mystery_bonus

September 22, 2019

0.0.1 UR Bootcamp student: Midlo-Marie

0.0.2 Homework #9: Postgres SQL and Python

0.1 *A Mystery in Two Parts*

0.2 Bonus (Optional)

- Import the SQL database into Pandas using SQLAlchemy
- Create a bar chart of average salary by title and compare salaries
- Find out why the boss asked you to check emp_no 499942

[1]:
```
# Create dependencies for Pandas, plotting, and numerical analysis
# Pandas
import pandas as pd

# Matplotlib
import matplotlib.pyplot as plt

# NumPy
import numpy as np
```

[2]:
```
# Make connection between Postgres database and Python

from sqlalchemy import create_engine

# Create Engine and Pass in Postgres Connection
# engine = create_engine('postgres://postgres:PostgresPW@localhost:5432/
 ↪mystery_lc_db')
engine = create_engine('postgres://postgres:Nikki2007@localhost:5432/
 ↪mystery_lc_db')
conn = engine.connect()
```

[7]:
```
# Query All Records in the Salaries Table
salaries_data = pd.read_sql("SELECT * FROM salaries", conn)
salaries_data.head()
```

[7]:

	emp_no	salary	from_date	to_date
0	10001	60117	1986-06-26	1987-06-26
1	10002	65828	1996-08-03	1997-08-03

```
2   10003   40006  1995-12-03  1996-12-02
3   10004   40054  1986-12-01  1987-12-01
4   10005   78228  1989-09-12  1990-09-12
```

[8]:
```
# Query All Records in the Titles Table
titles_data = pd.read_sql("SELECT * FROM titles", conn)
titles_data.head(10)
```

[8]:
```
   emp_no            title   from_date     to_date
0   10001  Senior Engineer  1986-06-26  9999-01-01
1   10002            Staff  1996-08-03  9999-01-01
2   10003  Senior Engineer  1995-12-03  9999-01-01
3   10004         Engineer  1986-12-01  1995-12-01
4   10004  Senior Engineer  1995-12-01  9999-01-01
5   10005     Senior Staff  1996-09-12  9999-01-01
6   10005            Staff  1989-09-12  1996-09-12
7   10006  Senior Engineer  1990-08-05  9999-01-01
8   10007     Senior Staff  1996-02-11  9999-01-01
9   10007            Staff  1989-02-10  1996-02-11
```

[9]:
```
# Merge the tables on emp_no using an INNER join
merge_tandsal = pd.merge(salaries_data, titles_data, on="emp_no", how="inner")
merge_tandsal.head()
```

[9]:
```
   emp_no  salary from_date_x   to_date_x            title from_date_y  \
0   10001   60117  1986-06-26  1987-06-26  Senior Engineer  1986-06-26
1   10002   65828  1996-08-03  1997-08-03            Staff  1996-08-03
2   10003   40006  1995-12-03  1996-12-02  Senior Engineer  1995-12-03
3   10004   40054  1986-12-01  1987-12-01         Engineer  1986-12-01
4   10004   40054  1986-12-01  1987-12-01  Senior Engineer  1995-12-01

    to_date_y
0  9999-01-01
1  9999-01-01
2  9999-01-01
3  1995-12-01
4  9999-01-01
```

[10]:
```
# Compute the average salary grouped by title
grouped_merge = merge_tandsal.groupby("title").mean().round()
grouped_merge.head(100)
```

[10]:
```
                      emp_no   salary
title
Assistant Engineer  251495.0  48493.0
Engineer            252943.0  48540.0
Manager             110781.0  51531.0
Senior Engineer     253034.0  48507.0
Senior Staff        253423.0  58503.0
Staff               253399.0  58465.0
```

```
Technique Leader    251709.0  48581.0
```

[11]:
```
# Don't need the "average" employee number, so drop the emp_column and reindex
title_and_sal = grouped_merge.drop(columns="emp_no").sort_values(by="salary",␣
 ↪ascending=False)
title_and_sal.head(100)
```

[11]:
```
                     salary
title
Senior Staff        58503.0
Staff               58465.0
Manager             51531.0
Technique Leader    48581.0
Engineer            48540.0
Senior Engineer     48507.0
Assistant Engineer  48493.0
```

0.2.1 Finding: Comparison of average salaries

All of the **technical** positions are paid nearly equally, around 48K. First level management makes a bit more, about 51K. Staff and senior staff make the most at about 58K.

