Pandas Exercise

Welcome to a quick exercise for you to practice your pandas skills! You will be using the [SF Salaries Dataset] for practicing Pandas! Just follow along and complete the tasks outlined in bold below.

** Import pandas as pd.**

In [19]:
```
import pandas as pd
import numpy as np
```

** Read Salaries.csv as a dataframe called sal.**

In [4]:
```
df1=pd.read_csv('Salaries.csv')
df1
```

Out[4]:

	Id	EmployeeName	JobTitle	BasePay	OvertimePay	OtherPay	Benefits
0	1	NATHANIEL FORD	GENERAL MANAGER-METROPOLITAN TRANSIT AUTHORITY	167411.18	0.00	400184.25	NaN
1	2	GARY JIMENEZ	CAPTAIN III (POLICE DEPARTMENT)	155966.02	245131.88	137811.38	NaN
2	3	ALBERT PARDINI	CAPTAIN III (POLICE DEPARTMENT)	212739.13	106088.18	16452.60	NaN
3	4	CHRISTOPHER CHONG	WIRE ROPE CABLE MAINTENANCE MECHANIC	77916.00	56120.71	198306.90	NaN
4	5	PATRICK GARDNER	DEPUTY CHIEF OF DEPARTMENT, (FIRE DEPARTMENT)	134401.60	9737.00	182234.59	NaN
...	...	...	...	...	...	...	...
148649	148650	Roy I Tillery	Custodian	0.00	0.00	0.00	0.0
148650	148651	Not provided	Not provided	NaN	NaN	NaN	NaN
148651	148652	Not provided	Not provided	NaN	NaN	NaN	NaN
148652	148653	Not provided	Not provided	NaN	NaN	NaN	NaN
148653	148654	Joe Lopez	Counselor, Log Cabin Ranch	0.00	0.00	-618.13	0.0

148654 rows × 13 columns

** Check the head of the DataFrame. **

In [4]:
```
df1.head(5)
```

Out[4]:

	Id	EmployeeName	JobTitle	BasePay	OvertimePay	OtherPay	Benefits	TotalPay	To
0	1	NATHANIEL FORD	GENERAL MANAGER-METROPOLITAN TRANSIT AUTHORITY	167411.18	0.00	400184.25	NaN	567595.43	
1	2	GARY JIMENEZ	CAPTAIN III (POLICE DEPARTMENT)	155966.02	245131.88	137811.38	NaN	538909.28	
2	3	ALBERT PARDINI	CAPTAIN III (POLICE DEPARTMENT)	212739.13	106088.18	16452.60	NaN	335279.91	
3	4	CHRISTOPHER CHONG	WIRE ROPE CABLE MAINTENANCE MECHANIC	77916.00	56120.71	198306.90	NaN	332343.61	
4	5	PATRICK GARDNER	DEPUTY CHIEF OF DEPARTMENT, (FIRE DEPARTMENT)	134401.60	9737.00	182234.59	NaN	326373.19	

** Use the .info() method to find out how many entries there are.**

In [5]:
```
df1.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 148654 entries, 0 to 148653
Data columns (total 13 columns):
 #   Column            Non-Null Count   Dtype  
---  ------            --------------   -----  
 0   Id                148654 non-null  int64  
 1   EmployeeName      148654 non-null  object 
 2   JobTitle          148654 non-null  object 
 3   BasePay           148045 non-null  float64
 4   OvertimePay       148650 non-null  float64
 5   OtherPay          148650 non-null  float64
 6   Benefits          112491 non-null  float64
 7   TotalPay          148654 non-null  float64
 8   TotalPayBenefits  148654 non-null  float64
 9   Year              148654 non-null  int64  
 10  Notes             0 non-null       float64
 11  Agency            148654 non-null  object 
 12  Status            0 non-null       float64
dtypes: float64(8), int64(2), object(3)
memory usage: 14.7+ MB
```

What is the average BasePay ?

In [7]:
```
#66325.44884050643
df1["BasePay"].mean()
```

Out[7]:
```
66325.44884050643
```

** What is the highest amount of OvertimePay in the dataset ? **

In [8]:
```
#245131.88
df1["OvertimePay"].max()
```

Out[8]:
```
245131.88
```

** What is the job title of JOSEPH DRISCOLL ? Note: Use all caps, otherwise you may get an answer that doesn't match up (there is also a lowercase Joseph Driscoll). **

In [10]:
```
#24    CAPTAIN, FIRE SUPPRESSION
#Name: JobTitle, dtype: object
df1[df1['EmployeeName']=='JOSEPH DRISCOLL']
```

Out[10]:

	Id	EmployeeName	JobTitle	BasePay	OvertimePay	OtherPay	Benefits	TotalPay	To
24	25	JOSEPH DRISCOLL	CAPTAIN, FIRE SUPPRESSION	140546.86	97868.77	31909.28	NaN	270324.91	

** How much does JOSEPH DRISCOLL make (including benefits)? **

In [12]:
```
#24    270324.91
#Name: TotalPayBenefits, dtype: float64
df1[df1['EmployeeName']=='JOSEPH DRISCOLL']['TotalPayBenefits']
```

Out[12]:
```
24    270324.91
Name: TotalPayBenefits, dtype: float64
```

** What is the name of highest paid person (including benefits)?**

In [14]:
```
df1[df1['TotalPayBenefits']==df1['TotalPayBenefits'].max()]
```

Out[14]:

	Id	EmployeeName	JobTitle	BasePay	OvertimePay	OtherPay	Benefits	TotalPay	To
0	1	NATHANIEL FORD	GENERAL MANAGER-METROPOLITAN TRANSIT AUTHORITY	167411.18	0.0	400184.25	NaN	567595.43	

** What is the name of lowest paid person (including benefits)? Do you notice something strange about how much he or she is paid?**

In [15]:
```
df1[df1['TotalPayBenefits']==df1['TotalPayBenefits'].min()]
```

Out[15]:

	Id	EmployeeName	JobTitle	BasePay	OvertimePay	OtherPay	Benefits	TotalPay
148653	148654	Joe Lopez	Counselor, Log Cabin Ranch	0.0	0.0	-618.13	0.0	-618.13

** What was the average (mean) BasePay of all employees per year? (2011-2014) ? **

In [52]:
```
df1.groupby('Year').mean()['BasePay']
```

Out[52]:
```
Year
2011    63595.956517
2012    65436.406857
2013    69630.030216
2014    66564.421924
Name: BasePay, dtype: float64
```

** How many unique job titles are there? **

In [25]:
```
#2159
df1['JobTitle'].nunique()
```

Out[25]:
```
2159
```

** What are the top 5 most common jobs? **

In [29]:
```
df1['JobTitle'].value_counts().head()
```

Out[29]:
```
Transit Operator                7036
Special Nurse                   4389
Registered Nurse                3736
Public Svc Aide-Public Works    2518
Police Officer 3                2421
Name: JobTitle, dtype: int64
```

** How many Job Titles were represented by only one person in 2013? (e.g. Job Titles with only one occurence in 2013?) **

```
sum(df1[df1['Year']==2013]['JobTitle'].value_counts()==1)
```

** How many people have the word Chief in their job title? (This is pretty tricky) **

In [39]:
```
def chefname(jobtitle):
    if 'Chief' in jobtitle:
        return True
    else:
        return False
```

In [40]:
```
sum(df1['JobTitle'].apply(lambda x:chefname(x)))
```

Out[40]:
```
423
```

** Bonus: Is there a correlation between length of the Job Title string and Salary? **

In [49]:
```
df1['title_len']=df1['JobTitle'].apply(len)
```

In [50]:
```
df1[['title_len','TotalPayBenefits']].corr()
```

Out[50]:

	title_len	TotalPayBenefits
title_len	1.000000	-0.036878
TotalPayBenefits	-0.036878	1.000000

Great Job!