#About Dataset salaries dataset generally provides information about the employees of an organization in relation to their compensation. It typically includes details such as how much each employee is paid (their salary), their job titles, the departments they work in, and possibly additional information like their level of experience, education, and employment history within the organization.

Features

- 'Id'
- 'EmployeeName'
- 'JobTitle'
- 'BasePay'
- 'OvertimePay'
- 'OtherPay'
- 'Benefits'
- 'TotalPay' -> salary
- 'TotalPayBenefits'
- 'Year'
- 'Notes'
- 'Agency'
- 'Status'

Tasks

1. **Basic Data Exploration**: Identify the number of rows and columns in the dataset, determine the data types of each column, and check for missing values in each column.
2. **Descriptive Statistics**: Calculate basic statistics mean, median, mode, minimum, and maximum salary, determine the range of salaries, and find the standard deviation.
3. **Data Cleaning**: Handle missing data by suitable method with explain why you use it.
4. **Basic Data Visualization**: Create histograms or bar charts to visualize the distribution of salaries, and use pie charts to represent the proportion of employees in different departments.
5. **Grouped Analysis**: Group the data by one or more columns and calculate summary statistics for each group, and compare the average salaries across different groups.
6. **Simple Correlation Analysis**: Identify any correlation between salary and another numerical column, and plot a scatter plot to visualize the relationship.
7. **Summary of Insights**: Write a brief report summarizing the findings and insights from the analyses.

Very Important Note

There is no fixed or singular solution for this assignment, so if anything is not clear, please do what you understand and provide an explanation.

In [1]:

```
import pandas as pd
import numpy as np

# Load your dataset
df = pd.read_csv('Salaries.csv')
df.head()
```

Out[1]:

	Id	EmployeeName	JobTitle	BasePay	OvertimePay	OtherPay	Benefits	TotalP
0	1	NATHANIEL FORD	GENERAL MANAGER-METROPOLITAN TRANSIT AUTHORITY	167411.18	0.00	400184.25	NaN	567595.
1	2	GARY JIMENEZ	CAPTAIN III (POLICE DEPARTMENT)	155966.02	245131.88	137811.38	NaN	538909.
2	3	ALBERT PARDINI	CAPTAIN III (POLICE DEPARTMENT)	212739.13	106088.18	16452.60	NaN	335279.
3	4	CHRISTOPHER CHONG	WIRE ROPE CABLE MAINTENANCE MECHANIC	77916.00	56120.71	198306.90	NaN	332343.
4	5	PATRICK GARDNER	DEPUTY CHIEF OF DEPARTMENT,(FIRE DEPARTMENT)	134401.60	9737.00	182234.59	NaN	326373.

In [2]:

```
df.columns
```

Out[2]:

```
Index(['Id', 'EmployeeName', 'JobTitle', 'BasePay', 'OvertimePay', 'Oth
erPay',
       'Benefits', 'TotalPay', 'TotalPayBenefits', 'Year', 'Notes', 'Ag
ency',
       'Status'],
      dtype='object')
```

In [3]:

```
row,col=df.shape
print(row , col)
```

```
148654 13
```

In [4]:

```
df.isnull().sum()
```

Out[4]:

```
Id                       0
EmployeeName             0
JobTitle                 0
BasePay                609
OvertimePay              4
OtherPay                 4
Benefits             36163
TotalPay                 0
TotalPayBenefits         0
Year                     0
Notes               148654
Agency                   0
Status              148654
dtype: int64
```

In [5]:

```
df.describe()
```

Out[5]:

	Id	BasePay	OvertimePay	OtherPay	Benefits	
count	148654.000000	148045.000000	148650.000000	148650.000000	112491.000000	14865
mean	74327.500000	66325.448841	5066.059886	3648.767297	25007.893151	7476
std	42912.857795	42764.635495	11454.380559	8056.601866	15402.215858	5051
min	1.000000	-166.010000	-0.010000	-7058.590000	-33.890000	-61
25%	37164.250000	33588.200000	0.000000	0.000000	11535.395000	3616
50%	74327.500000	65007.450000	0.000000	811.270000	28628.620000	7142
75%	111490.750000	94691.050000	4658.175000	4236.065000	35566.855000	10583
max	148654.000000	319275.010000	245131.880000	400184.250000	96570.660000	56759

In [6]:

```
column_type=df.dtypes
print(column_type)
```

```
Id                    int64
EmployeeName         object
JobTitle             object
BasePay             float64
OvertimePay         float64
OtherPay            float64
Benefits            float64
TotalPay            float64
TotalPayBenefits    float64
Year                  int64
Notes               float64
Agency               object
Status              float64
dtype: object
```

In [7]:

```
mean_salary=df["TotalPay"].mean()
median_salary=df["TotalPay"].median()
mode_salary=df["TotalPay"].mode()
min_salary=df["TotalPay"].min()
max_salary=df["TotalPay"].max()
std_salary=df["TotalPay"].std()
range_salary=max_salary-min_salary
```

In [8]:

```
print(mean_salary )
print(median_salary )
print(mode_salary )
print(min_salary )
print(max_salary )
print(std_salary )
print(range_salary )
```

```
74768.321971703
71426.60999999999
0    0.0
dtype: float64
-618.13
567595.43
50517.005273949944
568213.56
```

In [11]:

```
df.fillna(df.mean(),inplace=True)
# I use this method for several reasons:
# Preserving the data by filling missing values with reasonable alternati
# we can maintain the size of the dataset and avoid losing important info
# limiting the impact on results ....
df.dropna(axis=1)
# I have deleted columns that contain many null values
```

Out[11]:

	Id	EmployeeName	JobTitle	BasePay	OvertimePay	Other
0	1	NATHANIEL FORD	GENERAL MANAGER-METROPOLITAN TRANSIT AUTHORITY	167411.180000	0.000000	400184.250
1	2	GARY JIMENEZ	CAPTAIN III (POLICE DEPARTMENT)	155966.020000	245131.880000	137811.380
2	3	ALBERT PARDINI	CAPTAIN III (POLICE DEPARTMENT)	212739.130000	106088.180000	16452.600
3	4	CHRISTOPHER CHONG	WIRE ROPE CABLE MAINTENANCE MECHANIC	77916.000000	56120.710000	198306.900
4	5	PATRICK GARDNER	DEPUTY CHIEF OF DEPARTMENT,(FIRE DEPARTMENT)	134401.600000	9737.000000	182234.590
...	...	...	...	...	...	
148649	148650	Roy I Tillery	Custodian	0.000000	0.000000	0.000
148650	148651	Not provided	Not provided	66325.448841	5066.059886	3648.767
148651	148652	Not provided	Not provided	66325.448841	5066.059886	3648.767
148652	148653	Not provided	Not provided	66325.448841	5066.059886	3648.767
148653	148654	Joe Lopez	Counselor, Log Cabin Ranch	0.000000	0.000000	-618.130

148654 rows × 11 columns

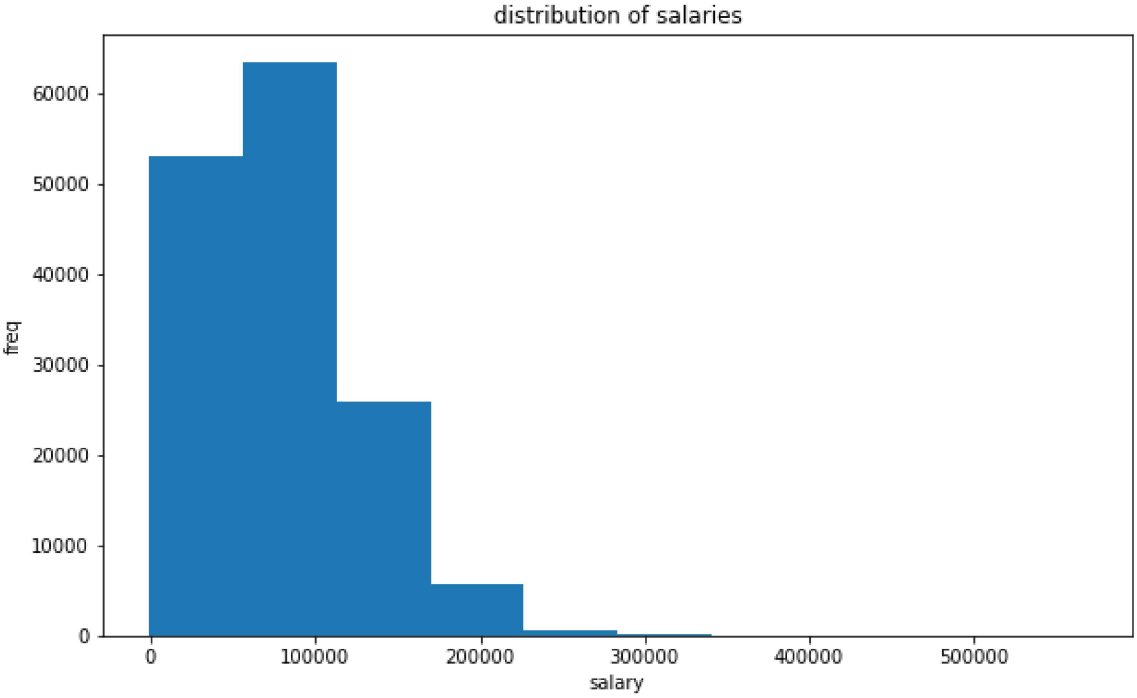
In [ ]:

In [9]:

```
import matplotlib.pyplot as plt

# Histogram

plt.figure(figsize=(10,6))
plt.hist(df["TotalPay"])
plt.title("distribution of salaries")
plt.xlabel("salary")
plt.ylabel("freq")
plt.show()
```

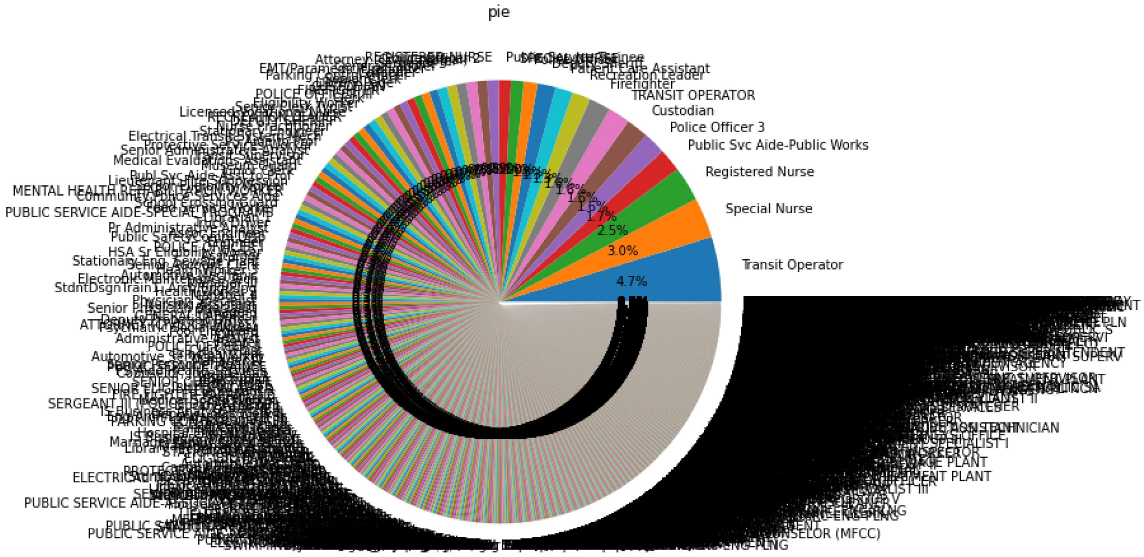


In [ ]:

```
# Bar chart
import matplotlib.pyplot as plt
plt.figure(figsize=(10,6))
plt.bar(df["JobTitle"],df["TotalPay"])
plt.title("distribution of salaries by department")
plt.xlabel("dept")
plt.ylabel("salary")
plt.show()
```

In [19]:

```
# Pie chart
import matplotlib.pyplot as plt
dept_count=df["JobTitle"].value_counts()
plt.figure(figsize=(8,8))
plt.pie(dept_count,labels=dept_count.index,autopct="%1.1f%%")
plt.title("pie")
plt.show()
```



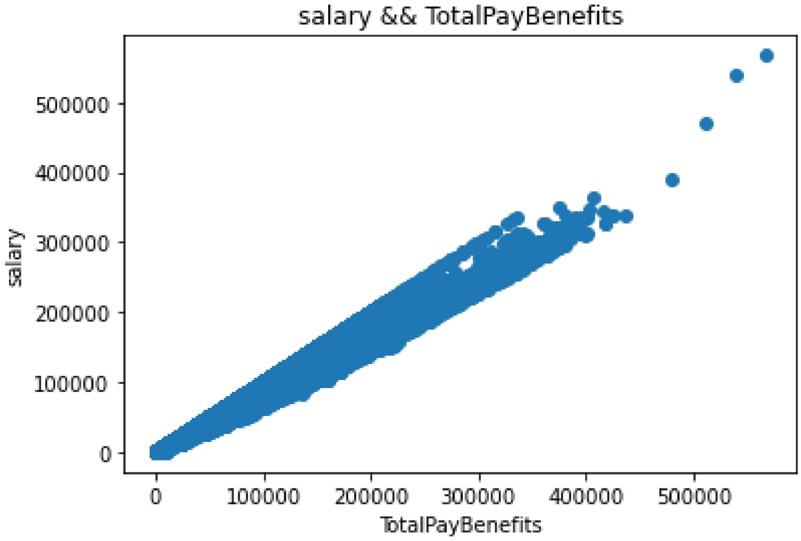
In [10]:

```
sal_dept=df.groupby("JobTitle")["TotalPay"].mean()
print(sal_dept)
```

```
JobTitle
ACCOUNT CLERK                                     44035.664337
ACCOUNTANT                                        47429.268000
ACCOUNTANT INTERN                                 29031.742917
ACPO,JuvP, Juv Prob (SFERS)                       62290.780000
ACUPUNCTURIST                                     67594.400000
                                                      ...
X-RAY LABORATORY AIDE                             52705.880385
X-Ray Laboratory Aide                             50823.942700
YOUTH COMMISSION ADVISOR, BOARD OF SUPERVISORS    53632.870000
Youth Comm Advisor                                41414.307500
ZOO CURATOR                                       66686.560000
Name: TotalPay, Length: 2159, dtype: float64
```

In [12]:

```
import matplotlib.pyplot as plt
correlation=df["TotalPay"].corr(df["TotalPayBenefits"])
plt.scatter(df["TotalPayBenefits"],df["TotalPay"])
plt.title("salary && TotalPayBenefits")
plt.xlabel("TotalPayBenefits")
plt.ylabel("salary")
plt.show()
```



In [ ]:

```
# Summary
#The analyses included a summary of average salaries by department, a cor
#and visualizations to understand the relationships between variables.

#Average Salary by Department:
#The analysis of average salaries by department revealed significant vari

#Correlation Analysis:
#The correlation analysis between salary and another numerical column sho
#correlation between salary and the other variable.
#This suggests that there is a tendency for higher salaries to be associa

#Visualizations:
#The scatter plot visualizations provided a clear depiction of the relati
#the other numerical column. The plots showed a discernible pattern indic
#and strength of the relationship, supporting the findings of the correla


#The analyses conducted on the employee dataset have provided valuable in
#as well as the relationships between salary and other numerical variable
#The findings highlight the importance of considering departmental variat
#for further exploration of the factors influencing salary levels within

#Overall, the analyses have offered a deeper understanding of the salary
#within the organization, laying the groundwork for informed decision-mak
#further investigation into compensation-related matters.
```

Good Luck!