Analysis of Highest IT Paying Jobs In India

This project anlayzed the highest paying IT jobs in India, focusing on job positions, location, salaries, education, and experience levels. It provides insights into salary trends, correltations and distributions, helping to understand the IT sector's job market and the factors influencing compensation.

Example of some insights we're analyzing in this project

1. Calculate the mean, median, and mode of the salaries in the dataset. Which measure (mean, median, or mode) best represents the central tendency of the salaries?
2. Calculate the variance and standard deviation for the salaries of employees with different levels of education (e.g., Bachelor's, Master's, Ph.D.). What does the standard deviation indicate about the spread of salaries within each education group?
3. Plot the distribution of salaries. Is the distribution positively or negatively skewed? What does this indicate about the salary structure in the IT industry in India?
4. What is the median salary based on different levels of experience (e.g., 0-3 years, 4-7 years, 8+ years)?

And others.

[1]:
```
# Import required libraries
import pandas as pd
import numpy as np
```

[3]:
```
# Import Dataset

df = pd.read_csv(r"C:\Users\Aayush\Documents\SQL Server Management Studio\31 Day of Data Analytic Project\Day 3 Analysis of Highe
df.head()
```

[3]:

	Position	Location	Gender	Education	Experience (Years)	Salary
0	QNXT Configuration QA/Testing SME	Ghaziabad	Female	B.Tech/B.E.	11	2014510
1	Provider Data Management	New Delhi	Female	B.Tech/B.E.	24	1624349
2	Accessibility Engineer QA	Noida	Female	BCA	25	1926223
3	Senior Software Engineer	Jalandhar	Male	NaN	27	2403560
4	Java Developer/Spring Boot	Meerut	Male	B.A	11	1128404

In [4]:
```
# Display the summary
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3467 entries, 0 to 3466
Data columns (total 6 columns):
 #   Column              Non-Null Count  Dtype 
---  ------              --------------  ----- 
 0   Position            3448 non-null   object
 1   Location            3467 non-null   object
 2   Gender              3443 non-null   object
 3   Education           3391 non-null   object
 4   Experience (Years)  3467 non-null   int64 
 5   Salary              3467 non-null   int64 
dtypes: int64(2), object(4)
memory usage: 162.6+ KB
```

In [5]:
```
# Check if there is any null value
df.isnull().sum()
```

Out[5]:
```
Position              19
Location               0
Gender                24
Education             76
Experience (Years)     0
Salary                 0
dtype: int64
```

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

Out[7]:
```
Position              808
Location               88
Gender                  2
Education              20
Experience (Years)     24
Salary               3465
dtype: int64
```

In [8]:
```
# Display the descriptive statistics

df.describe()
```

Out[8]:

	Experience (Years)	Salary
count	3467.000000	3.467000e+03
mean	16.639746	1.487459e+06
std	6.935259	5.776934e+05
min	5.000000	5.026890e+05
25%	11.000000	9.818155e+05
50%	17.000000	1.475343e+06
75%	23.000000	1.985568e+06
max	28.000000	2.499925e+06

In [10]:
```
df[df.duplicated()]
```

Out[10]:

Position	Location	Gender	Education	Experience (Years)	Salary

Data Cleaning

Handling Missing Values From Education Column

In [11]:
```
df['Education'].unique()
```

Out[11]:
```
array(['B.Tech/B.E.', 'BCA', nan, 'B.A', 'B.Com', 'Diploma', 'B.Sc',
       'B.B.A/ B.M.S', 'Not Pursuing Graduation', 'BJMC',
       'course on computer concepts', 'B.Des.', 'LLB', 'BBA', 'BTECH',
       'HSC', 'bachelor of arts', 'B.PHARMACY', 'B.Pharma', 'BBM',
       'Mtech'], dtype=object)
```

In [12]:
```
distinct_education = df['Education'].dropna().unique()

distinct_education
```

Out[12]:
```
array(['B.Tech/B.E.', 'BCA', 'B.A', 'B.Com', 'Diploma', 'B.Sc',
       'B.B.A/ B.M.S', 'Not Pursuing Graduation', 'BJMC',
       'course on computer concepts', 'B.Des.', 'LLB', 'BBA', 'BTECH',
       'HSC', 'bachelor of arts', 'B.PHARMACY', 'B.Pharma', 'BBM',
       'Mtech'], dtype=object)
```

