

Name :-Yash Patil A176 Mba Tech

Task1:

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

```
df = pd.read_csv('/content/company_employee_details.csv')
```

```
print(df)
```

```
➡ Unnamed: 0  company  department  employee_id  age  age_when_joined  \
0            0  Glasses      BigData         16   41                38
1            1  Glasses        AI          56   44                42
2            2  Cheerper    Support        59   41                40
3            3  Glasses      Design        40   39                32
4            4  Glasses  Search Engine        47   30                22
...         ...      ...      ...      ...   ...                ...
4995        4995  Cheerper  Search Engine        37   42                35
4996        4996  Glasses        AI           1   36                25
4997        4997  Glasses  Search Engine        20   35                30
4998        4998    Pear      Sales           9   49                48
4999        4999  Glasses        AI          36   43                38
```

```
      years_in_the_company      salary  annual_bonus  \
0                3  68074.971354  17664.104103
1                2  49092.147458  21551.141079
2                1  40000.000000  23698.878851
3                7  108192.465506  21570.929657
4                9   71291.191132  21565.505715
...         ...      ...      ...
4995            7   76541.841502  17366.896122
4996            9  113651.485089  20888.144794
4997            5   77560.047560  24792.910000
4998            1  153000.000000  11700.850325
4999            5   91443.504951  13529.040663
```

```
      prior_years_experience  full_time  part_time  contractor
0                3      0.000000      0.000000      1.000000
```

| | | | | |
|------|-----|----------|----------|----------|
| 1 | 2 | 0.000000 | 0.184991 | 1.000000 |
| 2 | 2 | 0.000000 | 0.919887 | 0.662706 |
| 3 | 1 | 0.930396 | 0.546206 | 0.000000 |
| 4 | 2 | 0.525432 | 0.938412 | 0.000000 |
| ... | ... | ... | ... | ... |
| 4995 | 2 | 0.269842 | 0.691977 | 0.038156 |
| 4996 | 1 | 0.356024 | 0.098686 | 0.545462 |
| 4997 | 2 | 0.179350 | 0.361727 | 0.459085 |
| 4998 | 4 | 1.000000 | 0.000000 | 0.585987 |
| 4999 | 4 | 0.755094 | 0.000000 | 0.462368 |

[5000 rows x 13 columns]

```
df.isnull().sum()
```



| | |
|------------------------|---|
| | 0 |
| Unnamed: 0 | 0 |
| company | 0 |
| department | 0 |
| employee_id | 0 |
| age | 0 |
| age_when_joined | 0 |
| years_in_the_company | 0 |
| salary | 0 |
| annual_bonus | 0 |
| prior_years_experience | 0 |
| full_time | 0 |
| part_time | 0 |
| contractor | 0 |

```
df = df.where(df!='-',0)
```

```
print(df.size,df.shape,df.dtypes)
```

```
↔ 65000 (5000, 13) Unnamed: 0          int64  
   company                          object  
   department                       object  
   employee_id                     int64  
   age                             int64  
   age_when_joined                 int64  
   years_in_the_company            int64  
   salary                         float64  
   annual_bonus                   float64  
   prior_years_experience          int64  
   full_time                      float64  
   part_time                      float64  
   contractor                     float64  
   dtype: object
```

```
for col in df.columns:  
    print(df[col].unique)
```

```
↔
```

```

4997 2
4998 4
4999 4
Name: prior_years_experience, Length: 5000, dtype: int64>
<bound method Series.unique of 0      0.000000
1      0.000000
2      0.000000
3      0.930396
4      0.525432
...
4995 0.269842
4996 0.356024
4997 0.179350
4998 1.000000
4999 0.755094
Name: full_time, Length: 5000, dtype: float64>
<bound method Series.unique of 0      0.000000
1      0.184991
2      0.919887
3      0.546206
4      0.938412
...
4995 0.691977
4996 0.098686
4997 0.361727
4998 0.000000
4999 0.000000
Name: part_time, Length: 5000, dtype: float64>
<bound method Series.unique of 0      1.000000
1      1.000000
2      0.662706
3      0.000000
4      0.000000
...
4995 0.038156
4996 0.545462
4997 0.459085
4998 0.585987
4999 0.462368
Name: contractor, Length: 5000, dtype: float64>

```

```
print(df.nlargest(1,'salary'))
```

```

➡ Unnamed: 0  company department employee_id age age_when_joined \
21          21  Cheerper      AI           3  44           40

    years_in_the_company  salary  annual_bonus  prior_years_experience \
21                    4  153000.0    24792.91             1

    full_time  part_time  contractor
21  0.767667  0.727031      0.0

```

```
print(df.nsmallest(1,'salary'))
```

```

➡ Unnamed: 0  company department employee_id age age_when_joined \
2          2  Cheerper      Support      59  41           40

    years_in_the_company  salary  annual_bonus  prior_years_experience \
2                    1  40000.0  23698.878851             2

    full_time  part_time  contractor
2          0.0  0.919887  0.662706

```

```
print(len(df))
```

```
➡ 5000
```

```

med = (len(df)+1)/2
roundmed=round(med)
newrow=df.loc[roundmed]
df.loc[len(df)]=newrow
print(df)

```

```

➡ Unnamed: 0  company department employee_id age age_when_joined \
0          0  Glasses      BigData      16  41           38
1          1  Glasses      AI          56  44           42
2          2  Cheerper      Support      59  41           40
3          3  Glasses      Design      40  39           32
4          4  Glasses  Search Engine      47  30           22
...      ...      ...      ...      ...      ...
4996     4996  Glasses      AI           1  36           25
4997     4997  Glasses  Search Engine      20  35           30
4998     4998   Pear      Sales         9  49           48

```

| | | | | | | |
|------|------|----------|---------|----|----|----|
| 4999 | 4999 | Glasses | AI | 36 | 43 | 38 |
| 5000 | 2500 | Cheerper | BigData | 39 | 36 | 33 |

| | years_in_the_company | salary | annual_bonus | \ |
|------|----------------------|---------------|--------------|---|
| 0 | 3 | 68074.971354 | 17664.104103 | |
| 1 | 2 | 49092.147458 | 21551.141079 | |
| 2 | 1 | 40000.000000 | 23698.878851 | |
| 3 | 7 | 108192.465506 | 21570.929657 | |
| 4 | 9 | 71291.191132 | 21565.505715 | |
| ... | ... | ... | ... | |
| 4996 | 9 | 113651.485089 | 20888.144794 | |
| 4997 | 5 | 77560.047560 | 24792.910000 | |
| 4998 | 1 | 153000.000000 | 11700.850325 | |
| 4999 | 5 | 91443.504951 | 13529.040663 | |
| 5000 | 2 | 94371.736697 | 20042.303911 | |

| | prior_years_experience | full_time | part_time | contractor | Bonus | \ |
|------|------------------------|-----------|-----------|------------|-------|---|
| 0 | 3 | 0.000000 | 0.000000 | 1.000000 | 5000 | |
| 1 | 2 | 0.000000 | 0.184991 | 1.000000 | 7000 | |
| 2 | 2 | 0.000000 | 0.919887 | 0.662706 | 7000 | |
| 3 | 1 | 0.930396 | 0.546206 | 0.000000 | 5000 | |
| 4 | 2 | 0.525432 | 0.938412 | 0.000000 | 5000 | |
| ... | ... | ... | ... | ... | ... | |
| 4996 | 1 | 0.356024 | 0.098686 | 0.545462 | 5000 | |
| 4997 | 2 | 0.179350 | 0.361727 | 0.459085 | 5000 | |
| 4998 | 4 | 1.000000 | 0.000000 | 0.585987 | 5000 | |
| 4999 | 4 | 0.755094 | 0.000000 | 0.462368 | 5000 | |
| 5000 | 3 | 0.365225 | 0.907716 | 0.000000 | 5000 | |

| | Salary_with_Bonus |
|------|-------------------|
| 0 | 73074.971354 |
| 1 | 56092.147458 |
| 2 | 47000.000000 |
| 3 | 113192.465506 |
| 4 | 76291.191132 |
| ... | ... |
| 4996 | 118651.485089 |
| 4997 | 82560.047560 |
| 4998 | 158000.000000 |
| 4999 | 96443.504951 |
| 5000 | 99371.736697 |

[5001 rows x 15 columns]

```
print(df.groupby('department')['salary'].sum())
```

```
↔ department
AI          1.034900e+08
BigData     5.978510e+07
Design      7.153055e+07
Sales       7.347354e+07
Search Engine 8.822031e+07
Support     3.944371e+07
Name: salary, dtype: float64
```

```
print(df.groupby('gender')['salary'].sum())
```

```
def find_salaries(df):
    department_groups = df.groupby('department')['salary']
    results = {}
    for dept, salaries in department_groups:
        sorted_salaries = salaries.sort_values(ascending=False)
        results[dept] = {
            '2nd_Largest': sorted_salaries.iloc[1] if len(sorted_salaries) > 1 else None,
            '3rd_Largest': sorted_salaries.iloc[2] if len(sorted_salaries) > 2 else None,
            'Smallest': sorted_salaries.min()
        }
    return pd.DataFrame(results).T

salaries_summary = find_salaries(df)

print(salaries_summary)
```

```
↔
```

| | 2nd_Largest | 3rd_Largest | Smallest |
|---------------|---------------|---------------|----------|
| AI | 153000.000000 | 153000.000000 | 40000.0 |
| BigData | 153000.000000 | 146971.942754 | 40000.0 |
| Design | 153000.000000 | 153000.000000 | 40000.0 |
| Sales | 153000.000000 | 153000.000000 | 40000.0 |
| Search Engine | 152381.693866 | 150847.229633 | 40000.0 |
| Support | 132665.632536 | 131381.236592 | 40000.0 |

```
df['Bonus'] = np.where(df['salary'] > 50000, 5000, 7000)
df['Salary_with_Bonus'] = df['salary'] + df['Bonus']
print(df)
```

```

↳ Unnamed: 0  company  department  employee_id  age  age_when_joined  \
0            0  Glasses      BigData          16  41             38
1            1  Glasses          AI          56  44             42
2            2  Cheerper    Support          59  41             40
3            3  Glasses      Design          40  39             32
4            4  Glasses  Search Engine          47  30             22
...         ...      ...      ...      ...      ...      ...
4995        4995  Cheerper    Search Engine          37  42             35
4996        4996  Glasses          AI           1  36             25
4997        4997  Glasses  Search Engine          20  35             30
4998        4998    Pear      Sales           9  49             48
4999        4999  Glasses          AI          36  43             38

```

```

      years_in_the_company      salary  annual_bonus  \
0                3  68074.971354  17664.104103
1                2  49092.147458  21551.141079
2                1  40000.000000  23698.878851
3                7  108192.465506  21570.929657
4                9   71291.191132  21565.505715
...             ...      ...      ...
4995             7   76541.841502  17366.896122
4996             9  113651.485089  20888.144794
4997             5   77560.047560  24792.910000
4998             1  153000.000000  11700.850325
4999             5   91443.504951  13529.040663

```

```

      prior_years_experience  full_time  part_time  contractor  Bonus  \
0                3   0.000000   0.000000   1.000000   5000
1                2   0.000000   0.184991   1.000000   7000
2                2   0.000000   0.919887   0.662706   7000
3                1   0.930396   0.546206   0.000000   5000
4                2   0.525432   0.938412   0.000000   5000
...             ...      ...      ...      ...      ...
4995             2   0.269842   0.691977   0.038156   5000
4996             1   0.356024   0.098686   0.545462   5000
4997             2   0.179350   0.361727   0.459085   5000
4998             4   1.000000   0.000000   0.585987   5000
4999             4   0.755094   0.000000   0.462368   5000

```


| | Salary_with_Bonus |
|------|-------------------|
| 0 | 73074.971354 |
| 1 | 56092.147458 |
| 2 | 47000.000000 |
| 3 | 113192.465506 |
| 4 | 76291.191132 |
| ... | ... |
| 4995 | 81541.841502 |
| 4996 | 118651.485089 |
| 4997 | 82560.047560 |
| 4998 | 158000.000000 |
| 4999 | 96443.504951 |

[5000 rows x 15 columns]

```
plt.figure(figsize=(12, 6))
plt.subplot(1, 2, 1)
sns.boxplot(x='department', y='salary', data=df, palette='viridis')
plt.title('Salary Distribution by Department')
plt.xlabel('Department')
plt.ylabel('Salary')
plt.subplot(1, 2, 2)
sns.barplot(x='employee_id', y='Salary_with_Bonus', data=df, palette='magma')
plt.title('Salary with Bonus per Employee')
plt.xlabel('Employee Name')
plt.ylabel('Salary with Bonus')
plt.tight_layout()
plt.show()
```

Insights:

- # 1. The boxplot shows the salary distribution in each department.
- # Insights can include identifying departments with wide variations or median salaries.
- # 2. The bar plot shows how the bonus impacts the salary per employee. You can highlight
- # that higher salaries (>50k) receive a smaller bonus (5k), while lower salaries (<50k) receive a larger bonus (7k).

<ipython-input-20-af712306f200>:3: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v0.14.0. Assign the `x` variable to `hue` and set `legend`

```
sns.boxplot(x='department', y='salary', data=df, palette='viridis')
```

<ipython-input-20-af712306f200>:8: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v0.14.0. Assign the `x` variable to `hue` and set `legend`

```
sns.barplot(x='employee_id', y='Salary_with_Bonus', data=df, palette='magma')
```



