In [2]:

```
import pandas as pd
import numpy as np
data = pd.read_csv('Salary Dataset.csv', thousands = ',')
data['Time'] = data['Salary'].str.split('/').str.get(1)
data['Currency'] = data['Salary'].str.split('/').str.get(0).str.replace(',', '').str.extract(r'(\D)')
data['Actual Salary'] = data['Salary'].str.split('/').str.get(0).str.replace(',', '').str.extract(r'(\d+)')
data.drop(columns = 'Unnamed: 5', inplace = True)
data['Actual Salary'] = data['Actual Salary'].astype('int')
data.loc[data['Time'] == 'mo', 'Actual Salary'] = data.loc[data['Time'] == 'mo', 'Actual Salary'] * 12
data.loc[data['Time'] == 'hr', 'Actual Salary'] = data.loc[data['Time'] == 'hr', 'Actual Salary'] * 24 * 365
data.loc[data['Currency'] == '₹', 'Actual Salary'] = data.loc[data['Currency'] == '₹', 'Actual Salary'] * 0.013
data.loc[data['Currency'] == '£', 'Actual Salary'] = data.loc[data['Currency'] == '£', 'Actual Salary'] * 1.36
data = data.rename(columns= {'Actual Salary' : 'Actual Salary in $'})
data
```

Out[2]:

	Company Name	Job Title	Salaries Reported	Location	Salary	Time	Currency	Actual Salary in $
0	Mu Sigma	Data Scientist	105.0	Bangalore	₹6,48,573/yr	yr	₹	8431.449
1	IBM	Data Scientist	95.0	Bangalore	₹11,91,950/yr	yr	₹	15495.350
2	Tata Consultancy Services	Data Scientist	66.0	Bangalore	₹8,36,874/yr	yr	₹	10879.362
3	Impact Analytics	Data Scientist	40.0	Bangalore	₹6,69,578/yr	yr	₹	8704.514
4	Accenture	Data Scientist	32.0	Bangalore	₹9,44,110/yr	yr	₹	12273.430
...	...	...	...	...	...	...	...	...
4339	TaiyōAI	Machine Learning Scientist	1.0	Mumbai	₹5,180/mo	mo	₹	808.080
4340	Decimal Point Analytics	Machine Learning Developer	1.0	Mumbai	₹7,51,286/yr	yr	₹	9766.718
4341	MyWays	Machine Learning Developer	1.0	Mumbai	₹4,10,952/yr	yr	₹	5342.376
4342	Market Pulse Technologies	Software Engineer - Machine Learning	1.0	Mumbai	₹16,12,324/yr	yr	₹	20960.212
4343	vPhrase	Machine Learning Engineer	1.0	Mumbai	₹9,39,843/yr	yr	₹	12217.959

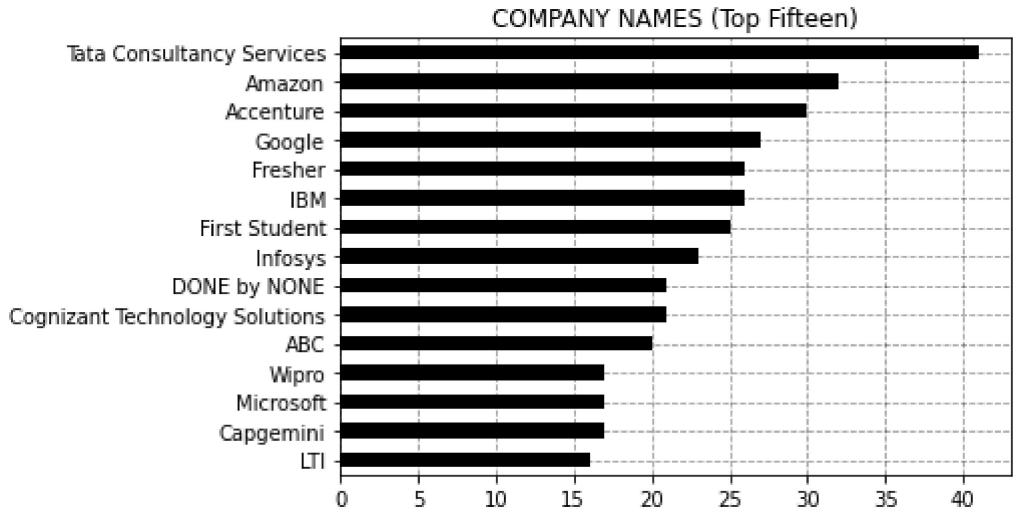
4344 rows × 8 columns

In [7]:

```
import matplotlib.pyplot as plt
data['Company Name'].value_counts().nlargest(15).plot(kind='barh', color = 'black').invert_yaxis()
plt.grid(linestyle='--', alpha = 0.4, color ='black')
plt.title('COMPANY NAMES (Top Fifteen)')
```

Out[7]:

```
Text(0.5, 1.0, 'COMPANY NAMES (Top Fifteen)')
```

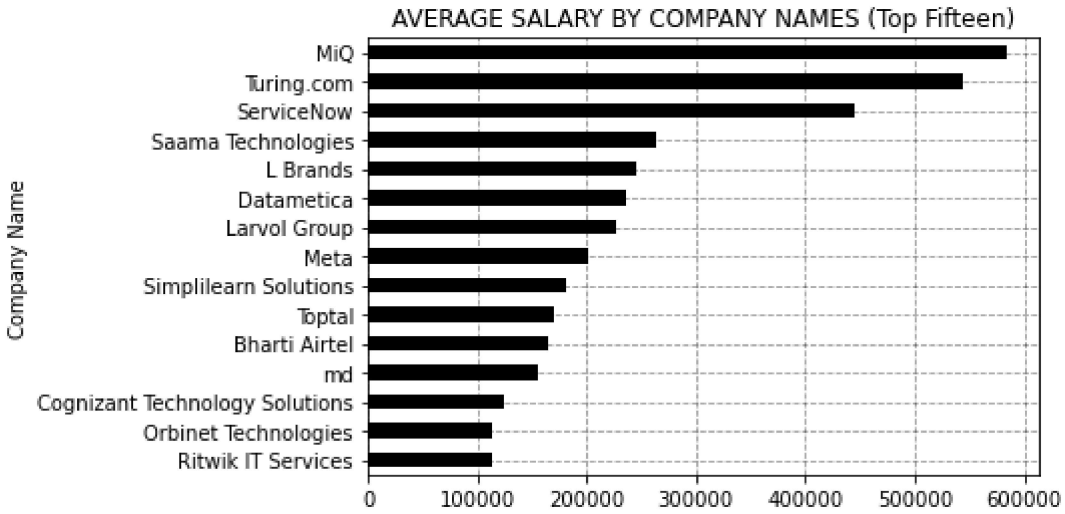


In [9]:

```
data['Company Name'].value_counts().nlargest(10)
data[data['Company Name']=='Tata Consultancy Services']
data.groupby('Company Name')['Actual Salary in $'].mean().nlargest(15).plot(kind = 'barh', color='black').invert_yaxis()
plt.grid(linestyle='--', alpha = 0.4, color ='black')
plt.title('AVERAGE SALARY BY COMPANY NAMES (Top Fifteen)')
```

Out[9]:

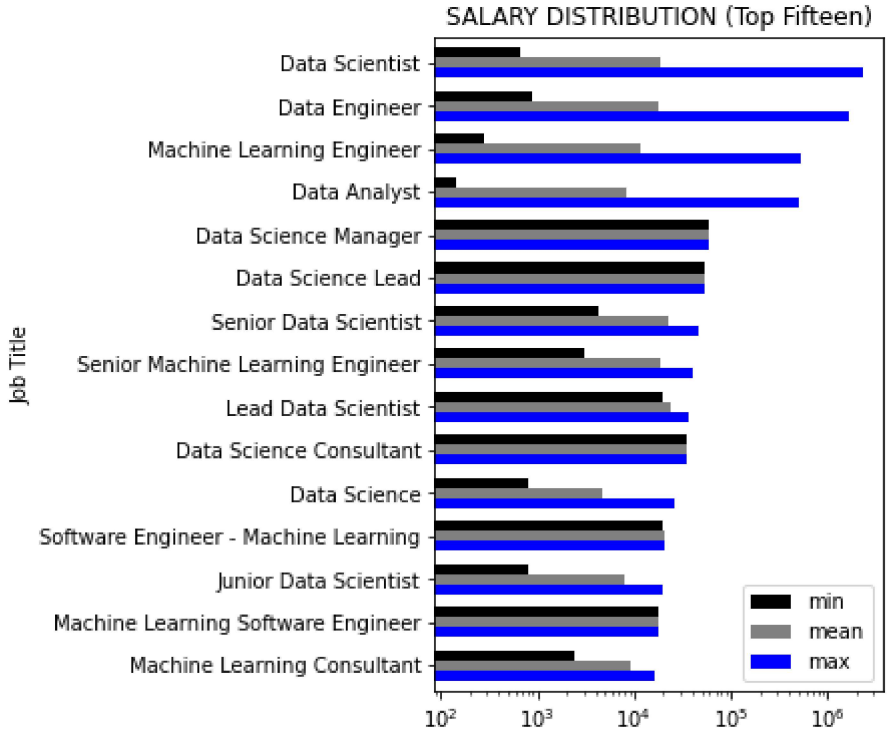
```
Text(0.5, 1.0, 'AVERAGE SALARY BY COMPANY NAMES (Top Fifteen)')
```



In [186…

```
JRS = data.groupby(data['Job Title'])['Actual Salary'].agg(['min', 'mean', 'max'])
JRS.sort_values('mean', ascending=False).head(10)
C = ['Black', 'Gray', 'Blue']
JRS.sort_values('max', ascending=False).head(15).plot.barh(logx=True, width=0.7, figsize=(4,6), color = C).invert_yaxis()
plt.title('SALARY DISTRIBUTION (Top Fifteen)')
```

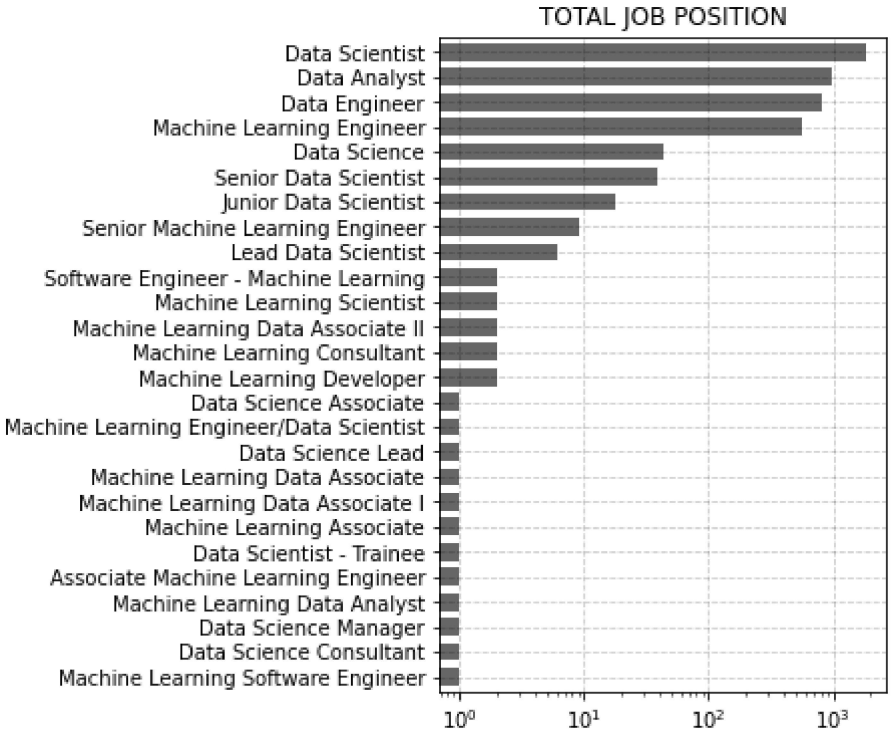
Out[186… Text(0.5, 1.0, 'SALARY DISTRIBUTION (Top Fifteen)')



In [305…

```
JT = data['Job Title'].value_counts()
JT.plot(kind='barh', logx=True, figsize=(4, 6), width=0.7, color='black', alpha = 0.6).invert_yaxis()
plt.grid(linestyle='--', alpha = 0.2, color ='black')
plt.title('TOTAL JOB POSITION')
```

Out[305… Text(0.5, 1.0, 'TOTAL JOB POSITION')

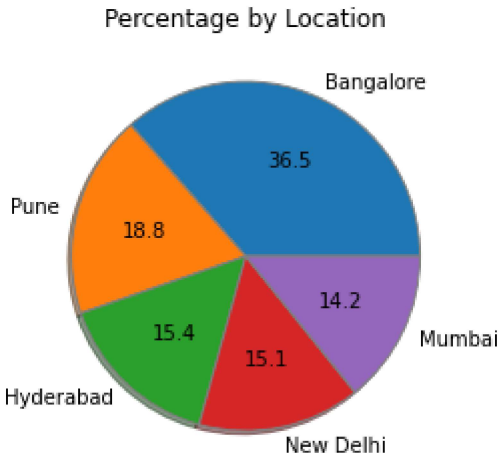


In [10]:

```
data.Location.value_counts().plot.pie(shadow=True, autopct='%0.1f', wedgeprops={'edgecolor':'gray'}, label = '')
plt.title('Percentage by Location')
```

Out[10]:

```
Text(0.5, 1.0, 'Percentage by Location')
```



In [5]:

```
data.to_csv('data.csv')
```

In [ ]: