Pandas Mini Project

- This data contains the names, job title, and compensation for San Francisco city employees on an annual basis from 2011 to 2014.

In [ ]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

In [ ]:
```
# loading the dataset
data=pd.read_csv('Salaries.csv',low_memory=False)
```

In [ ]:
```
# viewing the first 10 rows
data.head()
```

In [ ]:
```
# viewing last 10 rows
data.tail()
```

In [ ]:
```
# getting information
data.info()
```

In [ ]:
```
# shape of data
data.shape
```

In [ ]:
```
# calculating the null values
data.isnull().sum()
```

In [ ]:
```
# since benifit,status and notes have a significant number of null values w
data=data.drop(['Notes','Status','Benefits'],axis=1)
```

In [ ]:
```
# checking unique values in Agency column
data.Agency.nunique()
```

In [ ]:
```
# since Agency column has same entry for all we delete the column as we kno
data=data.drop(['Agency'],axis=1)
```

In [ ]:
```
data.head()
```

In [ ]:
```
data.set_index('Id',inplace=True)
```

In [ ]:
```
# checking for duplicates
data.duplicated()
```

In [ ]:
```
# removing duplicates
data.drop_duplicates('EmployeeName',inplace=True)
```

In [ ]:
```
# converting all string entries to title case
data['EmployeeName']=data['EmployeeName'].str.title()
data['JobTitle']=data['JobTitle'].str.title()
```

In [ ]:
```
data.describe()
```

In [ ]:
```
# since base pay overtime and other pay are not coming in the describe tabl
# ran pd.to_numeric(data['BasePay']) which showed string at poisition 11080
```

In [ ]:
```
# fixing error
data.iloc[110809]
```

In [ ]:
```
# deleting the row as information is not provided
data.drop(data.index[110809],axis=0, inplace= True)
```

In [ ]:
```
# converting to numeric
data['BasePay']= pd.to_numeric(data['BasePay'])
data['OvertimePay']=pd.to_numeric(data['OvertimePay'])
data['OtherPay']=pd.to_numeric(data['OtherPay'])
```

In [ ]:
```
# statistical summary
data.describe()
```

In [ ]:
```
# average base pay by year
year_group=data.groupby('Year')
u=pd.DataFrame(year_group['BasePay'].mean())
u
```

In [ ]:
```
# plotting
u.plot(kind='bar')
```

In [ ]:
```
# average base pay as per job title
round(data.groupby('JobTitle')['BasePay'].mean(),2)
```

In [ ]:
```
# complete infromation about person with highest Total pay
data.iloc[data['TotalPay'].idxmax()]
```

In [ ]:
```
# no. of job titles
data['JobTitle'].nunique()
```

In [ ]:
```
# top5 most common jobs
data['JobTitle'].value_counts().head(5)
```

In [ ]:
```
# which job title has highest overtime pay
data[data['OvertimePay'] == data['OvertimePay'].max()]['JobTitle']
```

In [ ]:
```
# Number of employees per year
num_employee=data.groupby('Year')
num_employ_per_year=pd.DataFrame(num_employee.nunique()['EmployeeName'])
num_employ_per_year
```

In [ ]:
```
# visualisation
num_employ_per_year.plot(kind='bar')
```

In [ ]:
```
# histogram that shows distribution of total pay
sns.histplot(data['TotalPay'])
```

In [ ]:
```
# list of people working in Police department
police_employees=data[data['JobTitle'].str.contains('Police')]
police_employees
```

In [ ]:
```

```