Lab1

In [ ]:

```
Name: Bhavesh Waghela
Student ID: N01639685
```

```
Read the Salaries.csv into a dataframe called df_data and use the head() method to check
that you have read in the data correctly. Make sure you import pandas.
```

In [2]:

```
#Write your code here

import pandas as pd

df_data = pd.read_csv('Salaries.csv')
df_data.head(10)
```

Out[2]:

	Id	EmployeeName	JobTitle	BasePay	OvertimePay	OtherPay	Benefits	TotalPay	TotalPayBenefits
0	1	NATHANIEL FORD	GENERAL MANAGER-METROPOLITAN TRANSIT AUTHORITY	167411.18	0.00	400184.25	NaN	567595.43	567595.43
1	2	GARY JIMENEZ	CAPTAIN III (POLICE DEPARTMENT)	155966.02	245131.88	137811.38	NaN	538909.28	538909.28
2	3	ALBERT PARDINI	CAPTAIN III (POLICE DEPARTMENT)	212739.13	106088.18	16452.60	NaN	335279.91	335279.91
3	4	CHRISTOPHER CHONG	WIRE ROPE CABLE MAINTENANCE MECHANIC	77916.00	56120.71	198306.90	NaN	332343.61	332343.61
4	5	PATRICK GARDNER	DEPUTY CHIEF OF DEPARTMENT,(FIRE DEPARTMENT)	134401.60	9737.00	182234.59	NaN	326373.19	326373.19
5	6	DAVID SULLIVAN	ASSISTANT DEPUTY CHIEF II	118602.00	8601.00	189082.74	NaN	316285.74	316285.74
6	7	ALSON LEE	BATTALION CHIEF, (FIRE DEPARTMENT)	92492.01	89062.90	134426.14	NaN	315981.05	315981.05
7	8	DAVID KUSHNER	DEPUTY DIRECTOR OF INVESTMENTS	256576.96	0.00	51322.50	NaN	307899.46	307899.46
8	9	MICHAEL MORRIS	BATTALION CHIEF, (FIRE DEPARTMENT)	176932.64	86362.68	40132.23	NaN	303427.55	303427.55
9	10	JOANNE HAYES-WHITE	CHIEF OF DEPARTMENT, (FIRE DEPARTMENT)	285262.00	0.00	17115.73	NaN	302377.73	302377.73

Use the dtypes attribute to view how each column is stored

In [4]:

```
#Write your code here
df_data.dtypes
```

Out[4]:

```
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
```

Slice the first two columns using .loc and store the result in a variable.

In [27]:

```
#Write you code here

df_empIdName = df_data.loc[:, ["Id", "EmployeeName"]]
df_empIdName
```

Out[27]:

	Id	EmployeeName
0	1	NATHANIEL FORD
1	2	GARY JIMENEZ
2	3	ALBERT PARDINI
3	4	CHRISTOPHER CHONG
4	5	PATRICK GARDNER
...	...	...
148649	148650	Roy I Tillery
148650	148651	Not provided
148651	148652	Not provided
148652	148653	Not provided
148653	148654	Joe Lopez

148654 rows × 2 columns

```
Slice the first two rows using .loc and store the result in a variable
```

In [19]:

```
#Write you code here

df_firstTwoRows = df_data.loc[0:1, :]
df_firstTwoRows
```

Out[19]:

	Id	EmployeeName	JobTitle	BasePay	OvertimePay	OtherPay	Benefits	TotalPay	TotalPayBenefits
0	1	NATHANIEL FORD	GENERAL MANAGER-METROPOLITAN TRANSIT AUTHORITY	167411.18	0.00	400184.25	NaN	567595.43	567595.43
1	2	GARY JIMENEZ	CAPTAIN III (POLICE DEPARTMENT)	155966.02	245131.88	137811.38	NaN	538909.28	538909.28

Slice the first two rows using .loc and store the result in a variable called result_2.

In [20]:

```
#Write you code here
result_2 = df_data.loc[0:1, :]
result_2
```

Out[20]:

	Id	EmployeeName	JobTitle	BasePay	OvertimePay	OtherPay	Benefits	TotalPay	TotalPayBenefits
0	1	NATHANIEL FORD	GENERAL MANAGER-METROPOLITAN TRANSIT AUTHORITY	167411.18	0.00	400184.25	NaN	567595.43	567595.43
1	2	GARY JIMENEZ	CAPTAIN III (POLICE DEPARTMENT)	155966.02	245131.88	137811.38	NaN	538909.28	538909.28

Slice the first four rows and the first five columns and store the result in a variable called result_3.

In [22]:

```
#Write you code here
result_3 = df_data.loc[0:3,"Id":"OvertimePay"]
result_3
```

Out[22]:

	Id	EmployeeName	JobTitle	BasePay	OvertimePay
0	1	NATHANIEL FORD	GENERAL MANAGER-METROPOLITAN TRANSIT AUTHORITY	167411.18	0.00
1	2	GARY JIMENEZ	CAPTAIN III (POLICE DEPARTMENT)	155966.02	245131.88
2	3	ALBERT PARDINI	CAPTAIN III (POLICE DEPARTMENT)	212739.13	106088.18
3	4	CHRISTOPHER CHONG	WIRE ROPE CABLE MAINTENANCE MECHANIC	77916.00	56120.71

Slice rows 0,4,6 and columns invoice time and price and store the result in variable called result_4.

In [26]:

```
#Write you code here
result_4 = df_data.loc[[0,4,6],["JobTitle", "BasePay"]]
result_4
```

Out[26]:

	JobTitle	BasePay
0	GENERAL MANAGER-METROPOLITAN TRANSIT AUTHORITY	167411.18
4	DEPUTY CHIEF OF DEPARTMENT,(FIRE DEPARTMENT)	134401.60
6	BATTALION CHIEF, (FIRE DEPARTMENT)	92492.01

In [ ]:

```
Store the number rows in a variable called num_rows.
```

In [34]:

```
#Write you code here
num_rows = len(df_data)
num_rows
```

Out[34]:

```
148654
```

Print out the last row of the data to dataframe. **Hint:** use the variable num_rows from the previous exercise.

In [38]:

```
#Write you code here
df_lastRow = df_data.tail(1)
df_lastRow
```

Out[38]:

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

In [ ]:

```
```

Compute the average and max TotalPay. Store the results in variables called avg_TotalPay and max_TotalPay

In [46]:

```
#Write your code here
avg_TotalPy = df_data['TotalPay'].mean()
max_TotalPy = df_data['TotalPay'].max()

print(f"""Max TotalPay: {max_TotalPy}
Average TotalPay: {avg_TotalPy}""")
```

```
Max TotalPay: 567595.43
Average TotalPay: 74768.321971703
```

```
Create a column called "final", which is BasePay*2.
```

In [48]:

```
#Write your code here
df_data['final'] = df_data['BasePay']*2
df_data.head(10)
```

Out[48]:

