SF Salaries Exercise

Welcome to a quick exercise for you to practice your pandas skills! We will be using the [SF Salaries Dataset] from Kaggle! Just follow along and complete the tasks outlined in bold below. The tasks will get harder and harder as you go along.

** Import pandas as pd.**

```
import pandas as pd
```

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

```
df=pd.read_csv("salaries1.csv")
```

** Check the head of the DataFrame. **

```
df.head()
```

	Id	EmployeeName	JobTitle	BasePay	OvertimePay	OtherPay	Benefits	TotalPay	TotalPayBenefits	Year	Nc
0	1	[illegible]	METROPOLITAN	167411.18	0.00	400184.25	NaN	567595.43	567595.43	2011	

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

```
df.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 ?

```
df["BasePay"].mean()
```

```
66325.44884050643
```

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

```
df["OvertimePay"].max()
```

```
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). **

```
df[df["EmployeeName"]=="JOSEPH DRISCOLL"]["JobTitle"]
```

```
24    CAPTAIN, FIRE SUPPRESSION
Name: JobTitle, dtype: object
```

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

```
df[df["EmployeeName"]=="JOSEPH DRISCOLL"]["TotalPayBenefits"]
```

```
24    270324.91
Name: TotalPayBenefits, dtype: float64
```

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

```
df[df["TotalPayBenefits"]==df["TotalPayBenefits"].max()]
```

Id	EmployeeName	JobTitle	BasePay	OvertimePay	OtherPay	Benefits	To

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

```
df[df["TotalPayBenefits"]==df["TotalPayBenefits"].min()]
```

Id	EmployeeName	JobTitle	BasePay	OvertimePay	OtherPay	Benefits	T

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

```
df.groupby("Year").mean()["BasePay"]
```

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

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

```
df["JobTitle"].nunique()
```

```
2159
```

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

```
jobs=df.groupby("JobTitle").count()
top=jobs.sort_values(by="Id", ascending=False)[:5]
top["Id"]
```

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

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

```
year=df[df["Year"]==2013]
group=year.groupby("JobTitle").count()
count=group[group["Id"]==1]
count.count()["Id"]
```

```
202
```

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

```
def fun(job_title):
    if "chief" in job_title.lower().split():
        return True
    else:
        return False
df=pd.read_csv("salaries1.csv")
sum(df["JobTitle"].apply(lambda x: fun(x)))
```

```
477
```

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

```
df["title_len"]=df["JobTitle"].apply(len)
df[["title_len","TotalPayBenefits"]].corr()
```

```
title_len  TotalPayBenefits
```

Great Job!