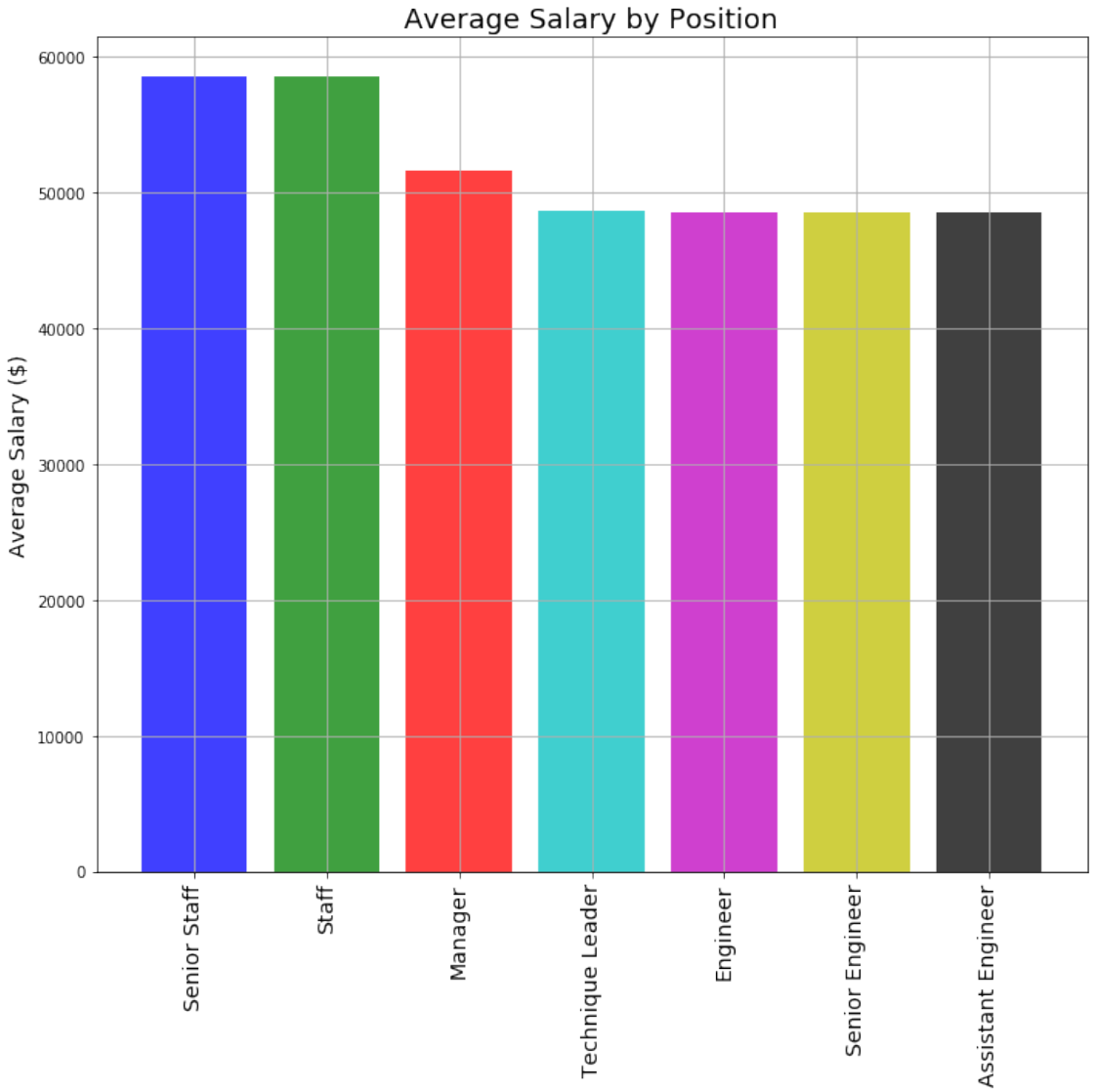
[34]:
```
# Make a bar graph of the salary values by title
# Initialize the plotting parameters for the graph

plt.figure(figsize=(10,10))
plt.title('Average Salary by Position', fontsize=18)
xdata = title_and_sal.index
# print(xlabels)
plt.xticks(np.arange(len(xdata)), xdata, fontsize=14, rotation="vertical")
plt.ylabel('Average Salary ($)', fontsize=14)
plt.grid()

ydata = title_and_sal["salary"]

plt.bar(xdata, ydata, align="center", alpha=0.75, color=['b', 'g', 'r', 'c',␣
 ↪'m', 'y', 'k', 'w'])

# Save the Figure
plt.tight_layout()
plt.savefig("./Average_salary.png")
```



0.2.2 Finding: What was the boss getting at?

Haha… funny guy. Jokes on us.

[36]:
```
# Query All Records in the Salaries Table
question_id = pd.read_sql("SELECT * FROM employees WHERE emp_no=499942", conn)
question_id
```

[36]:

	emp_no	birth_date	first_name	last_name	gender	hire_date
0	499942	1963-01-10	April	Foolsday	F	1997-02-10