	Id	EmployeeName	JobTitle	BasePay	OvertimePay	OtherPay	Benefits	TotalPay	TotalPayBenefits
0	1	NATHANIEL FORD	GENERAL MANAGER-METROPOLITAN TRANSIT AUTHORITY	167411.18	0.00	400184.25	NaN	567595.43	567595.43
1	2	GARY JIMENEZ	CAPTAIN III (POLICE DEPARTMENT)	155966.02	245131.88	137811.38	NaN	538909.28	538909.28
2	3	ALBERT PARDINI	CAPTAIN III (POLICE DEPARTMENT)	212739.13	106088.18	16452.60	NaN	335279.91	335279.91
3	4	CHRISTOPHER CHONG	WIRE ROPE CABLE MAINTENANCE MECHANIC	77916.00	56120.71	198306.90	NaN	332343.61	332343.61
4	5	PATRICK GARDNER	DEPUTY CHIEF OF DEPARTMENT,(FIRE DEPARTMENT)	134401.60	9737.00	182234.59	NaN	326373.19	326373.19
5	6	DAVID SULLIVAN	ASSISTANT DEPUTY CHIEF II	118602.00	8601.00	189082.74	NaN	316285.74	316285.74
6	7	ALSON LEE	BATTALION CHIEF, (FIRE DEPARTMENT)	92492.01	89062.90	134426.14	NaN	315981.05	315981.05
7	8	DAVID KUSHNER	DEPUTY DIRECTOR OF INVESTMENTS	256576.96	0.00	51322.50	NaN	307899.46	307899.46
8	9	MICHAEL MORRIS	BATTALION CHIEF, (FIRE DEPARTMENT)	176932.64	86362.68	40132.23	NaN	303427.55	303427.55
9	10	JOANNE HAYES-WHITE	CHIEF OF DEPARTMENT, (FIRE DEPARTMENT)	285262.00	0.00	17115.73	NaN	302377.73	302377.73

```
Use the drop() method to delete the column OvertimePay from the dataframe df_data.
```

In [50]:
```
#Write your code here
df_data.drop(['OvertimePay'], inplace=True, axis=1)
df_data.head(10)
```

Out[50]:

	Id	EmployeeName	JobTitle	BasePay	OtherPay	Benefits	TotalPay	TotalPayBenefits	Year	Notes
0	1	NATHANIEL FORD	GENERAL MANAGER-METROPOLITAN TRANSIT AUTHORITY	167411.18	400184.25	NaN	567595.43	567595.43	2011	NaN
1	2	GARY JIMENEZ	CAPTAIN III (POLICE DEPARTMENT)	155966.02	137811.38	NaN	538909.28	538909.28	2011	NaN
2	3	ALBERT PARDINI	CAPTAIN III (POLICE DEPARTMENT)	212739.13	16452.60	NaN	335279.91	335279.91	2011	NaN
3	4	CHRISTOPHER CHONG	WIRE ROPE CABLE MAINTENANCE MECHANIC	77916.00	198306.90	NaN	332343.61	332343.61	2011	NaN
4	5	PATRICK GARDNER	DEPUTY CHIEF OF DEPARTMENT,(FIRE DEPARTMENT)	134401.60	182234.59	NaN	326373.19	326373.19	2011	NaN
5	6	DAVID SULLIVAN	ASSISTANT DEPUTY CHIEF II	118602.00	189082.74	NaN	316285.74	316285.74	2011	NaN
6	7	ALSON LEE	BATTALION CHIEF, (FIRE DEPARTMENT)	92492.01	134426.14	NaN	315981.05	315981.05	2011	NaN
7	8	DAVID KUSHNER	DEPUTY DIRECTOR OF INVESTMENTS	256576.96	51322.50	NaN	307899.46	307899.46	2011	NaN
8	9	MICHAEL MORRIS	BATTALION CHIEF, (FIRE DEPARTMENT)	176932.64	40132.23	NaN	303427.55	303427.55	2011	NaN
9	10	JOANNE HAYES-WHITE	CHIEF OF DEPARTMENT, (FIRE DEPARTMENT)	285262.00	17115.73	NaN	302377.73	302377.73	2011	NaN

In this set of practice exercises, we will be working with a demographic data regarding the passengers aboard the Titanic. Read in the data frame and use the head() method to check that it was read in correctly.

In [51]:

```
import pandas as pd
#Write your code here
df_titanic_data = pd.read_csv('Titanic.csv')
df_titanic_data.head(10)
```

Out[51]:

	PassengerId	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
0	892	3	Kelly, Mr. James	male	34.5	0	0	330911	7.8292	NaN	Q
1	893	3	Wilkes, Mrs. James (Ellen Needs)	female	47.0	1	0	363272	7.0000	NaN	S
2	894	2	Myles, Mr. Thomas Francis	male	62.0	0	0	240276	9.6875	NaN	Q
3	895	3	Wirz, Mr. Albert	male	27.0	0	0	315154	8.6625	NaN	S
4	896	3	Hirvonen, Mrs. Alexander (Helga E Lindqvist)	female	22.0	1	1	3101298	12.2875	NaN	S
5	897	3	Svensson, Mr. Johan Cervin	male	14.0	0	0	7538	9.2250	NaN	S
6	898	3	Connolly, Miss. Kate	female	30.0	0	0	330972	7.6292	NaN	Q
7	899	2	Caldwell, Mr. Albert Francis	male	26.0	1	1	248738	29.0000	NaN	S
8	900	3	Abrahim, Mrs. Joseph (Sophie Halaut Easu)	female	18.0	0	0	2657	7.2292	NaN	C
9	901	3	Davies, Mr. John Samuel	male	21.0	2	0	A/4 48871	24.1500	NaN	S

Use the rename method to change the column "Name" to "Passenger_Name" and the column "Ticket" to "Ticket_Num".

In [53]:

```
#Write your code here
df_titanic_data.rename(columns={'Name':'Passenger_Name', 'Ticket':'Ticket_Num'}, inplace=Tru
df_titanic_data.head(10)
```

Out[53]:

	PassengerId	Pclass	Passenger_Name	Sex	Age	SibSp	Parch	Ticket_Num	Fare	Cabin	Embarked
0	892	3	Kelly, Mr. James	male	34.5	0	0	330911	7.8292	NaN	Q
1	893	3	Wilkes, Mrs. James (Ellen Needs)	female	47.0	1	0	363272	7.0000	NaN	S
2	894	2	Myles, Mr. Thomas Francis	male	62.0	0	0	240276	9.6875	NaN	Q
3	895	3	Wirz, Mr. Albert	male	27.0	0	0	315154	8.6625	NaN	S
4	896	3	Hirvonen, Mrs. Alexander (Helga E Lindqvist)	female	22.0	1	1	3101298	12.2875	NaN	S
5	897	3	Svensson, Mr. Johan Cervin	male	14.0	0	0	7538	9.2250	NaN	S
6	898	3	Connolly, Miss. Kate	female	30.0	0	0	330972	7.6292	NaN	Q
7	899	2	Caldwell, Mr. Albert Francis	male	26.0	1	1	248738	29.0000	NaN	S
8	900	3	Abrahim, Mrs. Joseph (Sophie Halaut Easu)	female	18.0	0	0	2657	7.2292	NaN	C
9	901	3	Davies, Mr. John Samuel	male	21.0	2	0	A/4 48871	24.1500	NaN	S

Select the name of passenger 896

In [56]:

```
#Write your code here
passenger_896 = df_titanic_data.loc[4,'Passenger_Name']
passenger_896
```

Out[56]:

```
'Hirvonen, Mrs. Alexander (Helga E Lindqvist)'
```

How many missing entries are there in the Age column?

In [71]:

```
#Write you code here
df_titanic_data.isnull().sum()["Age"]
```

Out[71]:

```
86
```

Compute the avg age of passengers ignoring the missing data.

In [77]:

```
#Write your code her
avg_age_df = df_titanic_data.Age.dropna()
avg_age_df.mean()
```

Out[77]:

```
30.272590361445783
```

Using the fillna() method replace the missing values in the Age column with the mean.

In [81]:

```
#Write your code here
fill_na_df =  df_titanic_data.Age.fillna(avg_age_df.mean())
fill_na_df
```

Out[81]:

```
0      34.50000
1      47.00000
2      62.00000
3      27.00000
4      22.00000
         ...   
413    30.27259
414    39.00000
415    38.50000
416    30.27259
417    30.27259
Name: Age, Length: 418, dtype: float64
```

In [ ]:

```
#Bonus: for students who wants to practice more
```

```
What is the average age of the 5 oldest passengers? The reset_index method will be helpful
here.
```

In [ ]:

```
#Write your code here
```

Read this xlsx into a dataframe called df_data and use the head() method to check that you have read in the data correctly. Make sure you import pandas. then fill the missing values if it has.

In [ ]: