EmployeesSQL

September 6, 2020

[ ]:
```
#import dependencies

from sqlalchemy import create_engine
import pandas as pd
import matplotlib.pyplot as plt
```

[2]:
```
#create a connection to the sql server/database

engine = create_engine('postgresql://postgres:password@localhost:5432/Sql_HW')
connection = engine.connect()
```

[3]:
```
#query the salaries table

salaries=pd.read_sql("SELECT * FROM salaries",connection)
salaries.head()
```

[3]:

	emp_no	salary
0	10001	60117
1	10002	65828
2	10003	40006
3	10004	40054
4	10005	78228

[4]:
```
#query the employees table

employees = pd.read_sql("select * from employees", connection)
employees.head()
```

[4]:

	emp_no	emp_title_id	birth_date	first_name	last_name	sex	hire_date
0	473302	s0001	1953-07-25	Hideyuki	Zallocco	M	1990-04-28
1	475053	e0002	1954-11-18	Byong	Delgrande	F	1991-09-07
2	57444	e0002	1958-01-30	Berry	Babb	F	1992-03-21
3	421786	s0001	1957-09-28	Xiong	Verhoeff	M	1987-11-26
4	282238	e0003	1952-10-28	Abdelkader	Baumann	F	1991-01-18

[5]:
```
#query the titles table
```

```
titles = pd.read_sql("select * from titles",connection)
titles.head()
```

[5]:
```
  title_id               title
0    s0001               Staff
1    s0002        Senior Staff
2    e0001  Assistant Engineer
3    e0002            Engineer
4    e0003     Senior Engineer
```

[6]:
```
#these are the three tables we will join
#first, join salaries to employees on emp_no

emp_sal = employees.merge(salaries, on = "emp_no")
emp_sal.head()
```

[6]:
```
    emp_no emp_title_id  birth_date  first_name  last_name sex   hire_date  \
0  473302        s0001  1953-07-25    Hideyuki   Zallocco   M  1990-04-28
1  475053        e0002  1954-11-18       Byong  Delgrande   F  1991-09-07
2   57444        e0002  1958-01-30       Berry       Babb   F  1992-03-21
3  421786        s0001  1957-09-28       Xiong   Verhoeff   M  1987-11-26
4  282238        e0003  1952-10-28  Abdelkader    Baumann   F  1991-01-18

   salary
0   40000
1   53422
2   48973
3   40000
4   40000
```

[7]:
```
#next, join the titles table to the newly created db on emp_title_id and␣
 ↪title_id

final_db = emp_sal.merge(titles, left_on = "emp_title_id", right_on =␣
 ↪"title_id")
final_db.head()
```

[7]:
```
    emp_no emp_title_id  birth_date  first_name   last_name sex   hire_date  \
0  473302        s0001  1953-07-25    Hideyuki    Zallocco   M  1990-04-28
1  421786        s0001  1957-09-28       Xiong    Verhoeff   M  1987-11-26
2  273487        s0001  1957-04-14   Christoph     Parfitt   M  1991-06-28
3  246449        s0001  1958-03-23       Subbu  Bultermann   F  1988-03-25
4   48085        s0001  1964-01-19  Venkatesan        Gilg   M  1993-06-28

   salary title_id  title
0   40000    s0001  Staff
1   40000    s0001  Staff
```

```
2   56087    s0001  Staff
3   87084    s0001  Staff
4   63016    s0001  Staff
```

[8]:
```
#now we can extract a db of only the titles and salaries

sal_title_db = final_db[['salary','title']]
sal_title_db.head()
```

[8]:
```
   salary  title
0   40000  Staff
1   40000  Staff
2   56087  Staff
3   87084  Staff
4   63016  Staff
```

[9]:
```
#just to check the number of rows

sal_title_db.count()
```

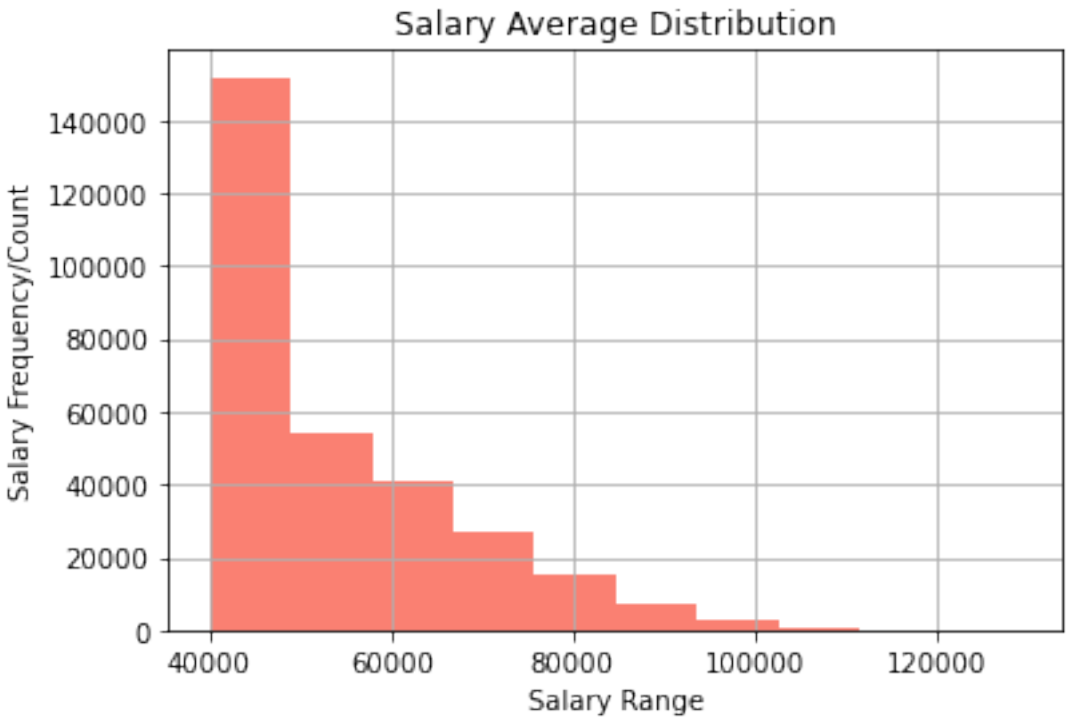
[9]:
```
salary    300024
title     300024
dtype: int64
```

