Human Resources Analysis

Human Resources Analysis

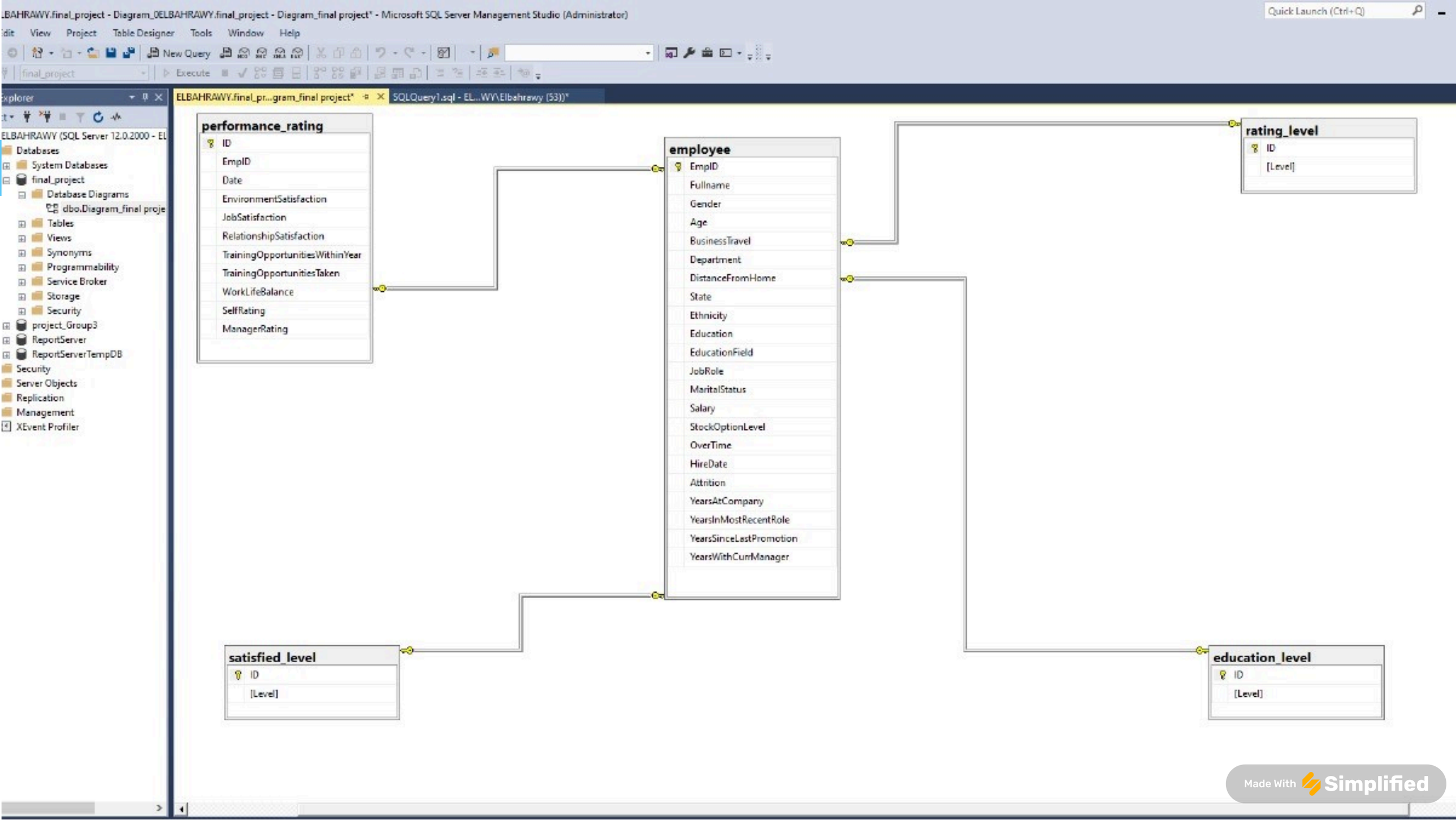
Team
Mahmoud Ramadan
Marwan Adnan Hamza Abd El Wahab
Mohamed Medhat Elbahrawy
Sara Ahmed Mohamed Taha Samaha

Team actions

Mohamed/ Build Data Model, Data Cleaning, and Preprocessing.
Marawan/Analysis Questions Phase.
Mahmoud/Forecasting Questions Phase.
Sara/Visualization Dashboard
and Final Presentation.

1-SQL

Build Data Model

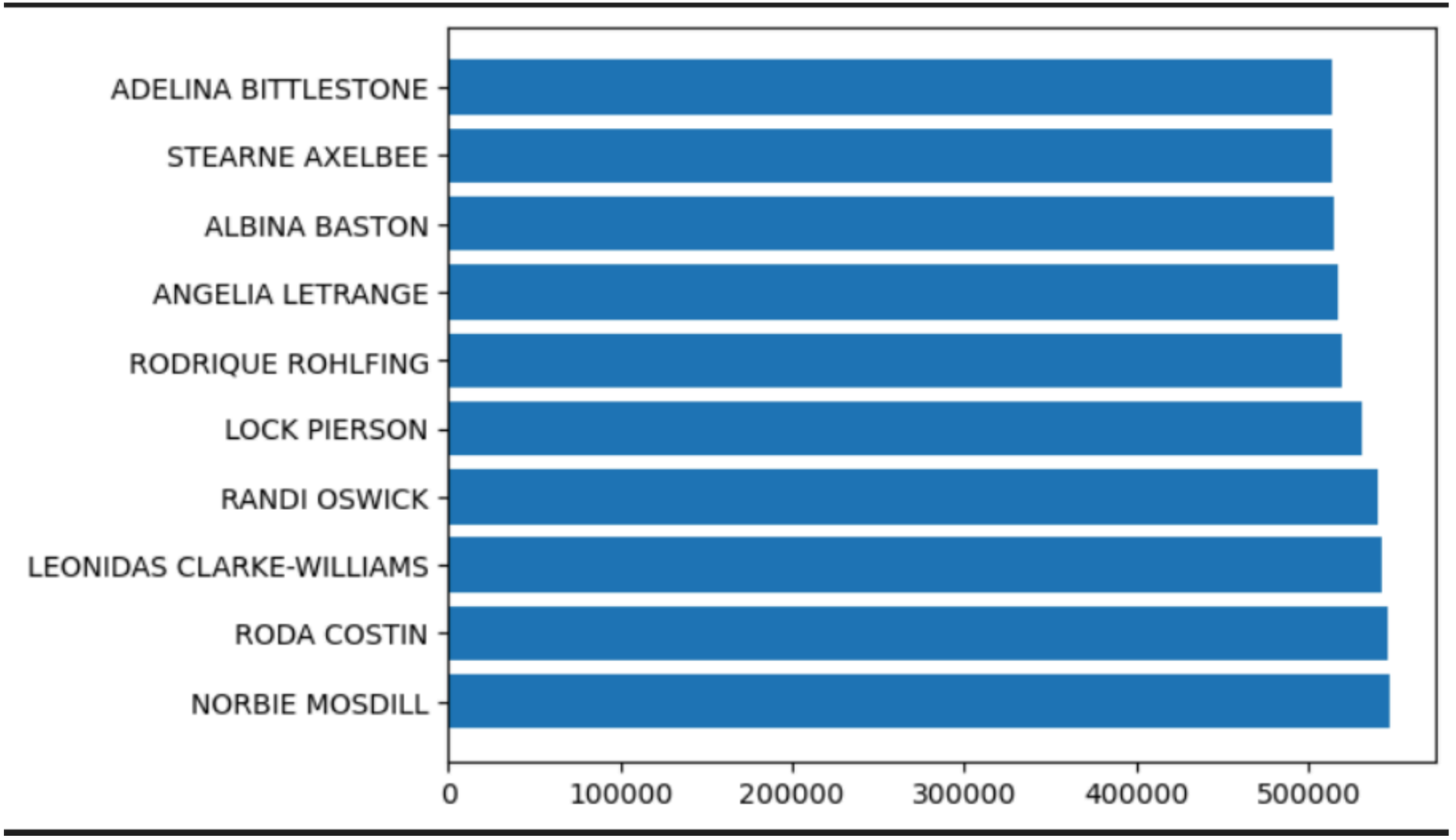

performance_rating
ID
EmpID
Date
EnvironmentSatisfaction
JobSatisfaction
RelationshipSatisfaction
TrainingOpportunitiesWithinYear
TrainingOpportunitiesTaken
WorkLifeBalance
SelfRating
ManagerRating
employee
EmpID
Fullname
Gender
Age
BusinessTravel
Department
DistanceFromHome
State
Ethnicity
Education
EducationField
JobRole
MaritalStatus
Salary
StockOptionLevel
OverTime
HireDate
Attrition
YearsAtCompany
YearsInMostRecentRole
YearsSinceLastPromotion
YearsWithCurrManager
rating_level
ID
[Level]
satisfied_level
ID
[Level]
education_level
ID
[Level]

2-Data Cleaning, and Preprocessing

3- Analysis Questions Phase By Python

Who Are The Top 10 Earning Employees?

	Fullname	Gender	Department	Salary
385	NORBIE MOSDILL	Male	Technology	547204
397	RODA COSTIN	Female	Technology	546549
65	LEONIDAS CLARKE-WILLIAMS	Non-Binary	Technology	542695
1167	RANDI OSWICK	Female	Technology	539998
1054	LOCK PIERSON	Male	Technology	531629
961	RODRIQUE ROHLFING	Male	Technology	519361
1031	ANGELIA LETRANGE	Female	Technology	517695
159	ALBINA BASTON	Female	Technology	514945
239	STEARNE AXELBEE	Male	Technology	513608
934	ADELINA BITTLESTONE	Female	Technology	513325



How Many Employees Are There In Each Department?

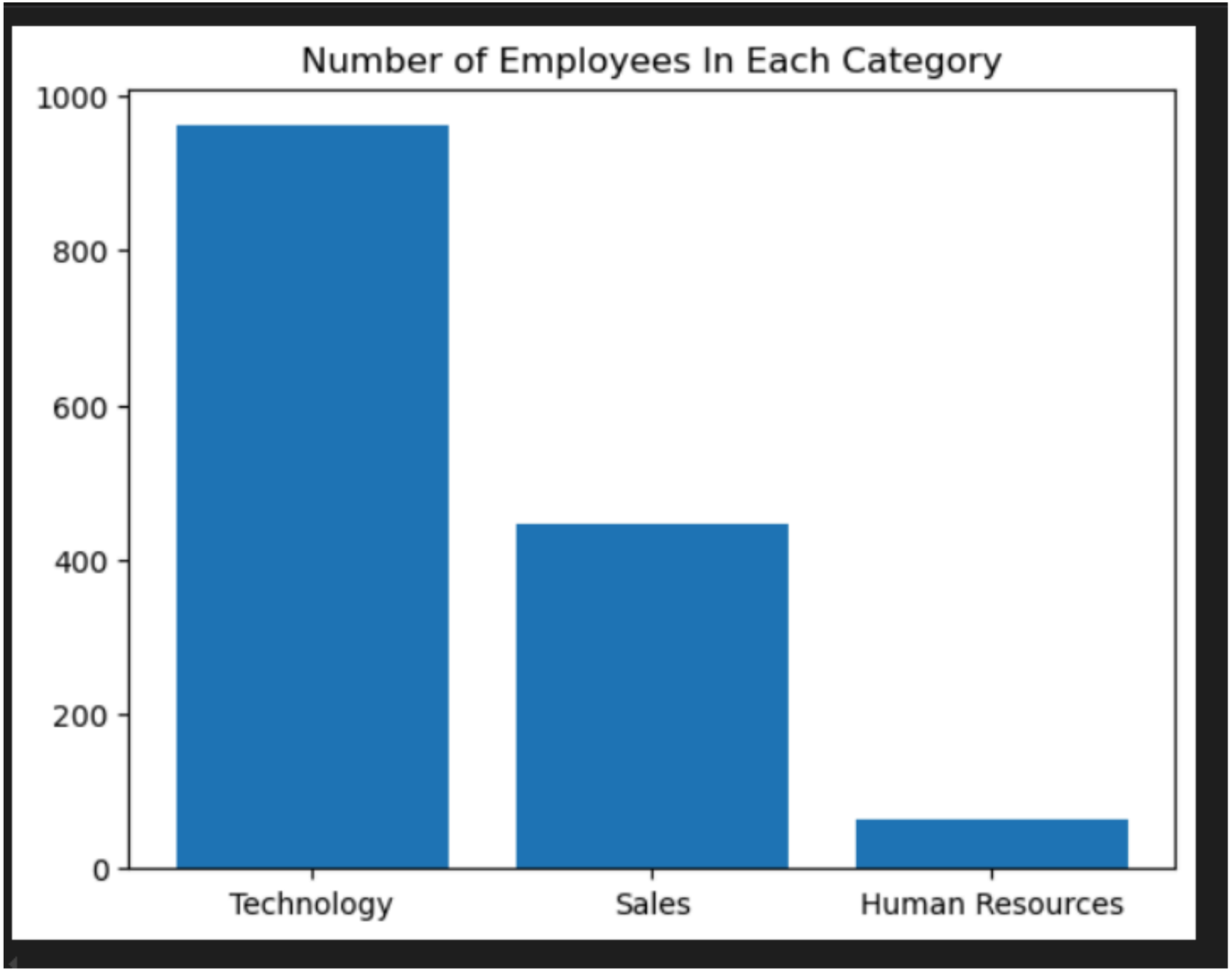
How Many Employees Are There In Each Department?

[151]
```
No_of_employees = employee['Department'].value_counts()
No_of_employees
```

```
Department
Technology         961
Sales              446
Human Resources     63
Name: count, dtype: int64
```

[153]
```
department = No_of_employees.index
No_of_employees = No_of_employees.values
```

[155]
```
# Creating the bar plot
plt.bar(department, No_of_employees)
plt.title("Number of Employees In Each Category")
plt.show()
```



Is There Any Gender Discrimination Regarding The Hiring Process?

Is There Any Gender Discrimination Regarding The Hiring Process?

[213]
```
gender_counts = employee['Gender'].value_counts()
gender_counts
```

```
Gender
Female                675
Male                  651
Non-Binary            124
Prefer Not To Say      20
Name: count, dtype: int64
```

[209]
```
gender_counts.plot.pie(autopct='%1.1f%%', figsize=(6, 6), startangle=90, colors=['#ff9999','#66b3ff','#99ff99',"#FFFF00"])

plt.ylabel('')
plt.title('Gender Distribution of Employees')
plt.show()
```

What Are The Satisfaction Trends?

What Are The Satisfaction Trends?

+ Code + Markdown

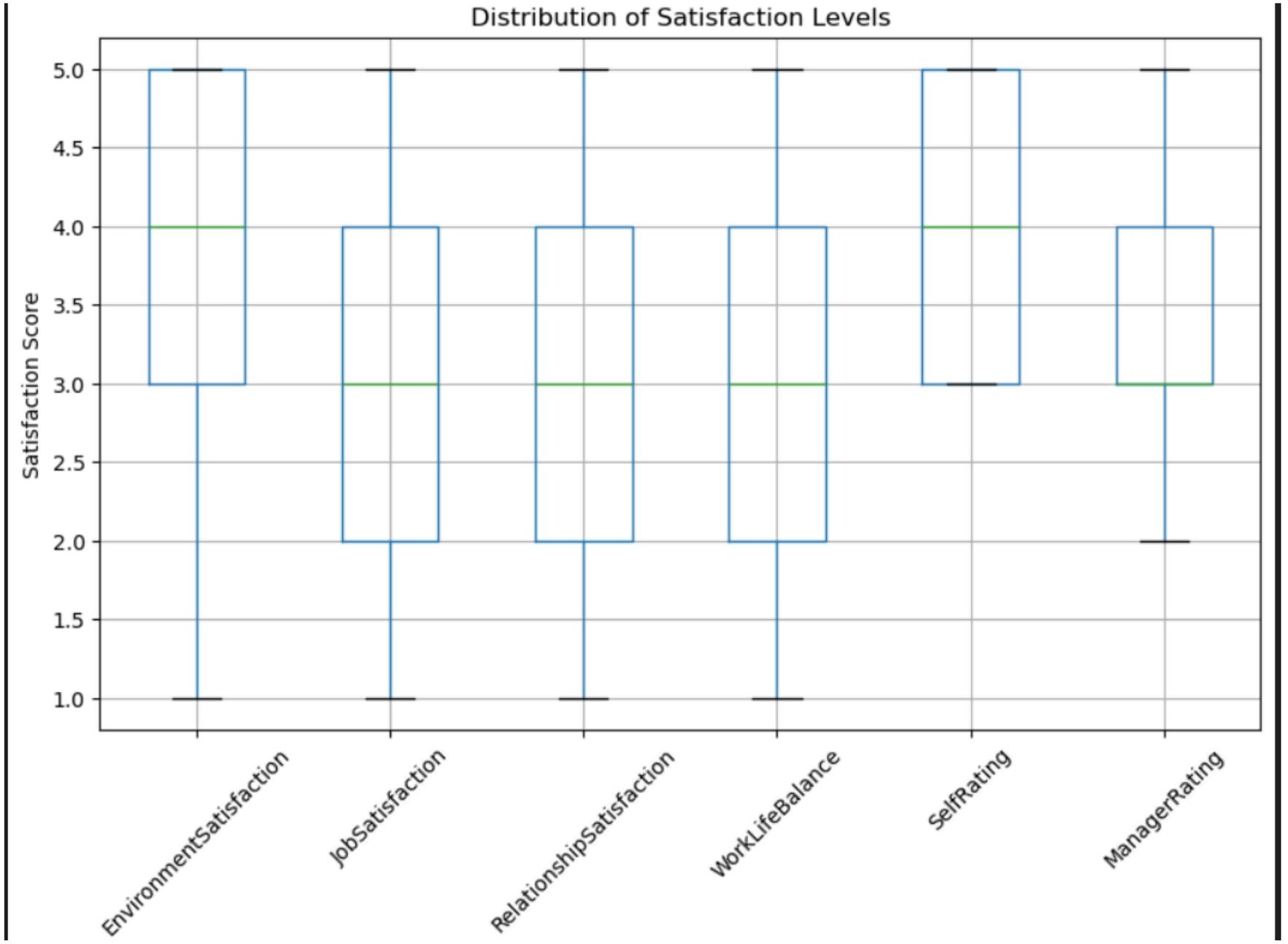
```
avg_satisfaction = performance_rating[['EnvironmentSatisfaction', 'JobSatisfaction',
                                       'RelationshipSatisfaction', 'WorkLifeBalance',
                                       'SelfRating', 'ManagerRating']].mean()
avg_satisfaction
```

[274]

```
EnvironmentSatisfaction     3.872559
JobSatisfaction             3.430616
RelationshipSatisfaction    3.427336
WorkLifeBalance             3.414667
SelfRating                  3.984051
ManagerRating               3.473394
dtype: float64
```

```
performance_rating[['EnvironmentSatisfaction', 'JobSatisfaction',
                    'RelationshipSatisfaction', 'WorkLifeBalance',
                    'SelfRating', 'ManagerRating']].boxplot(figsize=(10, 6))
plt.title('Distribution of Satisfaction Levels')
plt.ylabel('Satisfaction Score')
plt.xticks(rotation=45)
plt.show()
```

[197]



What Is The Correlation Between Satisfaction and Ratings?

What Is The Correlation Between Satisfaction and Ratings?

```
satisfaction_rating_corr = performance_rating[['EnvironmentSatisfaction', 'JobSatisfaction', 'RelationshipSatisfaction', 'SelfRating', 'ManagerRating']].corr()
satisfaction_rating_corr
```

[283] Python

	EnvironmentSatisfaction	JobSatisfaction	RelationshipSatisfaction	SelfRating	ManagerRating
EnvironmentSatisfaction	1.000000	0.102046	0.095539	0.000459	-0.005643
JobSatisfaction	0.102046	1.000000	0.048128	-0.008701	-0.015821
RelationshipSatisfaction	0.095539	0.048128	1.000000	0.018425	0.018999
SelfRating	0.000459	-0.008701	0.018425	1.000000	0.854107
ManagerRating	-0.005643	-0.015821	0.018999	0.854107	1.000000

Group by Department and calculate average satisfaction levels

Group by Department and calculate average satisfaction levels

```
satisfaction_by_department = performance_rating.merge(employee, left_on='EmployeeID', right_on='EmpID').groupby('Department')[['EnvironmentSatisfaction', 'JobS
satisfaction_by_department
```

[285] Python

Department	EnvironmentSatisfaction	JobSatisfaction	RelationshipSatisfaction
Human Resources	3.861386	3.435644	3.353135
Sales	3.892508	3.422057	3.451838
Technology	3.863284	3.434578	3.420249

#Merge performance data with employee data and group by Attrition

Merge performance data with employee data and group by Attrition

```
[223]
satisfaction_vs_attrition = performance_rating.merge(employee, left_on='EmployeeID', right_on='EmpID').groupby('Attrition')[['JobSatisfaction']].mean()
satisfaction_vs_attrition
```

Pytho

```
            JobSatisfaction
Attrition
No                 3.419739
Yes                3.452012
```

Correlation between work-life balance and satisfaction levels

Correlation between work-life balance and satisfaction levels

```
work_life_balance_corr = performance_rating[['WorkLifeBalance', 'EnvironmentSatisfaction', 'JobSatisfaction', 'RelationshipSatisfaction']].corr()
work_life_balance_corr
```

[225] Python

```
                          WorkLifeBalance  EnvironmentSatisfaction  \
WorkLifeBalance                  1.000000                 0.098293
EnvironmentSatisfaction          0.098293                 1.000000
JobSatisfaction                  0.041724                 0.102046
RelationshipSatisfaction         0.055782                 0.095539

                          JobSatisfaction  RelationshipSatisfaction
WorkLifeBalance                  0.041724                 0.055782
EnvironmentSatisfaction          0.102046                 0.095539
JobSatisfaction                  1.000000                 0.048128
RelationshipSatisfaction         0.048128                 1.000000
```

Group by TrainingOpportunitiesTaken and calculate average satisfaction

Group by TrainingOpportunitiesTaken and calculate average satisfaction

```
[227]
satisfaction_by_training = performance_rating.groupby('TrainingOpportunitiesTaken')[['JobSatisfaction', 'RelationshipSatisfaction']].mean()
satisfaction_by_training
```

Python

```
                              JobSatisfaction  RelationshipSatisfaction
TrainingOpportunitiesTaken
0                                    3.440135                  3.439713
1                                    3.415680                  3.436913
2                                    3.419255                  3.399068
3                                    3.478333                  3.400000
```

Group by Department and calculate average ratings

Group by Department and calculate average ratings

```
rating_by_department = performance_rating.merge(employee, left_on='EmployeeID', right_on='EmpID').groupby('Department')[['SelfRating', 'ManagerRating']].mean(
rating_by_department
```

```
                SelfRating  ManagerRating
Department
Human Resources    3.990099       3.442244
Sales              3.973476       3.449977
Technology         3.988959       3.487432
```

Who Are The Most Performing Employees?

Who Are The Most Performing Employees?

```
merged_data = performance_rating.merge(employee, left_on='EmployeeID', right_on='EmpID')
top_performing_employees = merged_data.sort_values(by=['ManagerRating', 'SelfRating', 'Salary'], ascending=[False, False, False])
top_performing_employees[['Fullname', 'Salary']].head()
```

[261] Python

	Fullname	Salary
1699	NORBIE MOSDILL	547204
3608	NORBIE MOSDILL	547204
2042	RODA COSTIN	546549
1588	LEONIDAS CLARKE-WILLIAMS	542695
2217	RANDI OSWICK	539998

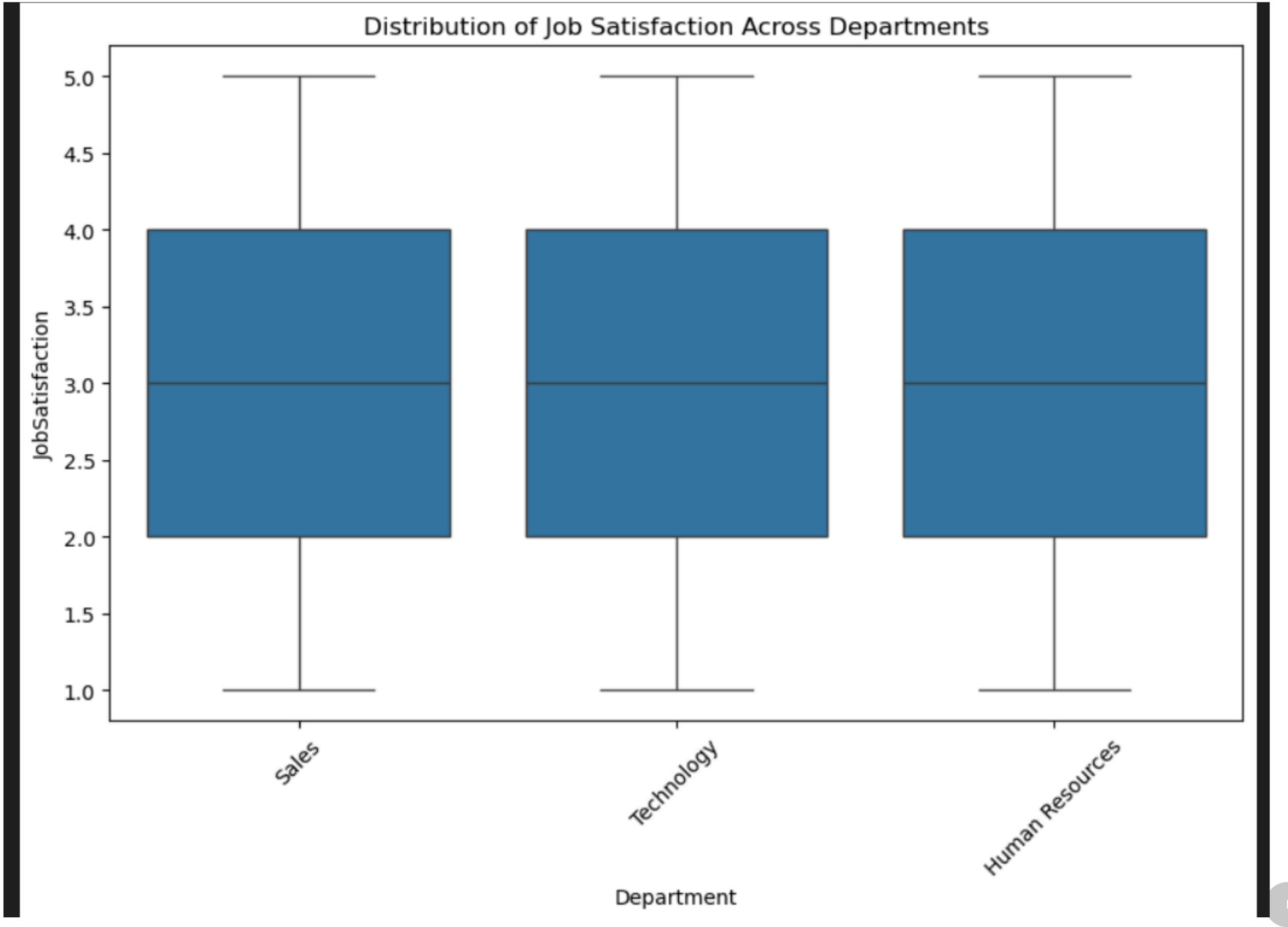
Distribution of Job Satisfaction Across Departments

Distribution of Job Satisfaction Across Departments

+ Code + Markdown

```
import seaborn as sns
import matplotlib.pyplot as plt

# Merge performance_rating with employee data on EmployeeID and EmpID
performance_rating_with_employee = performance_rating.merge(employee, left_on='EmployeeID', right_on='EmpID')
# Plotting Job Satisfaction by Department
plt.figure(figsize=(10, 6))
sns.boxplot(x='Department', y='JobSatisfaction', data=performance_rating_with_employee)
plt.title('Distribution of Job Satisfaction Across Departments')
plt.xticks(rotation=45)
plt.show()
```



Distribution of Job Satisfaction Across Departments

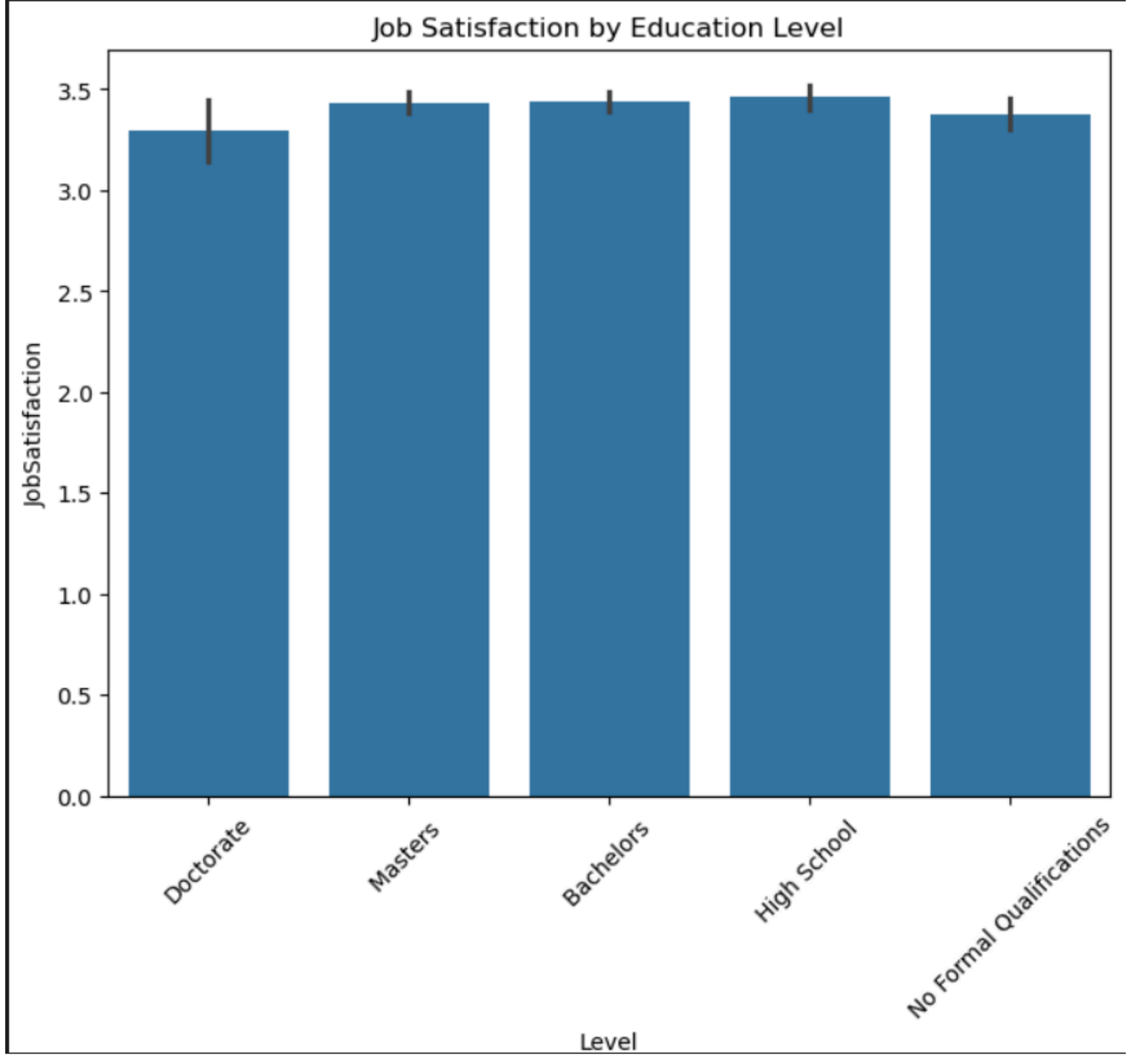
Distribution of Job Satisfaction Across Departments

```
education_data = employee.merge(education_level, left_on='Education', right_on='ID'

plt.figure(figsize=(8, 6))
sns.barplot(x='Level', y='JobSatisfaction', data=education_data.merge(performance_rating, left_on='EmpID', right_on='EmployeeID'))
plt.title('Job Satisfaction by Education Level')
plt.xticks(rotation=45)
plt.show()
```

[243]

Pytho

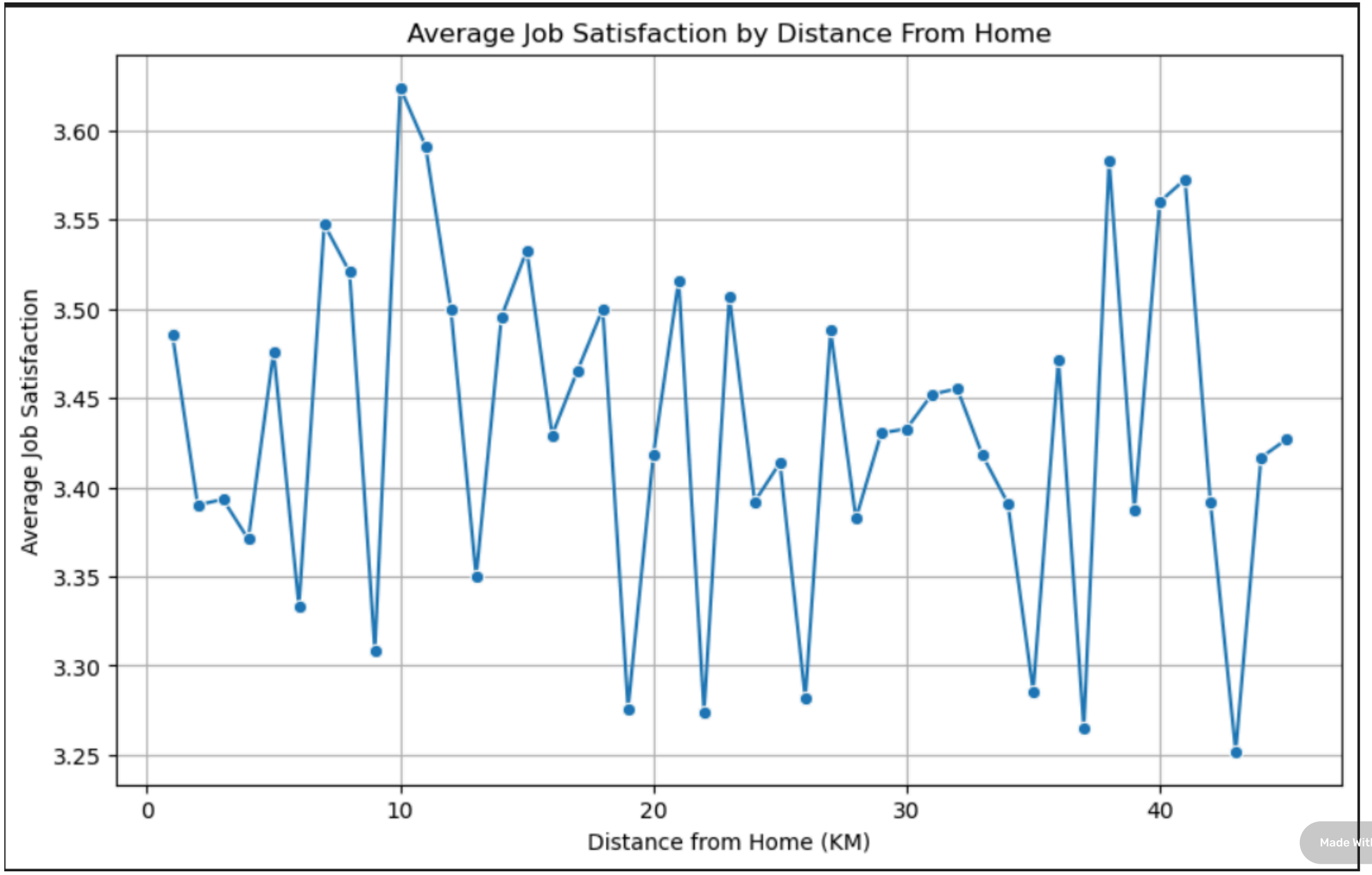


Grouping the data by DistanceFromHome and calculating the average Job Satisfaction

```
# Grouping the data by DistanceFromHome and calculating the average Job Satisfaction
average_satisfaction = merged_data.groupby('DistanceFromHome (KM)')['JobSatisfaction'].mean().reset_index()

# Line plot to show the trend of Job Satisfaction over DistanceFromHome
plt.figure(figsize=(10, 6))
sns.lineplot(x='DistanceFromHome (KM)', y='JobSatisfaction', data=average_satisfaction, marker='o')
plt.title('Average Job Satisfaction by Distance From Home')
plt.xlabel('Distance from Home (KM)')
plt.ylabel('Average Job Satisfaction')
plt.grid(True)
plt.show()
```

257]



Forecasting Questions Phase.