HR Analytics Project

Project Description

This Jupyter Notebook contains the analysis and findings of the HR Analytics project. The project's main objective is to analyze employee attrition within the organization and identify factors that contribute to attrition. We explore various aspects such as employee satisfaction, career progression, work-life balance, and more to gain insights into attrition patterns.

The analysis includes data cleaning, data visualization, and recommendations for HR strategies based on the findings.

LinkedIn: Asad Ali Mulla (https://www.linkedin.com/in/asad-ali-mulla-044240262/)

GitHub: Asadxio (https://github.com/Asadxio)

Import Libraries

In [1]:

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

Load the Data

In [2]:

```
df = pd.read_csv("C:/Users/Asad Xio/Desktop/Data Analyst Intern/Project 3 - HR Analytics/
```

Data Exploration

In [3]:

```
df.head()
```

Out[3]:

	Age	Attrition	BusinessTravel	DailyRate	Department	DistanceFromHome	Education	E
0	41	Yes	Travel_Rarely	1102	Sales	1	2	
1	49	No	Travel_Frequently	279	Research & Development	8	1	
2	37	Yes	Travel_Rarely	1373	Research & Development	2	2	
3	33	No	Travel_Frequently	1392	Research & Development	3	4	
4	27	No	Travel_Rarely	591	Research & Development	2	1	

5 rows × 35 columns

In [4]:

```
df.tail()
```

Out[4]:

	Age	Attrition	BusinessTravel	DailyRate	Department	DistanceFromHome	Education
1465	36	No	Travel_Frequently	884	Research & Development	23	2
1466	39	No	Travel_Rarely	613	Research & Development	6	1
1467	27	No	Travel_Rarely	155	Research & Development	4	3
1468	49	No	Travel_Frequently	1023	Sales	2	3
1469	34	No	Travel_Rarely	628	Research & Development	8	3

5 rows × 35 columns

In [5]:

```
df.describe()
```

Out[5]:

	Age	DailyRate	DistanceFromHome	Education	EmployeeCount	Employee
count	1470.000000	1470.000000	1470.000000	1470.000000	1470.0	147(
mean	36.923810	802.485714	9.192517	2.912925	1.0	1024
std	9.135373	403.509100	8.106864	1.024165	0.0	602
min	18.000000	102.000000	1.000000	1.000000	1.0	1
25%	30.000000	465.000000	2.000000	2.000000	1.0	491
50%	36.000000	802.000000	7.000000	3.000000	1.0	102(
75%	43.000000	1157.000000	14.000000	4.000000	1.0	155
max	60.000000	1499.000000	29.000000	5.000000	1.0	206

8 rows × 26 columns

In [6]:

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1470 entries, 0 to 1469
Data columns (total 35 columns):
 #   Column                    Non-Null Count  Dtype 
---  ------                    --------------  ----- 
 0   Age                       1470 non-null   int64 
 1   Attrition                 1470 non-null   object
 2   BusinessTravel            1470 non-null   object
 3   DailyRate                 1470 non-null   int64 
 4   Department                1470 non-null   object
 5   DistanceFromHome          1470 non-null   int64 
 6   Education                 1470 non-null   int64 
 7   EducationField            1470 non-null   object
 8   EmployeeCount             1470 non-null   int64 
 9   EmployeeNumber            1470 non-null   int64 
 10  EnvironmentSatisfaction   1470 non-null   int64 
 11  Gender                    1470 non-null   object
 12  HourlyRate                1470 non-null   int64 
 13  JobInvolvement            1470 non-null   int64 
 14  JobLevel                  1470 non-null   int64 
 15  JobRole                   1470 non-null   object
 16  JobSatisfaction           1470 non-null   int64 
 17  MaritalStatus             1470 non-null   object
 18  MonthlyIncome             1470 non-null   int64 
 19  MonthlyRate               1470 non-null   int64 
 20  NumCompaniesWorked        1470 non-null   int64 
 21  Over18                    1470 non-null   object
 22  OverTime                  1470 non-null   object
 23  PercentSalaryHike         1470 non-null   int64 
 24  PerformanceRating         1470 non-null   int64 
 25  RelationshipSatisfaction  1470 non-null   int64 
 26  StandardHours             1470 non-null   int64 
 27  StockOptionLevel          1470 non-null   int64 
 28  TotalWorkingYears         1470 non-null   int64 
 29  TrainingTimesLastYear     1470 non-null   int64 
 30  WorkLifeBalance           1470 non-null   int64 
 31  YearsAtCompany            1470 non-null   int64 
 32  YearsInCurrentRole        1470 non-null   int64 
 33  YearsSinceLastPromotion   1470 non-null   int64 
 34  YearsWithCurrManager      1470 non-null   int64 
dtypes: int64(26), object(9)
memory usage: 402.1+ KB
```

Data Cleaning and Preprocessing

In [7]:

```
df.dropna(inplace=True)
```

In [8]:

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

Out[8]:

```
Age                         0
Attrition                   0
BusinessTravel              0
DailyRate                   0
Department                  0
DistanceFromHome            0
Education                   0
EducationField              0
EmployeeCount               0
EmployeeNumber              0
EnvironmentSatisfaction     0
Gender                      0
HourlyRate                  0
JobInvolvement              0
JobLevel                    0
JobRole                     0
JobSatisfaction             0
MaritalStatus               0
MonthlyIncome               0
MonthlyRate                 0
NumCompaniesWorked          0
Over18                      0
OverTime                    0
PercentSalaryHike           0
PerformanceRating           0
RelationshipSatisfaction    0
StandardHours               0
StockOptionLevel            0
TotalWorkingYears           0
TrainingTimesLastYear       0
WorkLifeBalance             0
YearsAtCompany              0
YearsInCurrentRole          0
YearsSinceLastPromotion     0
YearsWithCurrManager        0
dtype: int64
```

In [9]:

```
df = df.drop(["EmployeeCount", "Over18", "StandardHours"], axis=1)
```

In [10]:

```
df = df.rename(columns={"Attrition": "Attrition", "JobSatisfaction": "Job_Satisfaction",}
```

In [11]:

```
df = df.dropna()
```

In [12]:

```
df.head()
```

Out[12]:

	Age	Attrition	BusinessTravel	DailyRate	Department	DistanceFromHome	Education	E
0	41	Yes	Travel_Rarely	1102	Sales	1	2	
1	49	No	Travel_Frequently	279	Research & Development	8	1	
2	37	Yes	Travel_Rarely	1373	Research & Development	2	2	
3	33	No	Travel_Frequently	1392	Research & Development	3	4	
4	27	No	Travel_Rarely	591	Research & Development	2	1	

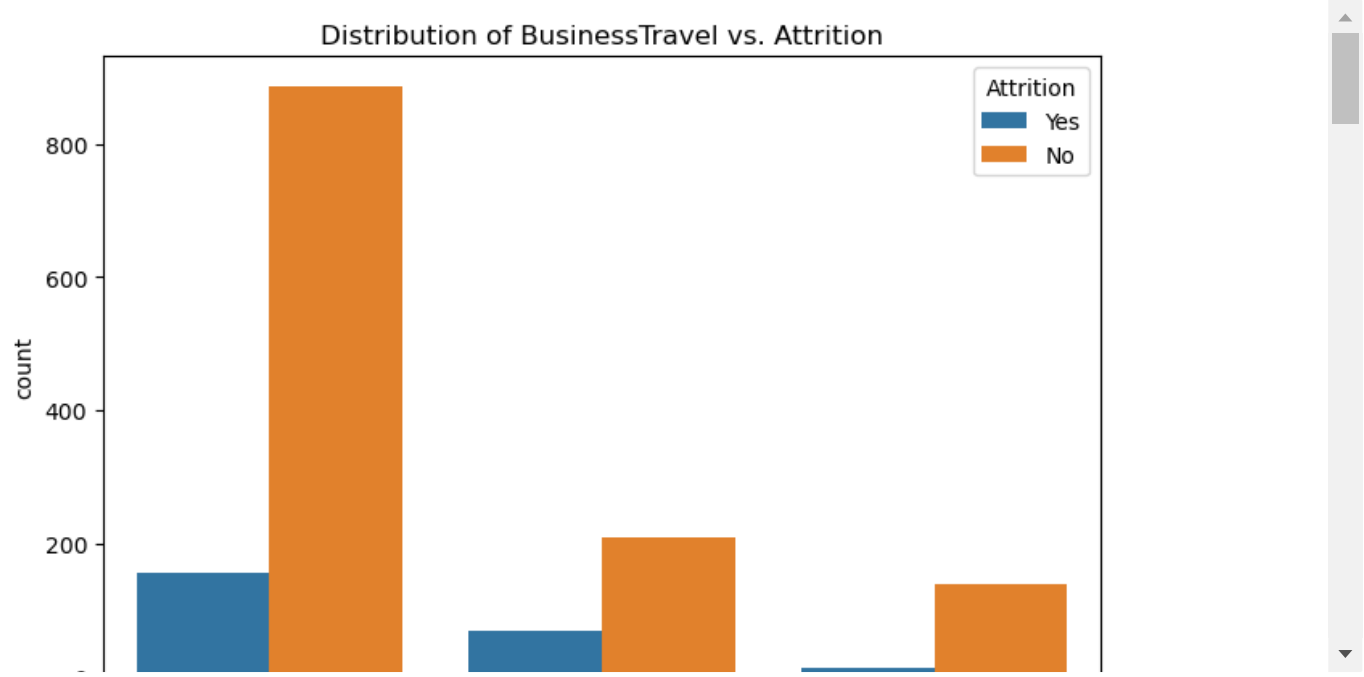
5 rows × 32 columns

In [13]:

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

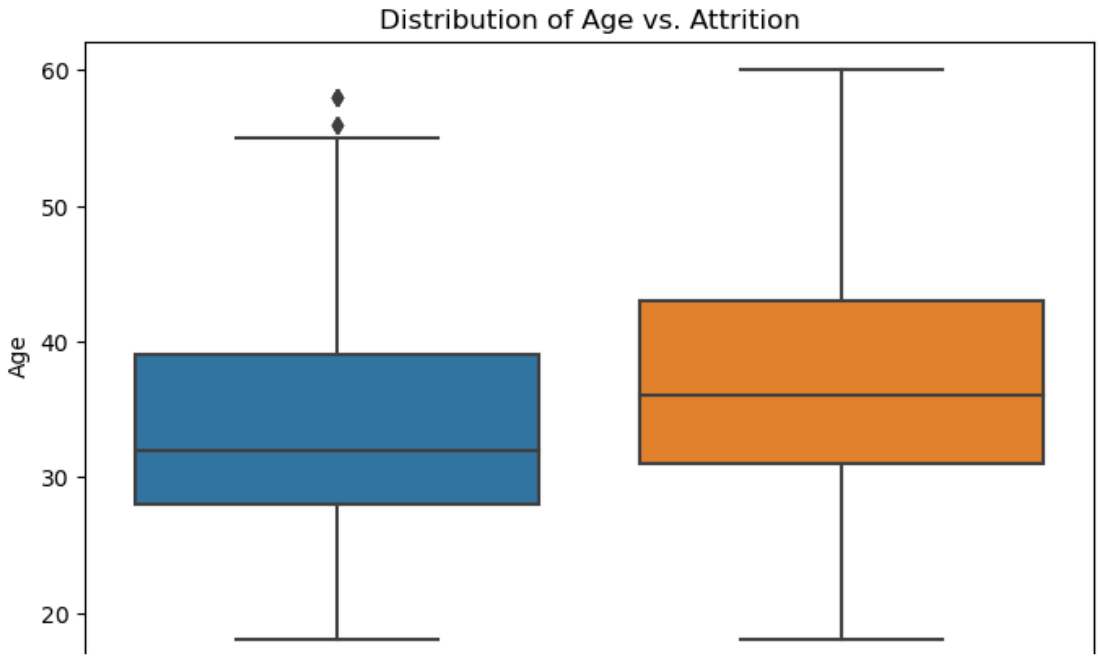
In [14]:

```
categorical_vars = ["BusinessTravel", "Department", "Gender", "JobRole", "MaritalStatus"]
for var in categorical_vars:
    plt.figure(figsize=(8, 5))
    sns.countplot(data=df, x=var, hue="Attrition")
    plt.title(f"Distribution of {var} vs. Attrition")
    plt.xticks(rotation=45)
    plt.show()
```



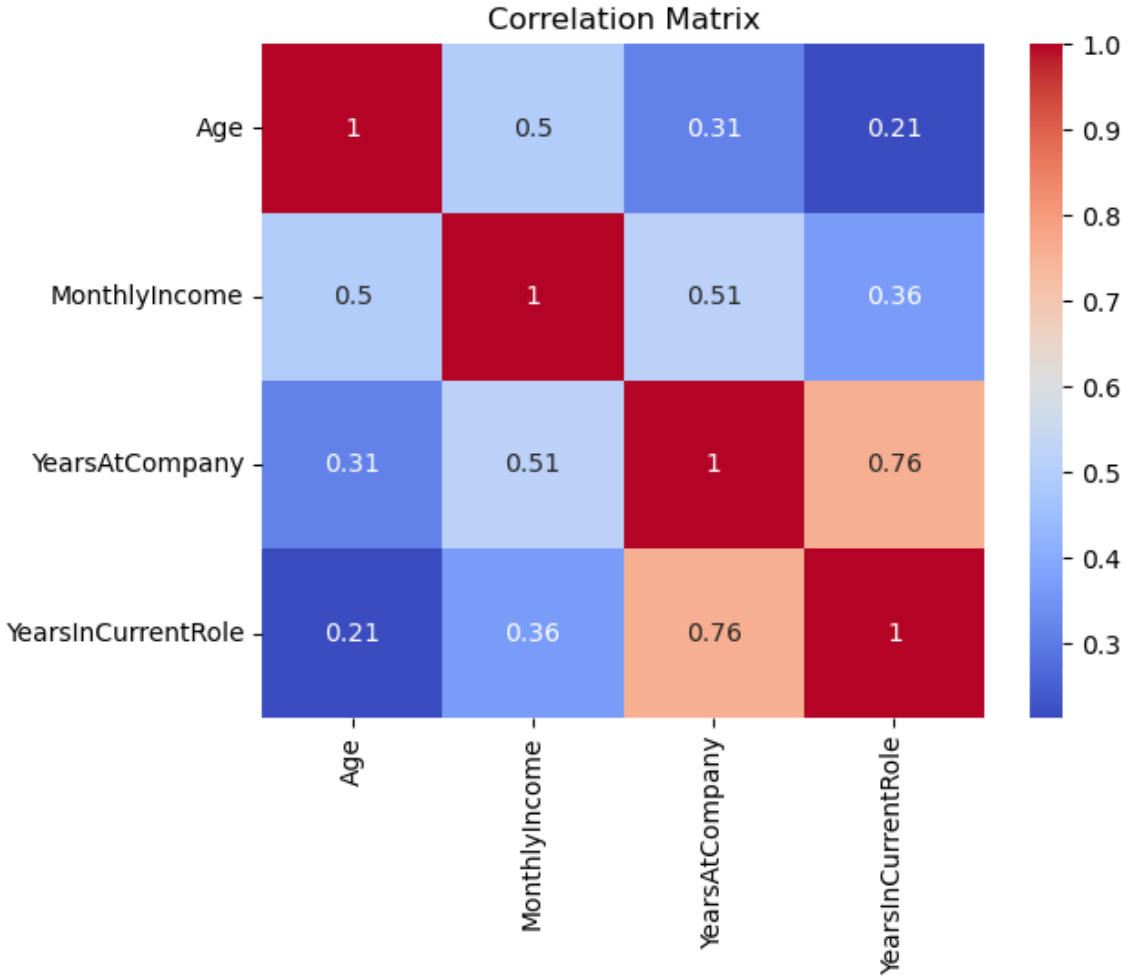
In [15]:

```
# Explore the distribution of numeric variables
numeric_vars = ["Age", "MonthlyIncome", "YearsAtCompany", "YearsInCurrentRole"]
for var in numeric_vars:
    plt.figure(figsize=(8, 5))
    sns.boxplot(data=df, y=var, x="Attrition")
    plt.title(f"Distribution of {var} vs. Attrition")
    plt.show()
```



In [16]:

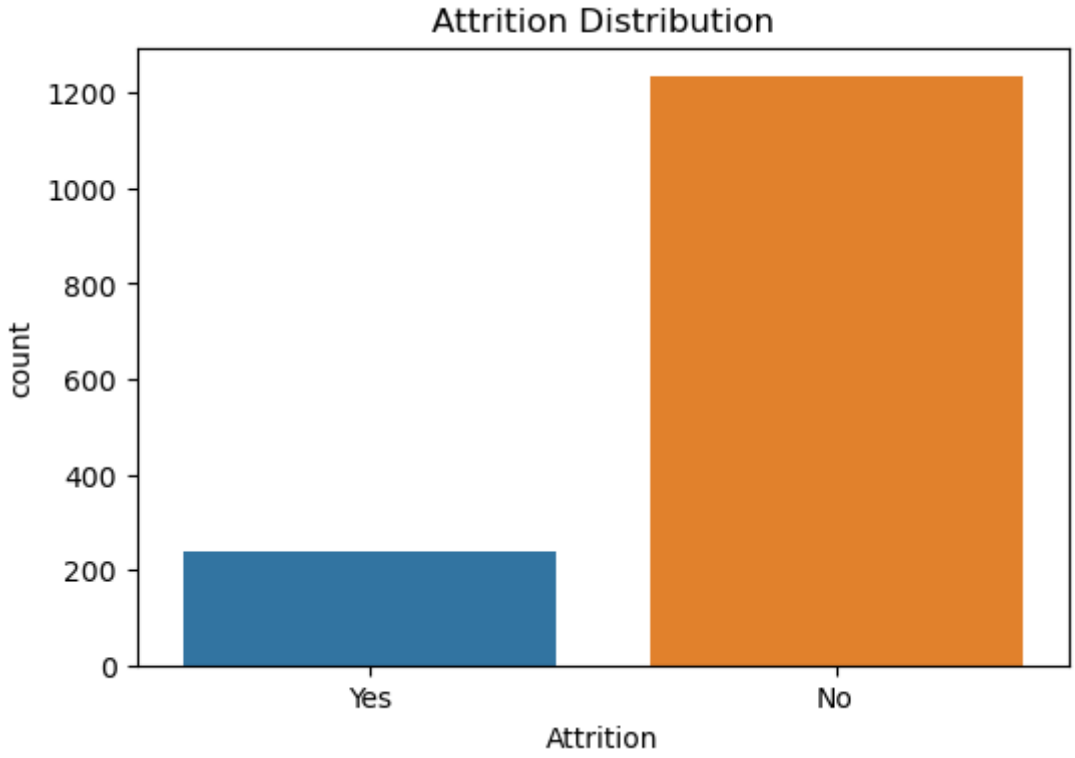
```
# Calculate and visualize the correlation between numeric variables
correlation_matrix = df[numeric_vars].corr()
sns.heatmap(correlation_matrix, annot=True, cmap="coolwarm")
plt.title("Correlation Matrix")
plt.show()
```



Analyze Attrition

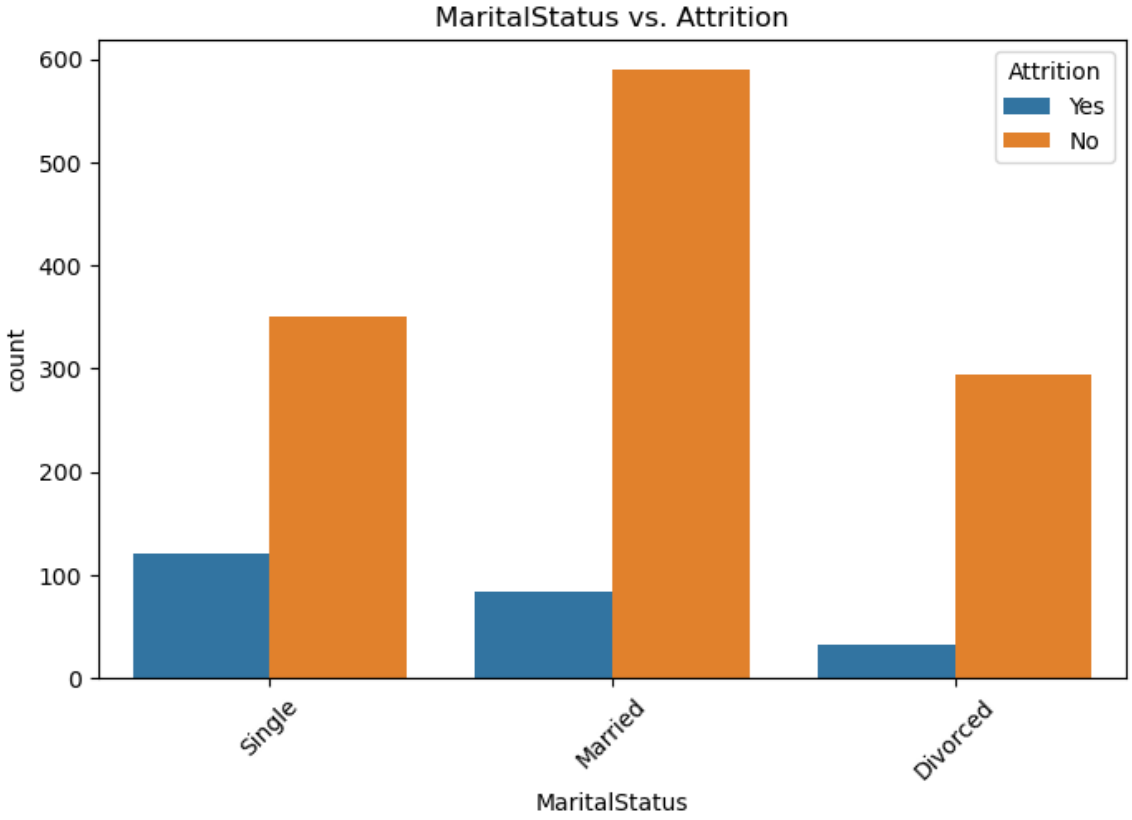
In [17]:

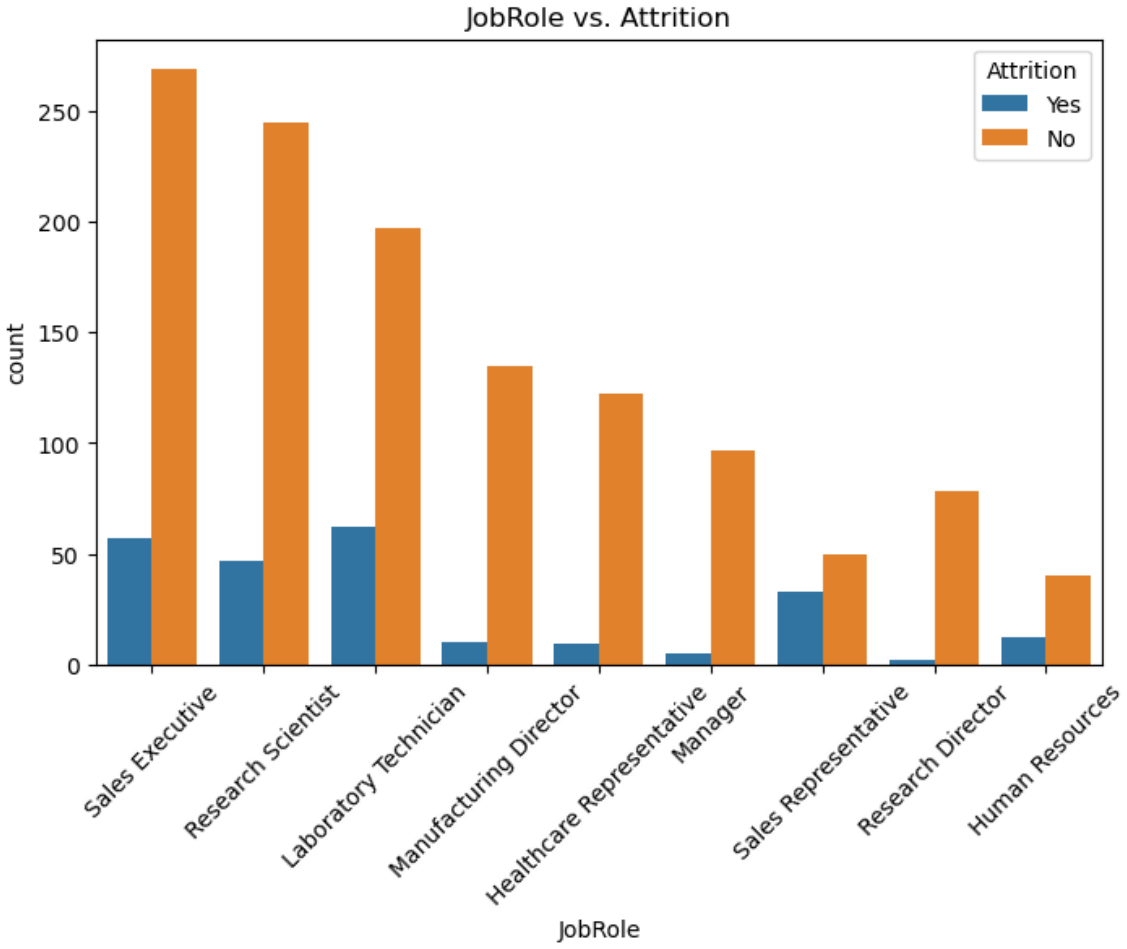
```
# Understand the distribution of Attrition
plt.figure(figsize=(6, 4))
sns.countplot(data=df, x="Attrition")
plt.title("Attrition Distribution")
plt.show()
```

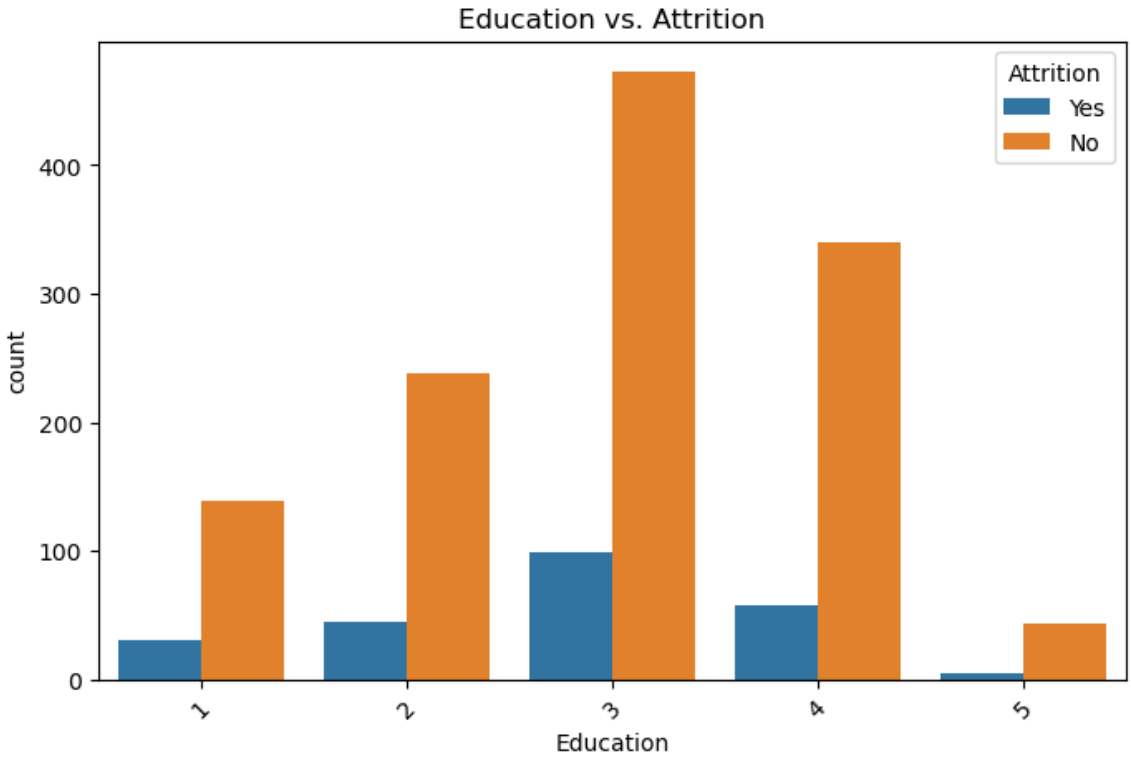


In [18]:

```
# Explore factors related to attrition
attrition_factors = ["MaritalStatus", "JobRole", "Education"]
for var in attrition_factors:
    plt.figure(figsize=(8, 5))
    sns.countplot(data=df, x=var, hue="Attrition")
    plt.title(f"{var} vs. Attrition")
    plt.xticks(rotation=45)
    plt.show()
```



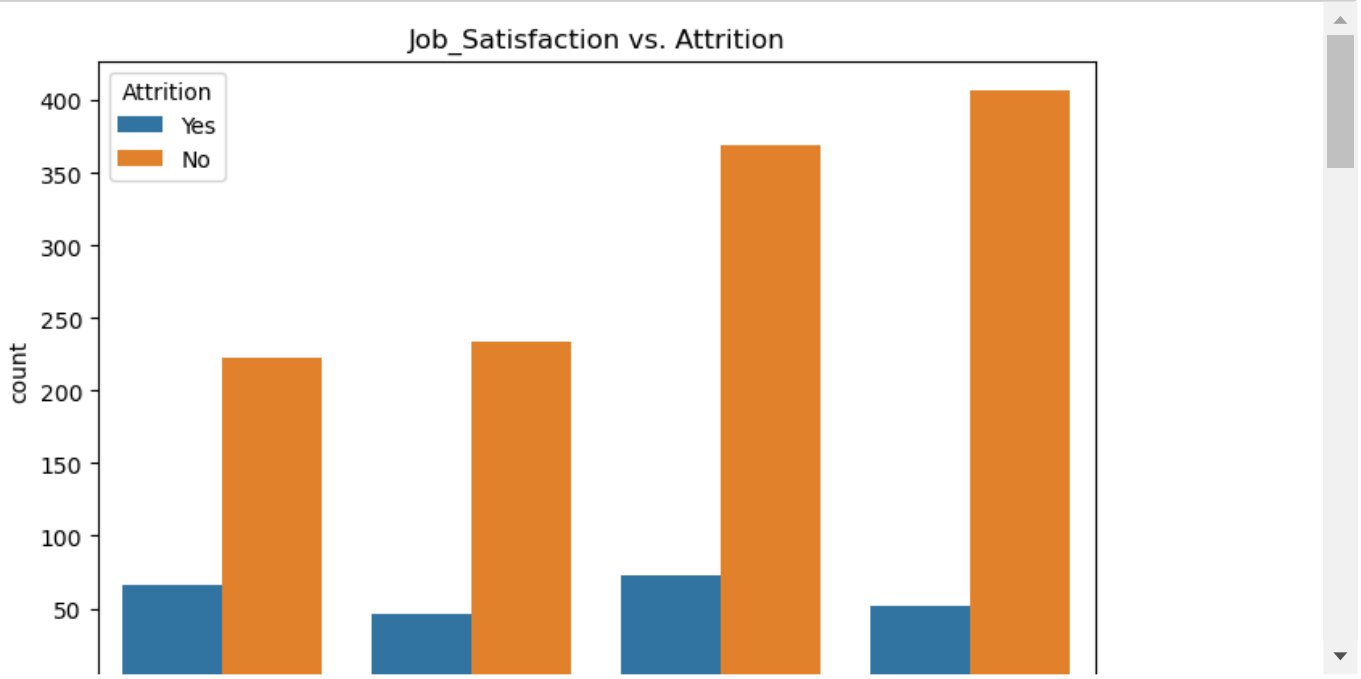

JobRole vs. Attrition
Attrition
Yes
No
count
0
50
100
150
200
250
Sales Executive
Research Scientist
Laboratory Technician
Manufacturing Director
Healthcare Representative
Manager
Sales Representative
Research Director
Human Resources
JobRole


Education vs. Attrition
Attrition
Yes
No
count
0
100
200
300
400
1
2
3
4
5
Education

Employee Satisfaction and Engagement

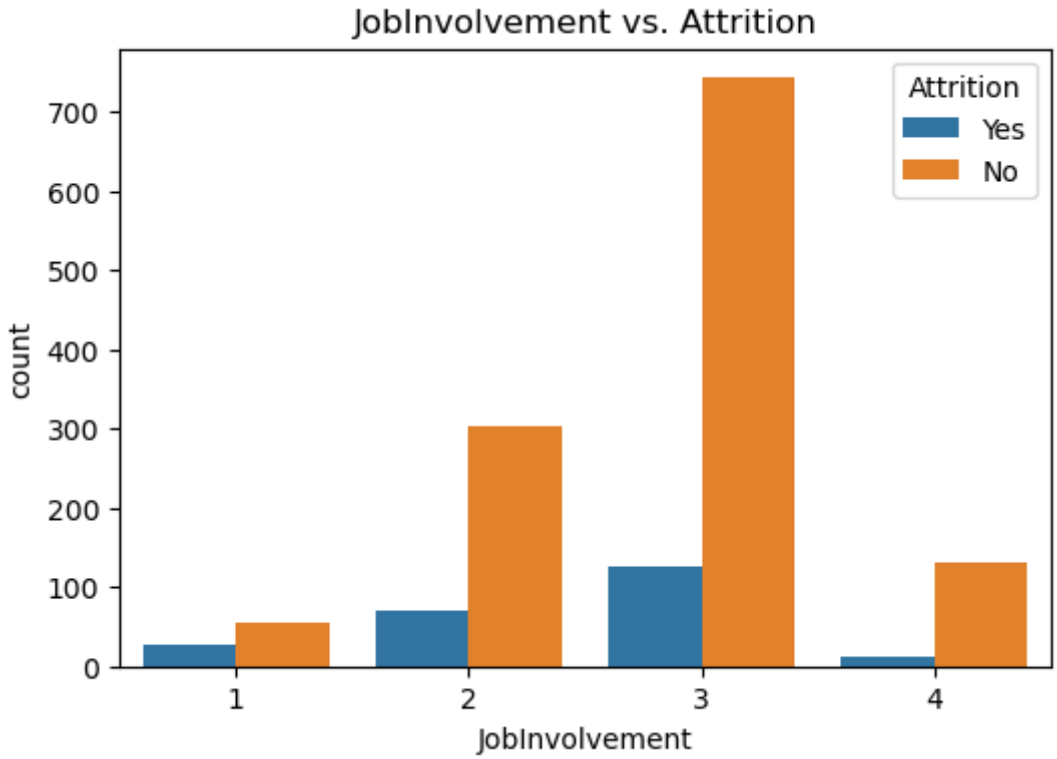
In [19]:

```
# Analyze employee satisfaction and engagement
satisfaction_vars = ["Job_Satisfaction", "EnvironmentSatisfaction", "RelationshipSatisfac
for var in satisfaction_vars:
    plt.figure(figsize=(8, 5))
    sns.countplot(data=df, x=var, hue="Attrition")
    plt.title(f"{var} vs. Attrition")
    plt.show()
```



In [20]:

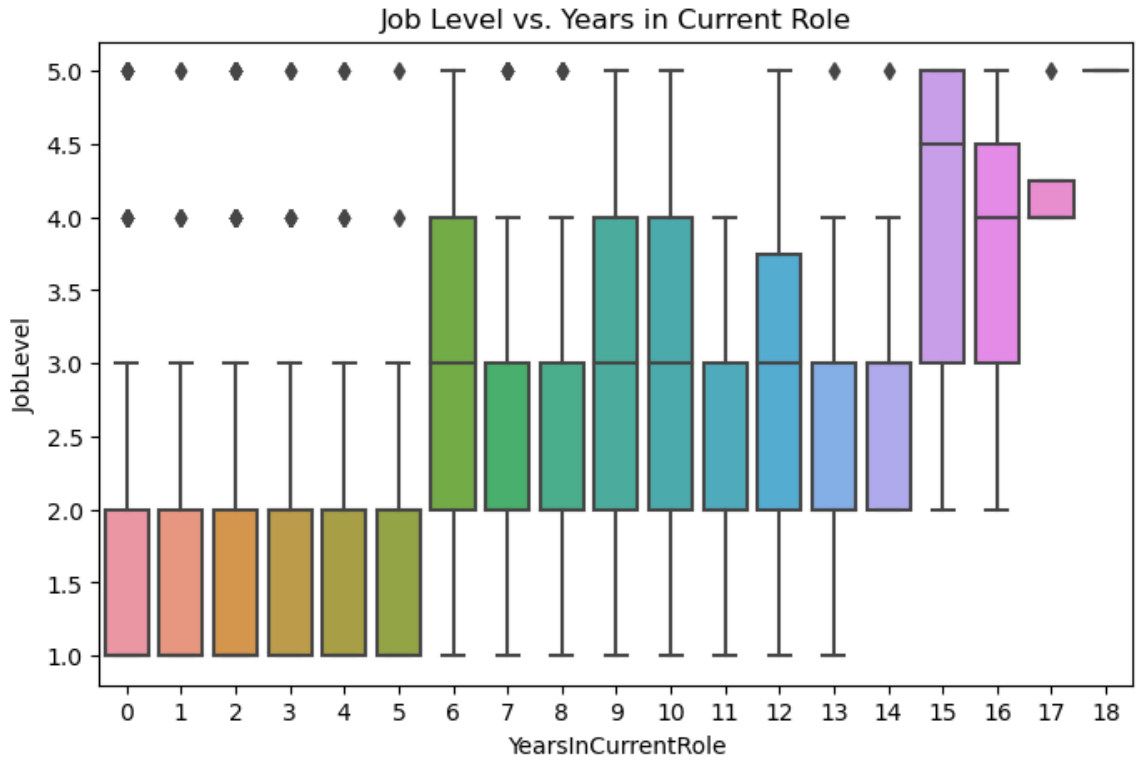
```
# Investigate the relationship between JobInvolvement and Attrition
plt.figure(figsize=(6, 4))
sns.countplot(data=df, x="JobInvolvement", hue="Attrition")
plt.title("JobInvolvement vs. Attrition")
plt.show()
```



Career Progression

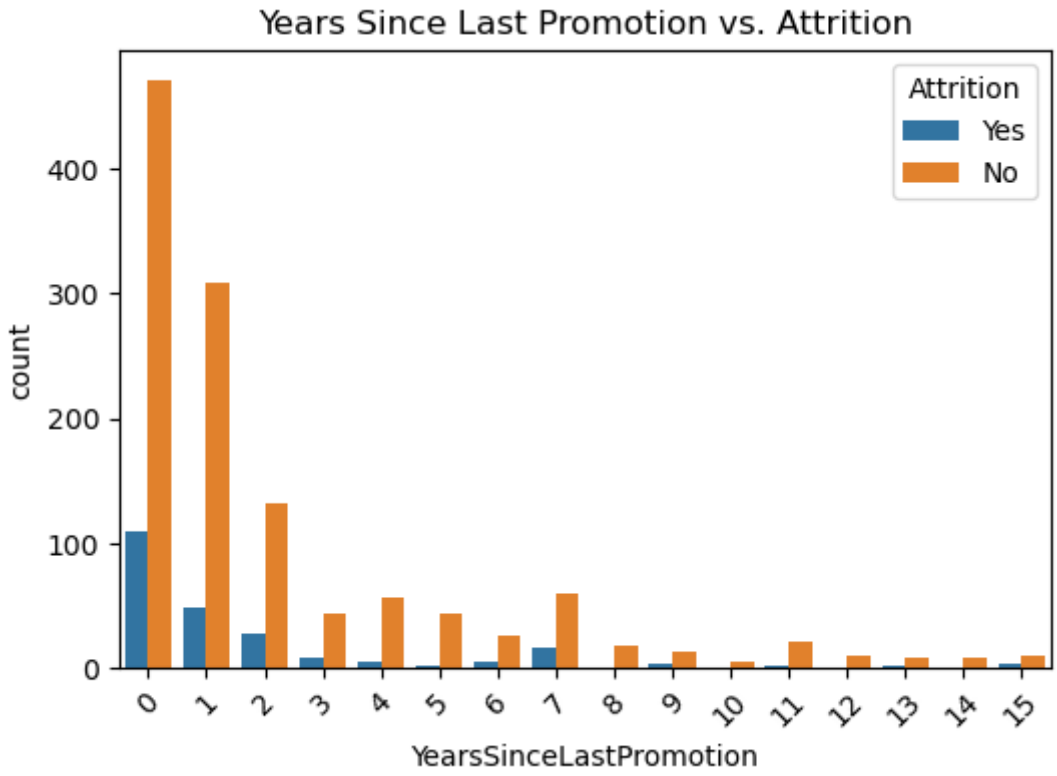
In [21]:

```
# Analyze employee career progression
plt.figure(figsize=(8, 5))
sns.boxplot(data=df, y="JobLevel", x="YearsInCurrentRole")
plt.title("Job Level vs. Years in Current Role")
plt.show()
```



In [22]:

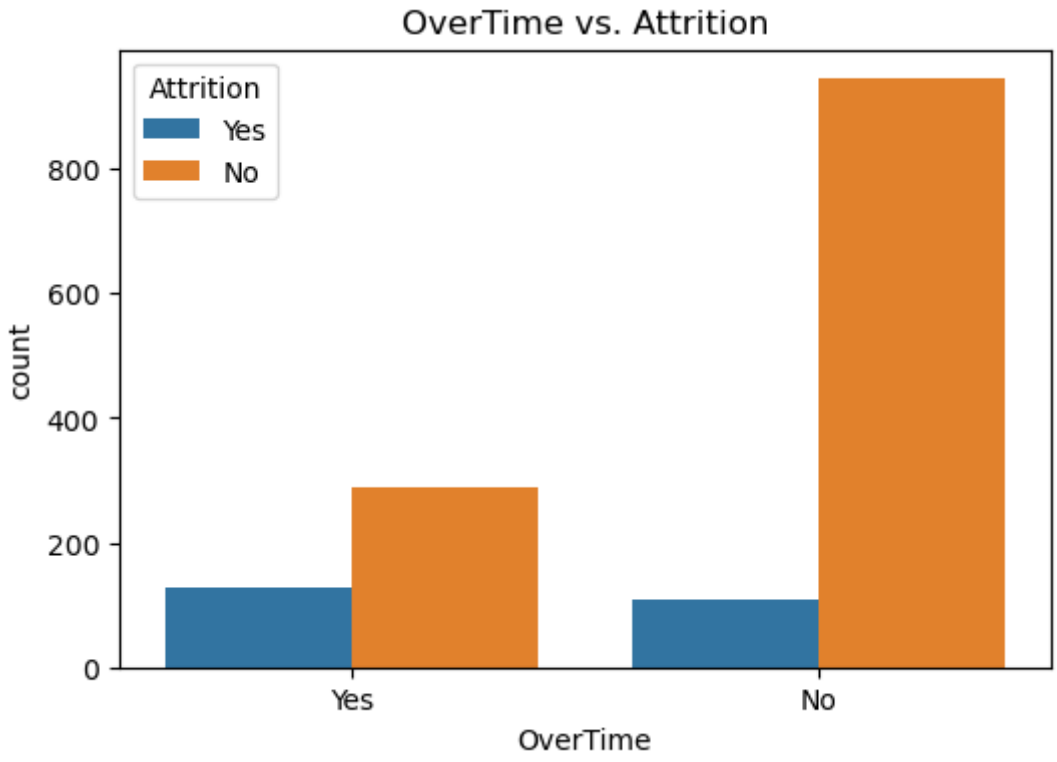
```
# Investigate the relationship between promotions and attrition
plt.figure(figsize=(6, 4))
sns.countplot(data=df, x="YearsSinceLastPromotion", hue="Attrition")
plt.title("Years Since Last Promotion vs. Attrition")
plt.xticks(rotation=45)
plt.show()
```



Work-Life Balance and Overtime

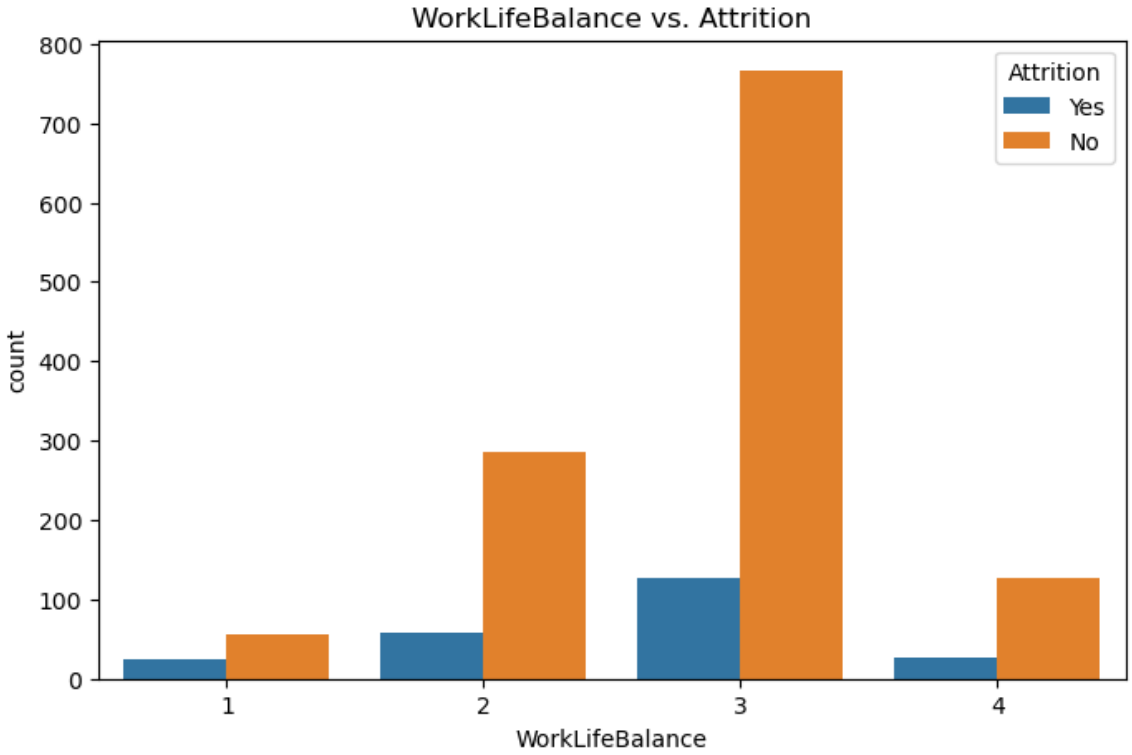
In [23]:

```
# Analyze the impact of work-life balance and overtime on attrition
plt.figure(figsize=(6, 4))
sns.countplot(data=df, x="OverTime", hue="Attrition")
plt.title("OverTime vs. Attrition")
plt.show()
```



In [24]:

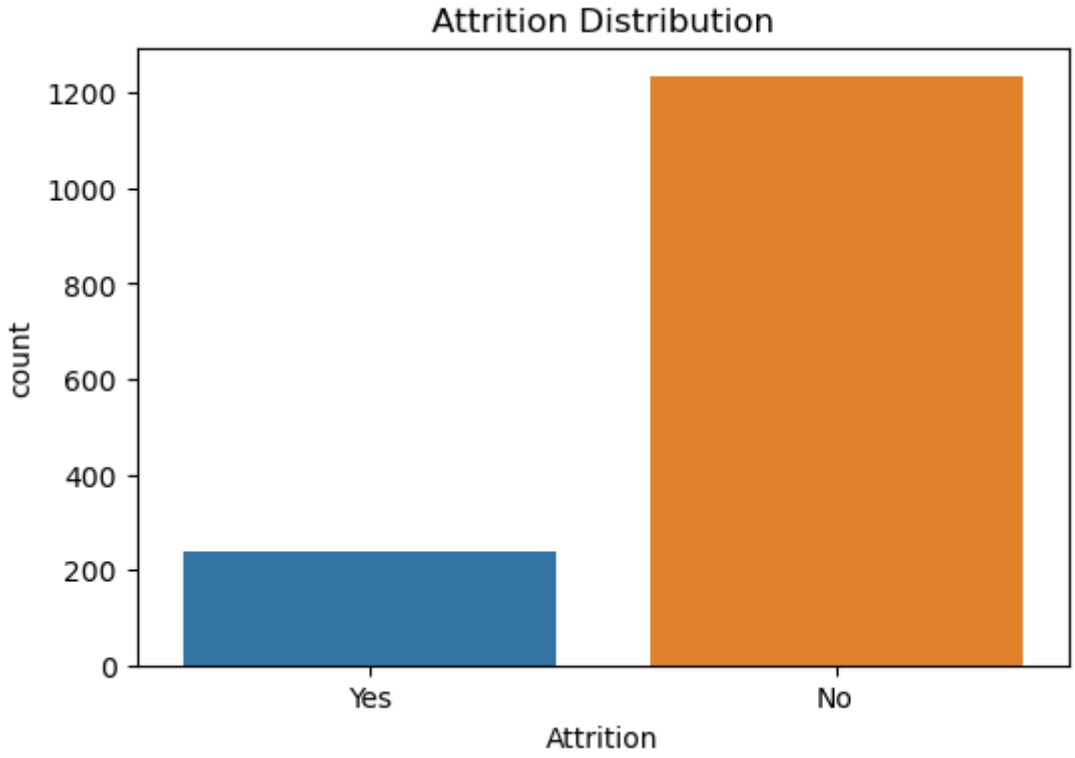
```
plt.figure(figsize=(8, 5))
sns.countplot(data=df, x="WorkLifeBalance", hue="Attrition")
plt.title("WorkLifeBalance vs. Attrition")
plt.show()
```



Analyze Attrition

In [25]:

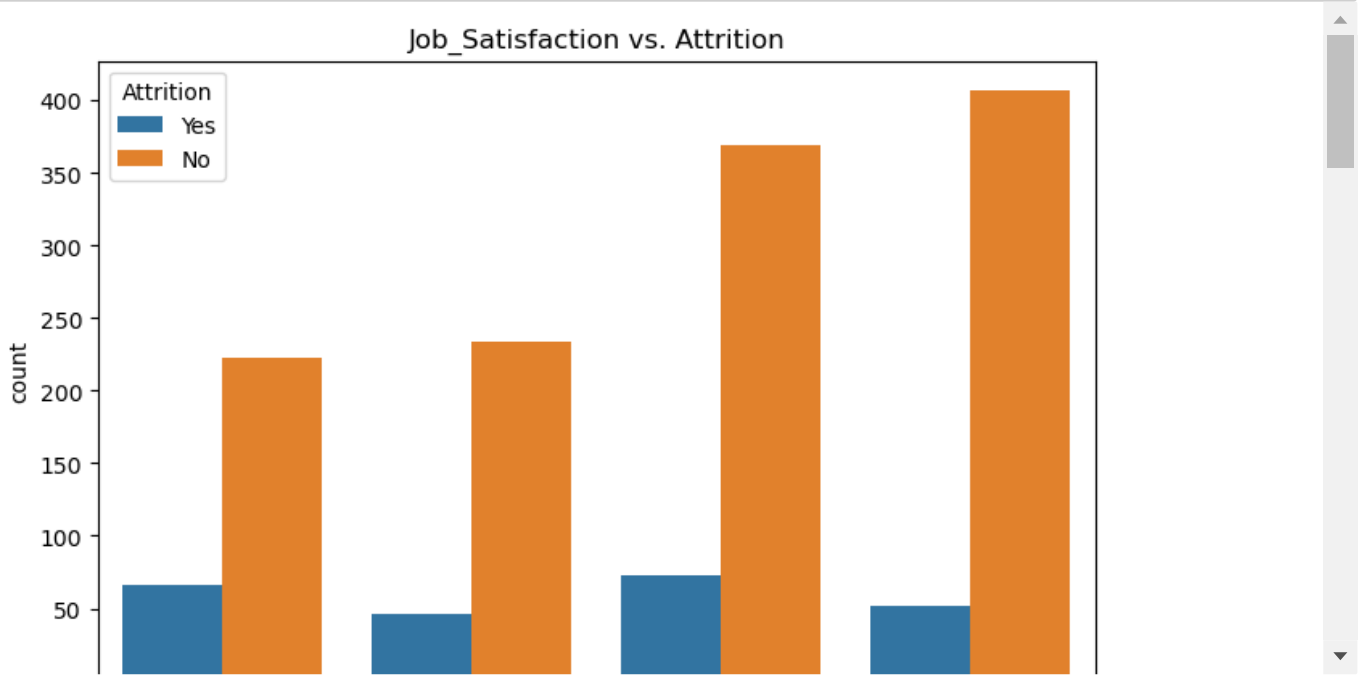
```
# Understand the distribution of Attrition
plt.figure(figsize=(6, 4))
sns.countplot(data=df, x="Attrition")
plt.title("Attrition Distribution")
plt.show()
```



Explore factors related to attrition

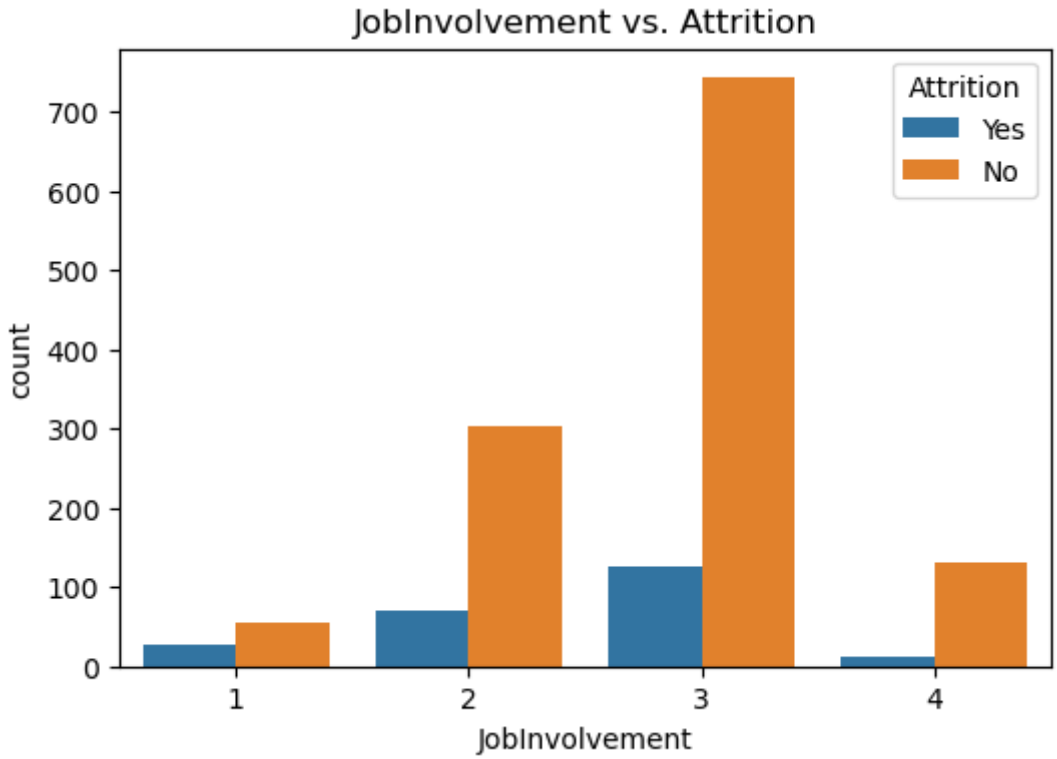
In [26]:

```
# Analyze employee satisfaction and engagement
satisfaction_vars = ["Job_Satisfaction", "EnvironmentSatisfaction", "RelationshipSatisfac
for var in satisfaction_vars:
    plt.figure(figsize=(8, 5))
    sns.countplot(data=df, x=var, hue="Attrition")
    plt.title(f"{var} vs. Attrition")
    plt.show()
```



In [27]:

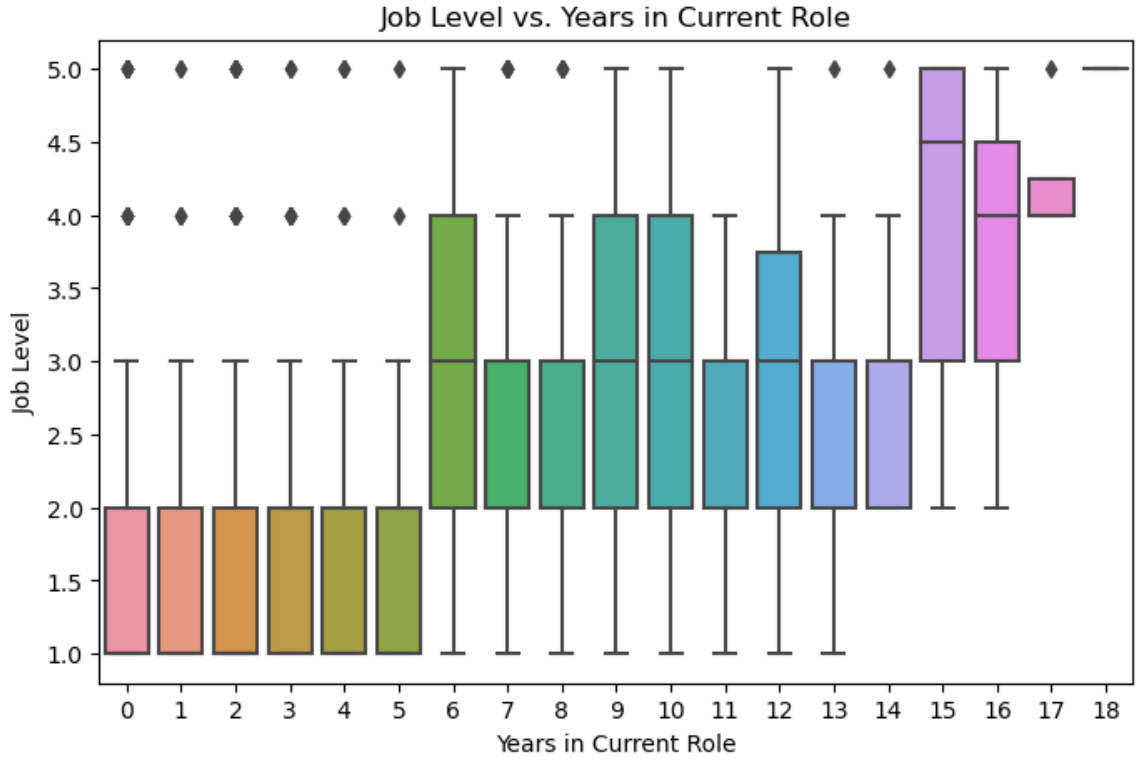
```
# Investigate the relationship between JobInvolvement and Attrition
plt.figure(figsize=(6, 4))
sns.countplot(data=df, x="JobInvolvement", hue="Attrition")
plt.title("JobInvolvement vs. Attrition")
plt.show()
```



Career Progression

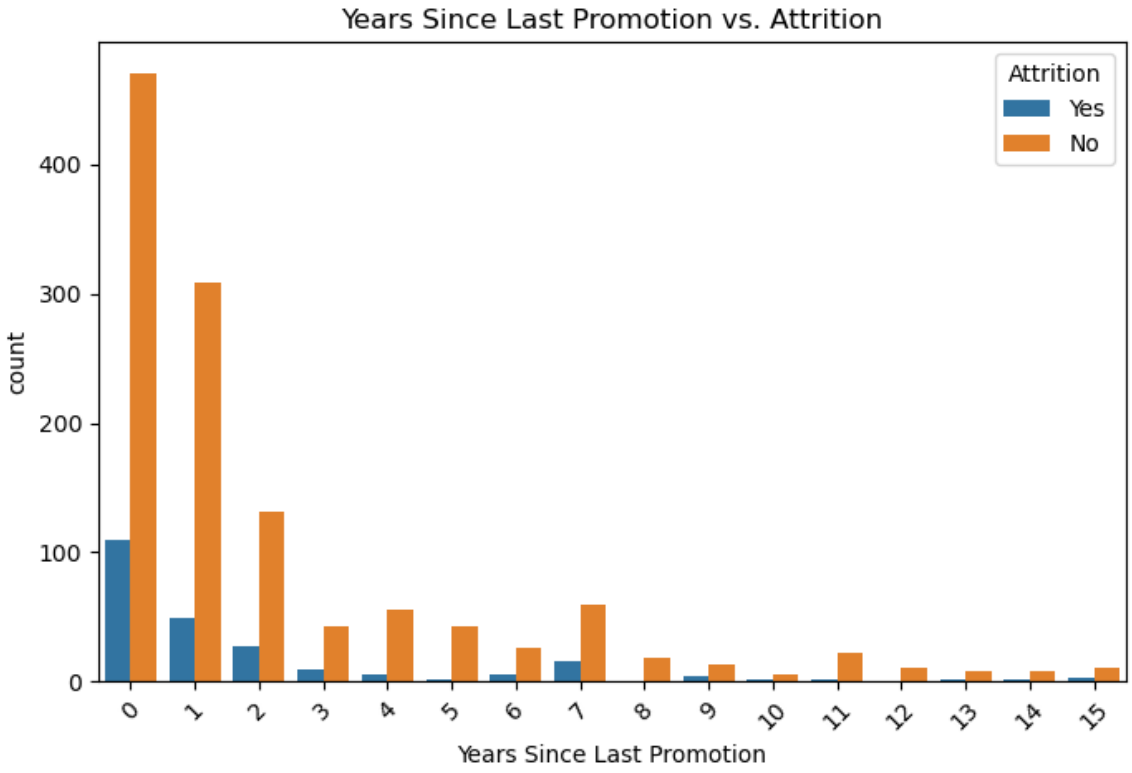
In [28]:

```
# Analyze employee career progression
plt.figure(figsize=(8, 5))
sns.boxplot(data=df, y="JobLevel", x="YearsInCurrentRole")
plt.title("Job Level vs. Years in Current Role")
plt.xlabel("Years in Current Role")
plt.ylabel("Job Level")
plt.show()
```



In [29]:

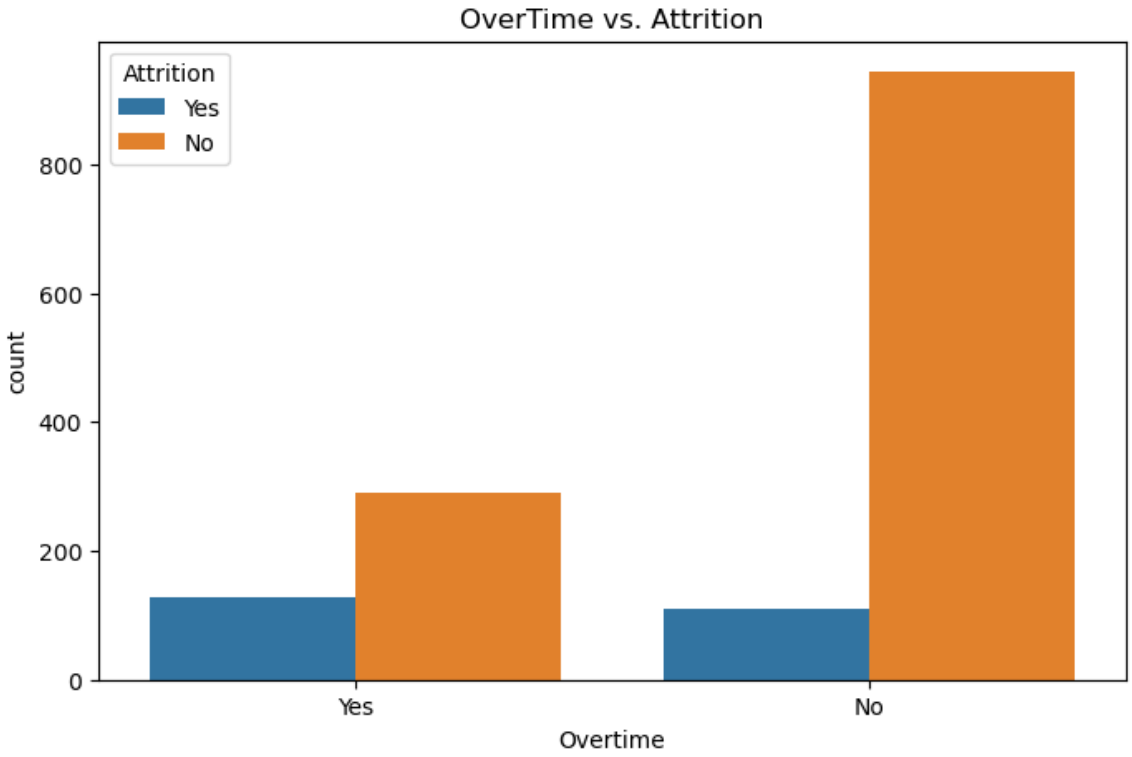
```
# Investigate the relationship between promotions and attrition
plt.figure(figsize=(8, 5))
sns.countplot(data=df, x="YearsSinceLastPromotion", hue="Attrition")
plt.title("Years Since Last Promotion vs. Attrition")
plt.xlabel("Years Since Last Promotion")
plt.xticks(rotation=45)
plt.show()
```



Work-Life Balance and Overtime

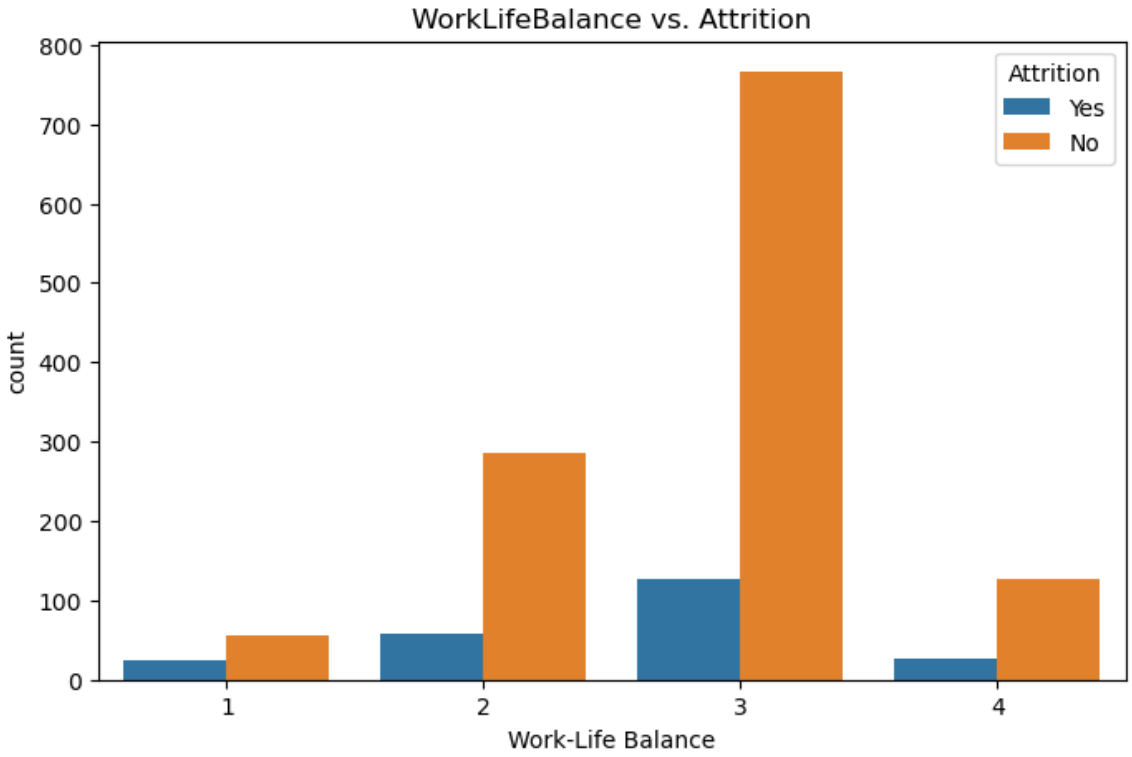
In [30]:

```
# Analyze the impact of work-life balance on attrition
plt.figure(figsize=(8, 5))
sns.countplot(data=df, x="OverTime", hue="Attrition")
plt.title("OverTime vs. Attrition")
plt.xlabel("Overtime")
plt.show()
```



In [31]:

```
# Investigate the relationship between work-life balance and attrition
plt.figure(figsize=(8, 5))
sns.countplot(data=df, x="WorkLifeBalance", hue="Attrition")
plt.title("WorkLifeBalance vs. Attrition")
plt.xlabel("Work-Life Balance")
plt.show()
```



Conclusion and Recommendations

In [32]:

```
# Summarize your findings
print("Summary of Findings:")
print("- Attrition Distribution:")
attrition_counts = df['Attrition'].value_counts()
print(attrition_counts)
```

```
Summary of Findings:
- Attrition Distribution:
No     1233
Yes     237
Name: Attrition, dtype: int64
```

In [33]:

```
# Provide recommendations
print("\nRecommendations:")
print("- Consider improving work-life balance to reduce attrition among employees.")
print("- Monitor the impact of overtime work on attrition and take necessary actions to m
print("- Focus on career development opportunities, such as promotions and skill developm
print("- Conduct exit interviews with departing employees to gather more insights into at
```

```
Recommendations:
- Consider improving work-life balance to reduce attrition among employee
s.
- Monitor the impact of overtime work on attrition and take necessary acti
ons to manage workload.
- Focus on career development opportunities, such as promotions and skill
development, to enhance job satisfaction.
- Conduct exit interviews with departing employees to gather more insights
into attrition reasons.
```

In [34]:

```
# Overall Conclusion
print("\nOverall Conclusion:")
print("Based on the analysis, we have identified several factors that are related to attr
```

```
Overall Conclusion:
Based on the analysis, we have identified several factors that are related
to attrition within the organization. It's important for the company to ad
dress these factors in order to improve employee retention and satisfactio
n. By implementing the recommended actions, the company can work towards r
educing attrition and creating a more positive work environment.
```

In [ ]:

```

```