In [ ]:

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import random as rd
```

In [ ]:

```
dataset = pd.read_csv('D:/DSML/assignments/abc.csv')
```

In [ ]:

```
dataset["Height"] = np.random.randint(150,180,size=len(dataset) )
```

In [ ]:

```
dataset.duplicated().sum()
```

Out[ ]:

```
0
```

In [ ]:

```
dataset.isnull().sum()
```

Out[ ]:

```
Name         0
Team         0
Number       0
Position     0
Age          0
Height       0
Weight       0
College     84
Salary      11
dtype: int64
```

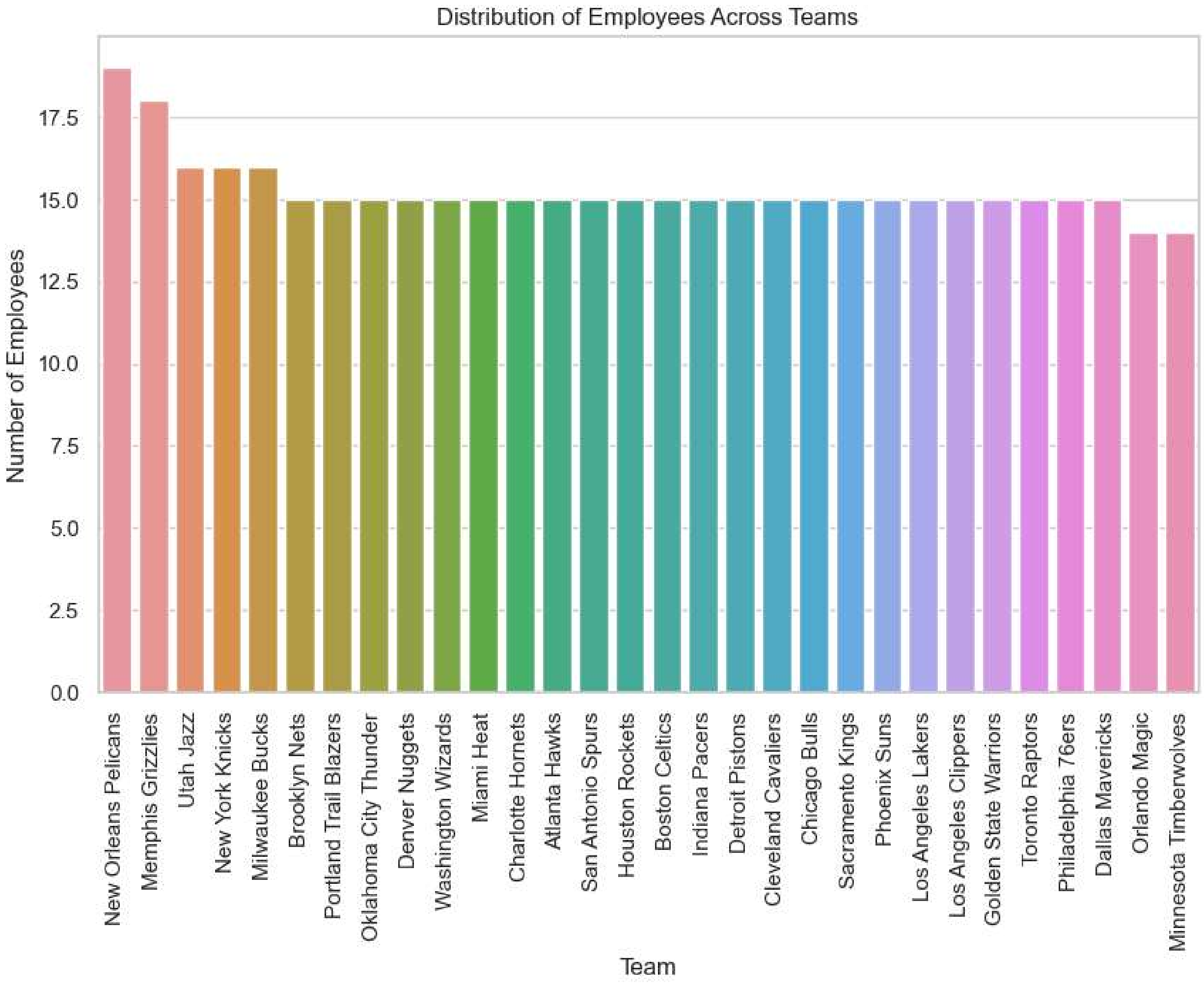
In [ ]:

```
#distribution of employees across each team
total_team = dataset['Team'].value_counts()
print(total_team)
```

```
New Orleans Pelicans      19
Memphis Grizzlies         18
Utah Jazz                 16
New York Knicks           16
Milwaukee Bucks           16
Brooklyn Nets             15
Portland Trail Blazers    15
Oklahoma City Thunder     15
Denver Nuggets            15
Washington Wizards        15
Miami Heat                15
Charlotte Hornets         15
Atlanta Hawks             15
San Antonio Spurs         15
Houston Rockets           15
Boston Celtics            15
Indiana Pacers            15
Detroit Pistons           15
Cleveland Cavaliers       15
Chicago Bulls             15
Sacramento Kings          15
Phoenix Suns              15
Los Angeles Lakers        15
Los Angeles Clippers      15
Golden State Warriors     15
Toronto Raptors           15
Philadelphia 76ers        15
Dallas Mavericks          15
Orlando Magic             14
Minnesota Timberwolves    14
Name: Team, dtype: int64
```

In [ ]:

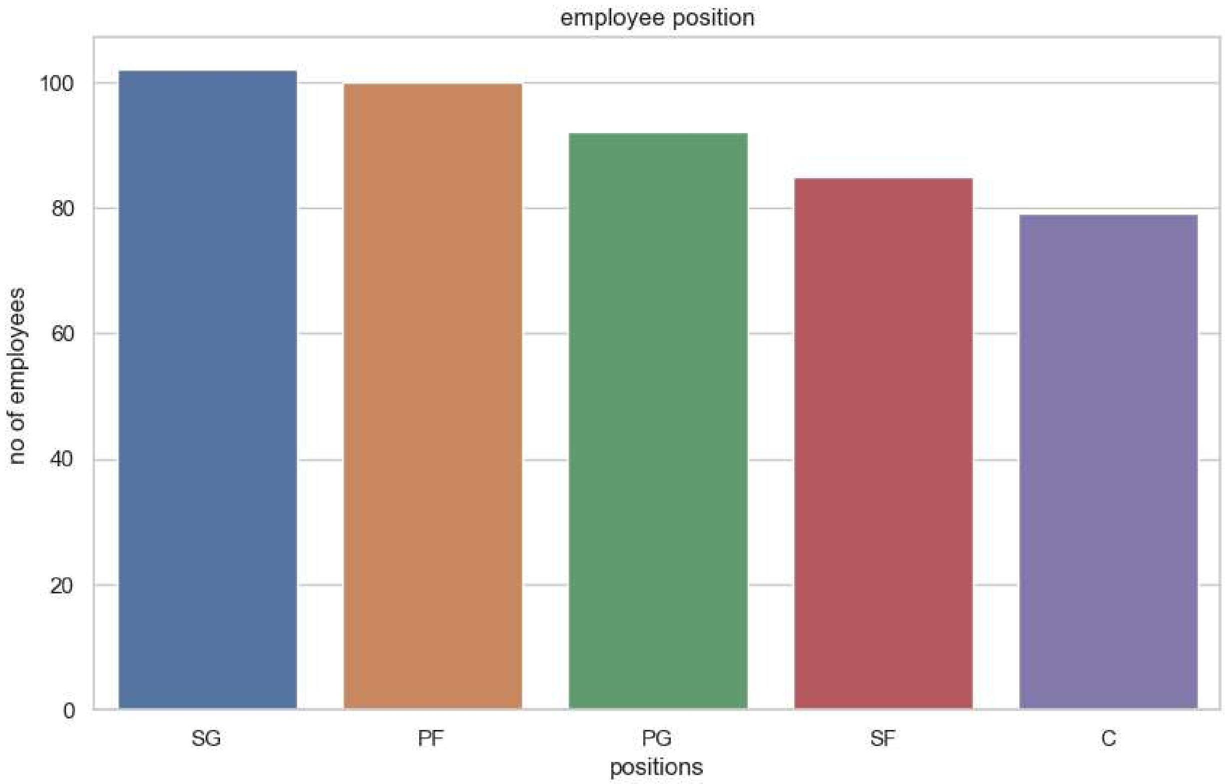
```
# calculate the percentage split relative to the total number of employees
total_employees = len(dataset)
team_percentage = (total_team/total_employees)*100
#graphical representation
plt.figure(figsize=(10,6))
sns.barplot(x=total_team.index, y=total_team.values)
plt.title('Distribution of Employees Across Teams')
plt.xlabel('Team')
plt.ylabel('Number of Employees')
plt.xticks(rotation=90)
plt.show()
```



The majority of employees are concentrated in certain teams, indicating potential areas of focus for the company. For example, if 'New Orleans Pelicans' has the largest number of employees, it could be a key driver of company operations.

In [ ]:

```
#Segregate employees based on their positions within the company
employee_position = dataset['Position'].value_counts()
plt.figure(figsize=(10,6))
sns.barplot(x= employee_position.index, y = employee_position.values)
plt.title('employee position')
plt.xlabel('positions')
plt.ylabel('no of employees')
plt.show()
```

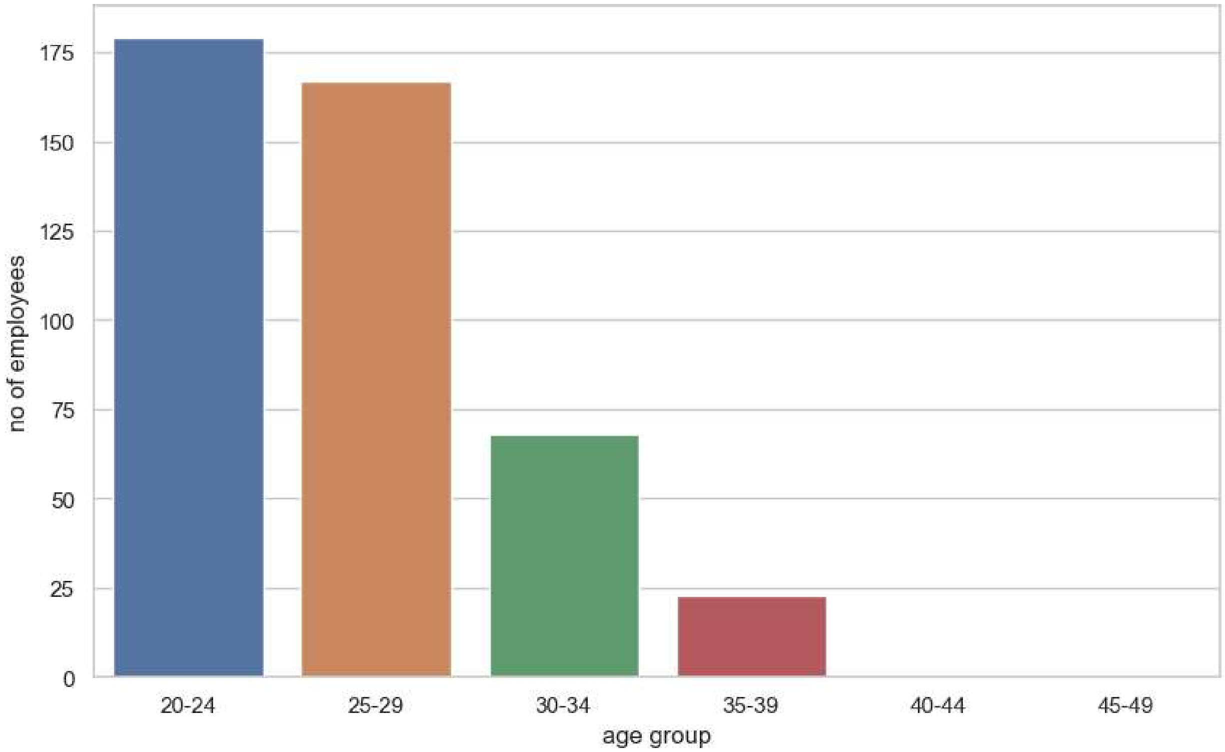


The distribution across positions reveals which roles are most common in the organization. If 'Position SG' is predominant, it may be a critical role for the company's success.

In [ ]:

```
#predominant age group
age_value = dataset['Age'].value_counts()
age_bins = [20, 25, 30, 35, 40, 45, 50]
age_labels = ['20-24', '25-29', '30-34', '35-39', '40-44', '45-49']
dataset['age_group'] = pd.cut(dataset['Age'], bins=age_bins, labels=age_labels)
print(dataset['age_group'])
age_group = dataset['age_group'].value_counts()
print(age_group)
# graphical representation
plt.figure(figsize=(10,6))
sns.barplot(x = age_group.index, y = age_group.values)
plt.xlabel('age group')
plt.ylabel('no of employees')
plt.show()
```

```
0      20-24
1      20-24
2      25-29
3      20-24
4      25-29
       ...
453    25-29
454    20-24
455    25-29
456    25-29
457    20-24
Name: age_group, Length: 458, dtype: category
Categories (6, object): ['20-24' < '25-29' < '30-34' < '35-39' < '40-44' < '45-49']
20-24    179
25-29    167
30-34     68
35-39     23
40-44      0
45-49      0
Name: age_group, dtype: int64
```



The most common age group provides insight into the demographic profile of the workforce. This can inform HR strategies, particularly in areas such as employee engagement and retention

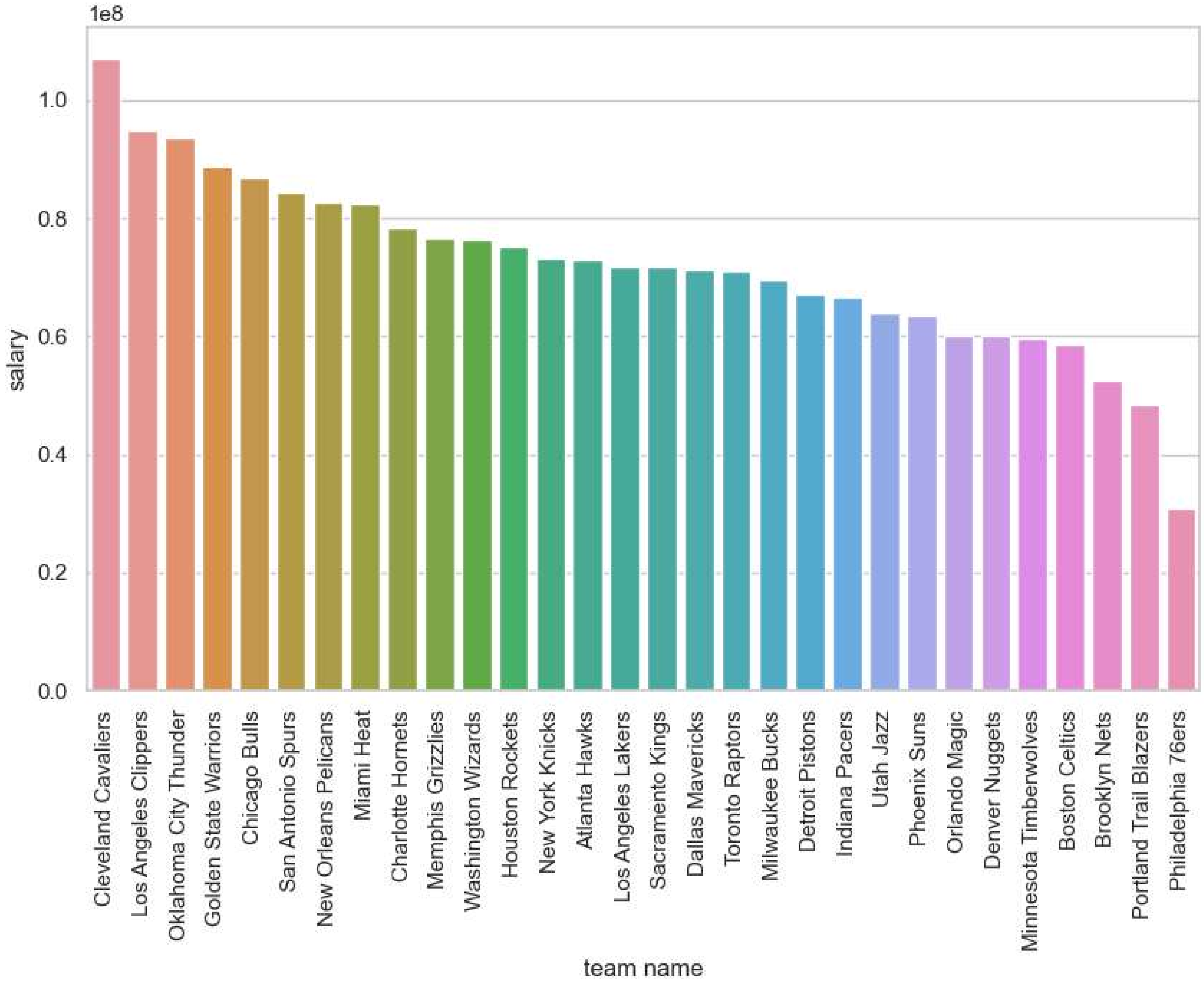
In [ ]:

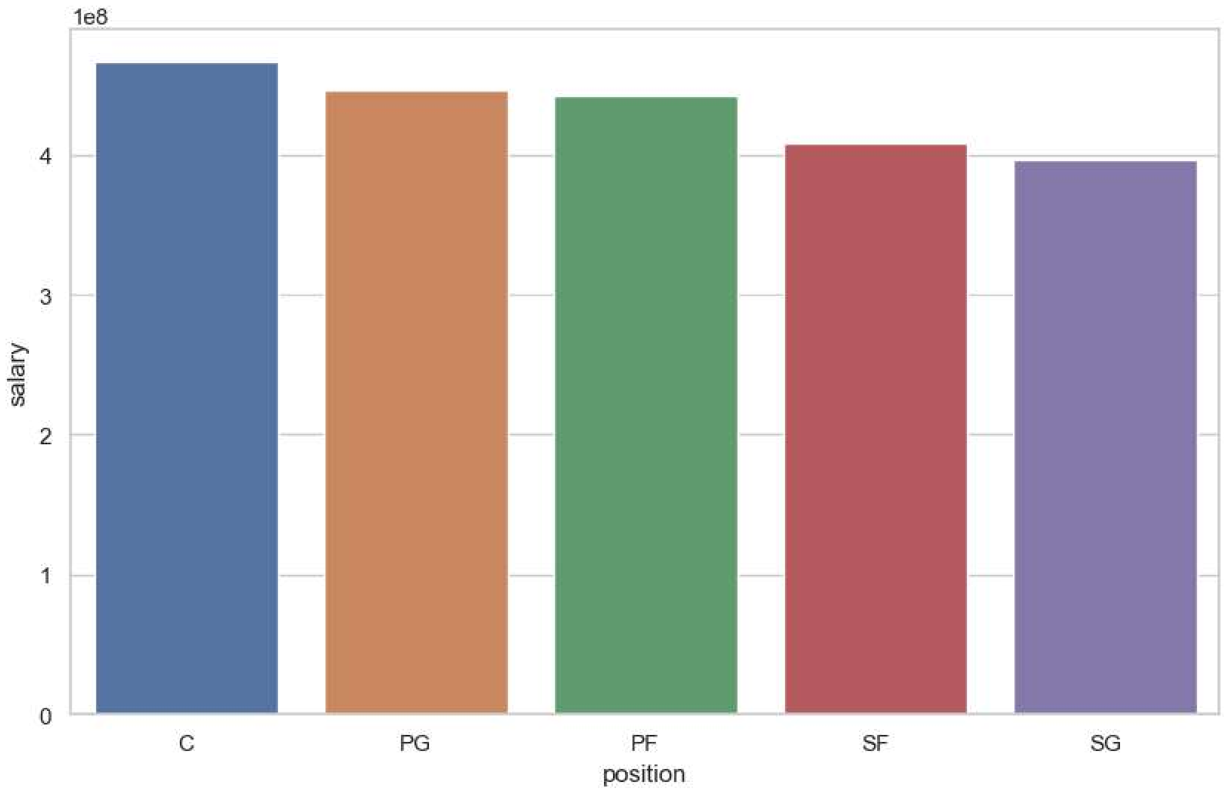
```
team = dataset.groupby('Team')['Salary'].sum().sort_values(ascending= False)
position = dataset.groupby('Position')['Salary'].sum().sort_values(ascending=False)

plt.figure(figsize=(10,6))
sns.barplot(x = team.index , y = team.values)
plt.xlabel('team name')
plt.ylabel('salary')
plt.xticks(rotation = 90)
plt.show()

plt.figure(figsize=(10,6))
```

```
sns.barplot(x =position.index , y = position.values)
plt.xlabel('position')
plt.ylabel('salary')
plt.show()
```



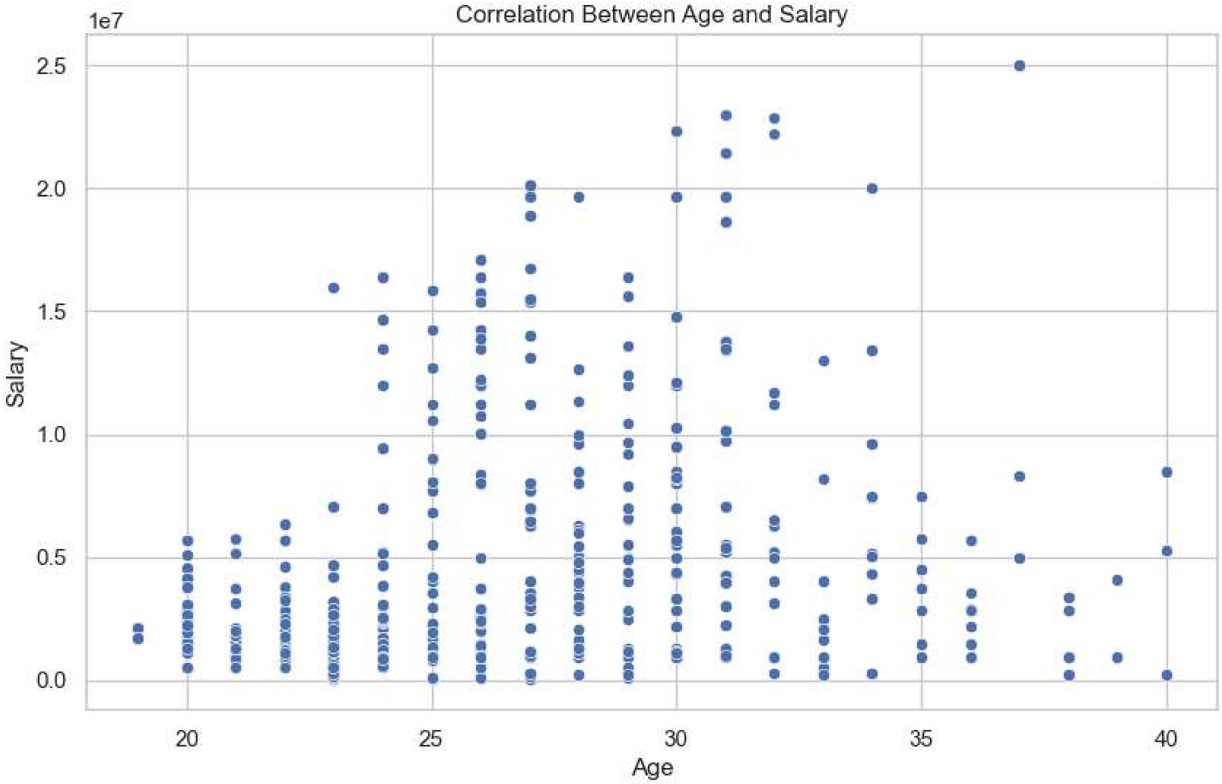


Teams and positions with the highest salary expenditures highlight areas where the company is making significant financial investments.These could be critical roles or teams driving key projects.

In [ ]:

```
correlation = dataset[['Age', 'Salary']].corr()
print(correlation)
plt.figure(figsize=(10, 6))
sns.scatterplot(x='Age', y='Salary',data=dataset)
plt.title('Correlation Between Age and Salary')
plt.xlabel('Age')
plt.ylabel('Salary')
plt.show()
```

```
             Age    Salary
Age     1.000000  0.214009
Salary  0.214009  1.000000
```



The correlation (or lack thereof) between age and salary can indicate how compensation is structured within the company. If there's a strong postive correlation, older employees might generally be earning more, which could reflect seniority-based pay scales. But here it has a weak correlation between age and salary which indicates that salary cant be salary can predicted by age

```
In [ ]:
```