In [1]:
```
import os
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

In [2]:
```
current_directory = os.getcwd()
print("Current Directory:", current_directory)
files = os.listdir()
print("Files in Directory:", files)
```

```
Current Directory: /Users/amarachiordor/Documents/HNG Data Analysis/Oshok
e's Project
Files in Directory: ['Indicino Project.ipynb', 'Indicino project.xlsx - At
trition_data.csv', 'Indicino project.xlsx', '.ipynb_checkpoints']
```

In [3]:
```
#What This Code Does:
#Loads the dataset into Python.
#Prints the structure of the dataset (how many rows and columns it has).
#Checks if any data is missing (e.g., empty cells).
#Gives summary statistics (e.g., average salary, minimum and maximum age)
#Shows unique values in text-based columns like “JobRole” and “Department

df = pd.read_csv("Indicino project.xlsx - Attrition_data.csv")

# Display basic information
print(df.info())

# Display the first few rows
print(df.head())

# Check for missing values
print(df.isna().sum())

# Summary statistics
print(df.describe())
```

```
<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   object
 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(25), object(10)
memory usage: 402.1+ KB
None
   Age Attrition     BusinessTravel  DailyRate              Department  \
0   41       Yes      Travel_Rarely       1102                   Sales   
1   49        No  Travel_Frequently        279  Research & Development   
2   37       Yes      Travel_Rarely       1373  Research & Development   
3   33        No  Travel_Frequently       1392  Research & Development   
4   27        No      Travel_Rarely        591  Research & Development   

   DistanceFromHome  Education EducationField  EmployeeCount  EmployeeNumb
er  \
0                 1          2  Life Sciences              1             
1                 8          1  Life Sciences              1             
2                 2          2          Other              1             
4                 3          4  Life Sciences              1             
5
```

```
4                 2          1        Medical              1
7

   ...  RelationshipSatisfaction StandardHours  StockOptionLevel  \
0  ...                         1            80                 0
1  ...                         4            80                 1
2  ...                         2            80                 0
3  ...                         3            80                 0
4  ...                         4            80                 1

  TotalWorkingYears  TrainingTimesLastYear WorkLifeBalance  YearsAtCompany
\
0                 8                      0               1               6
1                10                      3               3              10
2                 7                      3               3               0
3                 8                      3               3               8
4                 6                      3               3               2

  YearsInCurrentRole  YearsSinceLastPromotion  YearsWithCurrManager
0                  4                        0                     5
1                  7                        1                     7
2                  0                        0                     0
3                  7                        3                     0
4                  2                        2                     2

[5 rows x 35 columns]
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
                Age    DailyRate  DistanceFromHome    Education  EmployeeCo
unt  \
count  1470.000000  1470.000000       1470.000000  1470.000000         147
0.0
mean     36.923810   802.485714          9.192517     2.912925
1.0
std       9.135373   403.509100          8.106864     1.024165
0.0
min      18.000000   102.000000          1.000000     1.000000
1.0
25%      30.000000   465.000000          2.000000     2.000000
1.0
50%      36.000000   802.000000          7.000000     3.000000
1.0
75%      43.000000  1157.000000         14.000000     4.000000
1.0
max      60.000000  1499.000000         29.000000     5.000000
1.0

       EmployeeNumber  EnvironmentSatisfaction   HourlyRate     JobLevel
\
count     1470.000000              1470.000000  1470.000000  1470.000000
mean      1024.865306                 2.721769    65.891156     2.063946
std        602.024335                 1.093082    20.329428     1.106940
min          1.000000                 1.000000    30.000000     1.000000
25%        491.250000                 2.000000    48.000000     1.000000
50%       1020.500000                 3.000000    66.000000     2.000000
75%       1555.750000                 4.000000    83.750000     3.000000
max       2068.000000                 4.000000   100.000000     5.000000

       JobSatisfaction  ...  RelationshipSatisfaction  StandardHours  \
count      1470.000000  ...               1470.000000         1470.0
mean          2.728571  ...                  2.712245           80.0
std           1.102846  ...                  1.081209            0.0
min           1.000000  ...                  1.000000           80.0
25%           2.000000  ...                  2.000000           80.0
50%           3.000000  ...                  3.000000           80.0
75%           4.000000  ...                  4.000000           80.0
max           4.000000  ...                  4.000000           80.0

       StockOptionLevel  TotalWorkingYears  TrainingTimesLastYear  \
count       1470.000000        1470.000000            1470.000000
mean           0.793878          11.279592               2.799320
std            0.852077           7.780782               1.289271
min            0.000000           0.000000               0.000000
25%            0.000000           6.000000               2.000000
50%            1.000000          10.000000               3.000000
75%            1.000000          15.000000               3.000000
max            3.000000          40.000000               6.000000

       WorkLifeBalance  YearsAtCompany  YearsInCurrentRole  \
count      1470.000000     1470.000000         1470.000000
mean          2.761224        7.008163            4.229252
std           0.706476        6.126525            3.623137
min           1.000000        0.000000            0.000000
25%           2.000000        3.000000            2.000000
50%           3.000000        5.000000            3.000000
75%           3.000000        9.000000            7.000000
```

```
max           4.000000       40.000000           18.000000

       YearsSinceLastPromotion  YearsWithCurrManager
count              1470.000000           1470.000000
mean                  2.187755              4.123129
std                   3.222430              3.568136
min                   0.000000              0.000000
25%                   0.000000              2.000000
50%                   1.000000              3.000000
75%                   3.000000              7.000000
max                  15.000000             17.000000

[8 rows x 25 columns]
```

In [4]:
```
df.shape
```

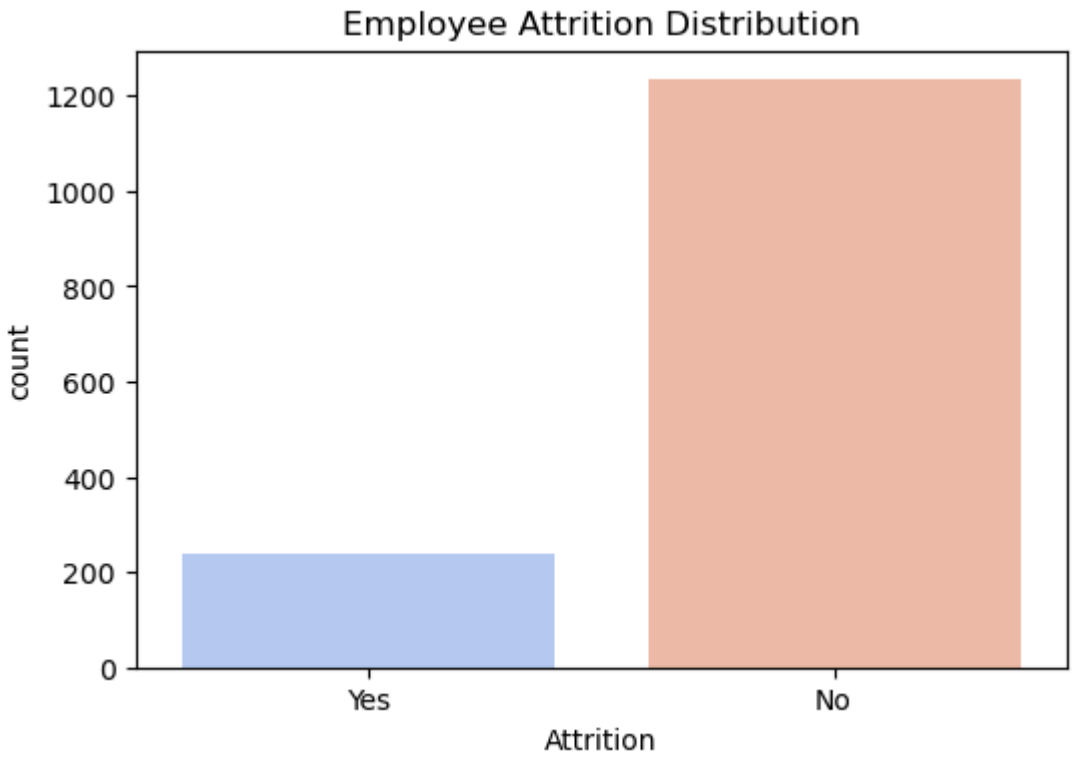
Out[4]:
```
(1470, 35)
```

In [5]:
```
# Check unique values in categorical columns
categorical_cols = df.select_dtypes(include=['object']).columns
for col in categorical_cols:
    print(f"{col}: {df[col].unique()}")
```

```
Attrition: ['Yes' 'No']
BusinessTravel: ['Travel_Rarely' 'Travel_Frequently' 'Non-Travel']
Department: ['Sales' 'Research & Development' 'Human Resources']
EducationField: ['Life Sciences' 'Other' 'Medical' 'Marketing' 'Technical
Degree'
 'Human Resources']
Gender: ['Female' 'Male']
 JobInvolvement : [' $ 3 ' ' $ 2 ' ' $ 4 ' ' $ 1 ']
JobRole: ['Sales Executive' 'Research Scientist' 'Laboratory Technician'
 'Manufacturing Director' 'Healthcare Representative' 'Manager'
 'Sales Representative' 'Research Director' 'Human Resources']
MaritalStatus: ['Single' 'Married' 'Divorced']
Over18: ['Y']
OverTime: ['Yes' 'No']
```

In [6]:
```
# Drop EmployeeCount and StandardHours as they have constant values
df.drop(columns=['EmployeeCount', 'StandardHours'], inplace=True)
```

In [7]:
```
# Creates a simple bar chart to show how many employees left the company
plt.figure(figsize=(6,4))
sns.countplot(x='Attrition', hue='Attrition', data=df, palette='coolwarm'
plt.title("Employee Attrition Distribution")
plt.show()
```



In [8]:
```
# Convert categorical variables to numerical using Label Encoding
from sklearn.preprocessing import LabelEncoder

label_encoders = {}
for col in categorical_cols:
    le = LabelEncoder()
    df[col] = le.fit_transform(df[col])
    label_encoders[col] = le
```

In [9]:
```
df.head()
```

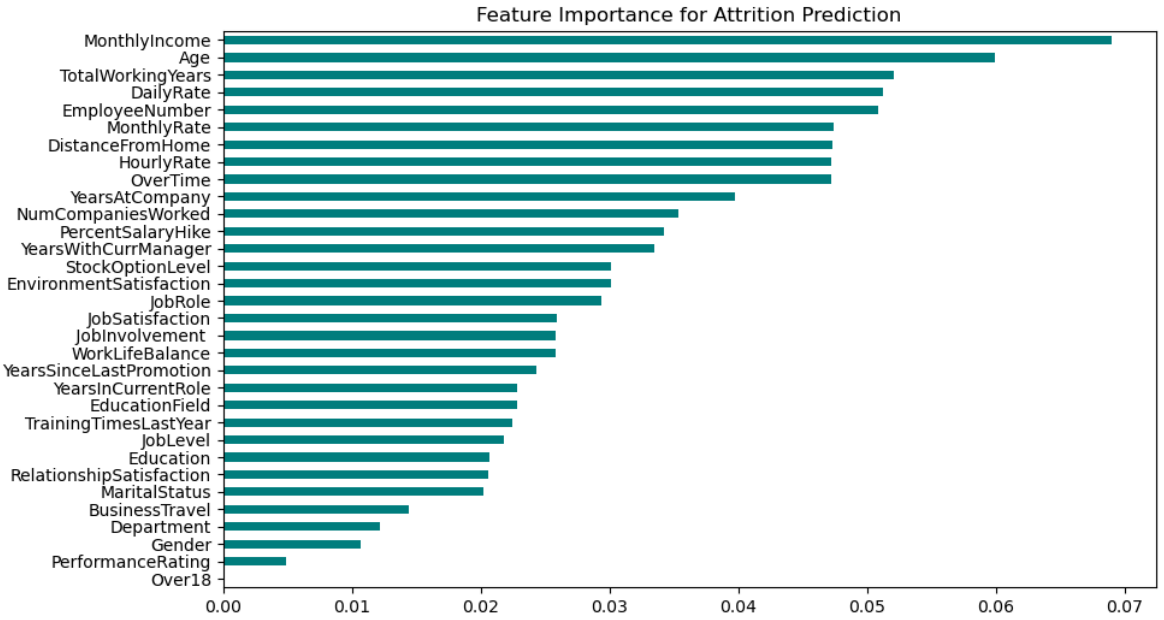
Out[9]:

	Age	Attrition	BusinessTravel	DailyRate	Department	DistanceFromHome	Edu
0	41	1	2	1102	2	1	
1	49	0	1	279	1	8	
2	37	1	2	1373	1	2	
3	33	0	1	1392	1	3	
4	27	0	2	591	1	2	

5 rows × 33 columns

In [10]:
```
#We want to find which factors (like salary, job satisfaction, etc.) are
#Uses a machine learning model to find the most important factors that af
#Creates a bar chart to show which features (salary, distance from home,

from sklearn.ensemble import RandomForestClassifier

# Define features (X) and target (y)
X = df.drop(columns=['Attrition'])  # Everything except Attrition
y = df['Attrition']  # The target we want to predict

# Train a model to find important factors
```

```
model = RandomForestClassifier(n_estimators=100, random_state=42)
model.fit(X, y)

# Show which factors are most important
feature_importances = pd.Series(model.feature_importances_, index=X.colum
feature_importances.sort_values(ascending=True).plot(kind='barh', figsize
plt.title("Feature Importance for Attrition Prediction")
plt.show()
```

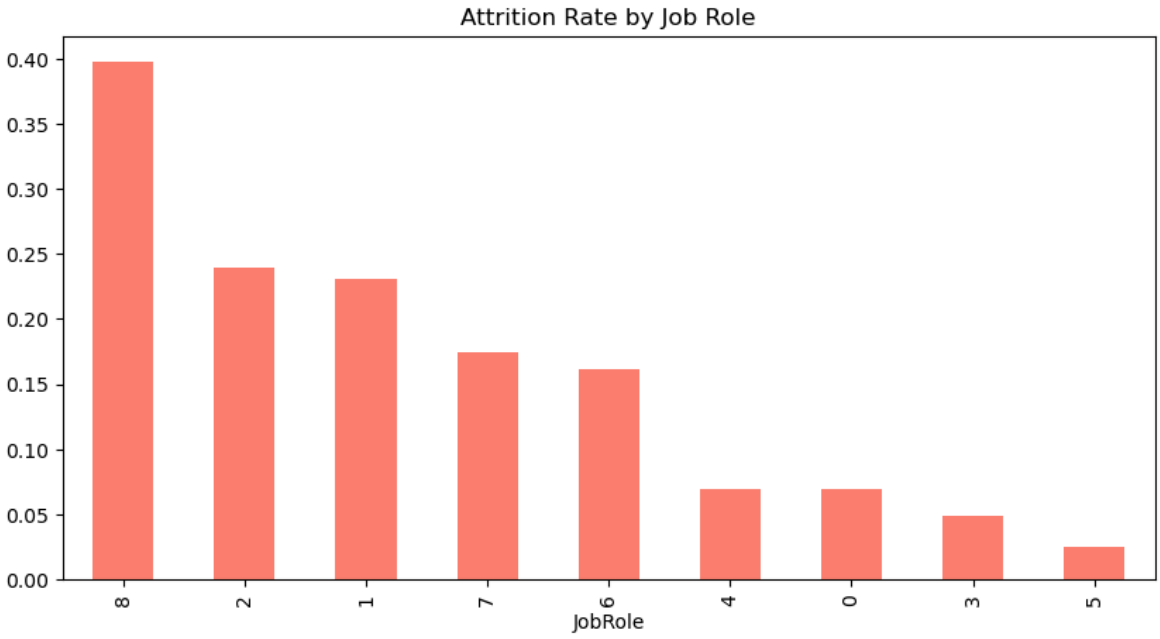


In [11]:
```
# We'll train a model to predict which job roles are at the highest risk

from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import classification_report
from sklearn.preprocessing import StandardScaler

X = df.drop(columns=["Attrition"])  # Drop the target variable
y = df["Attrition"]

# Train-test split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2,

# Scale the data
scaler = StandardScaler()
X_train_scaled = scaler.fit_transform(X_train)
X_test_scaled = scaler.transform(X_test)

# Train model with increased max_iter and saga solver
log_model = LogisticRegression(max_iter=2000, solver="saga")
log_model.fit(X_train_scaled, y_train)

# Predictions
y_pred = log_model.predict(X_test_scaled)

# Evaluation
print(classification_report(y_test, y_pred))
```

	precision	recall	f1-score	support
0	0.91	0.98	0.94	255
1	0.68	0.33	0.45	39
accuracy			0.89	294
macro avg	0.79	0.65	0.69	294
weighted avg	0.88	0.89	0.87	294

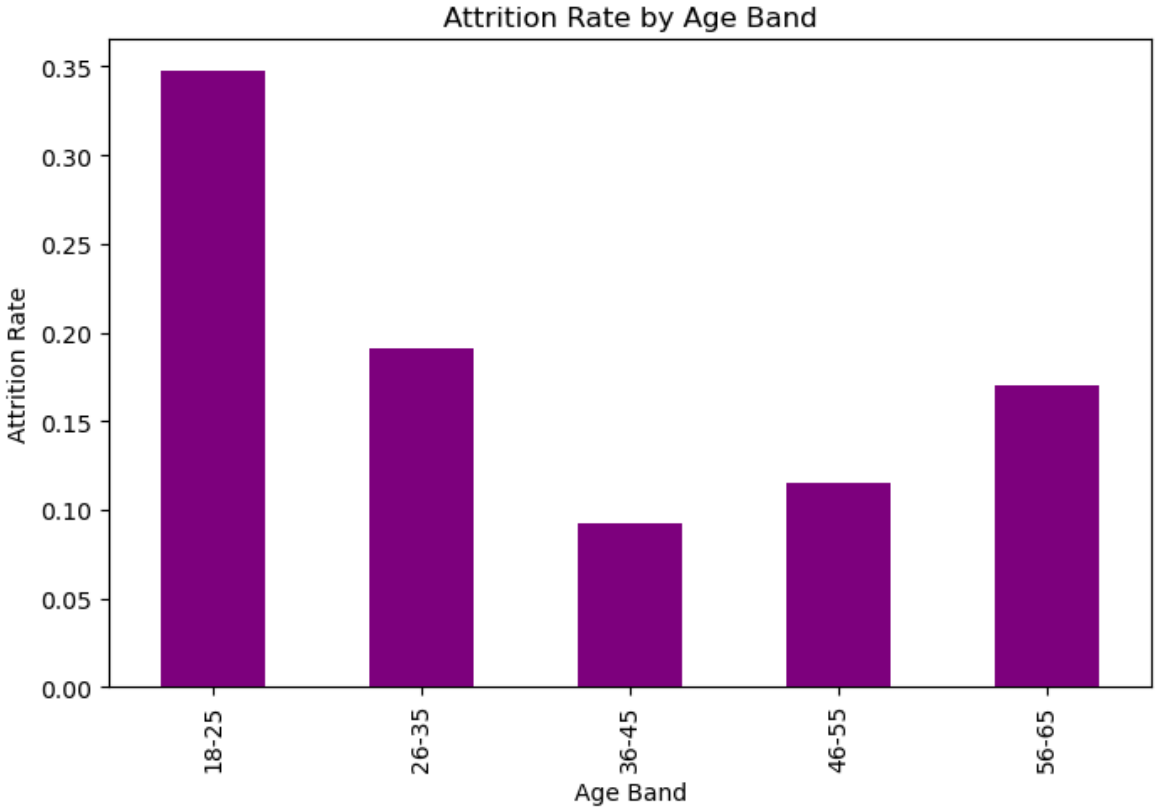
In [23]:
```
#Prints job roles with the highest resignation rates.
# Identify job roles with the highest attrition rate
job_role_attrition = df.groupby("JobRole")["Attrition"].mean().sort_value
print(job_role_attrition)

# Visualize job roles most affected
plt.figure(figsize=(10,5))
job_role_attrition.plot(kind='bar', color="salmon")
plt.title("Attrition Rate by Job Role")
plt.show()
```

```
JobRole
8    0.397590
2    0.239382
1    0.230769
7    0.174847
6    0.160959
4    0.068966
0    0.068702
3    0.049020
5    0.025000
Name: Attrition, dtype: float64
```

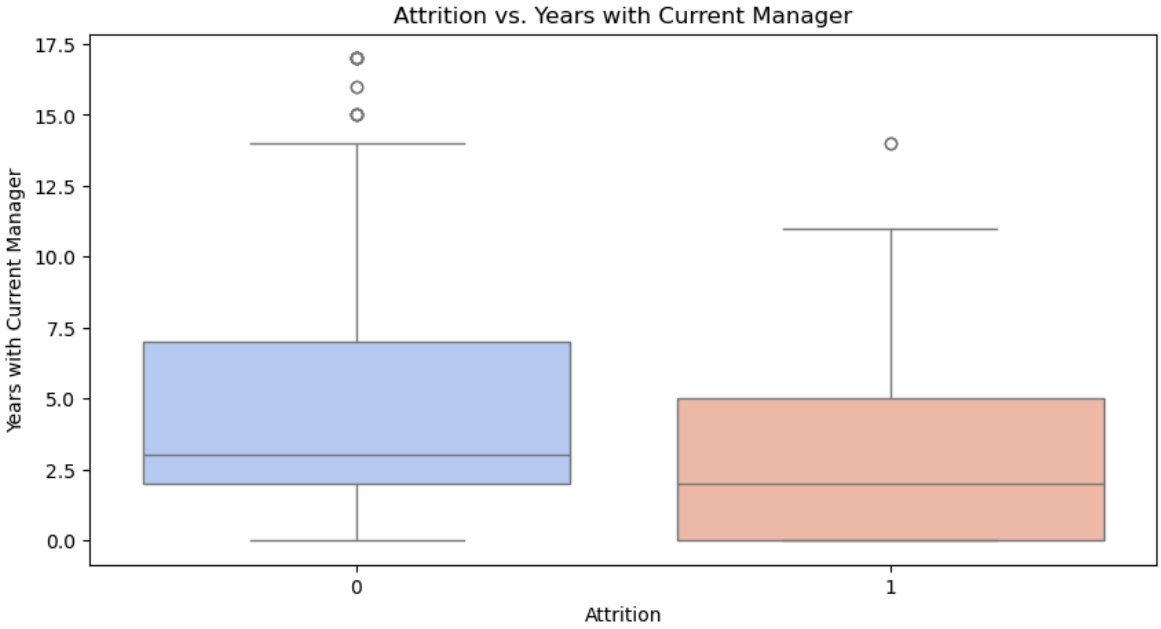


In [25]:
```
#Grouping employees into age bands to see which age group is most likely

# Create Age Bands
df['AgeBand'] = pd.cut(df['Age'], bins=[18,25,35,45,55,65], labels=['18-2

# Attrition by Age Band (fixing the FutureWarning)
age_attrition = df.groupby("AgeBand", observed=True)["Attrition"].mean()
```

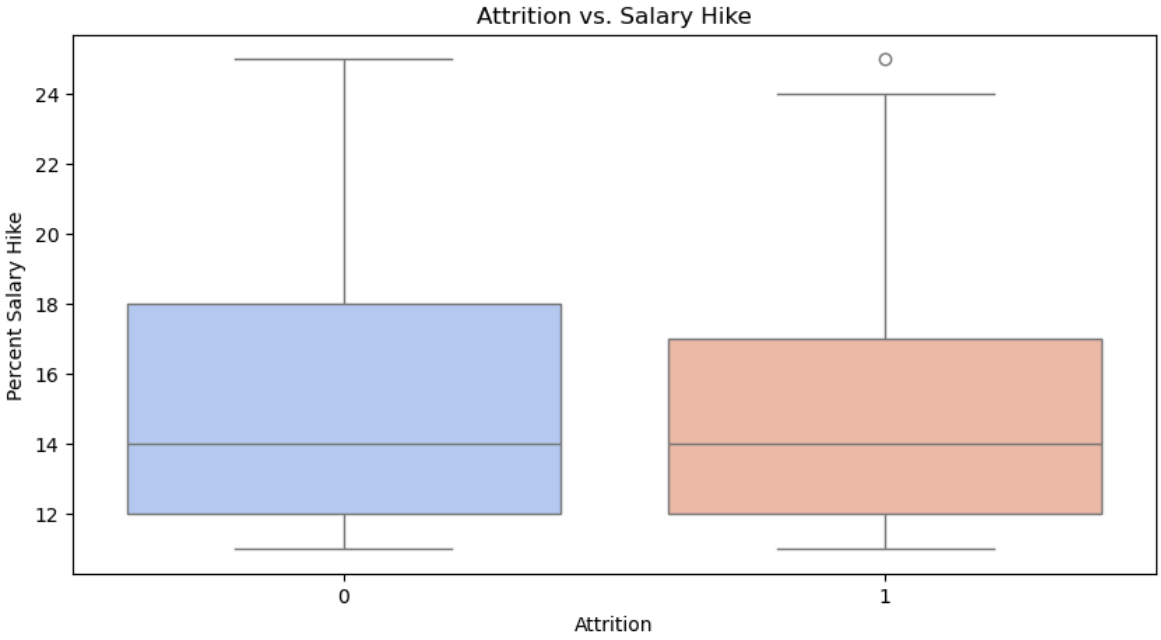
```
# Plot
plt.figure(figsize=(8,5))
age_attrition.plot(kind='bar', color="purple")
plt.title("Attrition Rate by Age Band")
plt.xlabel("Age Band")
plt.ylabel("Attrition Rate")
plt.show()
```

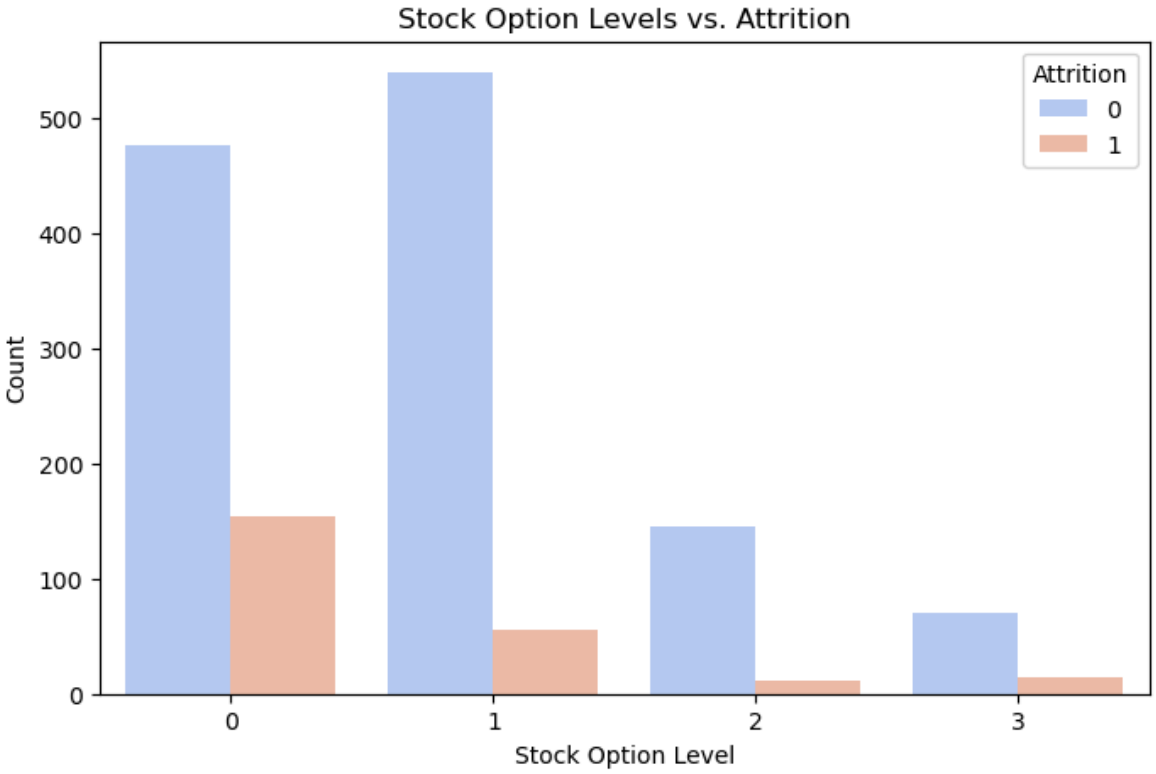


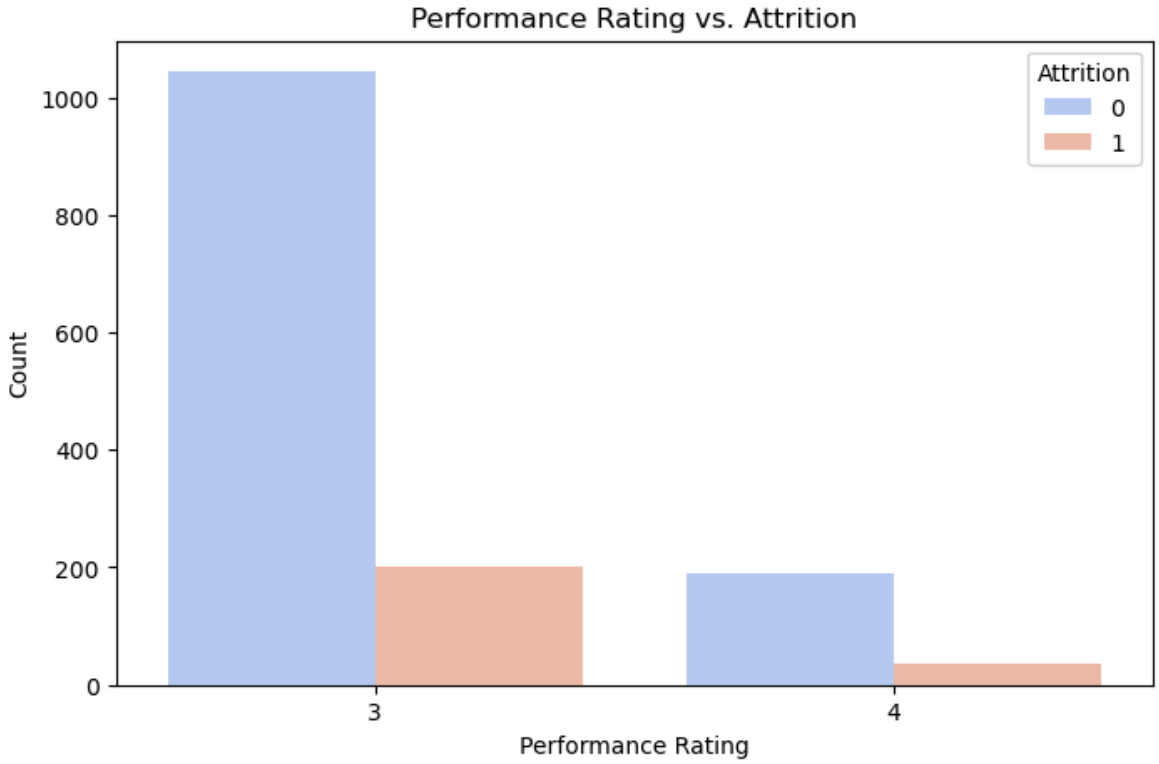
In [27]:
```
#Shows whether employees who stay with the same manager longer are more l

# Relationship between YearsWithCurrManager and Attrition
plt.figure(figsize=(10,5))
sns.boxplot(x='Attrition', y='YearsWithCurrManager', data=df, hue='Attrit
plt.title("Attrition vs. Years with Current Manager")
plt.xlabel("Attrition")
plt.ylabel("Years with Current Manager")
plt.show()
```



In [31]:
```
#Finds out if salary hikes, stock options, and performance ratings impact

# Salary Hike vs Attrition
plt.figure(figsize=(10,5))
sns.boxplot(x='Attrition', y='PercentSalaryHike', data=df, hue='Attrition
plt.title("Attrition vs. Salary Hike")
plt.xlabel("Attrition")
plt.ylabel("Percent Salary Hike")
plt.show()

# Stock Options vs Attrition
plt.figure(figsize=(8,5))
sns.countplot(x='StockOptionLevel', hue='Attrition', data=df, palette="co
plt.title("Stock Option Levels vs. Attrition")
plt.xlabel("Stock Option Level")
plt.ylabel("Count")
plt.show()

# Performance Ratings vs Attrition
plt.figure(figsize=(8,5))
sns.countplot(x='PerformanceRating', hue='Attrition', data=df, palette="c
plt.title("Performance Rating vs. Attrition")
plt.xlabel("Performance Rating")
plt.ylabel("Count")
plt.show()
```


Attrition vs. Salary Hike
Percent Salary Hike
24
22
20
18
16
14
12
0
1
Attrition


Stock Option Levels vs. Attrition
Attrition
0
1
Count
500
400
300
200
100
0
0
1
2
3
Stock Option Level


Performance Rating vs. Attrition
Attrition
0
1
Count
1000
800
600
400
200
0
3
4
Performance Rating

In [ ]: