

```
In [2]: import pandas as pd

In [3]: data = pd.read_csv('Salaries.csv')
data
C:\ProgramData\Anaconda3\lib\site-packages\IPython\core\interactiveshell.py:3165: DtypeWarning: Columns (3,4,5,6,12) have mixed types.Specify dtype option on import or set low_memory=False.
has raised = await self.run_ast_nodes(code_ast.body, cell_name,

Out[3]:
```

	Id	EmployeeName	JobTitle	BasePay	OvertimePay	OtherPay	Benefits	TotalPay	TotalPayBenefits	Year	Notes	Agency	Status	
	0	1	NATHANIEL FORD	GENERAL MANAGER-METROPOLITAN TRANSIT AUTHORITY	167411.18	0.0	400184.25	NaN	567595.43	567595.43	2011	NaN	San Francisco	NaN
	1	2	GARY JIMENEZ	CAPTAIN III (POLICE DEPARTMENT)	155966.02	245131.88	137811.38	NaN	538909.28	538909.28	2011	NaN	San Francisco	NaN
	2	3	ALBERT PARDINI	CAPTAIN III (POLICE DEPARTMENT)	212739.13	106088.18	16452.6	NaN	335279.91	335279.91	2011	NaN	San Francisco	NaN
	3	4	CHRISTOPHER CHONG	WIRE ROPE CABLE MAINTENANCE MECHANIC	77916.0	56120.71	198306.9	NaN	332343.61	332343.61	2011	NaN	San Francisco	NaN
	4	5	PATRICK GARDNER	DEPUTY CHIEF OF DEPARTMENT(FIRE DEPARTMENT)	134401.6	9737.0	182234.59	NaN	326373.19	326373.19	2011	NaN	San Francisco	NaN

	148649	148650	Roy I Tillery	Custodian	0.00	0.00	0.00	0.00	0.00	0.00	2014	NaN	San Francisco	PT
	148650	148651	Not provided	Not provided	Not Provided	Not Provided	Not Provided	0.00	0.00	2014	NaN	San Francisco	NaN	
	148651	148652	Not provided	Not provided	Not Provided	Not Provided	Not Provided	0.00	0.00	2014	NaN	San Francisco	NaN	
	148652	148653	Not provided	Not provided	Not Provided	Not Provided	Not Provided	0.00	0.00	2014	NaN	San Francisco	NaN	
	148653	148654	Joe Lopez	Counselor, Log Cabin Ranch	0.00	0.00	-618.13	0.00	-618.13	-618.13	2014	NaN	San Francisco	PT

148654 rows x 13 columns

```
In [4]: data.head(10)
```

```
Out[4]:
```

	Id	EmployeeName	JobTitle	BasePay	OvertimePay	OtherPay	Benefits	TotalPay	TotalPayBenefits	Year	Notes	Agency	Status
	0	1	NATHANIEL FORD	GENERAL MANAGER-METROPOLITAN TRANSIT AUTHORITY	167411.18	0.0	400184.25	NaN	567595.43	2011	NaN	San Francisco	NaN
	1	2	GARY JIMENEZ	CAPTAIN III (POLICE DEPARTMENT)	155966.02	245131.88	137811.38	NaN	538909.28	2011	NaN	San Francisco	NaN
	2	3	ALBERT PARDINI	CAPTAIN III (POLICE DEPARTMENT)	212739.13	106088.18	16452.6	NaN	335279.91	2011	NaN	San Francisco	NaN
	3	4	CHRISTOPHER CHONG	WIRE ROPE CABLE MAINTENANCE MECHANIC	77916.0	56120.71	198306.9	NaN	332343.61	2011	NaN	San Francisco	NaN
	4	5	PATRICK GARDNER	DEPUTY CHIEF OF DEPARTMENT(FIRE DEPARTMENT)	134401.6	9737.0	182234.59	NaN	326373.19	2011	NaN	San Francisco	NaN
	5	6	DAVID SULLIVAN	ASSISTANT DEPUTY CHIEF II	118902.0	8601.0	189082.74	NaN	316285.74	2011	NaN	San Francisco	NaN
	6	7	ALSON LEE	BATTALION CHIEF (FIRE DEPARTMENT)	92492.01	89062.9	134426.14	NaN	315861.05	2011	NaN	San Francisco	NaN
	7	8	DAVID KUSHNER	DEPUTY DIRECTOR OF INVESTMENTS	256576.96	0.0	51322.5	NaN	307899.46	2011	NaN	San Francisco	NaN
	8	9	MICHAEL MORRIS	BATTALION CHIEF (FIRE DEPARTMENT)	176932.64	86362.68	40132.23	NaN	303427.55	2011	NaN	San Francisco	NaN
	9	10	JOANNE HAYES-WHITE	CHIEF OF DEPARTMENT (FIRE DEPARTMENT)	285262.0	0.0	17115.73	NaN	302377.73	2011	NaN	San Francisco	NaN

```
In [5]: data.tail(10)
```

```
Out[5]:
```

	Id	EmployeeName	JobTitle	BasePay	OvertimePay	OtherPay	Benefits	TotalPay	TotalPayBenefits	Year	Notes	Agency	Status
	148644	148645	Randy D Wern	Stationary Eng. Sewage Plant	0.00	0.00	0.00	0.00	0.00	2014	NaN	San Francisco	PT
	148645	148646	Carolyn A Wilson	Human Services Technician	0.00	0.00	0.00	0.00	0.00	2014	NaN	San Francisco	PT
	148646	148647	Not provided	Not provided	Not Provided	Not Provided	Not Provided	0.00	0.00	2014	NaN	San Francisco	NaN
	148647	148648	Joann Anderson	Communications Dispatcher 2	0.00	0.00	0.00	0.00	0.00	2014	NaN	San Francisco	PT
	148648	148649	Leon Walker	Custodian	0.00	0.00	0.00	0.00	0.00	2014	NaN	San Francisco	PT
	148649	148650	Roy I Tillery	Custodian	0.00	0.00	0.00	0.00	0.00	2014	NaN	San Francisco	PT
	148650	148651	Not provided	Not provided	Not Provided	Not Provided	Not Provided	0.00	0.00	2014	NaN	San Francisco	NaN
	148651	148652	Not provided	Not provided	Not Provided	Not Provided	Not Provided	0.00	0.00	2014	NaN	San Francisco	NaN
	148652	148653	Not provided	Not provided	Not Provided	Not Provided	Not Provided	0.00	0.00	2014	NaN	San Francisco	NaN
	148653	148654	Joe Lopez	Counselor, Log Cabin Ranch	0.00	0.00	-618.13	0.00	-618.13	2014	NaN	San Francisco	PT

```
In [5]: data.shape
```

```
Out[5]: (148654, 13)
```

```
In [6]: print('Number of Rows', data.shape[0])
print('Number of Columns', data.shape[1])

Number of Rows 148654
Number of Columns 13
```

```
In [8]: data.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              148649 non-null  object
4   OvertimePay          148654 non-null  object
5   OtherPay             148654 non-null  object
6   Benefits             112495 non-null  object
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              38119 non-null   object
dtypes: float64(3), int64(2), object(8)
memory usage: 14.7+ MB
```

```
In [9]: data.isnull().sum()
```

```
Out[9]: Id                   0
EmployeeName              0
JobTitle                  0
BasePay                  665
OvertimePay              0
OtherPay                 0
Benefits                36159
TotalPay                 0
TotalPayBenefits         0
Year                     0
Notes                    148654
Agency                  0
Status                  110535
dtype: int64

Drop ID, Notes, Agency, and Status Columns
```

```
In [10]: data.columns
```

```
Out[10]: Index(['Id', 'EmployeeName', 'JobTitle', 'BasePay', 'OvertimePay', 'OtherPay',
        'Benefits', 'TotalPay', 'TotalPayBenefits', 'Year', 'Notes', 'Agency',
        'Status'],
        dtype='object')
```

```
In [13]: data = data.drop(['Id', 'Notes', 'Agency', 'Status'], axis=1)
```

```
In [19]: data.head(1)
```

```
Out[19]:
```

	EmployeeName	JobTitle	BasePay	OvertimePay	OtherPay	Benefits	TotalPay	TotalPayBenefits	Year
0	NATHANIEL FORD	GENERAL MANAGER-METROPOLITAN TRANSIT AUTHORITY	167411.18	0.0	400184.25	NaN	567595.43	567595.43	2011

```
In [20]: data.describe(include="all")
```

```
Out[20]:
```

	EmployeeName	JobTitle	BasePay	OvertimePay	OtherPay	Benefits	TotalPay	TotalPayBenefits	Year
count	148654	148654	148049.0	148654.0	148654.0	112495.0	148654.000000	148654.000000	148654.000000
unique	110811	2159	109900.0	66555.0	84968.0	99635.0	NaN	NaN	NaN
top	Kevin Lee	Transit Operator	0.0	0.0	0.0	0.0	NaN	NaN	NaN
freq	13	7036	875.0	66103.0	35218.0	1053.0	NaN	NaN	NaN
mean	NaN	NaN	NaN	NaN	NaN	NaN	74768.321972	93692.564911	2012.522643
std	NaN	NaN	NaN	NaN	NaN	NaN	50517.005274	62793.533483	1.117538
min	NaN	NaN	NaN	NaN	NaN	NaN	-618.130000	-618.130000	2011.000000
25%	NaN	NaN	NaN	NaN	NaN	NaN	36168.995000	44065.650000	2012.000000
50%	NaN	NaN	NaN	NaN	NaN	NaN	71425.610000	92404.090000	2013.000000
75%	NaN	NaN	NaN	NaN	NaN	NaN	105839.135000	132876.450000	2014.000000
max	NaN	NaN	NaN	NaN	NaN	NaN	567595.430000	567595.430000	2014.000000

Find Occurrence Of The Employee Names (Top 5)

```
In [21]: data.columns
```

```
Out[21]: Index(['EmployeeName', 'JobTitle', 'BasePay', 'OvertimePay', 'OtherPay',
        'Benefits', 'TotalPay', 'TotalPayBenefits', 'Year'],
        dtype='object')
```

```
In [30]: data['EmployeeName'].value_counts().head()
```

```
Out[30]: Kevin Lee      13
Steven Lee      11
Richard Lee     11
William Wong     11
Stanley Lee      9
Name: EmployeeName, dtype: int64

Find The Number of Unique Job Titles
```

```
In [31]: data.columns
```

```
Out[31]: Index(['EmployeeName', 'JobTitle', 'BasePay', 'OvertimePay', 'OtherPay',
        'Benefits', 'TotalPay', 'TotalPayBenefits', 'Year'],
        dtype='object')
```

```
In [36]: data['JobTitle'].nunique()
```

```
Out[36]: 2159

Total Number of Job Titles Contain Captain
```

```
In [37]: data.columns
```

```
Out[37]: Index(['EmployeeName', 'JobTitle', 'BasePay', 'OvertimePay', 'OtherPay',
        'Benefits', 'TotalPay', 'TotalPayBenefits', 'Year'],
        dtype='object')
```

```
In [62]: len(data[data['JobTitle'].str.contains('CAPTAIN', case=False)])
```

```
Out[62]: 552
```

```
In [63]: data[data['JobTitle'].str.contains('CAPTAIN', case=False)].count()
```

```
Out[63]: EmployeeName      552
JobTitle      551
BasePay      551
OvertimePay   552
OtherPay      552
Benefits      411
TotalPay      552
TotalPayBenefits  552
Year          552
dtype: int64

Display All the Employee Names From Fire Department
```

```
In [47]: data.columns
```

```
Out[47]: Index(['EmployeeName', 'JobTitle', 'BasePay', 'OvertimePay', 'OtherPay',
        'Benefits', 'TotalPay', 'TotalPayBenefits', 'Year'],
        dtype='object')
```

```
In [64]: data[data['JobTitle'].str.contains('FIRE DEPARTMENT', case=False)][['EmployeeName']]
```

```
Out[64]:
```

	EmployeeName
4	PATRICK GARDNER
6	ALSON LEE
8	MICHAEL MORRIS
9	JOANNE HAYES-WHITE
10	ARTHUR KENNEY
32623	JAMES BARDEN
36162	Joanne Hayes-White
72326	Joanne M Hayes-White
102303	Robert E Evans
235525	Joanne W Hayes-White

Name: EmployeeName, Length: 226, dtype: object

Find Minimum, Maximum, and Average BasePay

```
In [59]: data.columns
```

```
Out[59]: Index(['EmployeeName', 'JobTitle', 'BasePay', 'OvertimePay', 'OtherPay',
        'Benefits', 'TotalPay', 'TotalPayBenefits', 'Year'],
        dtype='object')
```

```
In [65]: data['BasePay'].describe()
```

```
Out[65]: count      148049.0
unique      109900.0
top          0.0
freq         875.0
Name: BasePay, dtype: float64

Replace 'Not Provided' in EmployeeName Column to NaN
```

```
In [66]: data.columns
```

```
Out[66]: Index(['EmployeeName', 'JobTitle', 'BasePay', 'OvertimePay', 'OtherPay',
        'Benefits', 'TotalPay', 'TotalPayBenefits', 'Year'],
        dtype='object')
```

```
In [69]: import numpy as np
data['EmployeeName']=data['EmployeeName'].replace('Not provided', np.nan)
```

```
In [70]: data['EmployeeName']
```

```
Out[70]:
```

	EmployeeName
0	NATHANIEL FORD
1	GARY JIMENEZ
2	ALBERT PARDINI
3	CHRISTOPHER CHONG
4	PATRICK GARDNER
...	...
148649	Roy I Tillery
148650	NaN
148651	NaN
148652	NaN
148653	Joe Lopez

Name: EmployeeName, Length: 148654, dtype: object

Drop The Rows Having 5 Missing Values

```
In [154]: data.isnull().sum(axis=1) == 5
```

```
Out[154]:
```

	EmployeeName	JobTitle	BasePay	OvertimePay	OtherPay	Benefits	TotalPay	TotalPayBenefits	Year
	ALBERT PARDINI	NaN	212739.13	106088.18	16452.6	NaN	335279.91	335279.91	2011

Find Job Title of ALBERT PARDINI

```
In [80]: data.columns
```

```
Out[80]: Index(['EmployeeName', 'JobTitle', 'BasePay', 'OvertimePay', 'OtherPay',
        'Benefits', 'TotalPay', 'TotalPayBenefits', 'Year'],
        dtype='object')
```

```
In [86]: data[data['EmployeeName'] == 'ALBERT PARDINI']['JobTitle']
```

```
Out[86]:
```

	JobTitle
2	CAPTAIN III (POLICE DEPARTMENT)

Name: JobTitle, dtype: object

How Much ALBERT PARDINI Make (Include Benefits)?

```
In [89]: data.columns
```

```
Out[89]: Index(['EmployeeName', 'JobTitle', 'BasePay', 'OvertimePay', 'OtherPay',
        'Benefits', 'TotalPay', 'TotalPayBenefits', 'Year'],
        dtype='object')
```

```
In [92]: data[data['EmployeeName'] == 'ALBERT PARDINI']['TotalPayBenefits']
```

```
Out[92]:
```

	TotalPayBenefits
2	335279.91

Name: TotalPayBenefits, dtype: float64

Display Name of The Person Having The Highest BasePay

```
In [93]: data.columns
```

```
Out[93]: Index(['EmployeeName', 'JobTitle', 'BasePay', 'OvertimePay', 'OtherPay',
        'Benefits', 'TotalPay', 'TotalPayBenefits', 'Year'],
        dtype='object')
```

```
In [113]: data['BasePay'].dtype
```

```
Out[113]: dtype('O')
```

```
In [116]: data['BasePay']=data['BasePay'].replace('Not Provided', np.nan)
```

```
In [118]: data['BasePay'].dtype
```

```
Out[118]: dtype('O')
```

```
In [121]: data['BasePay'] = data['BasePay'].astype('float')
```

```
In [126]: data[data['BasePay'].max()==data['BasePay']]
```

```
Out[126]:
```

	EmployeeName	JobTitle	BasePay	OvertimePay	OtherPay	Benefits	TotalPay	TotalPayBenefits	Year
72925	Gregory P Suhr	Chief of Police	319275.01	0.0	20007.06	86533.21	339282.07	425915.26	2013

Find Average BasePay of All Employee Per Year

```
In [127]: data.columns
```

```
Out[127]: Index(['EmployeeName', 'JobTitle', 'BasePay', 'OvertimePay', 'OtherPay',
        'Benefits', 'TotalPay', 'TotalPayBenefits', 'Year'],
        dtype='object')
```

```
In [130]: data.groupby('Year')['BasePay'].mean()
```

```
Out[130]: Year
2011    63595.956517
2012    65436.408857
2013    69630.620210
2014    65584.421924
Name: BasePay, dtype: float64

Find Average BasePay of All Employee Per JobTitle
```

```
In [131]: data.columns
```

```
Out[131]: Index(['EmployeeName', 'JobTitle', 'BasePay', 'OvertimePay', 'OtherPay',
        'Benefits', 'TotalPay', 'TotalPayBenefits', 'Year'],
        dtype='object')
```

```
In [134]: data.groupby('JobTitle')['BasePay'].mean()
```

```
Out[134]:
```

	JobTitle
	Transit Operator
	Special Nurse
	Registered Nurse
	Public Svc Aide-Public Works
	Police Officer 3

```
In [ ]:
```