

Startup - 2021 Data Analysis

From a Financial Newspaper

The government has recognised 41,061 startups as of December 23, 2020, according to the Economic Survey 2020-21 tabled in Parliament on Jan 29, 2021. Of this, more than 39,000 startups have reported 4,70,000 jobs, India currently houses the world's third largest startup ecosystem, with 38 firms being valued at over 1 billion dollars,

the government has taken several measures to support startups, including broadening the definition of startups, simplifying regulations, providing income tax exemptions and setting up a Rs 10,000 crore Fund of Funds for startups operated by the Small Industries Development Bank of India (Sidbi).

importing the necessary libraries

In [17]:

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline
```

Load data

In [18]:

```
#read the file
df=pd.read_csv(r"C:\Desktop\Data Analyst Project\startup 21\2021_registered_companies.csv")
```

In [19]:

```
#data information
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 54894 entries, 0 to 54893
Data columns (total 12 columns):
#   Column                Non-Null Count  Dtype
---  -
0   company_uid            54894 non-null  object
1   date_of_registration   54894 non-null  object
2   month_name             54894 non-null  object
3   state                  54894 non-null  object
4   roc                    54894 non-null  object
5   category               54894 non-null  object
6   class                  54894 non-null  object
7   company_type           54894 non-null  object
8   authorized_capital     54894 non-null  float64
9   paidup_capital         54894 non-null  float64
10  activity_code           54894 non-null  float64
11  activity_description    54894 non-null  object
dtypes: float64(3), object(9)
memory usage: 5.0+ MB
```

here we have 12 columns and 58893 rows in the dataset ,which contains 9 object datatype and 3 float data type

Column name meaning xplanation:

this dataset consists fo all the registered companies in period of January-21 to April-21

company_uid -a unique id given to every comapany registered.

date_of_registration - Date on which the company was registered.

month_name -Month on which the company was registered.

State -State in which the company was registered.

roc - The Registrar of Companies (ROC) is an office under the Ministry of Corporate Affairs (MCA), which is the body that deals with the administration of companies and Limited Liability Partnerships in India. Basically this column contains information about the city of ROC.

Category -Defines the Category of the company.

class -Defines class of a comapny.

comapny_type -define type of the comapny.

activity_description - defines what business the comapny is into.

In []:

In [20]:

```
df.describe()
```

Out[20]:

	authorized_capital	paidup_capital	activity_code
count	5.489400e+04	5.489400e+04	54894.000000
mean	1.617091e+06	5.860709e+05	52940.032718
std	5.226937e+07	1.725209e+07	26188.280366
min	0.000000e+00	0.000000e+00	1100.000000
25%	1.000000e+05	1.000000e+05	29308.000000
50%	5.000000e+05	1.000000e+05	60221.000000
75%	1.000000e+06	1.000000e+05	74300.000000
max	1.000000e+10	2.600000e+09	99000.000000

In [21]:

```
##checking the number of null values in each column  
df.isna().sum()
```

Out[21]:

```
company_uid          0  
date_of_registration 0  
month_name           0  
state                0  
roc                  0  
category             0  
class                0  
company_type         0  
authorized_capital    0  
paidup_capital        0  
activity_code         0  
activity_description  0  
dtype: int64
```

we can see here there is no null values in the data,

In [22]:

```
df.shape
```

Out[22]:

```
(54894, 12)
```

Changing the data type of columns from float to integer type

In [23]:

```
df['authorized_capital']=df['authorized_capital'].astype('int')
df['paidup_capital']=df['paidup_capital'].astype('int')
df['activity_code']=df['activity_code'].astype('int')
```

here we have successfully converted columns authorized capital & paidup capital to float to integer data type

In [24]:

```
df.columns
```

Out[24]:

```
Index(['company_uid', 'date_of_registration', 'month_name', 'state', 'ro  
c',  
      'category', 'class', 'company_type', 'authorized_capital',  
      'paidup_capital', 'activity_code', 'activity_description'],  
      dtype='object')
```

Removing the extra columns from data

In [25]:

```
df.drop('activity_code',axis=1,inplace=True)
```

In [26]:

```
#display top 10 records  
df.head(10)
```

Out[26]:

	company_uid	date_of_registration	month_name	state	roc	category	class
0	b928b473-d6d6-4b92-a63d-d262bf30c24b	19/01/21	Jan-21	Telangana	RoC-Hyderabad	Company limited by Shares	Private
1	9696190f-7478-4d61-81f1-fc4eb08cce13	25/01/21	Jan-21	Tamil Nadu	RoC-Chennai	Company limited by Shares	Private
2	59de4ca1-c15f-4d6d-8de3-03319793c884	14/01/21	Jan-21	Maharashtra	RoC-Pune	Company limited by Shares	Public
3	e28b4caa-b7f9-4faf-8bbe-b12028177417	25/01/21	Jan-21	Telangana	RoC-Hyderabad	Company limited by Shares	Private
4	e1861c6f-c2a0-4a89-81e9-b39f11b1b6c6	19/01/21	Jan-21	Jammu & Kashmir	RoC-Jammu	Company limited by Shares	Private
5	a8361638-b61b-4fbd-80ed-1be9dcf0c67a	30/01/21	Jan-21	Karnataka	RoC-Bangalore	Company limited by Shares	Private
6	857f6c55-dd51-4dd0-b8a9-303021afc90a	25/01/21	Jan-21	Maharashtra	RoC-Pune	Company limited by Shares	Private
7	37902446-b5a0-452a-8fd0-e486585fb0fd	04/01/21	Jan-21	Uttar Pradesh	RoC-Kanpur	Company limited by Shares	Private
8	17c46288-bed9-4a52-a63d-52ba8be9e40d	12/01/21	Jan-21	Maharashtra	RoC-Pune	Company limited by Shares	Private
9	8cff820-5e03-4741-929d-ee6b8c1f2e0f	08/01/21	Jan-21	Maharashtra	RoC-Mumbai	Company limited by Shares	Private



How many companies were registered each in months of 2021

In [27]:

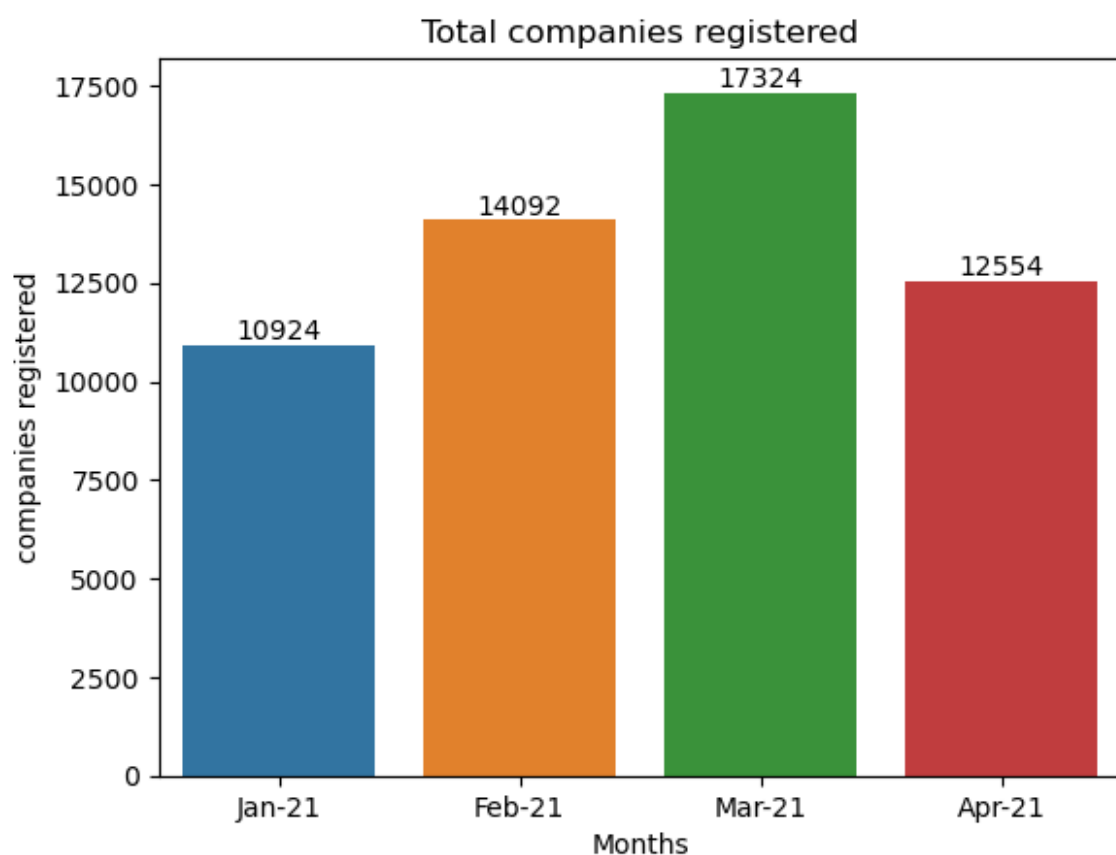
```
#group months  
df.groupby('month_name')['company_uid'].count()
```

Out[27]:

```
month_name  
Apr-21    12554  
Feb-21    14092  
Jan-21    10924  
Mar-21    17324  
Name: company_uid, dtype: int64
```

In [28]:

```
ax=sns.countplot(data=df,x='month_name')  
plt.title('Total companies registered')  
plt.xlabel('Months')  
plt.ylabel('companies registered')  
for bars in ax.containers:  
    ax.bar_label(bars)  
plt.show()
```



we can see here highest registration done on march month 17324, and after this feb, apr and january months the number registration are 14092, 12554 and 10924 respectively

Number of companies registered in each states

In [29]:

```
df.groupby('state')['company_uid'].count().keys()
```

Out[29]:

```
Index(['AN', 'AP', 'AR', 'AS', 'Andaman & Nicobar', 'Andhra Pradesh',  
      'Arunachal Pradesh', 'Assam', 'BR', 'Bihar', 'CH', 'CT', 'Chandigar  
h',  
      'Chattisgarh', 'DL', 'DN', 'Dadra & Nagar Haveli', 'Daman and Diu',  
      'Delhi', 'GA', 'GJ', 'Goa', 'Gujarat', 'HP', 'HR', 'Haryana',  
      'Himachal Pradesh', 'JH', 'JK', 'Jammu & Kashmir', 'Jharkhand', 'K  
A',  
      'KL', 'Karnataka', 'Kerala', 'LD', 'LH', 'Ladakh', 'Lakshadweep',  
      'MH',  
      'ML', 'MN', 'MP', 'MZ', 'Madhya Pradesh', 'Maharashtra', 'Manipur',  
      'Meghalaya', 'Mizoram', 'NL', 'Nagaland', 'OR', 'Orissa', 'PB', 'P  
Y',  
      'Pondicherry', 'Punjab', 'RJ', 'Rajasthan', 'TG', 'TN', 'TR',  
      'Tamil Nadu', 'Telangana', 'Tripura', 'UP', 'UR', 'Uttar Pradesh',  
      'Uttarakhand', 'WB', 'West Bengal'],  
      dtype='object', name='state')
```

as the state names are present in short form and full name first we need to change everyone in sigle format

In [30]:

```
#Denoting short form into Long form
```

```
short = {'MH': 'Maharashtra',  
, 'TG': 'Telangana',  
, 'GJ': 'Gujarat',  
, 'CH': 'Chandigarh',  
, 'DL': 'Delhi',  
, 'HR': 'Haryana',  
, 'UP': 'Uttar Pradesh',  
, 'RJ': 'Rajasthan',  
, 'CT': 'Chattisgarh',  
, 'KL': 'Kerala',  
, 'WB': 'West Bengal',  
, 'KA': 'Karnataka',  
, 'MP': 'Madhya Pradesh',  
, 'PB': 'Punjab',  
, 'BR': 'Bihar',  
, 'MN': 'Manipur',  
, 'TN': 'Tamil Nadu',  
, 'OR': 'Orissa',  
, 'HP': 'Himachal Pradesh',  
, 'UR': 'Uttarakhand',  
, 'JH': 'Jharkhand',  
, 'AP': 'Andhra Pradesh',  
, 'GA': 'Goa',  
, 'AS': 'Assam',  
, 'DN': 'Dadra & Nagar Haveli',  
, 'TR': 'Tripura',  
, 'JK': 'Jammu & Kashmir',  
, 'PY': 'Pondicherry',  
, 'MZ': 'Mizoram',  
, 'NL': 'Nagaland',  
, 'AN': 'Andaman & Nicobar',  
, 'AR': 'Arunachal Pradesh',  
, 'LD': 'Lakshadweep',  
, 'ML': 'Meghalaya',  
, 'LH': 'Jammu & Kashmir',  
, 'DD': 'Daman and Diu',  
}
```

In [31]:

```
#here merging the data
```

```
df.replace({'state':short},inplace=True)
```


In [32]:

```
s=df['state'].value_counts().sort_values(ascending=False)
s
```

Out[32]:

Maharashtra	10077
Uttar Pradesh	5772
Delhi	5449
Karnataka	4213
Telangana	3655
Tamil Nadu	3515
Gujarat	3057
Haryana	2705
West Bengal	2551
Kerala	2084
Bihar	1997
Rajasthan	1979
Madhya Pradesh	1578
Andhra Pradesh	1184
Orissa	1110
Punjab	771
Jharkhand	647
Assam	440
Chattisgarh	437
Uttarakhand	432
Jammu & Kashmir	332
Himachal Pradesh	244
Chandigarh	220
Goa	153
Manipur	83
Pondicherry	49
Tripura	39
Arunachal Pradesh	23
Nagaland	21
Dadra & Nagar Haveli	18
Meghalaya	18
Andaman & Nicobar	15
Daman and Diu	11
Mizoram	10
Lakshadweep	3
Ladakh	2

Name: state, dtype: int64

In [33]:

```
plt.figure(figsize=(8,20))
ax=sns.countplot(data=df,y='state')
for bars in ax.containers:
    ax.bar_label(bars)
plt.xlabel('number of startups')
```

Out[33]:

Text(0.5, 0, 'number of startups')

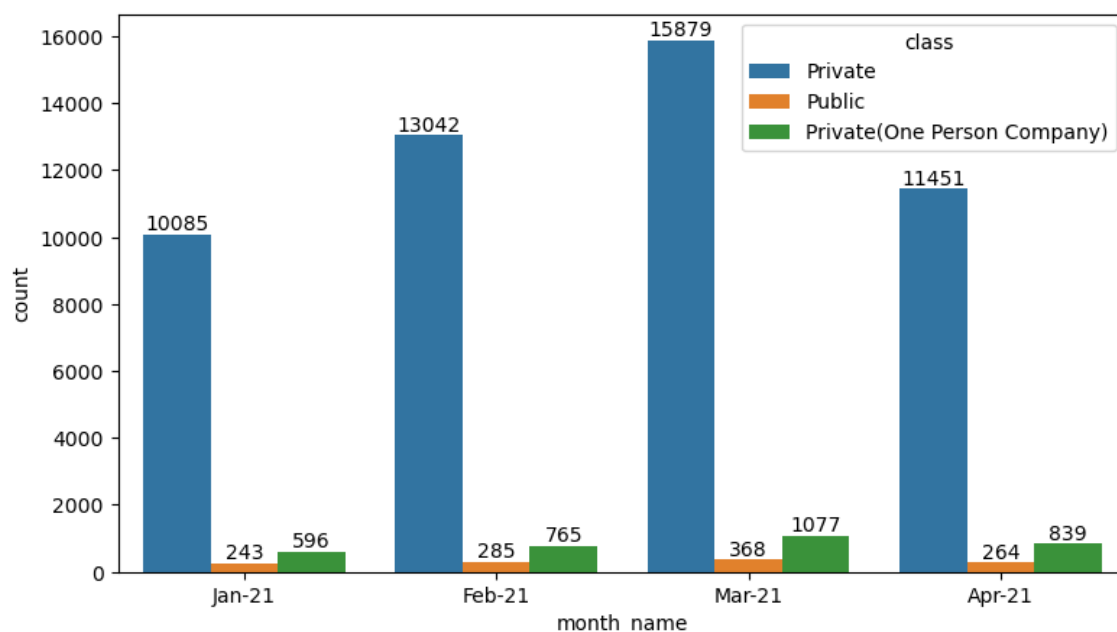
we can ,see here most startup registered state is maharashtra,with 10077 startups and in second place uttarpradesh with 5772 startups, Delhi,Karnataka & Telangana are from third to fifth place with 3655 to 5449

In []:

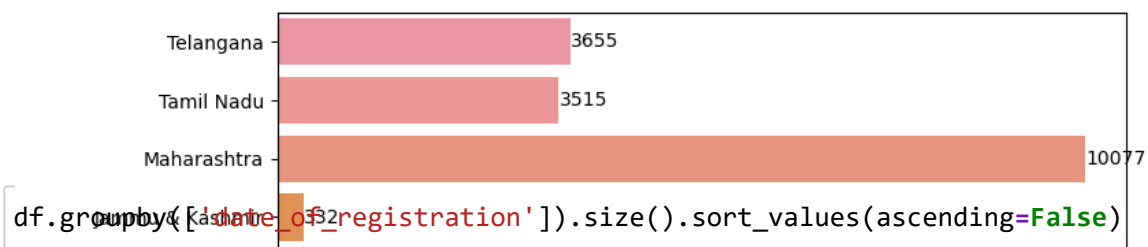
Explore the number of registration of each months based on class

In [34]:

```
plt.figure(figsize=(9,5))
ax=sns.countplot(data=df,x='month_name',hue='class')
for bars in ax.containers:
    ax.bar_label(bars)
```



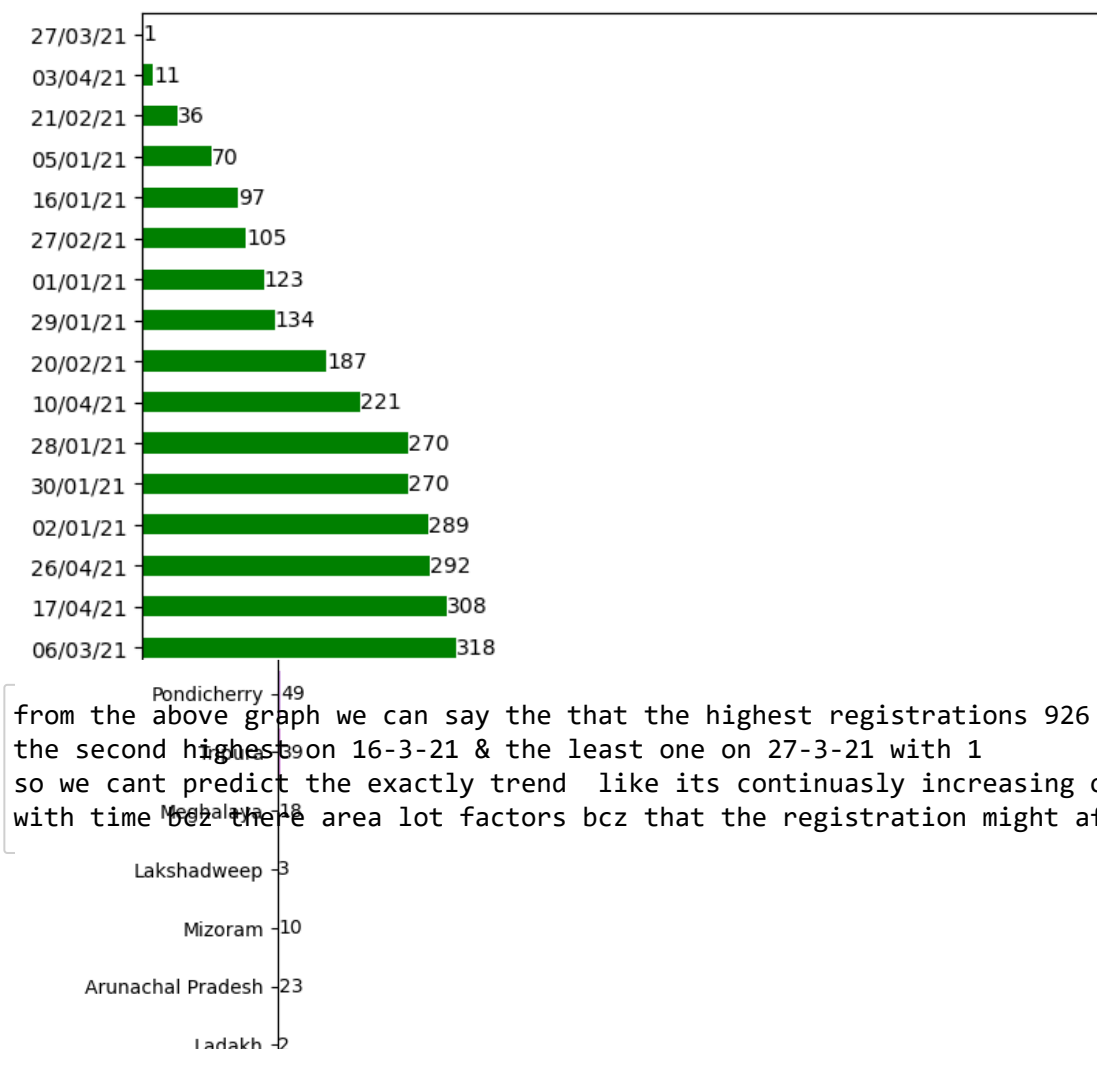
as showed in above graph, the startup type from jan-21 to apr-21, the classtype not increasing steadily with months but still the private startup values are much higher than the other two's in each month, the highest startups private 15879 ,public 368 and private (one person company) 1077 startups are on march . and the least one on jan -21 with private 10085 ,public 243 and private (one person company) 596 startups



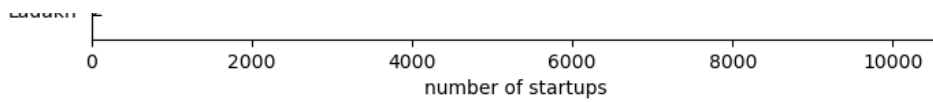
```
df.groupby(['date_of_registration']).size().sort_values(ascending=False)

Out[35]:
date_of_registration
08/02/21    926
16/03/21    925
10/02/21    912
03/03/21    877
23/03/21    870
...
16/01/21     97
05/01/21     70
21/02/21     36
03/04/21     11
27/03/21      1
Length: 99, dtype: int64

plt.figure(figsize=(8,34))
ax=df['date_of_registration'].value_counts().plot(kind='barh',color='green')
for bars in ax.containers:
    ax.bar_label(bars)
```

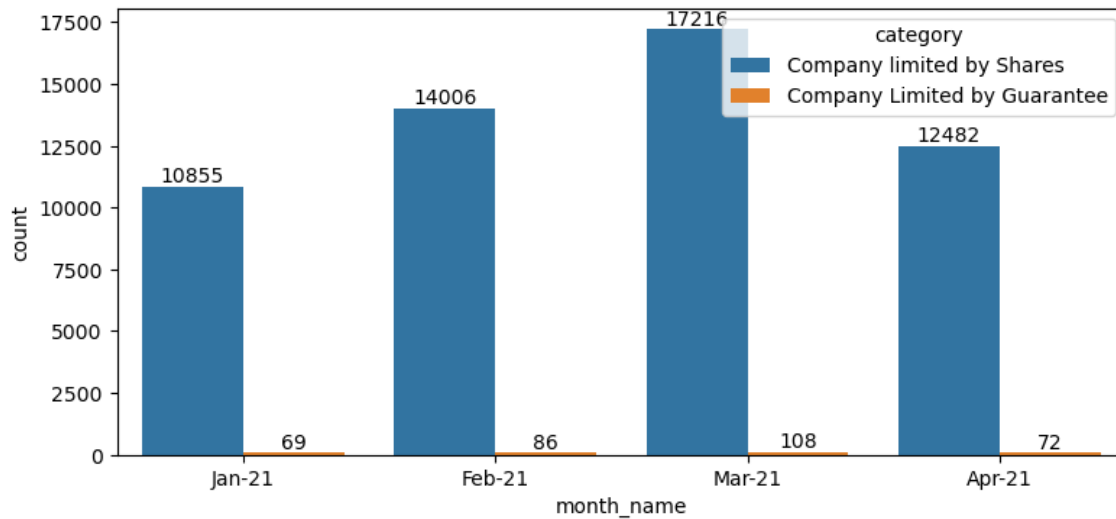


from the above graph we can say the that the highest registrations 926 were on 8-2-21, the second highest on 16-3-21 & the least one on 27-3-21 with 1 so we cant predict the exactly trend like its continasly increasing or decreasing with time bcz there area lot factors bcz that the registration might affect



In [37]:

```
plt.figure(figsize=(9,4))
ax=sns.countplot(data=df,x='month_name',hue='category')
for bars in ax.containers:
    ax.bar_label(bars)
```

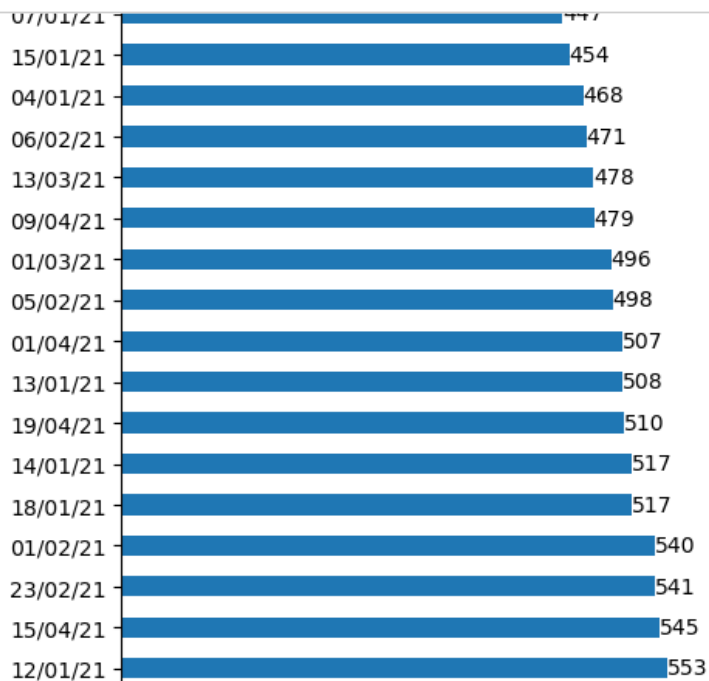


here we can see march month have highest registration in both category company limited by shares and company limited by guarantee, now coming to numerical term , in march company limited by share is 17216 and company limited by guarantee is 108 respectively

what is the Number of registration per day

In [38]:

```
plt.figure(figsize=(9,7))
plt.figure(figsize=(8,34))
ax=df['date_of_registration'].value_counts().plot(kind='barh')
for bars in ax.containers:
    ax.bar_label(bars)
```



Determine the Registration by company type?

In [39]:

```
com=df['company_type'].value_counts()
com
```

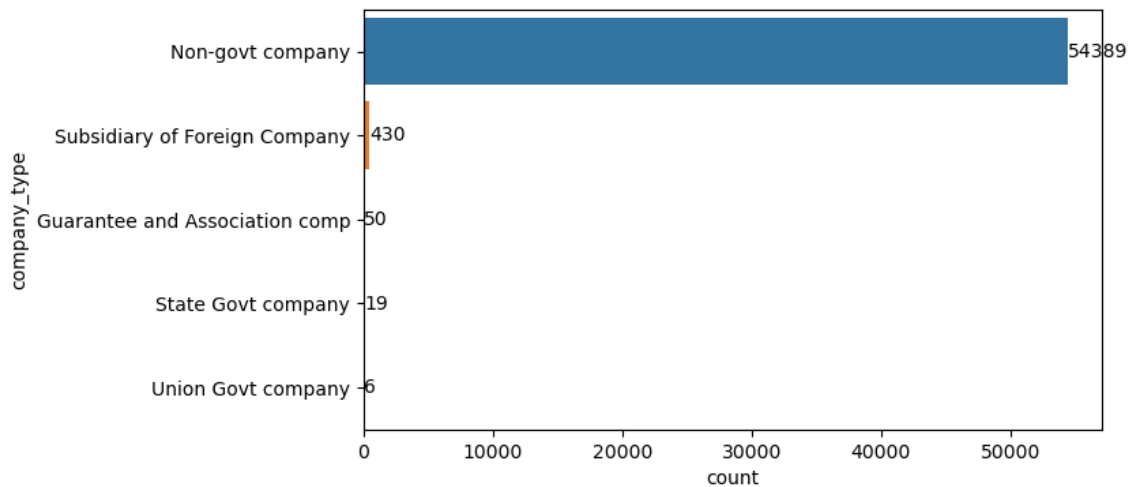
Out[39]:

Non-govt company	54389
Subsidiary of Foreign Company	430
Guarantee and Association comp	50
State Govt company	19
Union Govt company	6

Name: company_type, dtype: int64

In [40]:

```
plt.figure(figsize=(7,4))
ax=sns.countplot(data=df,y='company_type')
for bars in ax.containers:
    ax.bar_label(bars)
```



here we can see non govt companies have highest registration 54389 and after this subsidiary of foreign companies have 430 registration and with 6 registrations union govt company in least registration

determine monthly registrations of non government companies

In [41]:

```
df[df['company_type']=='Non-govt company']['month_name'].value_counts()
```

Out[41]:

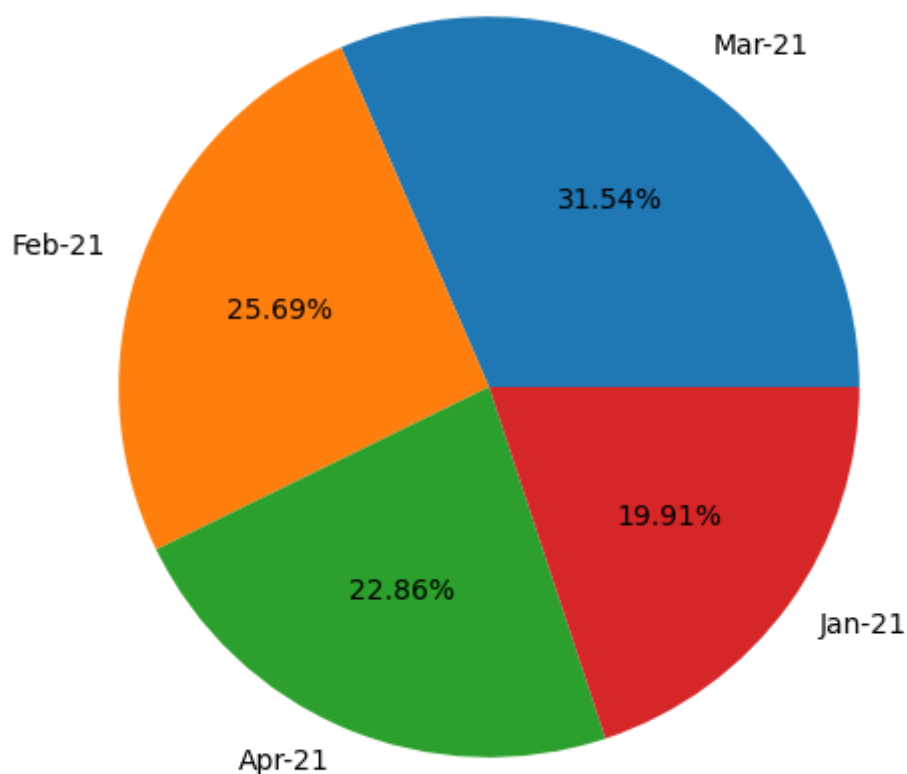
```
Mar-21    17155
Feb-21    13972
Apr-21    12432
Jan-21    10830
Name: month_name, dtype: int64
```

In [42]:

```
ngo=df[df['company_type']=='Non-govt company']
month=ngo['month_name'].value_counts()
plt.figure(figsize=(6,6))
plt.pie(month,autopct='%.2f%%',labels=month.index)
```

Out[42]:

```
([<matplotlib.patches.Wedge at 0x224b1a1ec70>,
 <matplotlib.patches.Wedge at 0x224b1a477c0>,
 <matplotlib.patches.Wedge at 0x224b1a47d30>,
 <matplotlib.patches.Wedge at 0x224b192c220>],
 [Text(0.6027315390792677, 0.9201710122575789, 'Mar-21'),
 Text(-1.0322689385490431, 0.38002741809878937, 'Feb-21'),
 Text(-0.4267488759128541, -1.0138468310879685, 'Apr-21'),
 Text(0.891700520459553, -0.6441041700006004, 'Jan-21')],
 [Text(0.32876265767960056, 0.5019114612314066, '31.54%'),
 Text(-0.5630557846631143, 0.20728768259933963, '25.69%'),
 Text(-0.23277211413428403, -0.5530073624116191, '22.86%'),
 Text(0.486382102068847, -0.3513295472730547, '19.91%')])
```



we can see here in graph highest number of registrations is 17155 from march. and then 13972,12432 and 10830 from from feb ,april and january respectively

Registraion Distribution in jan month

In [43]:

```
df[df['month_name']=='Jan-21']['company_type'].value_counts()
```

Out[43]:

Non-govt company	10830
Subsidiary of Foreign Company	80
Guarantee and Association comp	9
State Govt company	5

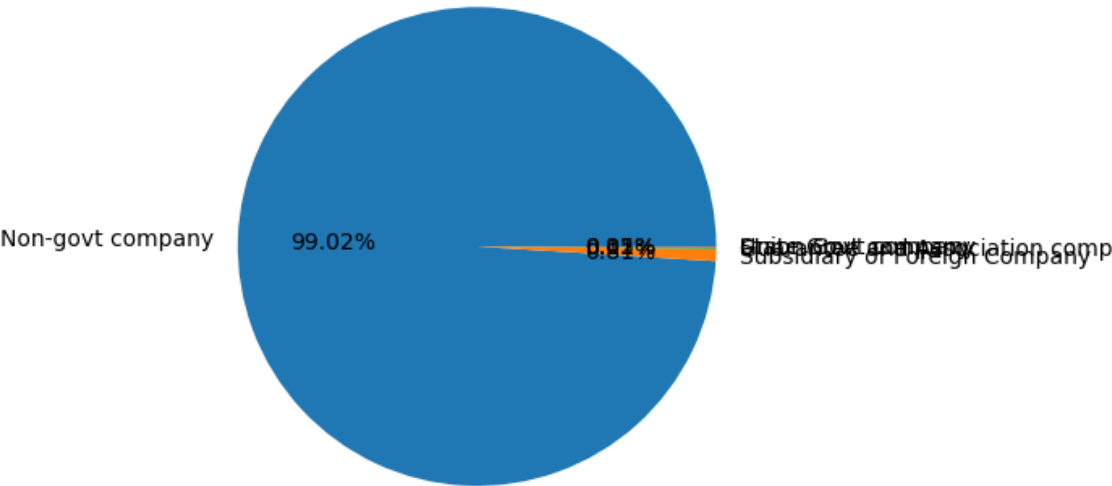
Name: company_type, dtype: int64

In [44]:

```
jan=df[df['month_name']=='Mar-21']
company=jan['company_type'].value_counts()
plt.pie(company,autopct='%.02f%%',labels=company.index)
```

Out[44]:

```
([<matplotlib.patches.Wedge at 0x224b19b56a0>,
<matplotlib.patches.Wedge at 0x224b19b5dc0>,
<matplotlib.patches.Wedge at 0x224b19af520>,
<matplotlib.patches.Wedge at 0x224b19afc40>,
<matplotlib.patches.Wedge at 0x224b19943a0>],
[Text(-1.0994834580624422, 0.03370646001961726, 'Non-govt company'),
Text(1.0992909938793063, -0.03948810929694213, 'Subsidiary of Foreign Co
mpany'),
Text(1.0999738824370378, -0.007580102663568347, 'Guarantee and Associati
on comp'),
Text(1.0999981912851713, -0.0019947855403939967, 'State Govt company'),
Text(1.0999999819112873, -0.00019948726149442196, 'Union Govt compan
y')],
[Text(-0.5997182498522412, 0.01838534182888214, '99.02%'),
Text(0.5996132693887124, -0.021538968707422976, '0.81%'),
Text(0.599985754056566, -0.004134601452855461, '0.12%'),
Text(0.5999990134282751, -0.0010880648402149071, '0.05%'),
Text(0.5999999901334294, -0.00010881123354241197, '0.01%')])
```



we can see here the highest number of registration in january month is 10830 from Non govt companies and then from subsidiary of foriegn companies from

number of registration from state govt companies

In [45]:

```
df[df['company_type']=='State Govt company']['state'].value_counts()
```

Out[45]:

Orissa	4
Tamil Nadu	4
Karnataka	2
Kerala	2
Rajasthan	1
Bihar	1
Andhra Pradesh	1
Gujarat	1
Meghalaya	1
Maharashtra	1
Uttar Pradesh	1

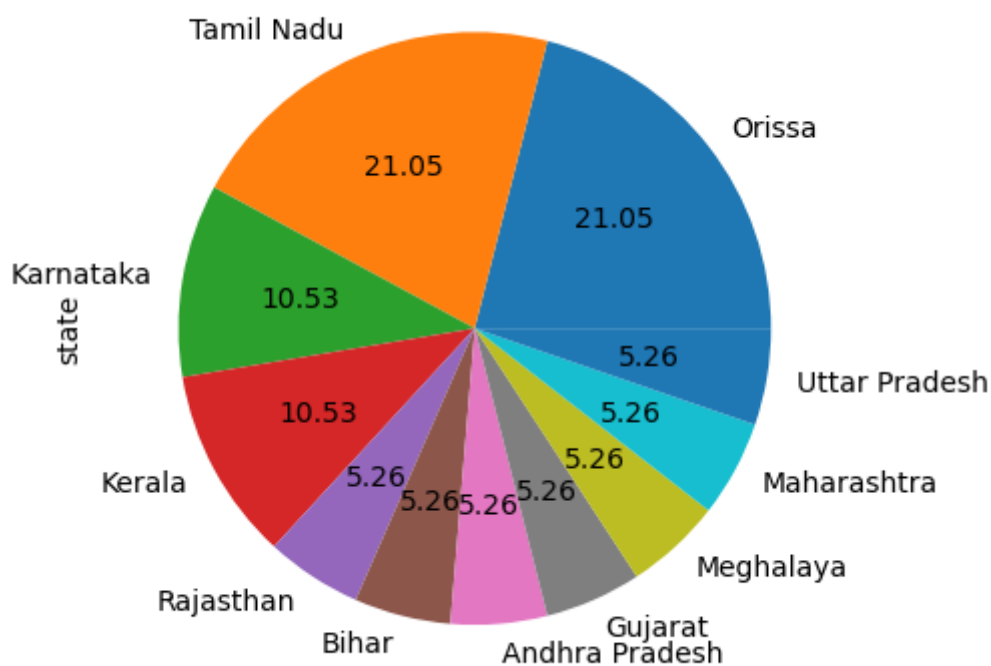
Name: state, dtype: int64

In [46]:

```
df[df['company_type']=='State Govt company']['state'].value_counts().plot(kind='pie',aut
```

Out[46]:

<AxesSubplot:ylabel='state'>



as we can see here orrisa and tamilnadu have 4,4 registration and karnataka & kerala have 2,2 ,after this remaining have 1 ,1 1, 1 .

which states have most foriegn companies

In [47]:

```
df[df['company_type']=='Subsidiary of Foreign Company']['state'].value_counts()
```

Out[47]:

Karnataka	105
Maharashtra	88
Delhi	62
Tamil Nadu	50
Telangana	33
Haryana	20
Uttar Pradesh	16
Gujarat	15
West Bengal	11
Andhra Pradesh	8
Kerala	6
Punjab	5
Chandigarh	2
Rajasthan	2
Jharkhand	1
Madhya Pradesh	1
Chattisgarh	1
Goa	1

as we can see here karnataka have 105 most of the foriegn company and after this Maharastra have 88 companies

Capital raised by foriegn company in india

In [48]:

```
df[df['company_type']=='Subsidiary of Foreign Company']['paidup_capital'].sum()
```

Out[48]:

4319932048

In [49]:

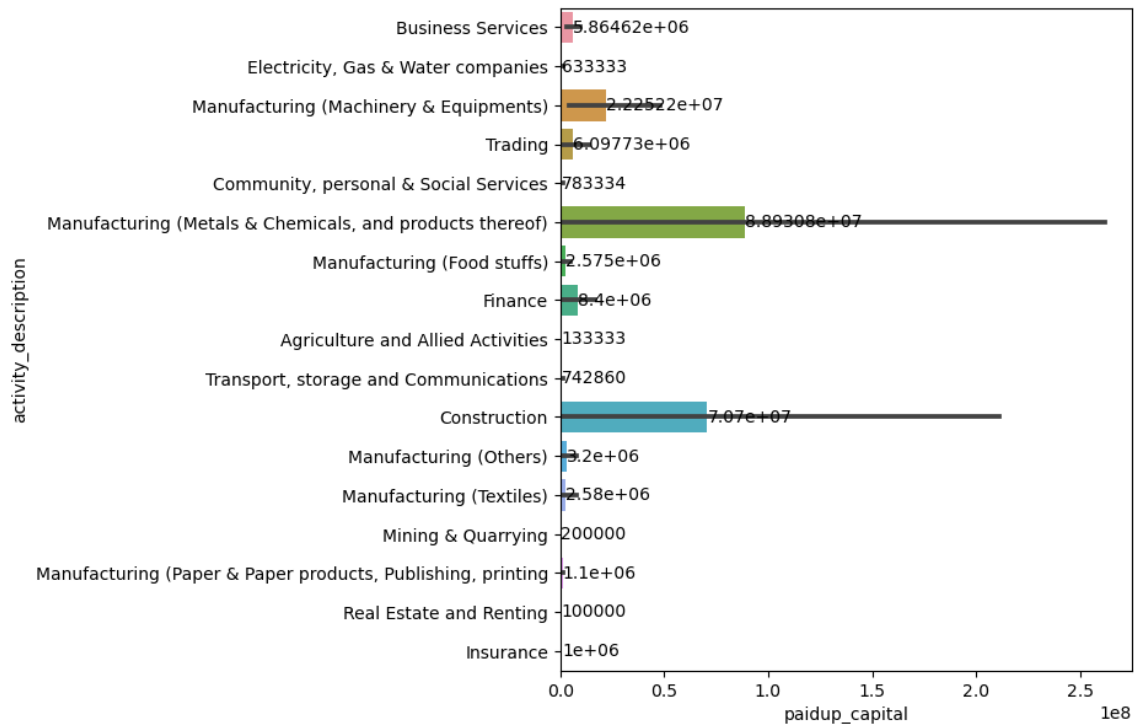
```
df.groupby(['company_type'])['paidup_capital'].sum()
```

Out[49]:

company_type	
Guarantee and Association comp	0.000000e+00
Non-govt company	1.863367e+10
State Govt company	1.556000e+08
Subsidiary of Foreign Company	4.319932e+09
Union Govt company	1.125000e+07
Name: paidup_capital, dtype: float64	

In [50]:

```
plt.figure(figsize=(6,7))  
  
ax=sns.barplot(data=df[df['company_type']=='Subsidiary of Foreign Company'],x='paidup_ca  
for bars in ax.containers:  
    ax.bar_label(bars)
```



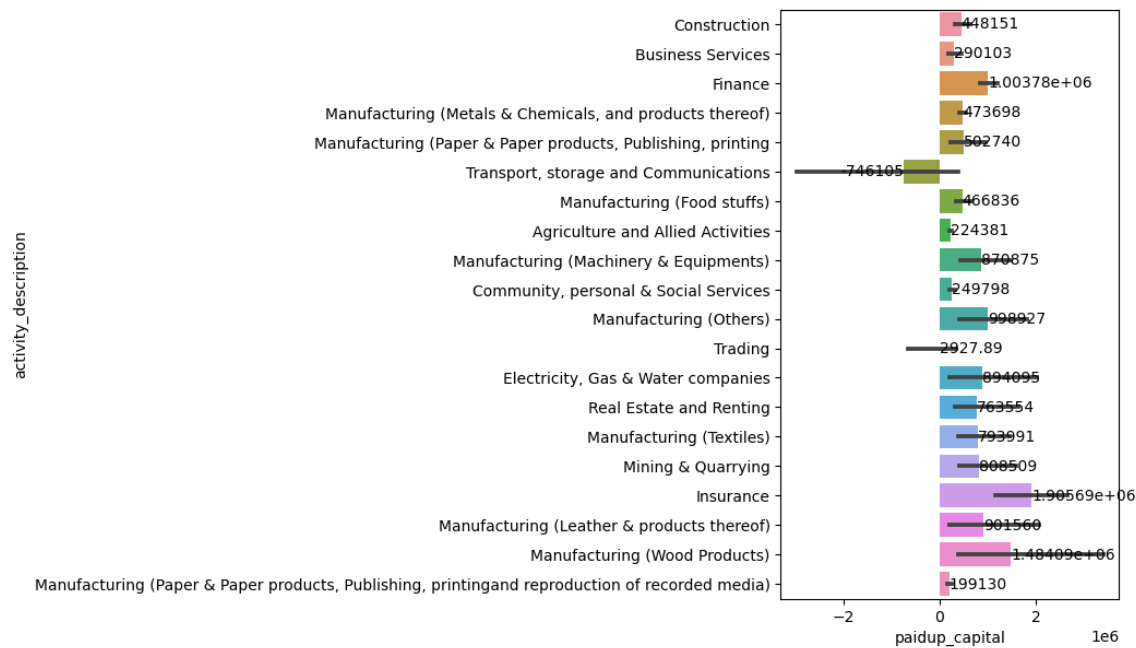
as we can see here non govt company have highest capital 10 billion but if we talk about foriegn companies they have 4 billion capital

distribution of non govt companies capital on different sectors

In [51]:

```
plt.figure(figsize=(4,7))

ax=sns.barplot(data=df[df['company_type']=='Non-govt company'],x='paidup_capital',y='act
for bars in ax.containers:
    ax.bar_label(bars)
```



which states have most guarantee and association companies

In [52]:

```
df[df['company_type']=='Guarantee and Association comp']['state'].value_counts()
```

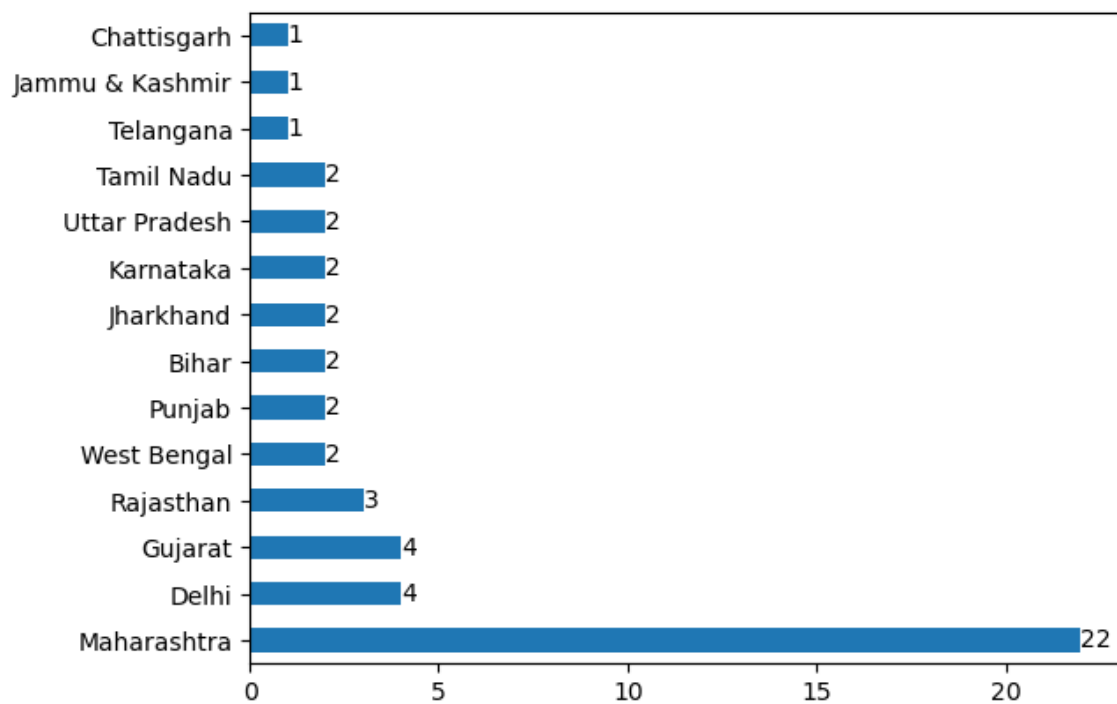
Out[52]:

Maharashtra	22
Delhi	4
Gujarat	4
Rajasthan	3
West Bengal	2
Punjab	2
Bihar	2
Jharkhand	2
Karnataka	2
Uttar Pradesh	2
Tamil Nadu	2
Telangana	1
Jammu & Kashmir	1
Chattisgarh	1

Name: state, dtype: int64

In [85]:

```
ax=df[df['company_type']=='Guarantee and Association comp']['state'].value_counts().plot  
for bars in ax.containers:  
    ax.bar_label(bars)
```



maharashtra have most of the guarantee and association companies

In [54]:

```
df['activity_description'].value_counts()
```

Out[54]:

```
Business Services
14805
Community, personal & Social Services
6937
Trading
6861
Agriculture and Allied Activities
4266
Manufacturing (Metals & Chemicals, and products thereof)
3842
Construction
3559
Manufacturing (Food stuffs)
2609
Manufacturing (Machinery & Equipments)
2560
Transport, storage and Communications
1977
Finance
1893
Real Estate and Renting
1848
Manufacturing (Textiles)
1158
Manufacturing (Others)
783
Electricity, Gas & Water companies
550
Manufacturing (Paper & Paper products, Publishing, printing
479
Mining & Quarrying
343
Manufacturing (Paper & Paper products, Publishing, printingand reproductio
n of recorded media)      154
Manufacturing (Leather & products thereof)
106
Manufacturing (Wood Products)
105
Insurance
59
Name: activity_description, dtype: int64
```

Registered companies of different sectors

In [55]:

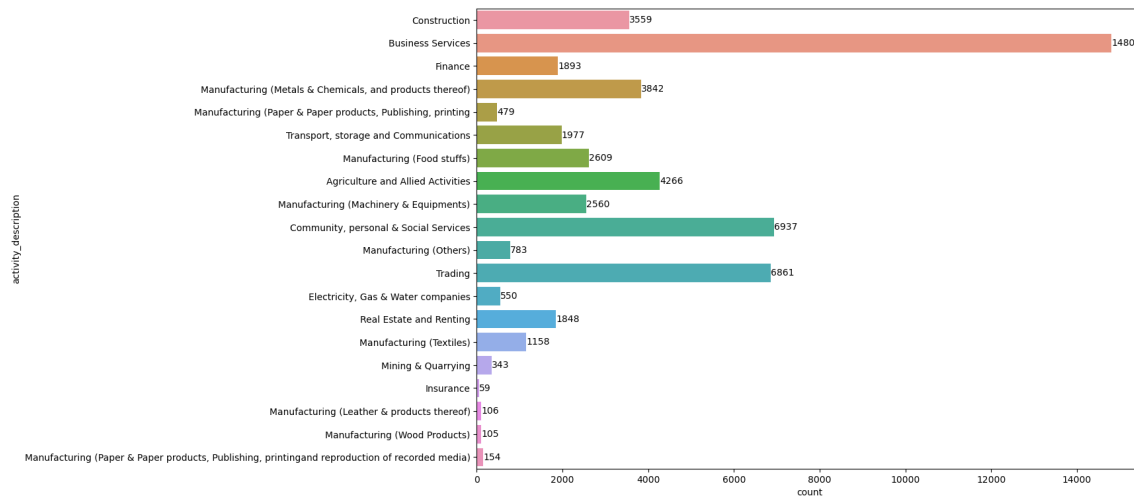
```
df['activity_description'].value_counts()
```

Out[55]:

```
Business Services
14805
Community, personal & Social Services
6937
Trading
6861
Agriculture and Allied Activities
4266
Manufacturing (Metals & Chemicals, and products thereof)
3842
Construction
3559
Manufacturing (Food stuffs)
2609
Manufacturing (Machinery & Equipments)
2560
Transport, storage and Communications
1977
Finance
1893
Real Estate and Renting
1848
Manufacturing (Textiles)
1158
Manufacturing (Others)
783
Electricity, Gas & Water companies
550
Manufacturing (Paper & Paper products, Publishing, printing
479
Mining & Quarrying
343
Manufacturing (Paper & Paper products, Publishing, printingand reproductio
n of recorded media)      154
Manufacturing (Leather & products thereof)
106
Manufacturing (Wood Products)
105
Insurance
59
Name: activity_description, dtype: int64
```


In [56]:

```
plt.figure(figsize=(13,9))
ax=sns.countplot(data=df,y='activity_description')
for bars in ax.containers:
    ax.bar_label(bars)
```



as we can see here higher number of companies were registered in business services which is around 15000 and in second place community personal and social services which is 7000 respectively

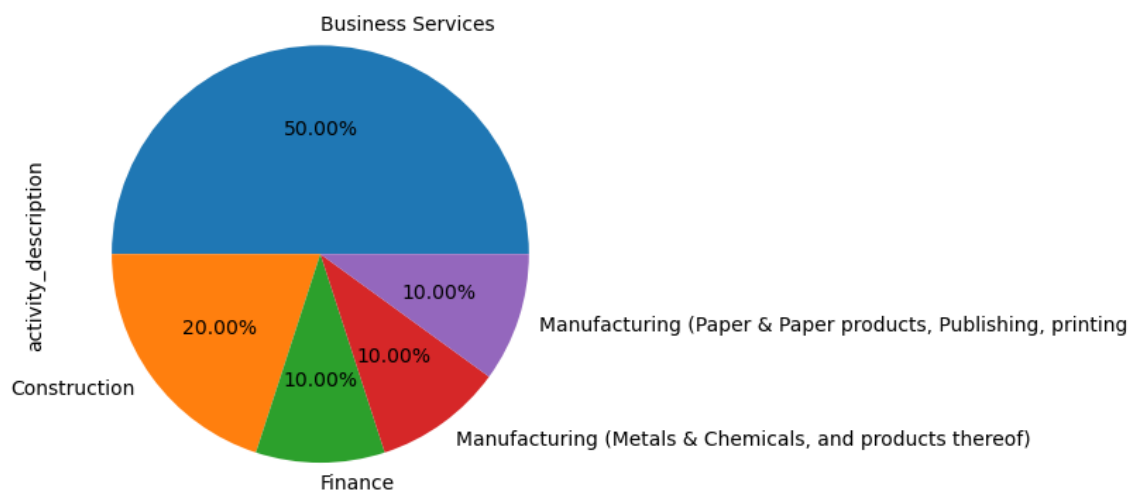
top 10 sectors with most companies registered

In [57]:

```
df['activity_description'][:10].value_counts().plot(kind='pie', autopct='%0.2f%%')
```

Out[57]:

<AxesSubplot:ylabel='activity_description'>



In which roc city higher number of registration done

In [58]:

```
df['roc'].value_counts()
```

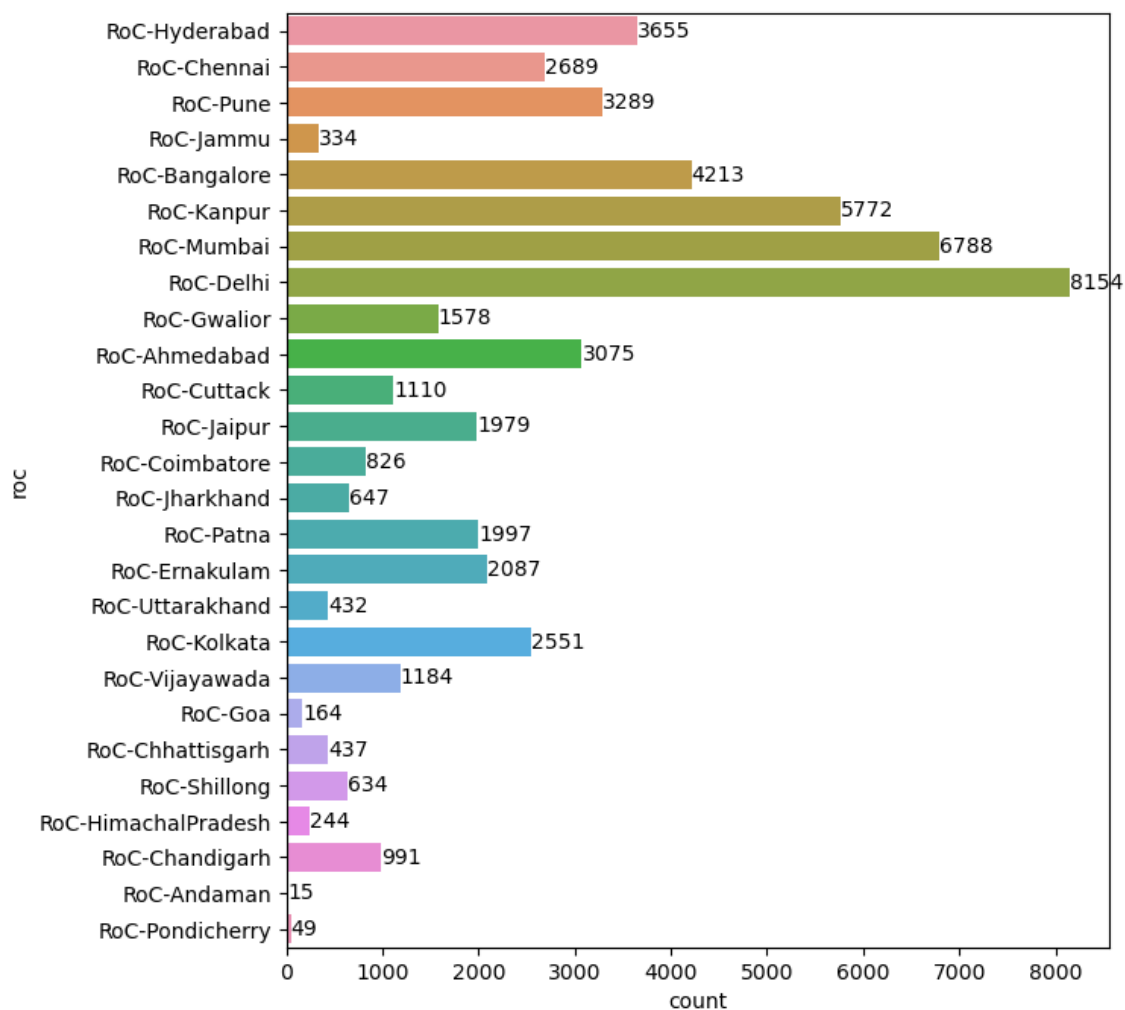
Out[58]:

RoC-Delhi	8154
RoC-Mumbai	6788
RoC-Kanpur	5772
RoC-Bangalore	4213
RoC-Hyderabad	3655
RoC-Pune	3289
RoC-Ahmedabad	3075
RoC-Chennai	2689
RoC-Kolkata	2551
RoC-Ernakulam	2087
RoC-Patna	1997
RoC-Jaipur	1979
RoC-Gwalior	1578
RoC-Vijayawada	1184
RoC-Cuttack	1110
RoC-Chandigarh	991
RoC-Coimbatore	826
RoC-Jharkhand	647
RoC-Shillong	634
RoC-Chhattisgarh	437
RoC-Uttarakhand	432
RoC-Jammu	334
RoC-HimachalPradesh	244
RoC-Goa	164
RoC-Pondicherry	49
RoC-Andaman	15

Name: roc, dtype: int64

In [59]:

```
plt.figure(figsize=(7,8))
ax=sns.countplot(data=df,y='roc')
for bars in ax.containers:
    ax.bar_label(bars)
```



we can see here delhi ios top place with 8154 and after this mumbai on second place with 6788 registrations

Registration based on class

In [60]:

```
df['class'].value_counts()
```

Out[60]:

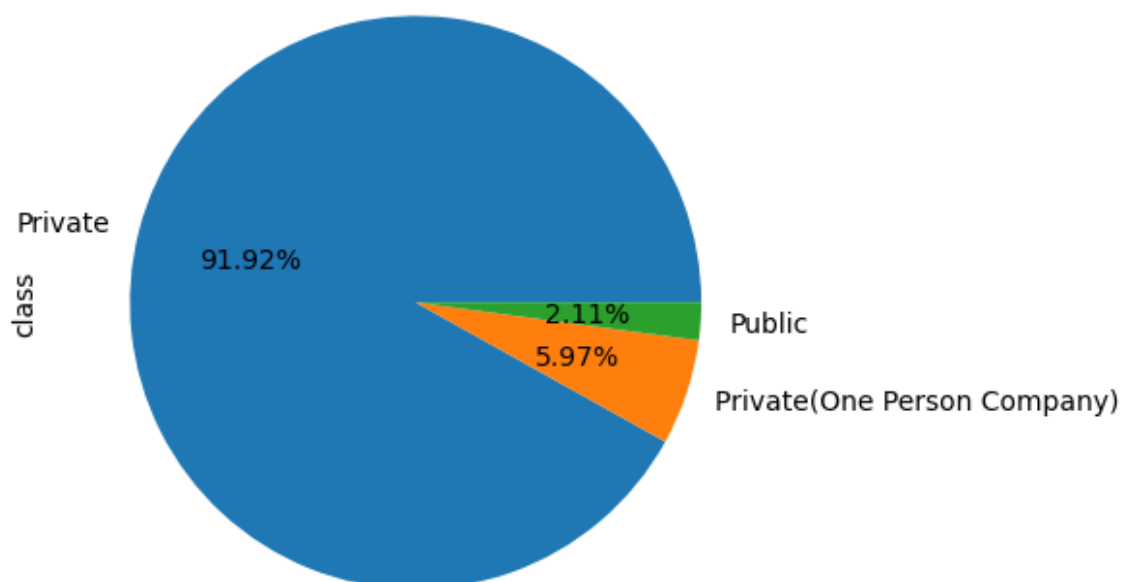
```
Private          50457
Private(One Person Company)  3277
Public           1160
Name: class, dtype: int64
```

In [61]:

```
df['class'].value_counts().plot(kind='pie', autopct='%0.2f%%')
```

Out[61]:

<AxesSubplot:ylabel='class'>



Registration based on company type

In [62]:

```
df['company_type'].value_counts()
```

Out[62]:

Non-govt company	54389
Subsidiary of Foreign Company	430
Guarantee and Association comp	50
State Govt company	19
Union Govt company	6

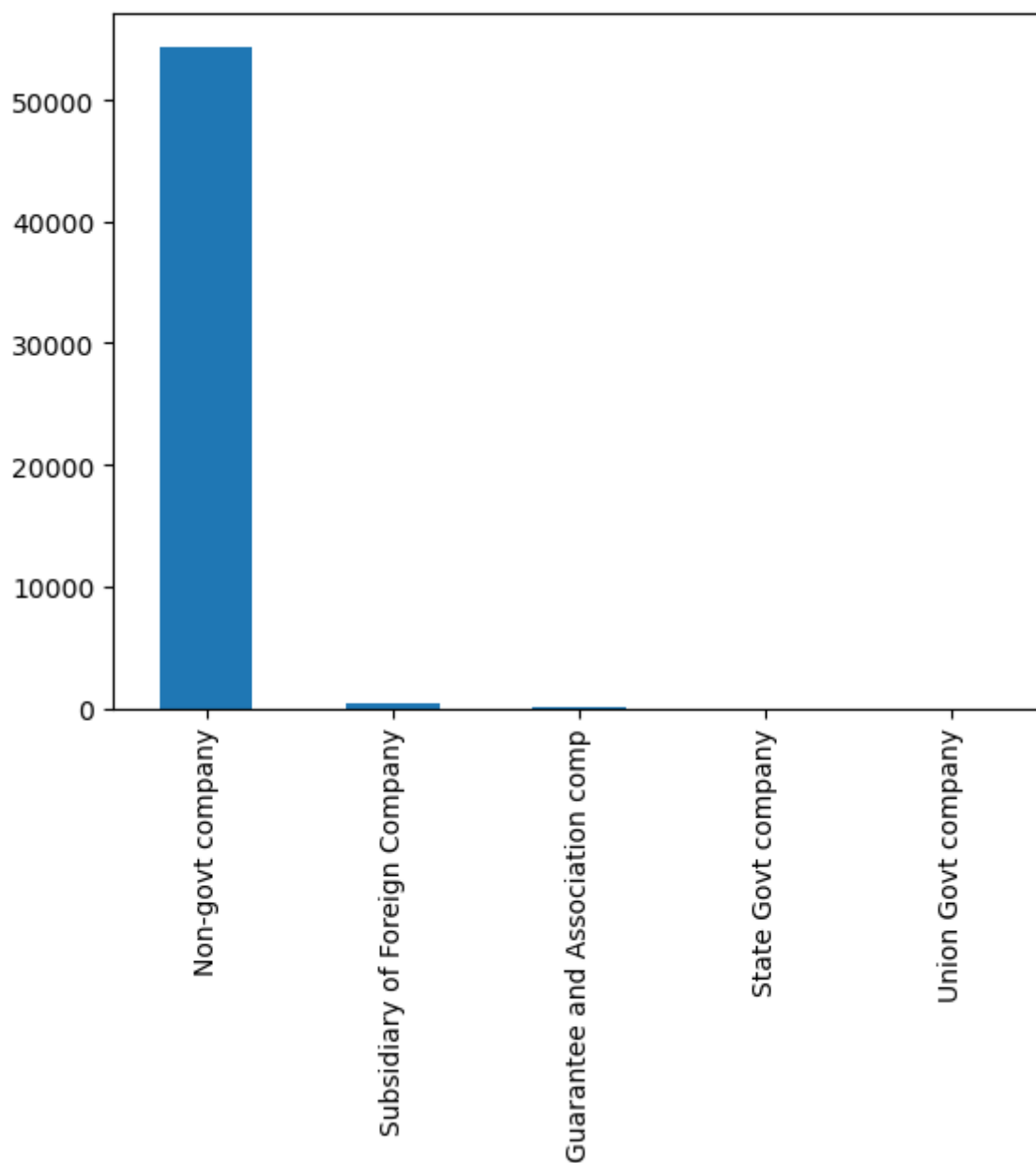
Name: company_type, dtype: int64

In [63]:

```
df['company_type'].value_counts().plot(kind='bar')
```

Out[63]:

<AxesSubplot:>



higher number of registrations in private type

In []:

In []:

In []: