In [2]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns

# Load Dataset
df = pd.read_excel("Employee Performance Data Set.xlsx")
df.head()
```

Out[2]:

	Age	Gender	EducationBackground	MaritalStatus	EmpDepartment	EmpJobRole	Busin
0	40	Male	Life Sciences	Married	Sales	Sales Executive	
1	30	Male	Marketing	Divorced	Sales	Sales Executive	
2	52	Male	Marketing	Married	Sales	Manager	
3	25	Female	Medical	Single	Sales	Sales Executive	
4	34	Male	Other	Single	Sales	Sales Executive	

5 rows × 27 columns

1. Basic Information

In [3]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 86 entries, 0 to 85
Data columns (total 27 columns):
 #   Column                        Non-Null Count  Dtype
---  ------                        --------------  -----
 0   Age                           86 non-null     int64
 1   Gender                        86 non-null     object
 2   EducationBackground           86 non-null     object
 3   MaritalStatus                 86 non-null     object
 4   EmpDepartment                 86 non-null     object
 5   EmpJobRole                    86 non-null     object
 6   BusinessTravelFrequency       86 non-null     object
 7   DistanceFromHome              86 non-null     int64
 8   EmpEducationLevel             86 non-null     int64
 9   EmpEnvironmentSatisfaction    86 non-null     int64
 10  EmpHourlyRate                 86 non-null     int64
 11  EmpJobInvolvement             86 non-null     int64
 12  EmpJobLevel                   86 non-null     int64
 13  EmpJobSatisfaction            86 non-null     int64
 14  NumCompaniesWorked            86 non-null     int64
 15  OverTime                      86 non-null     object
 16  EmpLastSalaryHikePercent      86 non-null     int64
 17  EmpRelationshipSatisfaction   86 non-null     int64
 18  TotalWorkExperienceInYears    86 non-null     int64
 19  TrainingTimesLastYear         86 non-null     int64
 20  EmpWorkLifeBalance            86 non-null     int64
 21  ExperienceYearsAtThisCompany  86 non-null     int64
 22  ExperienceYearsInCurrentRole  86 non-null     int64
 23  YearsSinceLastPromotion       86 non-null     int64
 24  YearsWithCurrManager          86 non-null     int64
 25  Attrition                     86 non-null     object
 26  PerformanceRating             86 non-null     int64
dtypes: int64(19), object(8)
memory usage: 18.3+ KB
```

In [4]:
```
df.describe(include="all")
```

Out[4]:

	Age	Gender	EducationBackground	MaritalStatus	EmpDepartment	EmpJobl
count	86.000000	86	86	86	86	
unique	NaN	2	6	3	4	
top	NaN	Male	Life Sciences	Married	Research & Development	S Execu
freq	NaN	53	35	36	41	
mean	37.209302	NaN	NaN	NaN	NaN	
std	9.577076	NaN	NaN	NaN	NaN	
min	18.000000	NaN	NaN	NaN	NaN	
25%	31.000000	NaN	NaN	NaN	NaN	
50%	37.000000	NaN	NaN	NaN	NaN	
75%	43.000000	NaN	NaN	NaN	NaN	
max	60.000000	NaN	NaN	NaN	NaN	

11 rows × 27 columns

2. Check Missing Values

In [5]:
```
df.isnull().sum()
```

Out[5]:
```
Age                           0
Gender                        0
EducationBackground           0
MaritalStatus                 0
EmpDepartment                 0
EmpJobRole                    0
BusinessTravelFrequency       0
DistanceFromHome              0
EmpEducationLevel             0
EmpEnvironmentSatisfaction    0
EmpHourlyRate                 0
EmpJobInvolvement             0
EmpJobLevel                   0
EmpJobSatisfaction            0
NumCompaniesWorked            0
OverTime                      0
EmpLastSalaryHikePercent      0
EmpRelationshipSatisfaction   0
TotalWorkExperienceInYears    0
TrainingTimesLastYear         0
EmpWorkLifeBalance            0
ExperienceYearsAtThisCompany  0
ExperienceYearsInCurrentRole  0
YearsSinceLastPromotion       0
YearsWithCurrManager          0
Attrition                     0
PerformanceRating             0
dtype: int64
```

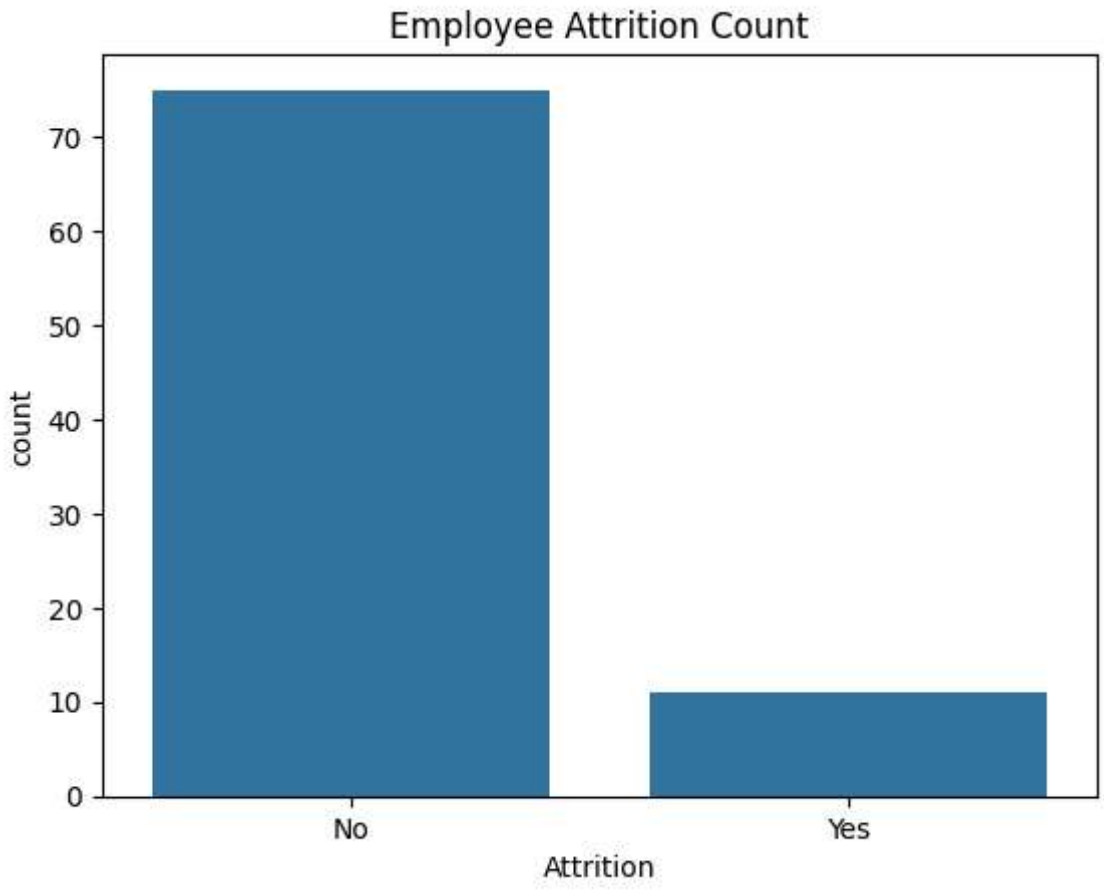
In [6]:
```
(df.isnull().sum() / len(df) * 100).sort_values(ascending=False)
```

Out[6]:
```
Age                           0.0
Gender                        0.0
EducationBackground           0.0
MaritalStatus                 0.0
EmpDepartment                 0.0
EmpJobRole                    0.0
BusinessTravelFrequency       0.0
DistanceFromHome              0.0
EmpEducationLevel             0.0
EmpEnvironmentSatisfaction    0.0
EmpHourlyRate                 0.0
EmpJobInvolvement             0.0
EmpJobLevel                   0.0
EmpJobSatisfaction            0.0
NumCompaniesWorked            0.0
OverTime                      0.0
EmpLastSalaryHikePercent      0.0
EmpRelationshipSatisfaction   0.0
TotalWorkExperienceInYears    0.0
TrainingTimesLastYear         0.0
EmpWorkLifeBalance            0.0
ExperienceYearsAtThisCompany  0.0
ExperienceYearsInCurrentRole  0.0
YearsSinceLastPromotion       0.0
YearsWithCurrManager          0.0
Attrition                     0.0
PerformanceRating             0.0
dtype: float64
```

3. Attrition Overview

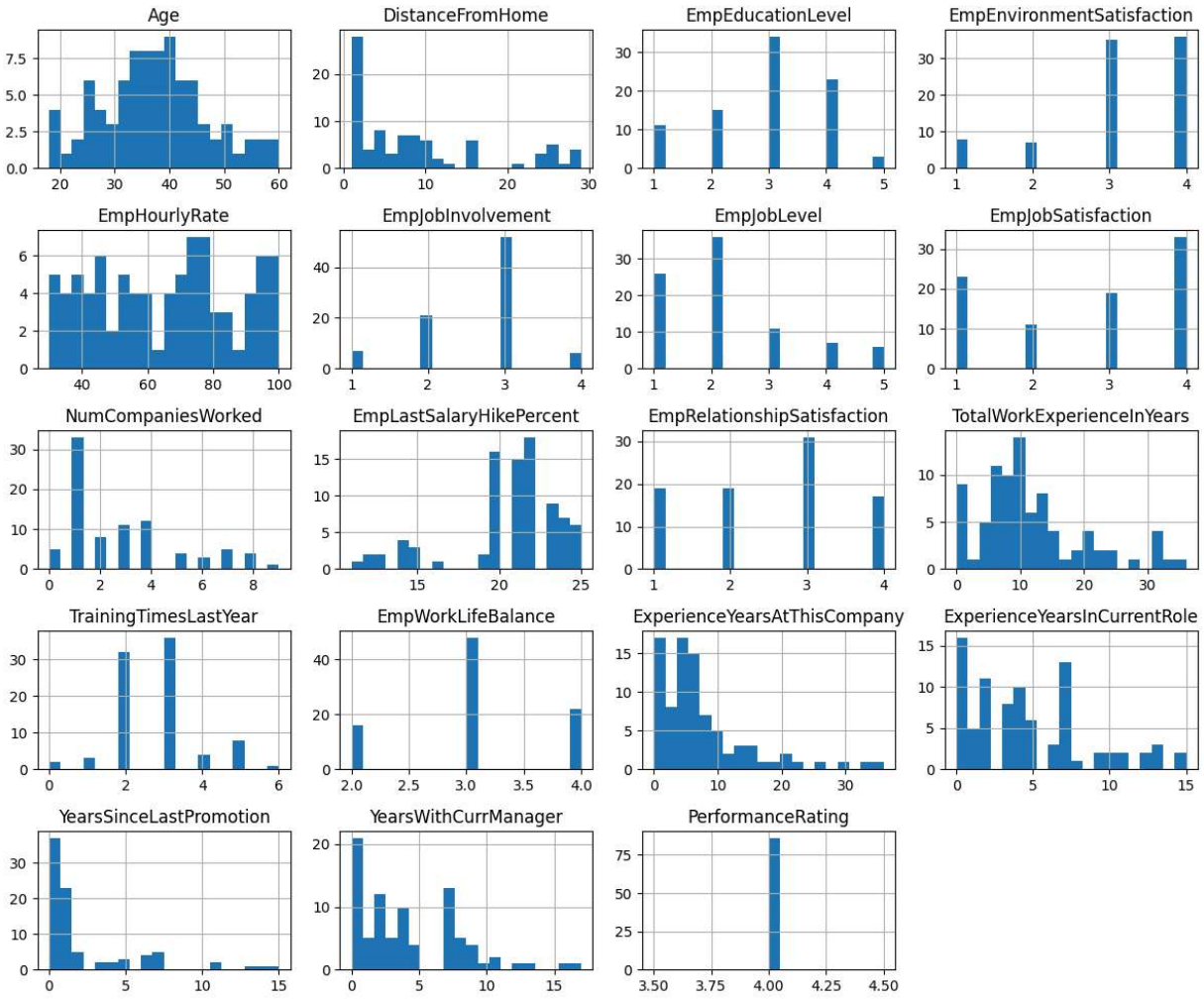
In [7]:
```
if "Attrition" in df.columns:
    print(df["Attrition"].value_counts())
    sns.countplot(data=df, x="Attrition")
    plt.title("Employee Attrition Count")
    plt.show()
else:
    print("No Attrition column found.")
```

```
Attrition
No     75
Yes    11
Name: count, dtype: int64
```



4. Numerical Column Distributions

In [8]:
```
numeric_cols = df.select_dtypes(include=np.number).columns

df[numeric_cols].hist(figsize=(12, 10), bins=20)
plt.tight_layout()
plt.show()
```



5. Categorical Column Value Counts

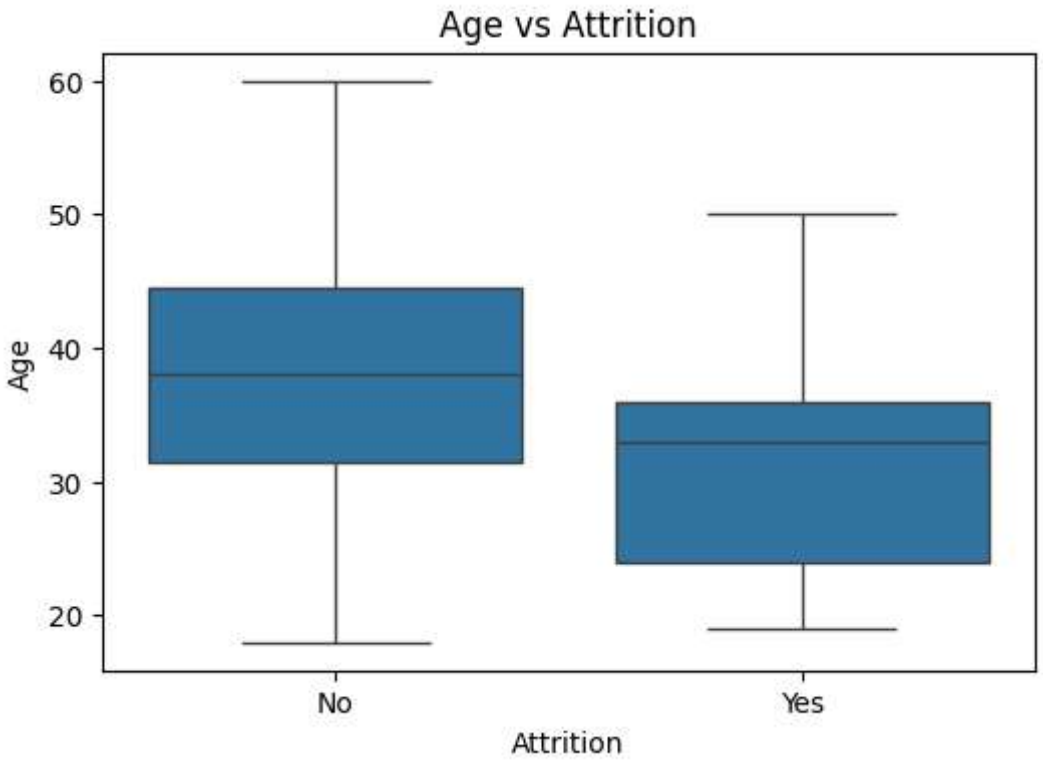
In [9]:
```
categorical_cols = df.select_dtypes(exclude=np.number).columns

for col in categorical_cols:
    print(f"---- {col} ----")
    print(df[col].value_counts())
    print("\n")
```

```
---- Gender ----
Gender
Male      53
Female    33
Name: count, dtype: int64


---- EducationBackground ----
EducationBackground
Life Sciences       35
Medical             23
Marketing           14
Technical Degree     7
Other                5
Human Resources      2
Name: count, dtype: int64


---- MaritalStatus ----
MaritalStatus
Married     36
Single      30
Divorced    20
Name: count, dtype: int64


---- EmpDepartment ----
EmpDepartment
Research & Development    41
Sales                     35
Human Resources            6
Finance                    4
Name: count, dtype: int64


---- EmpJobRole ----
EmpJobRole
Sales Executive              25
Research Scientist           14
Manager R&D                  11
Manager                      10
Laboratory Technician         5
Human Resources               5
Finance Manager               4
Sales Representative          3
Research Director             3
Healthcare Representative     3
Manufacturing Director        2
Senior Manager R&D            1
Name: count, dtype: int64


---- BusinessTravelFrequency ----
BusinessTravelFrequency
Travel_Rarely        53
Travel_Frequently    17
```

```
Non-Travel           16
Name: count, dtype: int64


---- OverTime ----
OverTime
No     62
Yes    24
Name: count, dtype: int64


---- Attrition ----
Attrition
No     75
Yes    11
Name: count, dtype: int64
```

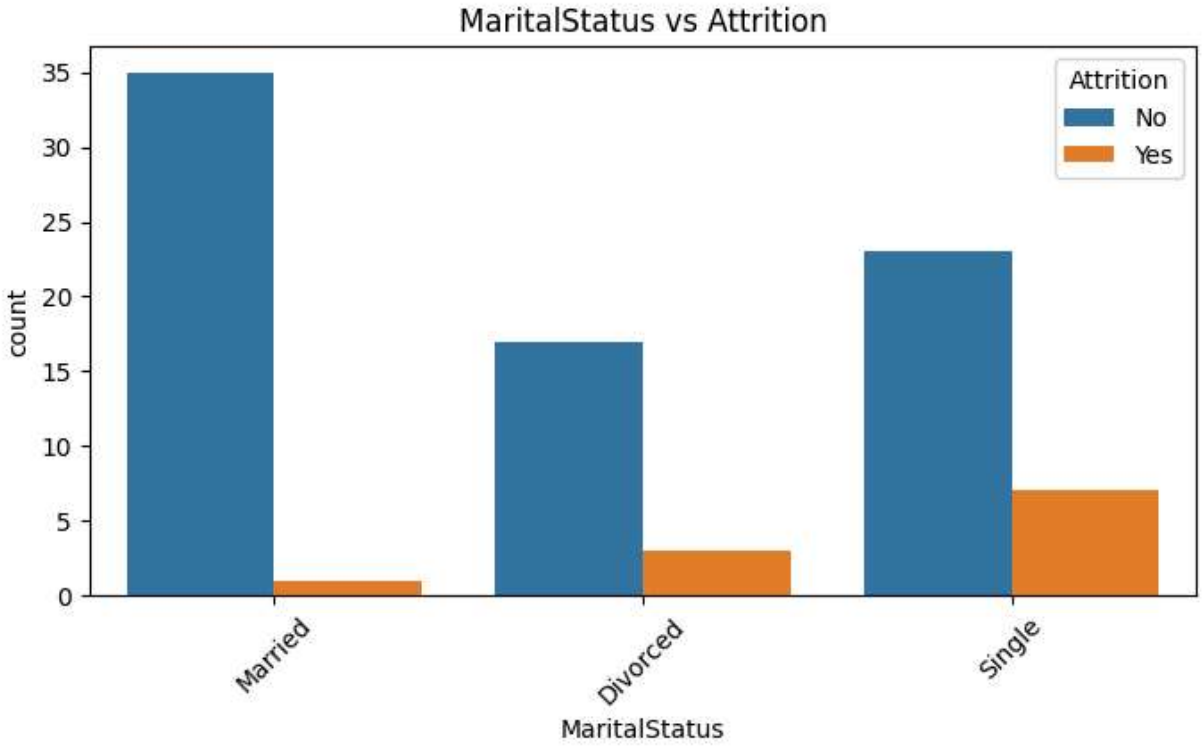
6. Attrition vs. Key Numeric Variables

In [11]:
```
important_cols = ["Age", "MonthlyIncome", "TotalWorkingYears"]

for col in important_cols:
    if col in df.columns:
        plt.figure(figsize=(6,4))
        sns.boxplot(data=df, x="Attrition", y=col)
        plt.title(f"{col} vs Attrition")
        plt.show()
```



7. Attrition vs. Categorical Variables

In [12]:
```
cat_to_check = ["JobRole", "MaritalStatus", "Department"]

for col in cat_to_check:
    if col in df.columns:
        plt.figure(figsize=(8,4))
        sns.countplot(data=df, x=col, hue="Attrition")
        plt.title(f"{col} vs Attrition")
        plt.xticks(rotation=45)
        plt.show()
```



In [ ]: