In [3]:
```
# Importing pandas library for the data transforming
import pandas as pd
```

In [4]:
```
# Reading Json file into the dataframe
df = pd.read_json('assign_chubb_engineering.json')
```

In [5]:
```
df
```

Out[5]:

	altran technologies india pvt ltd	L&T Technology Services Ltd.	
company_uuid	None		fbd567c5e7395b4697
companies_they_hire_from	{'company_name': {'L&T Technology Services Ltd...	{'company_name': {'HCL Technologies': {'count'...	{'company_name': {'T
employees	{'count': 1, 'freshers_count': 0, 'freshers_uu...	{'count': 6890, 'freshers_count': 2155, 'fresh...	{'count': 1711, 'fres
companies_where_their_ex_employees_work	{'count': 0, 'company_name': {}}	{'count': 2192, 'company_name': {'Robert Bosch...	{'count': 733, 'compa
is_score_calculated	False	True	
Department_Tier-1_College_Density_Score	0.00001	0.004499	
Department_Tier-1_College_Freshers_Density_Score	0.00001	0.00464	

7 rows × 60126 columns

In [6]:
```
# Transposing the entire dataframe
df = df.T
```

In [7]:
```
# Reseting the dataframe index
df = df.reset_index()
```

In [8]:
```
df.rename(columns = {'index' : 'Company Name'},inplace = True)
```

In [9]:
```
# Getting column names in the dataframe before modifing the dataframe
df.columns
```

Out[9]:
```
Index(['Company Name', 'company_uuid', 'companies_they_hire_from', 'employees',
       'companies_where_their_ex_employees_work', 'is_score_calculated',
       'Department_Tier-1_College_Density_Score',
       'Department_Tier-1_College_Freshers_Density_Score'],
      dtype='object')
```

In [10]:
```
no_of_rows = len(df.index)
```

In [11]:
```
Employees_Count = []
Fresher_Count = []
IIT_Count = []
IIT_Fresher_Count = []
NIT_Count = []
NIT_Fresher_Count = []
Employee_count_they_hired = []
Employee_count_their_ex_employees_work = []
Tier_1_College_Strength = []
Tier_1_College_Freshers_Strength = []
for i in range(no_of_rows):

    # To count the no. of employees in each company
    no_of_employees = df['employees'][i]['count']
    Employees_Count.append(no_of_employees)

    # To count the no. of fresher employees in each company
    no_of_fresher_employees = df['employees'][i]['freshers_count']
    Fresher_Count.append(no_of_fresher_employees)

    # To count the no. of employees from IIT in each company
    no_of_employees_iit = df['employees'][i]['from_iit']['count']
    IIT_Count.append(no_of_employees_iit)

    # To count the no. of fresher employees from IIT in each company
    no_of_fresher_employees_iit = df['employees'][i]['from_iit']['freshers_count']
    IIT_Fresher_Count.append(no_of_fresher_employees_iit)

    # To count the no. of employees from NIT in each company
    no_of_employees_nit = df['employees'][i]['from_nit']['count']
    NIT_Count.append(no_of_employees_nit)

    # To count the no. of fresher employees from NIT in each company
    no_of_fresher_employees_nit = df['employees'][i]['from_nit']['freshers_count']
    NIT_Fresher_Count.append(no_of_fresher_employees_nit)

    # To count the no. of employees hired from other companies
    no_of_employees_they_hired = df['companies_they_hire_from'][i]['count']
    Employee_count_they_hired.append(no_of_employees_they_hired)

    # To count the no. of ex employees
    ex_emp_count = df['companies_where_their_ex_employees_work'][i]['count']
    Employee_count_their_ex_employees_work.append(ex_emp_count)

    # Total strength companies hired from the tier 1 college
    strength_of_tier_1_college = df['employees'][i]['from_iit']['count'] + df['empl
    Tier_1_College_Strength.append(strength_of_tier_1_college)

    # Total Freshers strength companies hired from the tier 1 college
    Freshers_strength_of_tier_1_college = df['employees'][i]['from_iit']['freshers_
    Tier_1_College_Freshers_Strength.append(Freshers_strength_of_tier_1_college)
```

In [12]:
```
new_df = pd.DataFrame({'Company uuid' : df['company_uuid'],
                       'Company Name' : df['Company Name'],
                       'Employees Count' : Employees_Count,
                       'Fresher Count' : Fresher_Count,
                       'IIT Count' : IIT_Count,
                       'IIT Fresher Count' : IIT_Fresher_Count,
                       'NIT Count' : NIT_Count,
                       'NIT Fresher Count' : NIT_Fresher_Count,
```

```
                        'is score calculated' : df['is_score_calculated'],
                        'Employee count they hired' : Employee_count_they_hired,
                        'Employee count their ex employees work' : Employee_count_th
                        'Tier-1 College Density Score' : df['Department_Tier-1_Colle
                        'Tier1 College Freshers Density Score' : df['Department_Tier
                        'Tier-1 College Strength' : Tier_1_College_Strength,
                        'Tier-1 College Freshers Strength' : Tier_1_College_Freshers
                       })
```

In [13]:
```
# List of companies each company hire from

comp_hired_df = pd.DataFrame()
comp_hired_df['List of companies they hired from'] = None

for i in range(len(df)):
    comp_names = df['companies_they_hire_from'][i]['company_name']
    companies_hired = []
    for key,value in comp_names.items():
        companies_hired.append(key)
    comp_hired_df.loc[i, 'List of companies they hired from'] = companies_hired
```

In [14]:
```
# List of companies each ex-employee work from

ex_emp_comp_df = pd.DataFrame()
ex_emp_comp_df['List of companies their ex-employees work'] = None

for i in range(no_of_rows):
    ex_emp_comp = df['companies_where_their_ex_employees_work'][i]['company_name']
    ex_emp_comp_list = []
    for key,value in ex_emp_comp.items():
        ex_emp_comp_list.append(key)
    ex_emp_comp_df.loc[i, 'List of companies their ex-employees work'] = ex_emp_cor
```

In [15]:
```
# Concatenating all the three dataframes into a single dataframe
modified_df = pd.concat([new_df, comp_hired_df, ex_emp_comp_df],axis = 1)
```

In [16]:
```
modified_df
```

Out[16]:

	Company uuid	Company Name	Employees Count	Fresher Count	IIT Count	IIT Fresher Count	NI Coun
0	None	altran technologies india pvt ltd	1	0	0	0	
1		L&T Technology Services Ltd.	6890	2155	16	7	1
2	fbd567c5e7395b46974ece634a19ff2e	Amdocs	1711	28	4	0	
3	7aa9a32dd8f27a05646e7317af05959b	Xperia	6	0	0	0	
4		Ultra-Scan Corporation	1140	8	63	0	1
...	...	...	...	...	...	...	
60121	None	babu it jobs	1	0	0	0	
60122		capital honda service centre	0	0	0	0	
60123		systems plus llc	0	0	0	0	
60124		gemini traze rfid pvt ltd	0	0	0	0	
60125		radiate e-services pvt. ltd.	0	0	0	0	

60126 rows × 17 columns

In [17]:
```
# No. of columns after modifing the dataframe
len(modified_df.columns)
```

Out[17]: 17

In [184…
```
modified_df.to_excel('assignment.xlsx')
```

Okk...Lets Have some questions on the dataset

Find number of Null values in each column

In [18]:
```
modified_df.isnull().sum()
```

Out[18]:
```
Company uuid                                12449
Company Name                                    0
Employees Count                                 0
Fresher Count                                   0
IIT Count                                       0
IIT Fresher Count                               0
NIT Count                                       0
NIT Fresher Count                               0
is score calculated                             0
Employee count they hired                       0
Employee count their ex employees work          0
Tier-1 College Density Score                    0
Tier1 College Freshers Density Score            0
Tier-1 College Strength                         0
Tier-1 College Freshers Strength                0
List of companies they hired from               0
List of companies their ex-employees work       0
dtype: int64
```

List of companies where hiring companies hire from

In [19]:
```
modified_df['List of companies they hired from']
```

Out[19]:
```
0                            [L&T Technology Services Ltd.]
1        [HCL Technologies, Tech Mahindra, Wipro Techno...
2        [Tata Consultancy Services, Infosys, Cognizant...
3        [Ultra-Scan Corporation, Telescope Services AB...
4        [Wipro Technologies, Tata Consultancy Services...
                               ...
60121                                      [Tech Mahindra]
60122                                                   []
60123                                                   []
60124                                                   []
60125                                                   []
Name: List of companies they hired from, Length: 60126, dtype: object
```

Based on Number of employees classify them into 9 categories and add a column with category number

In [20]:
```
modified_df['Category'] = None
for i in range(no_of_rows):
    if modified_df['Employees Count'].iloc[i] <= 10 :
        modified_df.loc[i,'Category'] = 1
    elif modified_df['Employees Count'].iloc[i] >= 11 and modified_df['Employees Co
        modified_df.loc[i,'Category'] = 2
    elif modified_df['Employees Count'].iloc[i] >= 51 and modified_df['Employees Co
        modified_df.loc[i,'Category'] = 3
    elif modified_df['Employees Count'].iloc[i] >= 101 and modified_df['Employees C
        modified_df.loc[i,'Category'] = 4
    elif modified_df['Employees Count'].iloc[i] >= 251 and modified_df['Employees C
```

```
        modified_df.loc[i,'Category'] = 5
    elif modified_df['Employees Count'].iloc[i] >= 501 and modified_df['Employees C
        modified_df.loc[i,'Category'] = 6
    elif modified_df['Employees Count'].iloc[i] >= 1001 and modified_df['Employees
        modified_df.loc[i,'Category'] = 7
    elif modified_df['Employees Count'].iloc[i] >= 5001 and modified_df['Employees
        modified_df.loc[i,'Category'] = 8
    elif modified_df['Employees Count'].iloc[i] >= 10001:
        modified_df.loc[i,'Category'] = 9
```

In [21]:
```
modified_df
```

Out[21]:

	Company uuid	Company Name	Employees Count	Fresher Count	IIT Count	IIT Fresher Count	NI Cou
0	None	altran technologies india pvt ltd	1	0	0	0	
1		L&T Technology Services Ltd.	6890	2155	16	7	1
2	fbd567c5e7395b46974ece634a19ff2e	Amdocs	1711	28	4	0	
3	7aa9a32dd8f27a05646e7317af05959b	Xperia	6	0	0	0	
4		Ultra-Scan Corporation	1140	8	63	0	1
...	...	...	...	...	...	...	
60121	None	babu it jobs	1	0	0	0	
60122		capital honda service centre	0	0	0	0	
60123		systems plus llc	0	0	0	0	
60124		gemini traze rfid pvt ltd	0	0	0	0	
60125		radiate e-services pvt. ltd.	0	0	0	0	

60126 rows × 18 columns

Find number of companies for each category

In [22]:
```
comp_category_df = modified_df.loc[:,['Category', 'Company Name']]
```

In [23]:
```
comp_grouped = comp_category_df.groupby('Category')
```

In [24]:
```
print(comp_grouped.count())
```

```
          Company Name
Category
1                58055
2                 1427
3                  280
4                  216
5                   69
6                   38
7                   30
8                    4
9                    7
```

Find median and mean of tier-1 strength for each category

In [25]:
```
tier_1_strength_category_df = modified_df.loc[:,['Category', 'Tier-1 College Streng
```

In [26]:
```
tier_1_strength_category_df
```

Out[26]:

	Category	Tier-1 College Strength
0	1	0
1	8	31
2	7	9
3	1	0
4	7	82
...	...	...
60121	1	0
60122	1	0
60123	1	0
60124	1	0
60125	1	0

60126 rows × 2 columns

In [27]:
```
Tier_1_College_Strength_grouped = tier_1_strength_category_df.groupby('Category')
```

In [28]:
```
# Median of Tier-1 strength colleges
print(Tier_1_College_Strength_grouped.median())
```

```
          Tier-1 College Strength
Category
1                            0.0
2                            0.0
3                            0.0
4                            1.0
5                            4.0
6                            6.0
7                           15.5
8                           38.5
9                          113.0
```

In [29]:
```
# Mean of Tier-1 strength colleges
print(Tier_1_College_Strength_grouped.mean())
```

```
          Tier-1 College Strength
Category
1                        0.016019
2                        0.363700
3                        1.178571
4                        2.175926
5                        5.130435
6                        9.052632
7                       27.766667
8                       99.250000
9                      143.714286
```

Display top 5 and bottom 5 companies based on Tier-1 College Freshers Strength for each category

In [30]:
```
Tier_1_college_freshers_strength_category = modified_df.loc[:,['Category', 'Tier-1
```

In [31]:
```
Tier_1_college_freshers_strength_category
```

Out[31]:

	Category	Tier-1 College Freshers Strength
0	1	0
1	8	10
2	7	1
3	1	0
4	7	0
...	...	...
60121	1	0
60122	1	0
60123	1	0
60124	1	0
60125	1	0

60126 rows × 2 columns

In [32]:
```
Tier_1_college_grouped = Tier_1_college_freshers_strength_category.groupby('Categor
```

In [33]:
```
print(Tier_1_college_grouped.head().sort_values(by = 'Category'))
```

	Category	Tier-1 College Freshers Strength
0	1	0
3	1	0
5	1	0
7	1	0
9	1	0
46	2	0
10	2	0
43	2	0
37	2	0
18	2	0
118	3	1
109	3	0
71	3	0
49	3	0
38	3	0
31	4	0
70	4	0
13	4	0
6	4	0
55	4	0
8	5	0
159	5	0
33	5	0
21	5	1
59	5	0
65	6	0
113	6	1
169	6	0
23	6	0
142	6	0
187	7	0
240	7	0
4	7	0
327	7	0
2	7	1
1	8	10
307	8	14
260	8	55
117	8	4
157	9	30
102	9	24
15	9	8
36	9	97
22	9	153

In [34]:
```
print(Tier_1_college_grouped.tail().sort_values(by = 'Category'))
```

	Category	Tier-1 College Freshers Strength
60125	1	0
60121	1	0
60124	1	0
60123	1	0
60122	1	0
45495	2	0
33878	2	0
32846	2	0
28653	2	0
27889	2	0
14042	3	0
19311	3	2
25556	3	0
22305	3	0
23975	3	0
7505	4	1
6721	4	0
6430	4	0
6238	4	0
5843	4	0
5117	5	0
5746	5	0
5064	5	0
4832	5	0
4867	5	0
2807	6	1
3847	6	1
2773	6	2
4198	6	0
4063	6	0
4045	7	1
4015	7	0
3931	7	17
4179	7	10
4471	7	0
307	8	14
260	8	55
117	8	4
1	8	10
320	9	33
244	9	73
157	9	30
102	9	24
36	9	97

Save the data category wise into separate excel sheet and name it as category_k.xlsx where k is the category number

In [36]:
```
grouped = modified_df.groupby('Category')
```

In [37]:
```
for name, group in grouped:
    # Create the sheet name
    sheet_name = "category_{}.xlsx".format(name)

    # Use the to_excel() function to save the group to a new sheet in an Excel file
    group.to_excel(sheet_name, index=False)
```