[10]:
```
#in order to graph, the titles should be grouped using groupby and mean() for␣
 ↪the salaries
#edited to round() to zero digits as they don't add anything to the data

sal_title_db.groupby('title')['salary'].mean().round(0)
```

[10]:
```
title
Assistant Engineer    48564.0
Engineer              48535.0
Manager               51531.0
Senior Engineer       48507.0
Senior Staff          58550.0
Staff                 58465.0
Technique Leader      48583.0
Name: salary, dtype: float64
```

[16]:
```
#now we can graph these salaries

sal_title_db.hist(column='salary',color = 'salmon')
plt.xlabel('Salary Range')
plt.ylabel('Salary Frequency/Count')
plt.title('Salary Average Distribution')
```

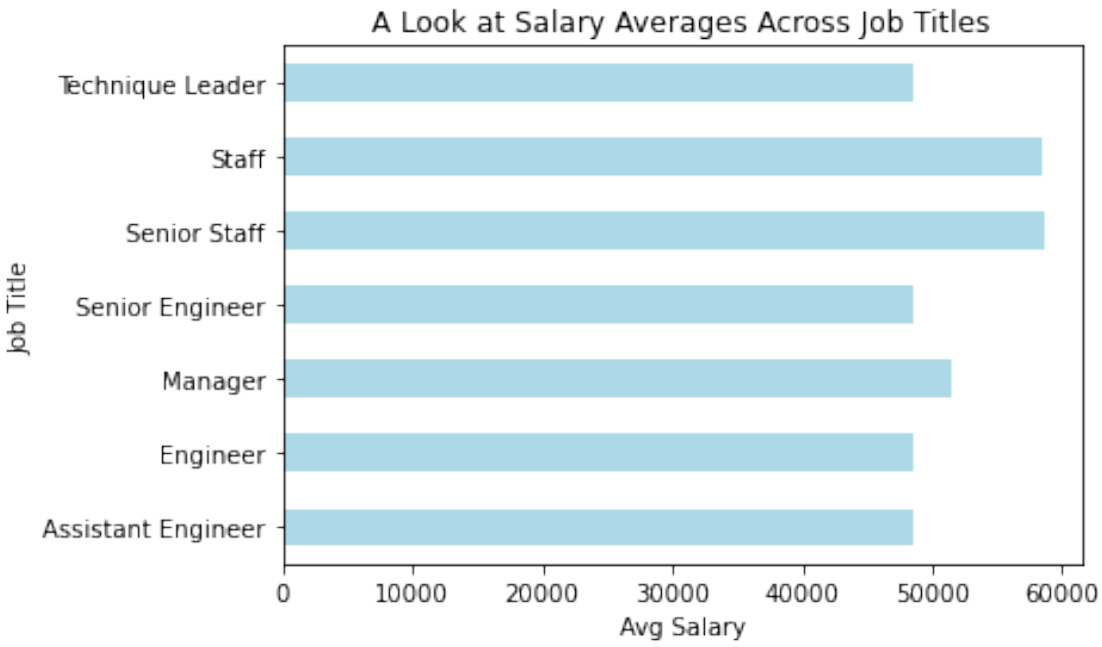
[16]: Text(0.5, 1.0, 'Salary Average Distribution')



1 This seems like an awful lot of salaries on the low end over two decades, but it is possible that these are low paying titles.

[19]:
```
# now we can take a look at the salaries by title

sal_title_db2 = sal_title_db.groupby(['title'])['salary'].mean()
sal_title_db2.plot.barh(color='lightblue')
plt.ylabel('Job Title')
plt.xlabel('Avg Salary')
plt.title('A Look at Salary Averages Across Job Titles')
plt.show()
```

[19]: Text(0.5, 1.0, 'A Look at Salary Averages Across Job Titles')



2 I'm 99.999999999999% sure that Senior Engineers do not make the lowest salaries on average. Most likely, this data is not real.

[ ]: