Application Platforms|Real Time|Social Network...	operating	USA	/organization/-qounter	round/22dacff496eb7
2	/organization/-qounter	:Qounter	Application Platforms|Real Time|Social Network...	operating	USA	/organization/-qounter	round/b44fbb94153f6
3	/organization/-the-one-of-them-inc-	(THE) ONE of THEM,Inc.	Apps|Games|Mobile	operating	NaN	/organization/-the-one-of-them-inc-	round/650b8f7044168(
4	/organization/0-6-com	0-6.com	Curated Web	operating	CHN	/organization/0-6-com	round/5727accaeaa574

In [110…]
```
round(100*(master_df.isnull().sum()/len(master_df.index)),2)
```

Out[110…
```
permalink                   0.00
name                        0.00
category_list               2.97
status                      0.00
country_code                7.55
company_permalink           0.00
funding_round_permalink     0.00
funding_round_type          0.00
funded_at                   0.00
raised_amount_usd          17.39
dtype: float64
```

In [111…
```
master_df['raised_amount_usd'].describe()
```

Out[111…
```
count    9.495900e+04
mean     1.042687e+07
std      1.148212e+08
min      0.000000e+00
25%      3.225000e+05
50%      1.680511e+06
75%      7.000000e+06
max      2.127194e+10
Name: raised_amount_usd, dtype: float64
```

In [124…
```
master_df = master_df[~master_df['raised_amount_usd'].isna()]
```

In [125…
```
master_df.isnull().sum()
```

Out[125…
```
permalink                   0
name                        1
category_list            1044
status                      0
country_code             5851
company_permalink           0
funding_round_permalink     0
funding_round_type          0
funded_at                   0
raised_amount_usd           0
dtype: int64
```

In [126…
```
round(100*(master_df.isnull().sum()/len(master_df.index)),2)
```

Out[126…
```
permalink                   0.00
name                        0.00
category_list               1.10
status                      0.00
country_code                6.16
company_permalink           0.00
funding_round_permalink     0.00
funding_round_type          0.00
funded_at                   0.00
raised_amount_usd           0.00
dtype: float64
```

In [127…
```
master_df['country_code'].value_counts()
```

Out[127…
```
USA    62049
GBR     5019
CAN     2616
CHN     1927
```

```
IND     1649
        ...
HND        1
GGY        1
TGO        1
MNE        1
SEN        1
Name: country_code, Length: 134, dtype: int64
```

As we can see that just 6% of 'country_code' column is having missing values.So we can remove those NA values

In [128…
```
master_df = master_df[~master_df['country_code'].isna()]
```

In [129…
```
round(100*(master_df.isnull().sum()/len(master_df.index)),2)
```

Out[129…
```
permalink                  0.00
name                       0.00
category_list              0.65
status                     0.00
country_code               0.00
company_permalink          0.00
funding_round_permalink    0.00
funding_round_type         0.00
funded_at                  0.00
raised_amount_usd          0.00
dtype: float64
```

As we can see that just 0.65% of 'Category_list' column is having missing values.So we can remove those NA values

In [130…
```
master_df = master_df[~master_df['category_list'].isna()]
```

In [131…
```
round(100*(master_df.isnull().sum()/len(master_df.index)),2)
```

Out[131…
```
permalink                  0.0
name                       0.0
category_list              0.0
status                     0.0
country_code               0.0
company_permalink          0.0
funding_round_permalink    0.0
funding_round_type         0.0
funded_at                  0.0
raised_amount_usd          0.0
dtype: float64
```

In [154…
```
master_df.to_csv("master_df.csv", index=False)
```

In [274…
```
df = pd.read_csv('master_df.csv',sep = ',',encoding = "ISO-8859-1")
```

Part3: Analysis

As per the objective, we are doing 3 types of analysis - Funding Type, country analysis and sector analysis

Funding Type Analysis

Let's compare the funding amounts across the funding types. Also, we need to impose the constraint that the investment amount should be between 5 and 15 million USD. We will choose the funding type such that the average investment amount falls in this range.

In [275…
```
df.head()
```

Out[275…

	permalink	name	category_list	status	country_code	company_permalink	
0	/organization/-fame	#fame	Media	operating	IND	/organization/-fame	round/9a01d0
1	/organization/-qounter	:Qounter	Application Platforms|Real Time|Social Network...	operating	USA	/organization/-qounter	round/b44fbb9
2	/organization/0-6-com	0-6.com	Curated Web	operating	CHN	/organization/0-6-com	round/5727acca
3	/organization/01games-technology	01Games Technology	Games	operating	HKG	/organization/01games-technology	round/7d53696
4	/organization/0ndine-biomedical-inc	Ondine Biomedical Inc.	Biotechnology	operating	CAN	/organization/0ndine-biomedical-inc	round/2b9d3a

In [276…
```
df['funding_round_type'].value_counts()
```

Out[276…
```
venture                  47809
seed                     21095
debt_financing            6506
angel                     4400
grant                     1939
private_equity            1820
undisclosed               1345
convertible_note          1320
equity_crowdfunding       1128
post_ipo_equity            598
product_crowdfunding       330
post_ipo_debt              151
non_equity_assistance       60
secondary_market            28
Name: funding_round_type, dtype: int64
```

In [277…
```
df['raised_amount_usd'].describe()
```

Out[277…
```
count    8.852900e+04
mean     1.047385e+07
std      1.118118e+08
min      0.000000e+00
25%      3.705180e+05
50%      1.800000e+06
75%      7.100000e+06
max      2.127194e+10
Name: raised_amount_usd, dtype: float64
```

In [278…
```
df = df[df.funding_round_type.isin(['venture','seed','angel','private_equity'])]
```

In [279…
```
df['funding_round_type'].value_counts()
```

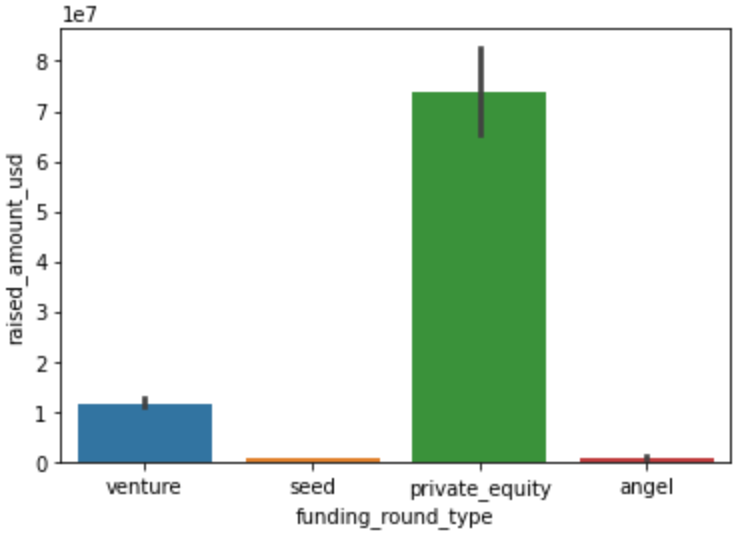
Out[279…
```
venture            47809
seed               21095
angel               4400
private_equity      1820
Name: funding_round_type, dtype: int64
```

In [280…
```
import seaborn as sns
import matplotlib.pyplot as plt
%matplotlib inline
```

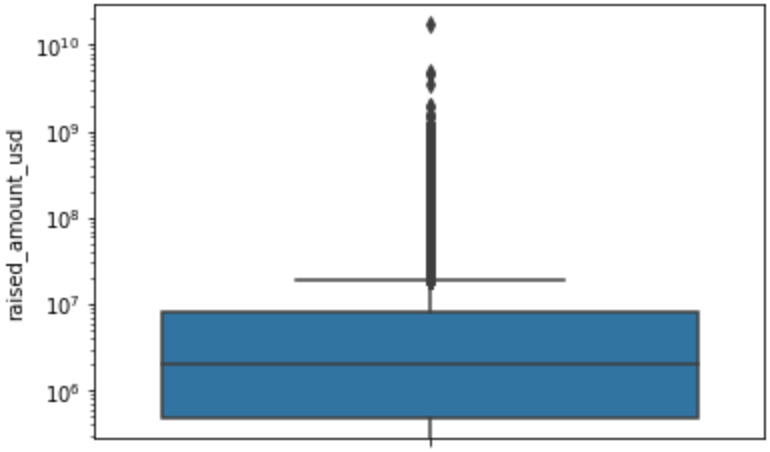
In [281…
```
sns.barplot(df['funding_round_type'],df['raised_amount_usd'])
```

```
C:\Users\ibeme\anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureWarning: Pass
the following variables as keyword args: x, y. From version 0.12, the only valid positiona
l argument will be `data`, and passing other arguments without an explicit keyword will re
sult in an error or misinterpretation.
  warnings.warn(
```

Out[281…
```
<AxesSubplot:xlabel='funding_round_type', ylabel='raised_amount_usd'>
```



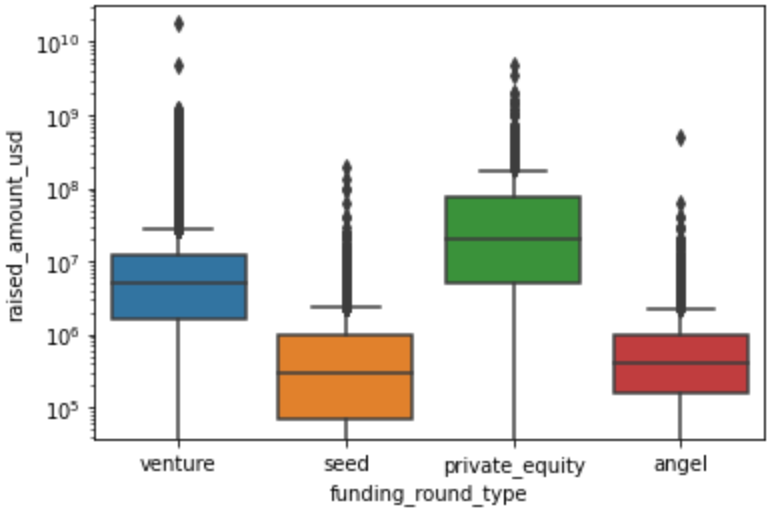
In [282…
```
sns.boxplot(y=df['raised_amount_usd'])
plt.yscale('log')
```



In [283…
```
sns.boxplot(df['funding_round_type'],df['raised_amount_usd'])
plt.yscale("log")
```

```
C:\Users\ibeme\anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureWarning: Pass
the following variables as keyword args: x, y. From version 0.12, the only valid positiona
```

```
l argument will be `data`, and passing other arguments without an explicit keyword will re
sult in an error or misinterpretation.
  warnings.warn(
```



In [284…
```
df.groupby(['funding_round_type'])['raised_amount_usd'].describe()
```

Out[284…

funding_round_type	count	mean	std	min	25%	50%	75%	max
angel	4400.0	9.715739e+05	7.710904e+06	0.0	152756.5	414906.0	1000000.0	4.945120e+08
private_equity	1820.0	7.393849e+07	2.017765e+08	0.0	5000000.0	20000000.0	75762572.0	4.745460e+09
seed	21095.0	7.477937e+05	2.288318e+06	0.0	68816.5	300000.0	1000000.0	2.000000e+08
venture	47809.0	1.172422e+07	8.821571e+07	0.0	1600000.0	5000000.0	12000000.0	1.760000e+10

As we can see huge difference betweebn mean and median, we chose median to be most representative values

In [285…
```
df.groupby(['funding_round_type'])['raised_amount_usd'].median().sort_values(ascending=Fal
```

Out[285…
```
funding_round_type
private_equity    20000000.0
venture            5000000.0
angel               414906.0
seed                300000.0
Name: raised_amount_usd, dtype: float64
```

Observations: Considering that Spark Funds wants to invest between 5 to 15 million USD per investment round, the "venture" investemnt type is well suited

Country Analysis

Spark Funds wants to invest in countries with the highest amount of funding for the chosen investment type. This is a part of its broader strategy to invest where most investments are occurring.

1. Spark Funds wants to see the top nine countries which have received the highest total funding (across ALL sectors for the chosen investment type)

2. For the chosen investment type, make a data frame named top9 with the top nine countries (based on the total investment amount each country has received)

Let's now compare the total investment amounts across countries. Note that we'll filter the data for only the

'venture' type investments and then compare the 'total investment' across countries.

In [286…
```
df_venture = df[df['funding_round_type']=='venture']
```

In [287…
```
df_venture.head()
```

Out[287…

	permalink	name	category_list	status	country_code	company_permalink	fun
0	/organization/-fame	#fame	Media	operating	IND	/organization/-fame	round/9a01d05418
2	/organization/0-6-com	0-6.com	Curated Web	operating	CHN	/organization/0-6-com	round/5727accaeaa57
5	/organization/0ndine-biomedical-inc	Ondine Biomedical Inc.	Biotechnology	operating	CAN	/organization/0ndine-biomedical-inc	round/954b9499724b
7	/organization/0xdata	H2O.ai	Analytics	operating	USA	/organization/0xdata	round/3bb2ee4a2d89
8	/organization/0xdata	H2O.ai	Analytics	operating	USA	/organization/0xdata	round/ae2a174c0651

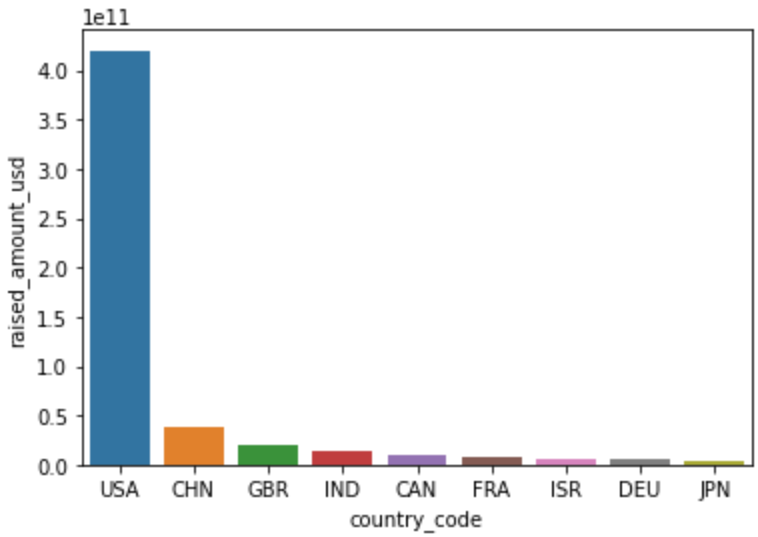
In [288…
```
#  top nine countries which have received the highest total funding (across ALL sectors fo

df_countries = df_venture.groupby('country_code')['raised_amount_usd'].sum().sort_values(a
```

In [289…
```
df_new =df_countries.reset_index()
```

In [290…
```
sns.barplot(df_new['country_code'],df_new['raised_amount_usd'])
```

```
C:\Users\ibeme\anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureWarning: Pass
the following variables as keyword args: x, y. From version 0.12, the only valid positiona
l argument will be `data`, and passing other arguments without an explicit keyword will re
sult in an error or misinterpretation.
  warnings.warn(
<AxesSubplot:xlabel='country_code', ylabel='raised_amount_usd'>
```

Out[290…



Identify the top three English-speaking countries in the data frame top9.

USA,GBR,IND

In [312…
```
df_english = df_venture[df_venture['country_code'].isin(['USA','GBR','IND'])]
```

Sector Analysis

First, we need to extract the main sector using the column `category_list` . The category_list column contains values such as 'Biotechnology|Health Care' - in this, 'Biotechnology' is the 'main category' of the company, which we need to use.

Let's extract the main categories in a new column.

In [313…
```
df_english.head()
```

Out[313…

	permalink	name	category_list	status	country_code	company_permalink	
0	/organization/-fame	#fame	Media	operating	IND	/organization/-fame	
7	/organization/0xdata	H2O.ai	Analytics	operating	USA	/organization/0xdata	rou
8	/organization/0xdata	H2O.ai	Analytics	operating	USA	/organization/0xdata	ro
9	/organization/0xdata	H2O.ai	Analytics	operating	USA	/organization/0xdata	
15	/organization/1-mainstream	1 Mainstream	Apps|Cable|Distribution|Software	acquired	USA	/organization/1-mainstream	rc

In [314…
```
df_english.loc[:,'main_category']=df_english['category_list'].apply(lambda x:x.split("|")
```

```
C:\Users\ibeme\AppData\Local\Temp\ipykernel_69116\866150497.py:1: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_gu
ide/indexing.html#returning-a-view-versus-a-copy
  df_english.loc[:,'main_category']=df_english['category_list'].apply(lambda x:x.split
("|")[0])
```

In [315…
```
df_english['main_category'].value_counts()
```

Out[315…
```
Biotechnology          5875
Software               3345
Advertising            1847
Health Care            1596
Enterprise Software    1560
                       ...
High Tech                 1
Rapidly Expanding         1
Video Conferencing        1
Debt Collecting           1
Task Management           1
Name: main_category, Length: 563, dtype: int64
```

In [316…
```
mapping = pd.read_csv(r"C:\Users\ibeme\Downloads\mapping.csv")
```

In [317…

```
mapping.head()
```

Out[317…

	category_list	Automotive & Sports	Blanks	Cleantech / Semiconductors	Entertainment	Health	Manufacturing	News, Search and Messaging	Others	A
0	NaN	0	1	0	0	0	0	0	0	
1	3D	0	0	0	0	0	1	0	0	
2	3D Printing	0	0	0	0	0	1	0	0	
3	3D Technology	0	0	0	0	0	1	0	0	
4	Accounting	0	0	0	0	0	0	0	0	

In [318…

```
mapping.isna().sum()
```

Out[318…

```
category_list                          1
Automotive & Sports                    0
Blanks                                 0
Cleantech / Semiconductors             0
Entertainment                          0
Health                                 0
Manufacturing                          0
News, Search and Messaging             0
Others                                 0
Social, Finance, Analytics, Advertising    0
dtype: int64
```

In [319…

```
mapping.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 688 entries, 0 to 687
Data columns (total 10 columns):
 #   Column                                 Non-Null Count  Dtype 
---  ------                                 --------------  ----- 
 0   category_list                          687 non-null    object
 1   Automotive & Sports                    688 non-null    int64 
 2   Blanks                                 688 non-null    int64 
 3   Cleantech / Semiconductors             688 non-null    int64 
 4   Entertainment                          688 non-null    int64 
 5   Health                                 688 non-null    int64 
 6   Manufacturing                          688 non-null    int64 
 7   News, Search and Messaging             688 non-null    int64 
 8   Others                                 688 non-null    int64 
 9   Social, Finance, Analytics, Advertising  688 non-null    int64 
dtypes: int64(9), object(1)
memory usage: 53.9+ KB
```

In [320…

```
mapping.Blanks.value_counts()
```

Out[320…

```
0    687
1      1
Name: Blanks, dtype: int64
```

In [321…

```
mapping = mapping[~mapping['category_list'].isna()]
```

In [322…
```
mapping.isna().sum()
```

Out[322…
```
category_list                          0
Automotive & Sports                    0
Blanks                                 0
Cleantech / Semiconductors             0
Entertainment                          0
Health                                 0
Manufacturing                          0
News, Search and Messaging             0
Others                                 0
Social, Finance, Analytics, Advertising    0
dtype: int64
```

In [323…
```
# As given we have to merge mapping and master files to map category to the company

df_english.head()
```

Out[323…

	permalink	name	category_list	status	country_code	company_permalink	
0	/organization/-fame	#fame	Media	operating	IND	/organization/-fame	
7	/organization/0xdata	H2O.ai	Analytics	operating	USA	/organization/0xdata	rou
8	/organization/0xdata	H2O.ai	Analytics	operating	USA	/organization/0xdata	ro
9	/organization/0xdata	H2O.ai	Analytics	operating	USA	/organization/0xdata	
15	/organization/1-mainstream	1 Mainstream	Apps|Cable|Distribution|Software	acquired	USA	/organization/1-mainstream	rc

In [324…
```
mapping['category_list'] = mapping['category_list'].str.lower()
```

In [325…
```
df_english['main_category']=df_english['main_category'].str.lower()
```

```
C:\Users\ibeme\AppData\Local\Temp\ipykernel_69116\3531690553.py:1: SettingWithCopyWarning:

A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_gu
ide/indexing.html#returning-a-view-versus-a-copy
  df_english['main_category']=df_english['main_category'].str.lower()
```

In [326…
```
df_english.head()
```

Out[326…

	permalink	name	category_list	status	country_code	company_permalink	
0	/organization/-fame	#fame	Media	operating	IND	/organization/-fame	
7	/organization/0xdata	H2O.ai	Analytics	operating	USA	/organization/0xdata	rou
8	/organization/0xdata	H2O.ai	Analytics	operating	USA	/organization/0xdata	ro

	permalink	name	category_list	status	country_code	company_permalink	
9	/organization/0xdata	H2O.ai	Analytics	operating	USA	/organization/0xdata	
15	/organization/1-mainstream	1 Mainstream	Apps|Cable|Distribution|Software	acquired	USA	/organization/1-mainstream	rc

To be able to merge all the `main_category` values with the mapping file's `category_list` column, all the values in the `main_category` column should be present in the `category_list` column of the mapping file.

Let's see if this is true.

In [327…
```
df_english[~df_english['main_category'].isin(mapping['category_list'])]
```

Out[327…

	permalink	name	category_list	status	country_code	company_permalink	
7	/organization/0xdata	H2O.ai	Analytics	operating	USA	/organization/0xdata	round/3bb2ee4a
8	/organization/0xdata	H2O.ai	Analytics	operating	USA	/organization/0xdata	round/ae2a174
9	/organization/0xdata	H2O.ai	Analytics	operating	USA	/organization/0xdata	round/e1cfcb
47	/organization/100plus	100Plus	Analytics	acquired	USA	/organization/100plus	round/b5facb0
136	/organization/1world-online	1World Online	Analytics|Big Data|Enterprise Software|Market …	operating	USA	/organization/1world-online	round/32936e58
…	…	…	…	…	…	…	
88270	/organization/zoopla	Zoopla	Property Management|Real Estate	ipo	GBR	/organization/zoopla	round/98da1f4
88291	/organization/zopa	Zopa	Finance|FinTech	operating	GBR	/organization/zopa	round/2a55d43
88292	/organization/zopa	Zopa	Finance|FinTech	operating	GBR	/organization/zopa	round/4b0740c
88293	/organization/zopa	Zopa	Finance|FinTech	operating	GBR	/organization/zopa	round/54dbfbd
88294	/organization/zopa	Zopa	Finance|FinTech	operating	GBR	/organization/zopa	round/720b9f2

2616 rows × 11 columns

In [328…
```
# values in the category_list column which are not in main_category column
mapping[~mapping['category_list'].isin(df_english['main_category'])]
```

Out[328…

category_list	Automotive & Sports	Blanks	Cleantech / Semiconductors	Entertainment	Health	Manufacturing	News, Search and Messaging	Others

	category_list	Automotive & Sports	Blanks	Cleantech / Semiconductors	Entertainment	Health	Manufacturing	News, Search and Messaging	Others
16	air pollution control	0	0	1	0	0	0	0	0
20	alter0tive medicine	0	0	0	0	1	0	0	0
22	a0lytics	0	0	0	0	0	0	0	0
33	aquaculture	0	0	1	0	0	0	0	0
49	b2b express delivery	0	0	0	0	0	0	0	0
...	...	...	...	...	...	...	...	...	...
670	virtual workforces	0	0	0	1	0	0	0	0
672	waste ma0gement	0	0	1	0	0	0	0	0
682	weddings	0	0	0	1	0	0	0	0
683	wholesale	0	0	0	0	0	0	0	1
686	women	0	0	0	0	0	0	0	1

175 rows × 10 columns

If you see carefully, you'll notice something fishy - there are sectors named *alter0tive medicine*, *a0lytics*, *waste ma0gement*, *veteri0ry*, etc. This is not a *random* quality issue, but rather a pattern. In some strings, the 'na' has been replaced by '0'. This is weird - maybe someone was trying to replace the 'NA' values with '0', and ended up doing this.

Let's treat this problem by replacing '0' with 'na' in the `category_list` column.

In [329…
```
mapping['category_list']=mapping['category_list'].apply(lambda x:x.replace('0','na'))
```

In [330…
```
# values in the category_list column which are not in main_category column
mapping[~mapping['category_list'].isin(df_english['main_category'])]
```

Out[330…

	category_list	Automotive & Sports	Blanks	Cleantech / Semiconductors	Entertainment	Health	Manufacturing	News, Search and Messaging	Others
16	air pollution control	0	0	1	0	0	0	0	0
33	aquaculture	0	0	1	0	0	0	0	0
49	b2b express delivery	0	0	0	0	0	0	0	0
64	biomass power generation	0	0	1	0	0	0	0	0

	category_list	Automotive & Sports	Blanks	Cleantech / Semiconductors	Entertainment	Health	Manufacturing	News, Search and Messaging	Others
69	boating industry	1	0	0	0	0	0	0	0
...	...	...	...	...	...	...	...	...	...
669	video streaming	0	0	0	1	0	0	0	0
670	virtual workforces	0	0	0	1	0	0	0	0
682	weddings	0	0	0	1	0	0	0	0
683	wholesale	0	0	0	0	0	0	0	1
686	women	0	0	0	0	0	0	0	1

134 rows × 10 columns

In [331…
```
df_english=df_english.drop('category_list',axis = 1)
```

In [332…
```
# merge the dfs
df_english = pd.merge(df_english, mapping, how='inner', left_on='main_category', right_on=
df_english.head()
```

Out[332…

	permalink	name	status	country_code	company_permalink	funding_round_permalink
0	/organization/-fame	#fame	operating	IND	/organization/-fame	/funding-round/9a01d05418af9f794eebff7ace91f638
1	/organization/90min	90min	operating	GBR	/organization/90min	/funding-round/21a2cbf6f2fb2a1c2a61e04bf930dfe6
2	/organization/90min	90min	operating	GBR	/organization/90min	/funding-round/bd626ed022f5c66574b1afe234f3c90c
3	/organization/90min	90min	operating	GBR	/organization/90min	/funding-round/fd4b15e8c97ee2ffc0acccdbe1a98810
4	/organization/all-def-digital	All Def Digital	operating	USA	/organization/all-def-digital	/funding-round/452a2342fe720285c3b92e9bd927d9ba

In [333…
```
df_english.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 38788 entries, 0 to 38787
Data columns (total 20 columns):
 #   Column                                  Non-Null Count  Dtype
---  ------                                  --------------  -----
 0   permalink                               38788 non-null  object
 1   name                                    38788 non-null  object
 2   status                                  38788 non-null  object
 3   country_code                            38788 non-null  object
```

```
 4   company_permalink                          38788 non-null  object
 5   funding_round_permalink                    38788 non-null  object
 6   funding_round_type                         38788 non-null  object
 7   funded_at                                  38788 non-null  object
 8   raised_amount_usd                          38788 non-null  float64
 9   main_category                              38788 non-null  object
 10  category_list                              38788 non-null  object
 11  Automotive & Sports                        38788 non-null  int64
 12  Blanks                                     38788 non-null  int64
 13  Cleantech / Semiconductors                 38788 non-null  int64
 14  Entertainment                              38788 non-null  int64
 15  Health                                     38788 non-null  int64
 16  Manufacturing                              38788 non-null  int64
 17  News, Search and Messaging                 38788 non-null  int64
 18  Others                                     38788 non-null  int64
 19  Social, Finance, Analytics, Advertising    38788 non-null  int64
dtypes: float64(1), int64(9), object(10)
memory usage: 6.2+ MB
```

Convert the wide dataframe into long Df

You'll notice that the columns representing the main category in the mapping file are originally in the 'wide' format - Automotive & Sports, Cleantech / Semiconductors etc.

They contain the value '1' if the company belongs to that category, else 0. This is quite redundant. We can as well have a column named 'sub-category' having these values.

Let's convert the df into the long format from the current wide format. First, we'll store the 'value variables' (those which are to be melted) in an array. The rest will then be the 'index variables'.

In [344…

```
#?pd.melt
df_english.iloc[:,11:19]
#df_english.loc[:,'permalink':'main_category']
```

Out[344…

	Automotive & Sports	Blanks	Cleantech / Semiconductors	Entertainment	Health	Manufacturing	News, Search and Messaging	Others
0	0	0	0	1	0	0	0	0
1	0	0	0	1	0	0	0	0
2	0	0	0	1	0	0	0	0
3	0	0	0	1	0	0	0	0
4	0	0	0	1	0	0	0	0
...	...	...	...	...	...	...	...	...
38783	0	0	0	0	0	0	0	1
38784	0	0	0	0	0	0	0	1
38785	0	0	0	0	0	0	0	1
38786	0	0	0	0	0	0	0	1
38787	0	0	0	0	0	0	0	1

38788 rows × 8 columns

In [346…

```
#Convert the wide dataframe into long Df
```

```
long_df = pd.melt(df_english,id_vars=df_english.iloc[:,range(10)],value_vars=df_english.il
```

In [349…

```
long_df.head()
```

Out[349…

	permalink	name	status	country_code	company_permalink	funding_round_permalink
0	/organization/-fame	#fame	operating	IND	/organization/-fame	/funding-round/9a01d05418af9f794eebff7ace91f638
1	/organization/90min	90min	operating	GBR	/organization/90min	/funding-round/21a2cbf6f2fb2a1c2a61e04bf930dfe6
2	/organization/90min	90min	operating	GBR	/organization/90min	/funding-round/bd626ed022f5c66574b1afe234f3c90c
3	/organization/90min	90min	operating	GBR	/organization/90min	/funding-round/fd4b15e8c97ee2ffc0acccdbe1a98810
4	/organization/all-def-digital	All Def Digital	operating	USA	/organization/all-def-digital	/funding-round/452a2342fe720285c3b92e9bd927d9ba

We can now get rid of the rows where the column 'value' is 0 and then remove that column altogether.

In [354…

```
long_df = long_df[long_df['value'] == 1]
```

In [355…

```
long_df=long_df.drop('value',axis = 1)
```

In [356…

```
long_df.head()
```

Out[356…

	permalink	name	status	country_code	company_permalink	funding_round_
25828	/organization/3d-robotics	3D Robotics	operating	USA	/organization/3d-robotics	round/2785595770e91ab8fd4854e
25829	/organization/3d-robotics	3D Robotics	operating	USA	/organization/3d-robotics	round/7ca0d4dc119b6d65eebfb35
25830	/organization/3d-robotics	3D Robotics	operating	USA	/organization/3d-robotics	round/d6221c11246b0a536ee2cac
25831	/organization/3d-robotics	3D Robotics	operating	USA	/organization/3d-robotics	round/ff3c1d1ae1c3486d775095b0
25832	/organization/cape-productions	Cape Productions	operating	USA	/organization/cape-productions	round/156e4fbce54aca39a8be9a1

In [357…

```
len(long_df)
```

Out[357…

```
30974
```

In [361…

```
# renaming the 'variable' column

long_df=long_df.rename(columns={'variable':'sector'})
```

In [362…

```
long_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 30974 entries, 25828 to 310303
Data columns (total 11 columns):
 #   Column                   Non-Null Count  Dtype
---  ------                   --------------  -----
 0   permalink                30974 non-null  object
 1   name                     30974 non-null  object
 2   status                   30974 non-null  object
 3   country_code             30974 non-null  object
 4   company_permalink        30974 non-null  object
 5   funding_round_permalink  30974 non-null  object
 6   funding_round_type       30974 non-null  object
 7   funded_at                30974 non-null  object
 8   raised_amount_usd        30974 non-null  float64
 9   main_category            30974 non-null  object
 10  sector                   30974 non-null  object
dtypes: float64(1), object(10)
memory usage: 2.8+ MB
```

In [365…

```
long_df.groupby('sector')['raised_amount_usd'].sum().sort_values(ascending=False)
```

Out[365…

```
sector
Cleantech / Semiconductors    1.257916e+11
Others                        9.361855e+10
News, Search and Messaging    5.029612e+10
Health                        3.328608e+10
Manufacturing                 2.659486e+10
Entertainment                 1.915332e+10
Automotive & Sports           1.366939e+10
Name: raised_amount_usd, dtype: float64
```

The dataframe now contains only venture type investments in countries USA, IND and GBR, and we have mapped each company to one of the eight main sectors (named 'sector' in the dataframe).

We can now compute the sector-wise number and the amount of investment in the three countries.

In [369…

```
#Taking rows whose investment amount is falling within the 5-15 million USD range as per t

df_final =long_df[long_df['raised_amount_usd'].between(5000000,15000000)]
```

In [370…

```

```

Out[370…

```
permalink                0
name                     0
status                   0
country_code             0
company_permalink        0
funding_round_permalink  0
funding_round_type       0
funded_at                0
raised_amount_usd        0
main_category            0
sector                   0
dtype: int64
```

In [375…

```
# groupby country, sector and compute the count and sum
df_final.groupby(['country_code','sector'])['raised_amount_usd'].agg(['sum','count'])
```

Out[375…

		sum	count
country_code	**sector**		

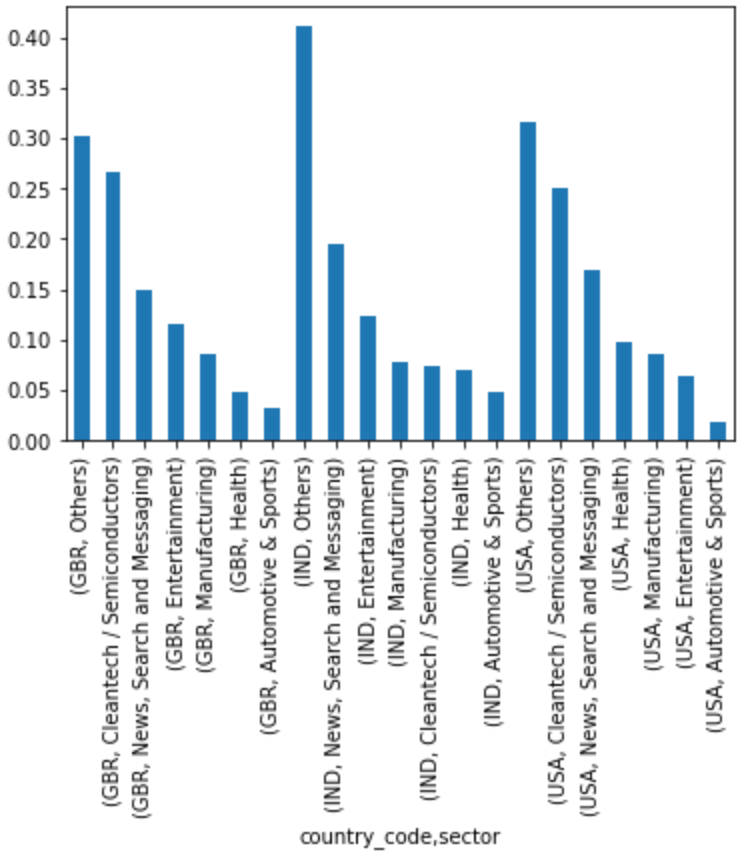
		sum	count
country_code	**sector**		
GBR	**Automotive & Sports**	1.670516e+08	16
	Cleantech / Semiconductors	1.163990e+09	130
	Entertainment	4.827847e+08	56
	Health	2.145375e+08	24
	Manufacturing	3.619403e+08	42
	News, Search and Messaging	6.157462e+08	73
	Others	1.283624e+09	147
IND	**Automotive & Sports**	1.369000e+08	13
	Cleantech / Semiconductors	1.653800e+08	20
	Entertainment	2.808300e+08	33
	Health	1.677400e+08	19
	Manufacturing	2.009000e+08	21
	News, Search and Messaging	4.338345e+08	52
	Others	1.013410e+09	110
USA	**Automotive & Sports**	1.454104e+09	167
	Cleantech / Semiconductors	2.163343e+10	2350
	Entertainment	5.099198e+09	591
	Health	8.211859e+09	909
	Manufacturing	7.258553e+09	799
	News, Search and Messaging	1.397157e+10	1583
	Others	2.632101e+10	2950

In [391…

```
# plotting sector-wise count and sum of investments in the three countries
df_final.groupby('country_code')['sector'].value_counts(normalize=True).plot.bar()
```

Out[391…

```
<AxesSubplot:xlabel='country_code,sector'>
```

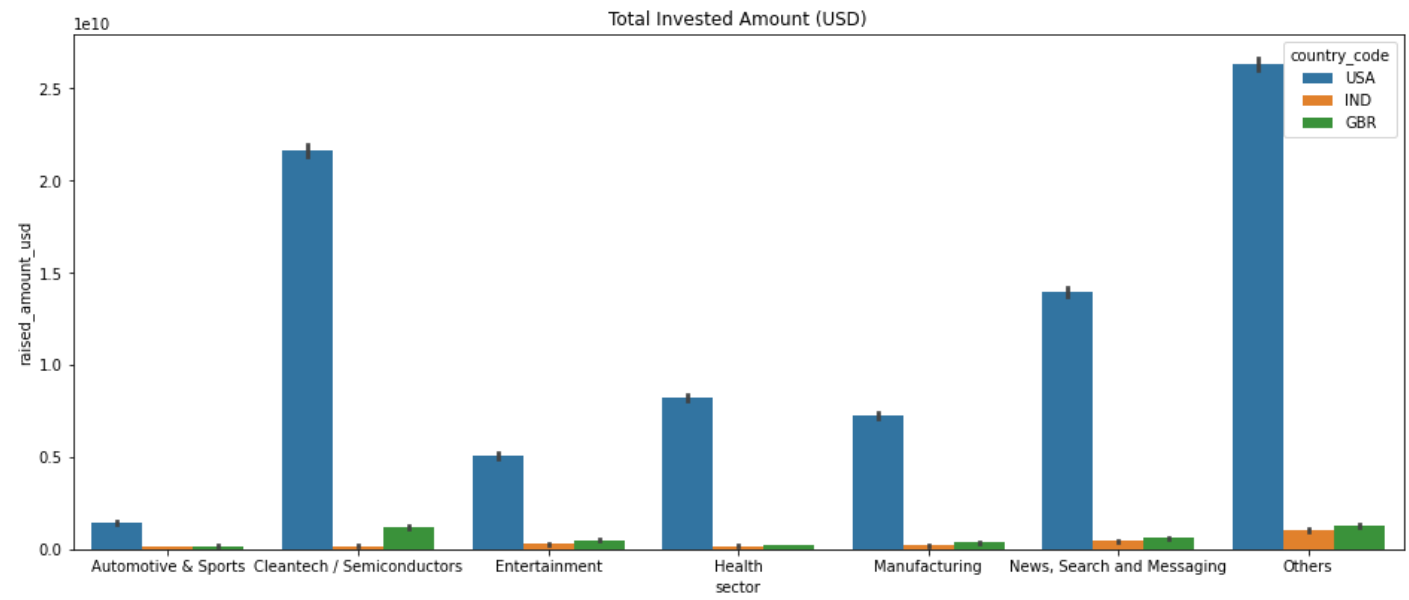


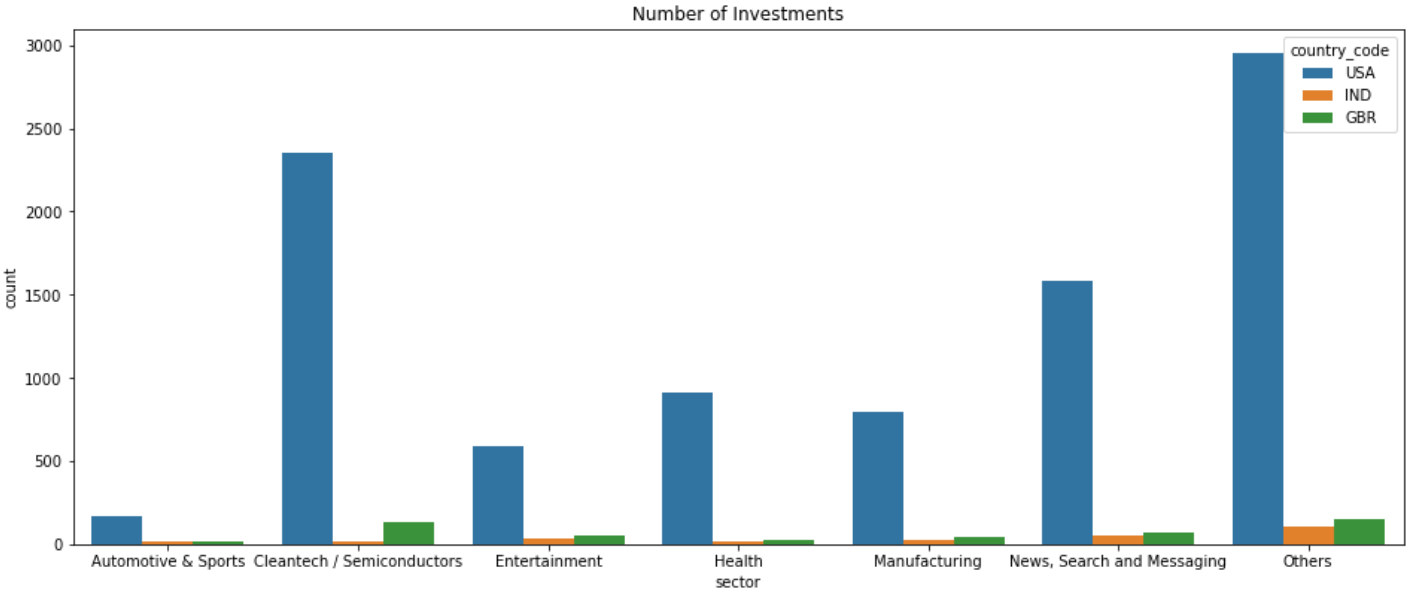
In [409…

```
import numpy as np
plt.figure(figsize=(16, 14))

plt.subplot(2, 1, 1)
p=sns.barplot(data = df_final,x=df_final['sector'],y=df_final['raised_amount_usd'],hue =df
#p.set_xticklabels(p.get_xticklabels(),rotation=30)
plt.title('Total Invested Amount (USD)')

plt.subplot(2, 1, 2)
q = sns.countplot(x='sector', hue='country_code', data=df_final)
#q.set_xticklabels(q.get_xticklabels(),rotation=30)
plt.title('Number of Investments')
```

Out[409…

```
Text(0.5, 1.0, 'Number of Investments')
```





In [415…

```
#Total investment in each company in Others sector in IND in ascending order
df_final[df_final['sector']=='Others'].groupby(['permalink'])['raised_amount_usd'].sum().s
```

Out[415…

```
permalink
/organization/seed-2                   5000000.0
/organization/m-six                    5000000.0
/organization/pancetera                5000000.0
/organization/lumeta                   5000000.0
/organization/vriti-infocom            5000000.0
                                            ...
/organization/black-duck-software     51000000.0
/organization/decarta                 52100000.0
/organization/airtight-networks       54201907.0
/organization/capella                 54968051.0
/organization/virtustream             64300000.0
Name: raised_amount_usd, Length: 2257, dtype: float64
```

Observations:

Thus, the top country in terms of the number of investments (and the total amount invested) is the USA. The sectors 'Others', 'Social, Finance, Analytics and Advertising' and 'Cleantech/Semiconductors' are the most heavily invested ones.

In case you don't want to consider 'Others' as a sector, 'News, Search and Messaging' is the next best sector.

In [ ]: