

WEB ANALYTICS DASHBOARD

```
In [1]: #IMPORT LIBRARIES
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
%matplotlib inline
import seaborn as sns
```

DATA COLLECTION

```
In [2]: df=pd.read_csv('Salary Dataset cleaned.csv',encoding='unicode_escape')
```

```
In [3]: df.head()
```

```
Out[3]:
```

	Company Name	Job Title	Salaries Reported	Location	Salary
0	Mu Sigma	Data Scientist	105.0	Bangalore	648573
1	IBM	Data Scientist	95.0	Bangalore	1191950
2	Tata Consultancy Services	Data Scientist	66.0	Bangalore	836874
3	Impact Analytics	Data Scientist	40.0	Bangalore	669578
4	Accenture	Data Scientist	32.0	Bangalore	944110

TO CHECK AND REMOVE NULL VALUES

```
In [4]: pd.isnull(df).sum()
```

```
Out[4]: Company Name      3
Job Title      0
Salaries Reported    2
Location      0
Salary      0
dtype: int64
```

```
In [5]: df.dropna(inplace=True)
```

```
In [6]: df.shape
```

```
Out[6]: (4339, 5)
```

```
In [7]: df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 4339 entries, 0 to 4343
Data columns (total 5 columns):
#   Column                Non-Null Count  Dtype
---  -
0   Company Name          4339 non-null  object
1   Job Title              4339 non-null  object
2   Salaries Reported      4339 non-null  float64
3   Location               4339 non-null  object
4   Salary                 4339 non-null  int64
dtypes: float64(1), int64(1), object(3)
memory usage: 203.4+ KB
```

```
In [8]: df.describe()
```

```
Out[8]:
```

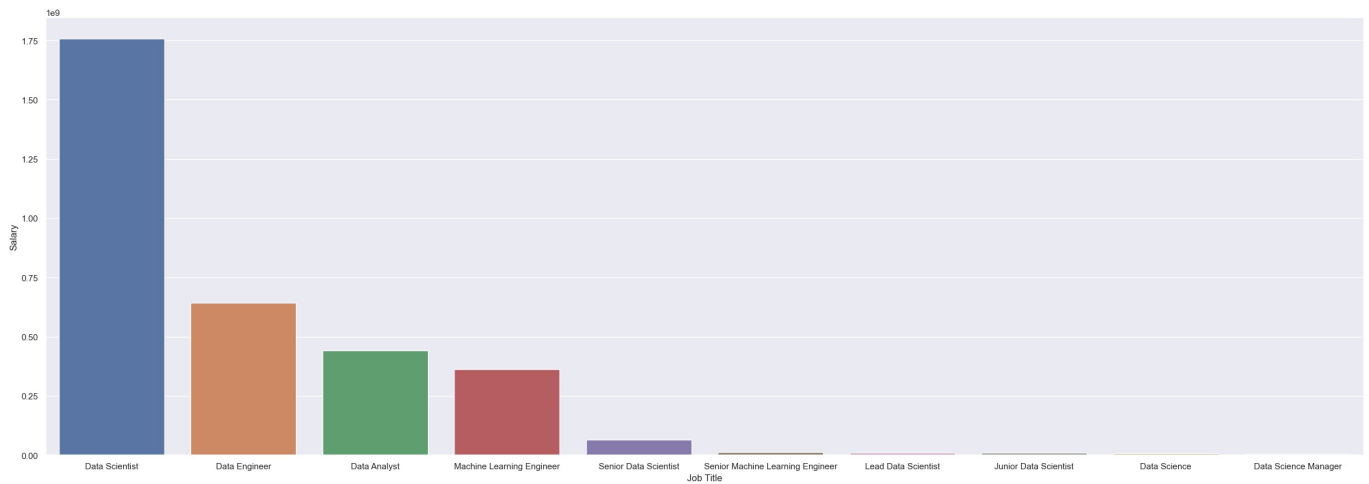
	Salaries Reported	Salary
count	4339.000000	4.339000e+03
mean	2.777138	7.680710e+05
std	5.148095	7.690025e+05
min	1.000000	2.600000e+01
25%	1.000000	1.087225e+05
50%	1.000000	6.199950e+05
75%	3.000000	1.098632e+06
max	105.000000	9.568943e+06

```
In [11]: df.columns
```

```
Out[11]: Index(['Company Name', 'Job Title', 'Salaries Reported', 'Location', 'Salary'], dtype='object')
```

```
In [18]: job_salary=df.groupby(['Job Title'],as_index=False)['Salary'].sum().sort_values(by='Salary',ascending=False).head(10)
sns.set(rc={'figure.figsize':(30,10)})
sns.barplot(x='Job Title',y='Salary',data=job_salary)
```

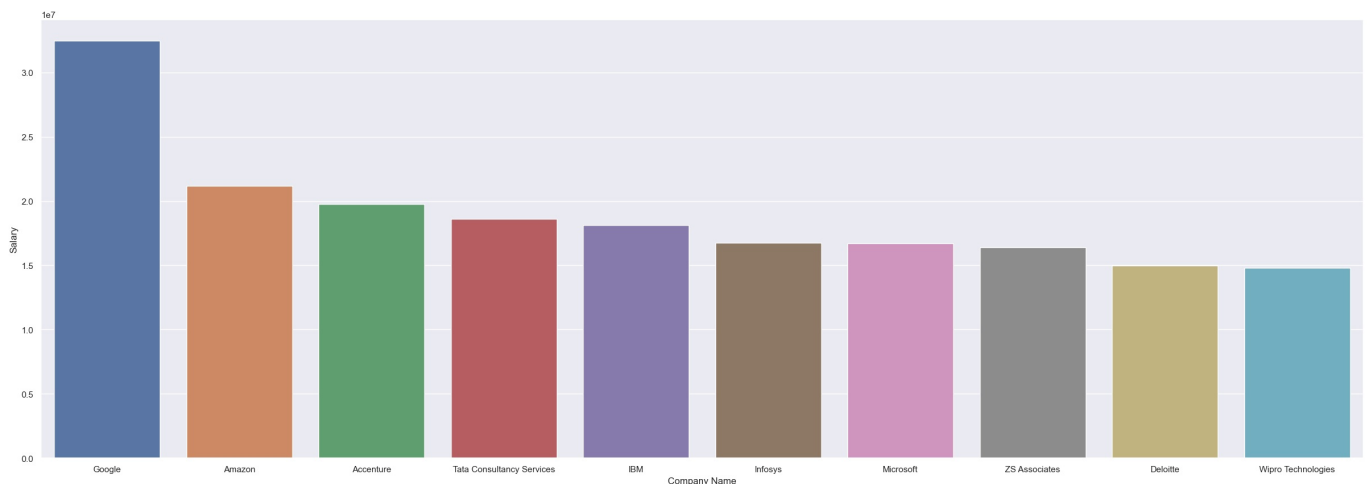
```
Out[18]: <Axes: xlabel='Job Title', ylabel='Salary'>
```



The graph illustrates that Data Scientist positions command the highest salaries across companies, followed by Data Engineers and Data Analysts, reflecting the increasing importance and demand for data-driven expertise in organizations.

```
In [20]: company_salary=df.groupby(['Company Name'],as_index=False)['Salary'].sum().sort_values(by='Salary',ascending=False).head(10)
sns.set(rc={'figure.figsize':(30,10)})
sns.barplot(x='Company Name',y='Salary',data=company_salary)
```

```
Out[20]: <Axes: xlabel='Company Name', ylabel='Salary'>
```



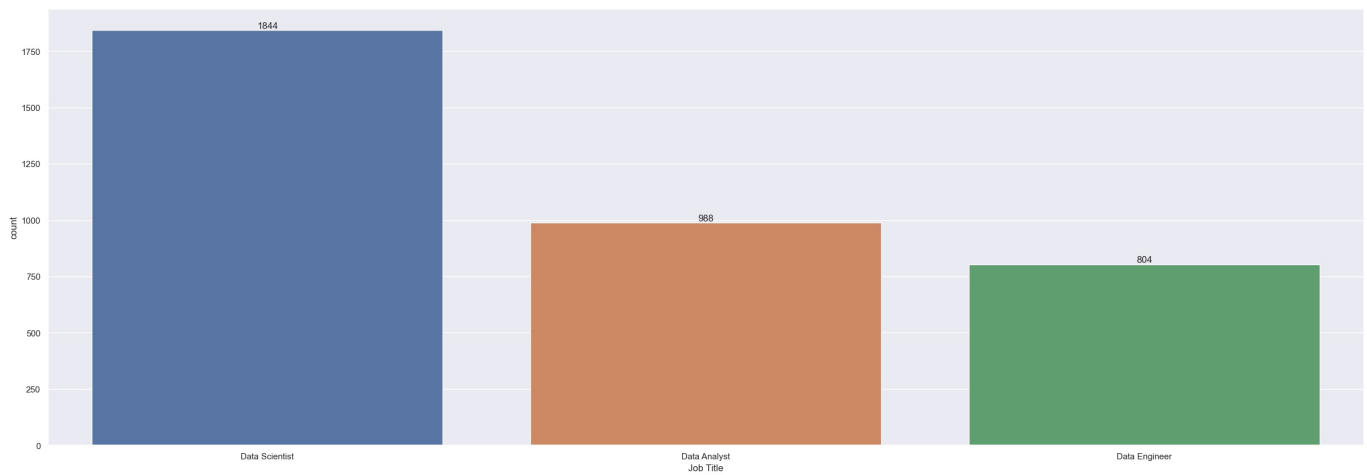
The analysis reveals that Google offers the highest salaries among the companies examined, with Amazon, Accenture, TCS, IBM, and others following suit, highlighting the competitive compensation packages provided by leading tech and consulting firms in the industry.

```
In [26]: # Select the top 3 most common job titles
top_job_titles = df['Job Title'].value_counts().head(3).index.tolist()

# Filter the DataFrame to include only the top 10 job titles
df_filtered = df[df['Job Title'].isin(top_job_titles)]

# Create the count plot
ax = sns.countplot(x='Job Title', data=df_filtered)

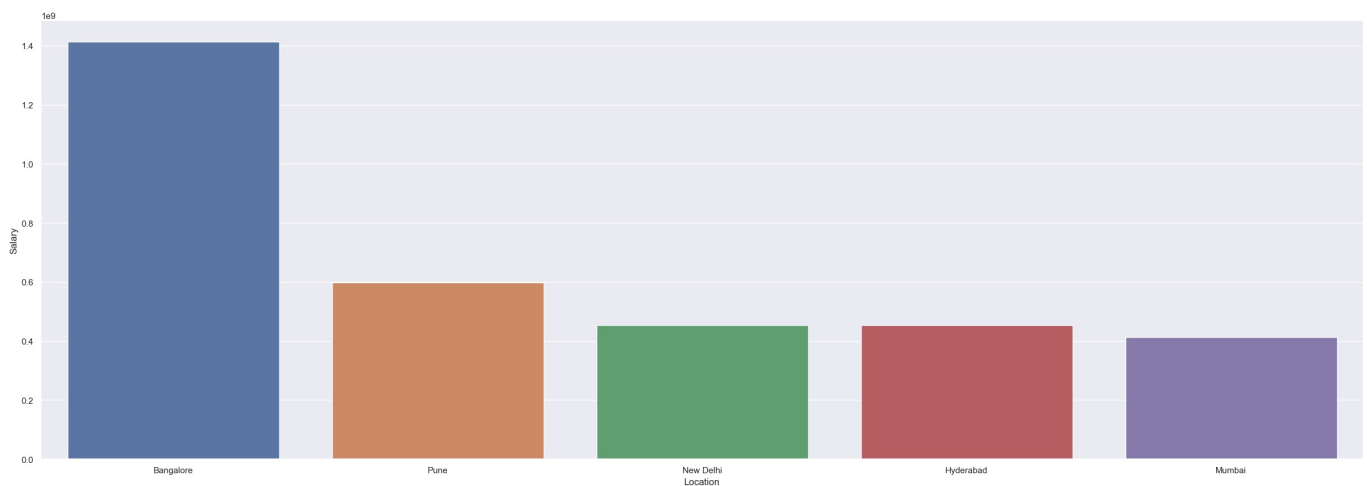
# Add data labels
for bars in ax.containers:
    ax.bar_label(bars)
```



The data indicates that Data Scientist, Data Analyst, and Data Engineer are the most prevalent job titles, underscoring the widespread demand for professionals skilled in data analysis, interpretation, and engineering, reflecting the central role of data-related expertise across various industries in today's digital landscape.

```
In [25]: LocationWise_salary=df.groupby(['Location'],as_index=False)['Salary'].sum().sort_values(by='Salary',ascending=False)
sns.set(rc={'figure.figsize':(30,10)})
sns.barplot(x='Location',y='Salary',data=LocationWise_salary)
```

```
Out[25]: <Axes: xlabel='Location', ylabel='Salary'>
```



The analysis indicates that Bangalore offers the most lucrative salary packages for various job roles, with Pune and New Delhi following closely behind. This suggests that Bangalore is a thriving hub for career opportunities in the analyzed job sectors, attracting top talent with competitive compensation, while Pune and New Delhi also present favorable prospects for professionals seeking rewarding employment in these fields.

```
In [29]: # Group by 'Job Title' and 'Company', and find the maximum salary in each group
max_salary_by_job_company = df.groupby(['Job Title', 'Company Name'])['Salary'].max().reset_index()

# Display the DataFrame containing maximum salaries by job title and company
print(max_salary_by_job_company)
```

	Job Title	Company Name	Salary
0	Associate Machine Learning Engineer	Techolution	464372
1	Data Analyst	159 Solutions	828777
2	Data Analyst	20K Group	42892
3	Data Analyst	72 Dragons	23043
4	Data Analyst	ABC	908078
...
3360	Senior Machine Learning Engineer	Snap2Insight	1452000
3361	Senior Machine Learning Engineer	Techolution	803907
3362	Senior Machine Learning Engineer	iSchoolConnect	1335445
3363	Software Engineer - Machine Learning	HSBC	1521236
3364	Software Engineer - Machine Learning	Market Pulse Technologies	1612324

[3365 rows x 3 columns]

```
In [32]: # Find the row with the maximum salary
max_salary_row = df.loc[df['Salary'].idxmax()]

# Extract job title, company name, and maximum salary
max_salary_job_title = max_salary_row['Job Title']
max_salary_company = max_salary_row['Company Name']
total_max_salary = max_salary_row['Salary']

# Display the results
print("Job Title with Maximum Salary:", max_salary_job_title)
```

```
print("Company with Maximum Salary:", max_salary_company)
print("Total Maximum Salary:", total_max_salary)
```

Job Title with Maximum Salary: Data Scientist
Company with Maximum Salary: Societ  G n rale
Total Maximum Salary: 9568943

```
In [38]: # Find the row with the minimum salary
min_salary_row = df.loc[df['Salary'].idxmin()]

# Extract job title, company name, and minimum salary
min_salary_job_title = min_salary_row['Job Title']
min_salary_company = min_salary_row['Company Name']
total_min_salary = min_salary_row['Salary']

# Display the results
print("Job Title with Minimum Salary:", min_salary_job_title)
print("Company with Minimum Salary:", min_salary_company)
print("Total Minimum Salary:", total_min_salary)
```

Job Title with Minimum Salary: Data Engineer
Company with Minimum Salary: Larvol Group
Total Minimum Salary: 26

THANK YOU!

CONNECT WITH ME:

LinkedIn: www.linkedin.com/in/harshita-sharma-b68154220

GitHub: <https://github.com/DATAPREDICTS>

Loading [MathJax]/jax/output/CommonHTML/fonts/TeX/fontdata.js