In [15]:
```
# Defining Ranges

def fill_education(row):
    if pd.isnull(row['Education']):
        if row['Experience (Years)'] <=2 or row['Salary'] <=500000:
            return 'B.Tech/B.E'
        elif 3 <= row['Experience (Years)'] <= 6 or row['Salary'] <= 1000000:
            return 'BCA'
        else:
            return 'Mtech'
    return row['Education']



# Apply the function to fill missing values
df['Education'] = df.apply(fill_education, axis = 1)

print('Data after Imputation Education Columun: ')
df
```

Out[15]:

	Position	Location	Gender	Education	Experience (Years)	Salary
0	QNXT Configuration QA/Testing SME	Ghaziabad	Female	B.Tech/B.E.	11	2014510
1	Provider Data Management	New Delhi	Female	B.Tech/B.E.	24	1624349
2	Accessibility Engineer QA	Noida	Female	BCA	25	1926223
3	Senior Software Engineer	Jalandhar	Male	Mtech	27	2403560
4	Java Developer/Spring Boot	Meerut	Male	B.A	11	1128404
...	...	...	...	...	...	...
3462	Salesforce developer	Hyderabad	Male	B.Tech/B.E.	21	816277
3463	Salesforce developer	Bengaluru	Male	BCA	20	1786298
3464	Salesforce developer	New Delhi	Female	Mtech	28	1050400
3465	Salesforce developer	Gurugram	Male	B.A	25	764525
3466	Salesforce developer	Pune	Male	B.Com	24	2252207

3467 rows × 6 columns

Handling Gender Column

In [34]:
```
gender_mode = df['Gender'].mode()[0]
df['Gender'].fillna(gender_mode)
```

Out[34]:
```
0       Female
1       Female
2       Female
3         Male
4         Male
         ...
3462      Male
3463      Male
3464    Female
3465      Male
3466      Male
Name: Gender, Length: 3448, dtype: object
```

In [22]:
```
df[df['Position'].isnull()]
```

Out[22]:

	Position	Location	Gender	Education	Experience (Years)	Salary
102	NaN	Mumbai	Male	BCA	9	695736
815	NaN	Pune	Female	B.Tech/B.E.	13	2175425
1703	NaN	Noida	Female	B.Tech/B.E.	8	504577
1845	NaN	Pune	Male	Diploma	23	2420933
1897	NaN	vishakapatnam - Andhra Pradesh	Male	B.A	12	541049
1989	NaN	Gurugram	Female	B.Tech/B.E.	19	957190
2011	NaN	Gurugram	Male	B.Com	5	1885497
2812	NaN	New Delhi	Female	B.A	14	1293038
2824	NaN	Noida	Female	BCA	25	908183
2880	NaN	New Delhi	Male	BCA	25	1689585
2883	NaN	New Delhi	Female	B.Sc	25	781545
2888	NaN	New Delhi	Male	B.Sc	26	1291237
2891	NaN	New Delhi	Male	B.Com	17	877531
2893	NaN	Noida	Female	B.Tech/B.E.	15	2278275
2897	NaN	Kolkata	Male	BCA	26	523782
2902	NaN	Noida	Male	B.Tech/B.E.	23	2073673
3081	NaN	Chennai	Female	B.Sc	21	1747409
3352	NaN	Noida	Male	B.Com	18	1927312
3412	NaN	Mumbai Suburban	Male	B.Tech/B.E.	6	1494398

In [28]:
```
df['Position'] = df['Position'].replace(['', ' '], pd.NA)
```

In [29]:
```
df = df.dropna(subset=['Position'])
```

Basic-Level Questions

How many unique job position are listed in the dataset?

In [36]:
```
df['Position'].unique()
```

```
'Snowflake Data Engineer', 'Data Analyst',
'React Native Architect', 'Prinicipal Data Eng',
'Service Operations Manager', 'NPD Trim design',
'Solution architect', 'Imanage Support',
'NPD Design-Chassis, Engineering Head', 'Distributed sup eng',
'Data Bricks engineer', 'Java with flowable',
'AWS CLOUD &amp;amp; DEVOPS ENGINEER', 'JAva with Flowable',
'Cloud Engeener', 'lead Consultant -power BI', 'UI Lead',
'Sr.Colud Data Engineer',
'NDP Design Chassis, Vehicle Integration and Chassis Design,',
'Data Engineering', 'AML', 'QA Manager Automation',
'Hydrallic Project manager', 'React native', 'Abinitio Lead',
'Linux Administrator,Openshift admin,cloud operations',
'ServiceNow SAM',
'Train Design - Car body shell Engineering at Alstom Transportation',
'Sr Developer', 'GCP Architect(Cloud architect)',
'Golang Developer/ Azure Devops', 'SAP SCM ,SAP MM',
'Senior consultant/ Software developer', 'Sr.Test Engineer',
'Microsoft Dyna. 365', 'PRINCIPLE DATA ENGINEER', 'AZURE VIRTUVAL',
```

In [37]:
```
df['Position'].nunique()
```

Out[37]:
```
808
```

What is the average salary of IT jobs in India?

In [39]:
```
average_salary = df['Salary'].mean()
print("Average Salary of IT jobs in India: ",average_salary)
```

```
Average Salary of IT jobs in India:  1488095.2665313226
```

Find the location with the highest number of job position

In [43]:
```
location_count = df.groupby('Location')['Position'].count()
```

In [45]:
```
location_position = location_count.idxmax()
max_position_count = location_count.max()


print("The location with the highest number of job position is ",location_position,"with ",max_position_count,"job position." )
```

```
The location with the highest number of job position is  New Delhi with  689 job position.
```

Intermediate Level Questions

Calculate the mean, median, and mode of the salaries in the dataset. Which measure (mean, median, or mode) best represents the central tendency of the salaries?

In [49]:
```
mean_salary = df['Salary'].mean()
median_salary = df['Salary'].median()
mode_salary = df['Salary'].mode()

print('The Mean of Salary is ', mean_salary)
print('The Median of Salary is ',median_salary)
print('The Mode of Salary is ',mode_salary.iloc[0] if not mode_salary.empty else "No mode") # If there are multiple modes, print
```

```
The Mean of Salary is  1488095.2665313226
The Median of Salary is  1475379.5
The Mode of Salary is  1199944
```

Find the range and interquartile range (IQR) of the Experience (Years) in the dataset. How do these values help in understanding the spread of experience across the employees?

In [54]:
```
print(df['Experience (Years)'].max())
print(df['Experience (Years)'].min())
```

```
28
5
```

In [51]:
```
# Calculate the range and interquartile range fo the Experience (years)

experience_range  = df['Experience (Years)'].max() - df['Experience (Years)'].min()

#Calculate Interquartile Range
Q1 = df['Experience (Years)'].quantile(0.25)
Q3 = df['Experience (Years)'].quantile(0.75)
InterQuartile_Range = Q3- Q1

print('Range: ',experience_range)
print('InterQuartile Range: ', InterQuartile_Range)
```

```
Range:  23
InterQuartile Range:  12.0
```

Range: 23

This mean that the difference between maximum and minimum years of experience (Years) is 23 years in our dataset. The maximum experience employee had is 28 years and minimum experience any employee had is 5 years. This indicate a wide spread in experience level of employee in the dataset.

Interquartile Range: 12

The IQR represent that the middle 50% of employees have experience years ranging from Q1 to Q3, and the difference between Q1 and Q3 is 12 years, showing that the majority of the employees have experience levels within this range.

Calculate the variance and standard deviation for the salaries of employees with different levels of education (e.g., Bachelor's, Master's, Ph.D.). What does the standard deviation indicate about the spread of salaries within each education group?

In [63]:
```
df.groupby('Education')['Salary'].agg(['mean','var','std', 'max','min'])
```

Out[63]:

Education	mean	var	std	max	min
B.A	1.513427e+06	3.311335e+11	575442.040646	2497196	502689
B.B.A/ B.M.S	1.482160e+06	3.614730e+11	601226.277355	2486222	536308
B.Com	1.481392e+06	3.006492e+11	548314.860649	2492390	506448
B.Des.	1.696488e+06	1.276928e+11	357341.349411	2107011	1038609
B.PHARMACY	1.979655e+06	4.152628e+10	203779.980119	2195626	1684091
B.Pharma	1.542135e+06	5.734843e+11	757287.455563	2314053	539149
B.Sc	1.467785e+06	3.479009e+11	589831.211732	2499925	503071
B.Tech/B.E.	1.478101e+06	3.308602e+11	575204.505047	2496341	502893
BBA	1.172199e+06	3.436439e+11	586211.483286	1978064	618482
BBM	1.428390e+06	2.295791e+11	479144.144862	1970939	846821
BCA	1.466479e+06	3.601868e+11	600155.652126	2499733	502818
BJMC	1.846517e+06	3.292398e+11	573794.240395	2489343	887682
BTECH	1.517533e+06	3.887524e+11	623500.117453	2152921	558228
Diploma	1.528728e+06	3.537124e+11	594737.227114	2493345	512503
HSC	1.445539e+06	2.140471e+11	462652.223480	1904500	683226
LLB	1.815877e+06	2.983905e+11	546251.271698	2466419	1006036
Mtech	1.793020e+06	1.925567e+11	438812.781807	2494639	1029270
Not Pursuing Graduation	1.239952e+06	6.170065e+10	248396.153180	1746532	929380
bachelor of arts	1.637977e+06	9.212620e+11	959823.916355	2316675	959279
course on computer concepts	1.357363e+06	5.344248e+11	731043.608285	2493588	551989

Determine the correlation between experience (years) and salary. Is there a positive, negative, or no correlation between these variables?

In [64]:
```
df1 = df[['Experience (Years)','Salary']]
df1.columns = ['Experience','Salary']
df1.head(8)
```

Out[64]:

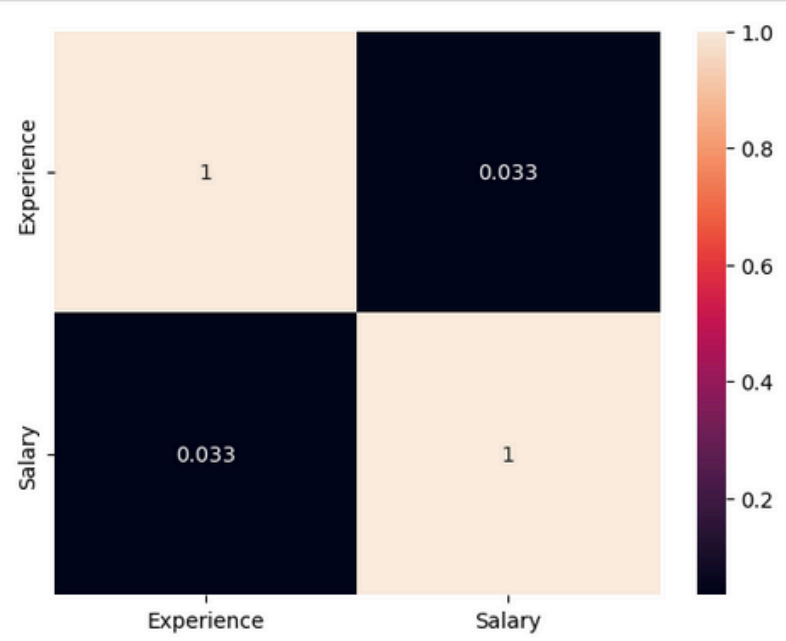
	Experience	Salary
0	11	2014510
1	24	1624349
2	25	1926223
3	27	2403560
4	11	1128404
5	15	2090495
6	21	1399850
7	8	881054

In [65]:
```
corr = df1.corr()
corr
```

Out[65]:

	Experience	Salary
Experience	1.000000	0.033471
Salary	0.033471	1.000000

In [70]:
```
sns.heatmap(corr, annot = True)
plt.show()
plt.savefig("Images/Correlation Betwwen Experience and Salary.jpg")
```



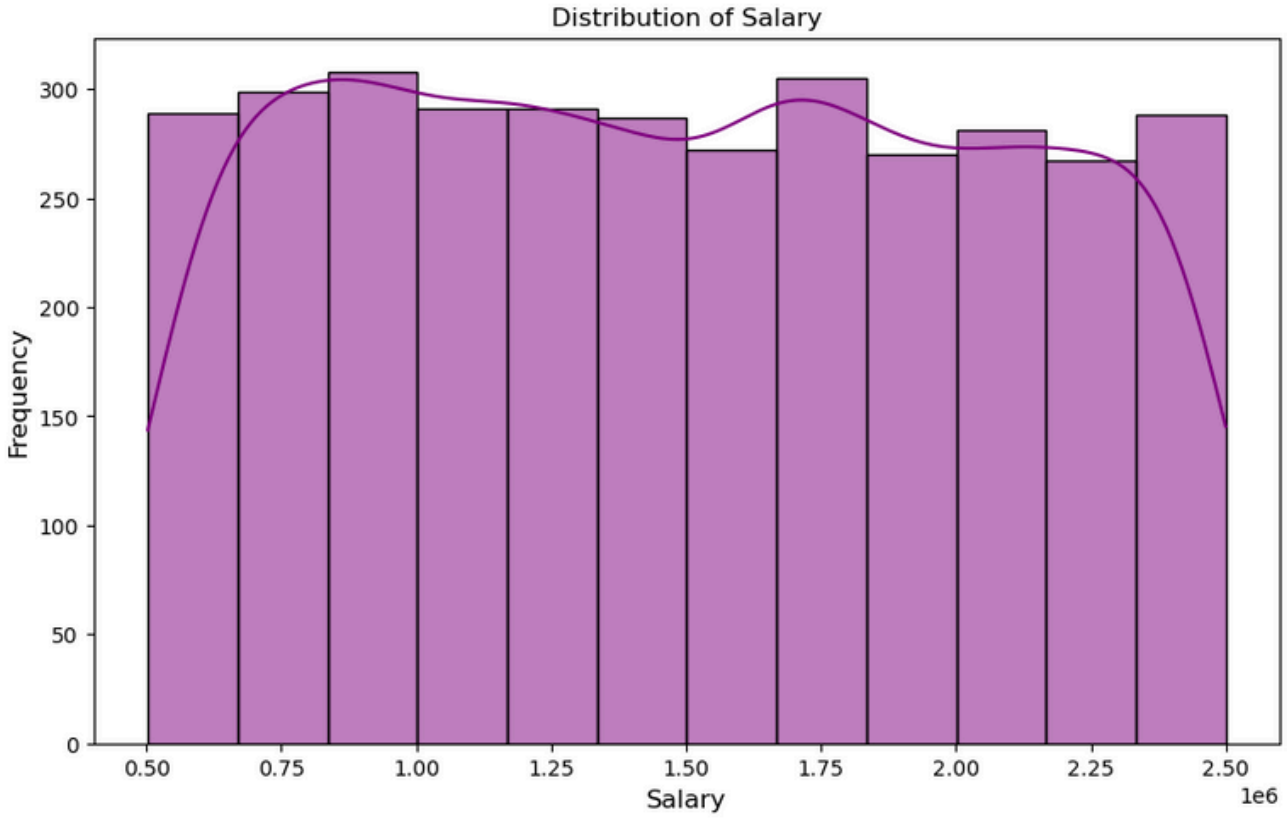

```
<Figure size 640x480 with 0 Axes>
```

Interpretation

- A correlation near 0 implies that experience and salay are not strongly related in our data. The weak positive correlation means that there is a slightly tendency for salary to increase as experience increases, but not a strong or consistent trend.

Plot the distribution of salaries. Is the distribution positively or negatively skewed? What does this indicate about the salary structure in the IT industry in India?

In [72]:
```
plt.figure(figsize=(10,6))
sns.histplot(df['Salary'], bins = 12,kde=True, color = 'Purple')
plt.title('Distribution of Salary')
plt.xlabel('Salary', fontsize=12)
plt.ylabel('Frequency', fontsize=12)
plt.savefig('Images/Distribution of Salary.jpg')
plt.show()
```



What is the median salary based on different levels of experience (e.g., 0-3 years, 4-7 years, 8+ years)?

[73]:
```
df['Salary'].median()
```

t[73]:
```
1475379.5
```

[80]:
```
bins = [0,3,7, float('inf')]
group_names = ['0-3 Years','4-7 Years','8+ Years']
```

[78]:
```
df['Experience Level'] = pd.cut(df['Experience (Years)'],bins = bins, labels = group_names, right=False)
```

```
C:\Users\Aayush\AppData\Local\Temp\ipykernel_9856\876975248.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_guide/indexing.html#returning-a-view-versus-a-copy
  df['Experience Level'] = pd.cut(df['Experience (Years)'],bins = bins, labels = group_names, right=False)
```

[79]:
```
median_salary_experience = df.groupby('Experience Level')['Salary'].median()
median_salary_experience
```

t[79]:
```
Experience Level
0-3 Years          NaN
4-7 Years    1472721.0
8+ Years     1475787.0
Name: Salary, dtype: float64
```