

```
import pandas as pd
import numpy as np
```

```
df=pd.read_csv('HR-Employee-Attrition.csv')
df
```

	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					
...	
...					
1465	36	No	Travel_Frequently	884	Research &
Development					
1466	39	No	Travel_Rarely	613	Research &
Development					
1467	27	No	Travel_Rarely	155	Research &
Development					
1468	49	No	Travel_Frequently	1023	
Sales					
1469	34	No	Travel_Rarely	628	Research &
Development					
	DistanceFromHome	Education	EducationField	EmployeeCount	\
0	1	2	Life Sciences	1	
1	8	1	Life Sciences	1	
2	2	2	Other	1	
3	3	4	Life Sciences	1	
4	2	1	Medical	1	
...	
1465	23	2	Medical	1	
1466	6	1	Medical	1	
1467	4	3	Life Sciences	1	
1468	2	3	Medical	1	
1469	8	3	Medical	1	
	EmployeeNumber	...	RelationshipSatisfaction	StandardHours	\
0	1	...	1	80	
1	2	...	4	80	
2	4	...	2	80	
3	5	...	3	80	
4	7	...	4	80	

...
1465	2061	...	3	80
1466	2062	...	1	80
1467	2064	...	2	80
1468	2065	...	4	80
1469	2068	...	1	80

	StockOptionLevel	TotalWorkingYears	TrainingTimesLastYear	\
0	0	8	0	
1	1	10	3	
2	0	7	3	
3	0	8	3	
4	1	6	3	
...	
1465	1	17	3	
1466	1	9	5	
1467	1	6	0	
1468	0	17	3	
1469	0	6	3	

	WorkLifeBalance	YearsAtCompany	YearsInCurrentRole	\
0	1	6	4	
1	3	10	7	
2	3	0	0	
3	3	8	7	
4	3	2	2	
...	
1465	3	5	2	
1466	3	7	7	
1467	3	6	2	
1468	2	9	6	
1469	4	4	3	

	YearsSinceLastPromotion	YearsWithCurrManager
0	0	5
1	1	7
2	0	0
3	3	0
4	2	2
...
1465	0	3
1466	1	7
1467	0	3
1468	0	8
1469	1	2

[1470 rows x 35 columns]

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
```

```
df.describe()
```

	Age	DailyRate	DistanceFromHome	Education
EmployeeCount \				
count	1470.000000	1470.000000	1470.000000	1470.000000
1470.0				
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
JobInvolvement \			
count	1470.000000	1470.000000	1470.000000
1470.000000			
mean	1024.865306	2.721769	65.891156
2.729932			
std	602.024335	1.093082	20.329428
0.711561			
min	1.000000	1.000000	30.000000
1.000000			
25%	491.250000	2.000000	48.000000
2.000000			
50%	1020.500000	3.000000	66.000000
3.000000			
75%	1555.750000	4.000000	83.750000
3.000000			
max	2068.000000	4.000000	100.000000
4.000000			

	JobLevel	...	RelationshipSatisfaction	StandardHours	\
count	1470.000000	...	1470.000000	1470.0	
mean	2.063946	...	2.712245	80.0	
std	1.106940	...	1.081209	0.0	
min	1.000000	...	1.000000	80.0	
25%	1.000000	...	2.000000	80.0	
50%	2.000000	...	3.000000	80.0	
75%	3.000000	...	4.000000	80.0	
max	5.000000	...	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 26 columns]

df.duplicated()

0	False
1	False
2	False
3	False
4	False

...	
1465	False
1466	False
1467	False
1468	False
1469	False

Length: 1470, dtype: bool

df.isnull()

	Age	Attrition	BusinessTravel	DailyRate	Department \
0	False	False	False	False	False
1	False	False	False	False	False
2	False	False	False	False	False
3	False	False	False	False	False
4	False	False	False	False	False
...
1465	False	False	False	False	False

1466	False	False	False	False	False
1467	False	False	False	False	False
1468	False	False	False	False	False
1469	False	False	False	False	False
	DistanceFromHome	Education	EducationField	EmployeeCount	\
0	False	False	False	False	
1	False	False	False	False	
2	False	False	False	False	
3	False	False	False	False	
4	False	False	False	False	
...	
1465	False	False	False	False	
1466	False	False	False	False	
1467	False	False	False	False	
1468	False	False	False	False	
1469	False	False	False	False	
	EmployeeNumber	...	RelationshipSatisfaction	StandardHours	\
0	False	...	False	False	
1	False	...	False	False	
2	False	...	False	False	
3	False	...	False	False	
4	False	...	False	False	
...	
1465	False	...	False	False	
1466	False	...	False	False	
1467	False	...	False	False	
1468	False	...	False	False	
1469	False	...	False	False	
	StockOptionLevel	TotalWorkingYears	TrainingTimesLastYear		\
0	False	False	False	False	
1	False	False	False	False	
2	False	False	False	False	
3	False	False	False	False	
4	False	False	False	False	
...	
1465	False	False	False	False	
1466	False	False	False	False	
1467	False	False	False	False	
1468	False	False	False	False	
1469	False	False	False	False	
	WorkLifeBalance	YearsAtCompany	YearsInCurrentRole		\
0	False	False	False	False	
1	False	False	False	False	
2	False	False	False	False	
3	False	False	False	False	
4	False	False	False	False	

...
1465	False	False	False
1466	False	False	False
1467	False	False	False
1468	False	False	False
1469	False	False	False

	YearsSinceLastPromotion	YearsWithCurrManager
0	False	False
1	False	False
2	False	False
3	False	False
4	False	False
...
1465	False	False
1466	False	False
1467	False	False
1468	False	False
1469	False	False

[1470 rows x 35 columns]

df.isnull().sum()

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
```

DATA CLEANING

1. Deleting redundant columns

```
df.drop(columns=['EmployeeNumber'], inplace=True)
```

```
df
```

	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					
...	
...					
1465	36	No	Travel_Frequently	884	Research &
Development					
1466	39	No	Travel_Rarely	613	Research &
Development					
1467	27	No	Travel_Rarely	155	Research &
Development					
1468	49	No	Travel_Frequently	1023	
Sales					
1469	34	No	Travel_Rarely	628	Research &
Development					

	DistanceFromHome	Education	EducationField	EmployeeCount	\
0	1	2	Life Sciences	1	
1	8	1	Life Sciences	1	
2	2	2	Other	1	
3	3	4	Life Sciences	1	
4	2	1	Medical	1	

...
1465	23	2	Medical	1
1466	6	1	Medical	1
1467	4	3	Life Sciences	1
1468	2	3	Medical	1
1469	8	3	Medical	1

EnvironmentSatisfaction ... RelationshipSatisfaction
StandardHours \

0	2	...	1
80			
1	3	...	4
80			
2	4	...	2
80			
3	4	...	3
80			
4	1	...	4
80			
...
...			
1465	3	...	3
80			
1466	4	...	1
80			
1467	2	...	2
80			
1468	4	...	4
80			
1469	2	...	1
80			

	StockOptionLevel	TotalWorkingYears	TrainingTimesLastYear	\
0	0	8	0	
1	1	10	3	
2	0	7	3	
3	0	8	3	
4	1	6	3	
...	
1465	1	17	3	
1466	1	9	5	
1467	1	6	0	
1468	0	17	3	
1469	0	6	3	

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

4	3	2	2
...
1465	3	5	2
1466	3	7	7
1467	3	6	2
1468	2	9	6
1469	4	4	3

	YearsSinceLastPromotion	YearsWithCurrManager
0	0	5
1	1	7
2	0	0
3	3	0
4	2	2
...
1465	0	3
1466	1	7
1467	0	3
1468	0	8
1469	1	2

[1470 rows x 34 columns]

2. Renaming the columns

```
new_column_names = {
    'Education': 'EducationLevel',
    'EducationField': 'FieldOfEducation',
    'EnvironmentSatisfaction': 'EnvSatisfaction'
}
df.rename(columns=new_column_names, inplace=True)
```

df

	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					
...	
...					
1465	36	No	Travel_Frequently	884	Research &
Development					

1466	39	No	Travel_Rarely	613	Research & Development
1467	27	No	Travel_Rarely	155	Research & Development
1468	49	No	Travel_Frequently	1023	Sales
1469	34	No	Travel_Rarely	628	Research & Development
	DistanceFromHome	EducationLevel	FieldOfEducation	EmployeeCount	\
0	1	2	Life Sciences	1	
1	8	1	Life Sciences	1	
2	2	2	Other	1	
3	3	4	Life Sciences	1	
4	2	1	Medical	1	
...	
1465	23	2	Medical	1	
1466	6	1	Medical	1	
1467	4	3	Life Sciences	1	
1468	2	3	Medical	1	
1469	8	3	Medical	1	
	EnvSatisfaction	...	RelationshipSatisfaction	StandardHours	\
0	2	...	1	80	
1	3	...	4	80	
2	4	...	2	80	
3	4	...	3	80	
4	1	...	4	80	
...	
1465	3	...	3	80	
1466	4	...	1	80	
1467	2	...	2	80	
1468	4	...	4	80	
1469	2	...	1	80	
	StockOptionLevel	TotalWorkingYears	TrainingTimesLastYear	\	
0	0	8	0		
1	1	10	3		
2	0	7	3		

3	0	8	3
4	1	6	3
...
1465	1	17	3
1466	1	9	5
1467	1	6	0
1468	0	17	3
1469	0	6	3

	WorkLifeBalance	YearsAtCompany	YearsInCurrentRole	\
0	1	6	4	
1	3	10	7	
2	3	0	0	
3	3	8	7	
4	3	2	2	
...	
1465	3	5	2	
1466	3	7	7	
1467	3	6	2	
1468	2	9	6	
1469	4	4	3	

	YearsSinceLastPromotion	YearsWithCurrManager
0	0	5
1	1	7
2	0	0
3	3	0
4	2	2
...
1465	0	3
1466	1	7
1467	0	3
1468	0	8
1469	1	2

[1470 rows x 34 columns]

3. Dropping duplicates

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

	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
...
...
1465	36	No	Travel_Frequently	884	Research & Development
1466	39	No	Travel_Rarely	613	Research & Development
1467	27	No	Travel_Rarely	155	Research & Development
1468	49	No	Travel_Frequently	1023	Sales
1469	34	No	Travel_Rarely	628	Research & Development
	DistanceFromHome	EducationLevel	FieldOfEducation	EmployeeCount	
\					
0	1	2	Life Sciences	1	
1	8	1	Life Sciences	1	
2	2	2	Other	1	
3	3	4	Life Sciences	1	
4	2	1	Medical	1	
...
1465	23	2	Medical	1	
1466	6	1	Medical	1	
1467	4	3	Life Sciences	1	
1468	2	3	Medical	1	
1469	8	3	Medical	1	
	EnvSatisfaction	...	RelationshipSatisfaction	StandardHours	\
0	2	...	1	80	
1	3	...	4	80	
2	4	...	2	80	
3	4	...	3	80	
4	1	...	4	80	
...
1465	3	...	3	80	

1466	4	...	1	80
1467	2	...	2	80
1468	4	...	4	80
1469	2	...	1	80

	StockOptionLevel	TotalWorkingYears	TrainingTimesLastYear	\
0	0	8	0	
1	1	10	3	
2	0	7	3	
3	0	8	3	
4	1	6	3	
...	
1465	1	17	3	
1466	1	9	5	
1467	1	6	0	
1468	0	17	3	
1469	0	6	3	

	WorkLifeBalance	YearsAtCompany	YearsInCurrentRole	\
0	1	6	4	
1	3	10	7	
2	3	0	0	
3	3	8	7	
4	3	2	2	
...	
1465	3	5	2	
1466	3	7	7	
1467	3	6	2	
1468	2	9	6	
1469	4	4	3	

	YearsSinceLastPromotion	YearsWithCurrManager
0	0	5
1	1	7
2	0	0
3	3	0
4	2	2
...
1465	0	3
1466	1	7
1467	0	3
1468	0	8
1469	1	2

[1470 rows x 34 columns]

4. Cleaning individual columns

```
df['JobRole'] = df['JobRole'].str.lower()
```

5. Remove NaN values from the dataset

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

	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					
...	
...					
1465	36	No	Travel_Frequently	884	Research &
Development					
1466	39	No	Travel_Rarely	613	Research &
Development					
1467	27	No	Travel_Rarely	155	Research &
Development					
1468	49	No	Travel_Frequently	1023	
Sales					
1469	34	No	Travel_Rarely	628	Research &
Development					

	DistanceFromHome	EducationLevel	FieldOfEducation	EmployeeCount
\				
0	1	2	Life Sciences	1
1	8	1	Life Sciences	1
2	2	2	Other	1
3	3	4	Life Sciences	1
4	2	1	Medical	1
...
1465	23	2	Medical	1
1466	6	1	Medical	1
1467	4	3	Life Sciences	1
1468	2	3	Medical	1

1469	8	3	Medical	1
	EnvSatisfaction	...	RelationshipSatisfaction	StandardHours \
0	2	...	1	80
1	3	...	4	80
2	4	...	2	80
3	4	...	3	80
4	1	...	4	80
...
1465	3	...	3	80
1466	4	...	1	80
1467	2	...	2	80
1468	4	...	4	80
1469	2	...	1	80
	StockOptionLevel	TotalWorkingYears	TrainingTimesLastYear	\
0	0	8	0	
1	1	10	3	
2	0	7	3	
3	0	8	3	
4	1	6	3	
...	
1465	1	17	3	
1466	1	9	5	
1467	1	6	0	
1468	0	17	3	
1469	0	6	3	
	WorkLifeBalance	YearsAtCompany	YearsInCurrentRole	\
0	1	6	4	
1	3	10	7	
2	3	0	0	
3	3	8	7	
4	3	2	2	
...	
1465	3	5	2	
1466	3	7	7	
1467	3	6	2	
1468	2	9	6	
1469	4	4	3	
	YearsSinceLastPromotion	YearsWithCurrManager		
0	0	5		
1	1	7		
2	0	0		
3	3	0		
4	2	2		
...		
1465	0	3		

1466	1	7
1467	0	3
1468	0	8
1469	1	2

[1470 rows x 34 columns]

6. Check for some more transformations

```
from sklearn.preprocessing import StandardScaler

scaler = StandardScaler()
numeric_columns = ['Age', 'TotalWorkingYears', 'MonthlyIncome']
df[numeric_columns] = scaler.fit_transform(df[numeric_columns])
```

DATA VISUALIZATION

1. Correlation map for all numeric variables:

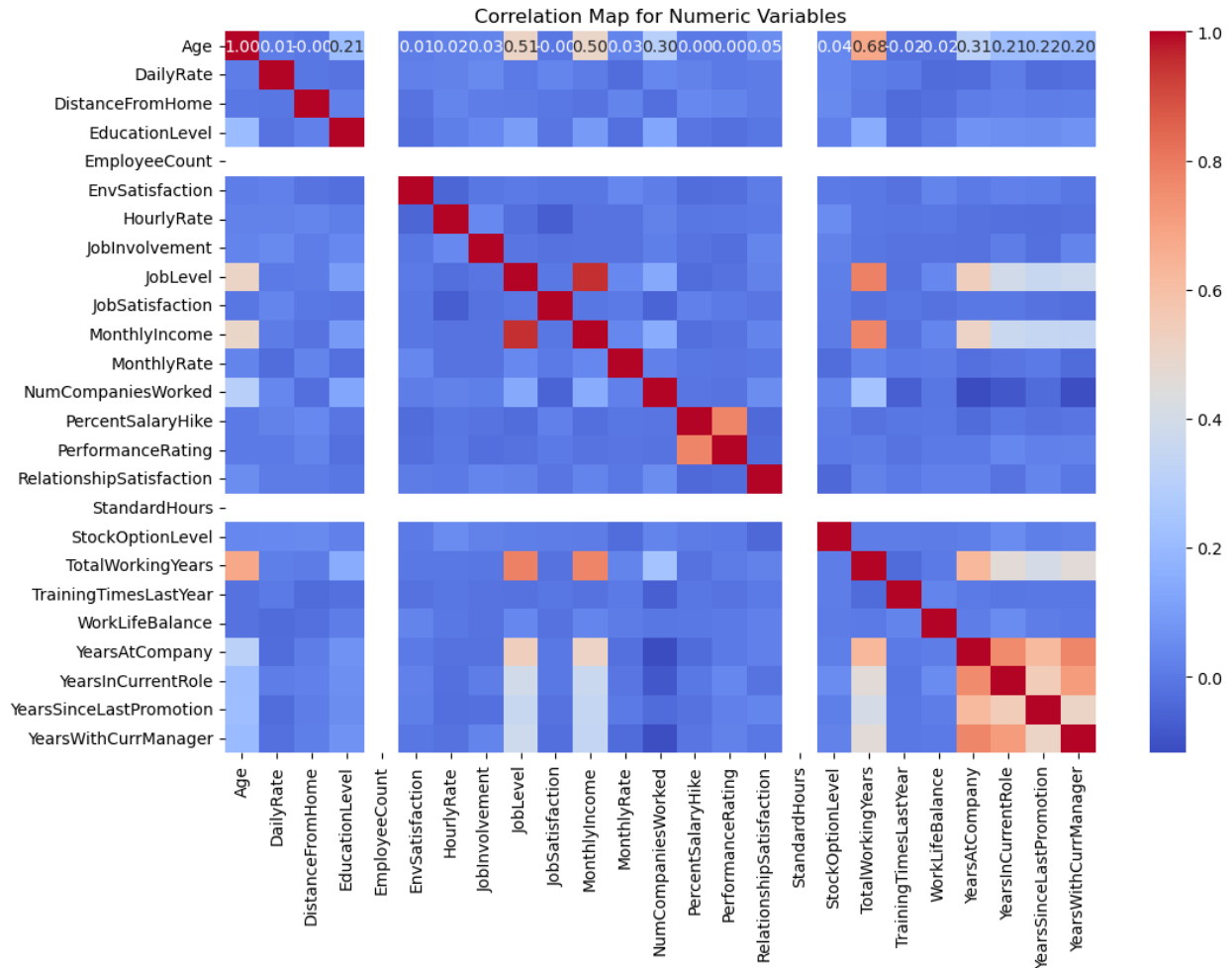
```
import seaborn as sns
import matplotlib.pyplot as plt

numeric_columns = df.select_dtypes(include=['int64', 'float64'])

correlation_matrix = numeric_columns.corr()

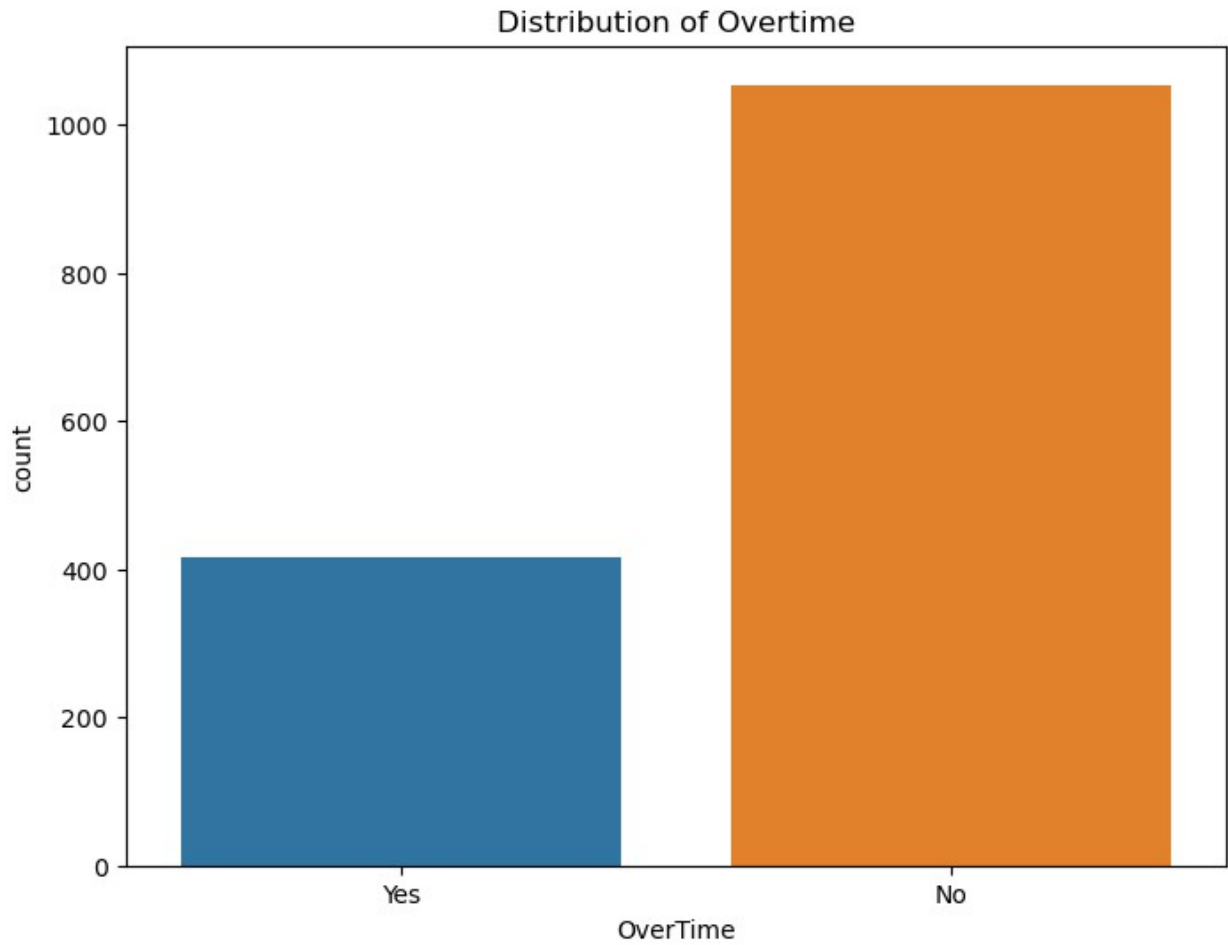
plt.figure(figsize=(12, 8))
sns.heatmap(correlation_matrix, annot=True, cmap='coolwarm',
            fmt=".2f")
plt.title('Correlation Map for Numeric Variables')
plt.show()
```

```
C:\Users\MSI\anaconda3\Lib\site-packages\seaborn\matrix.py:260:
FutureWarning: Format strings passed to MaskedConstant are ignored,
but in future may error or produce different behavior
    annotation = ("{" + self.fmt + "}").format(val)
```



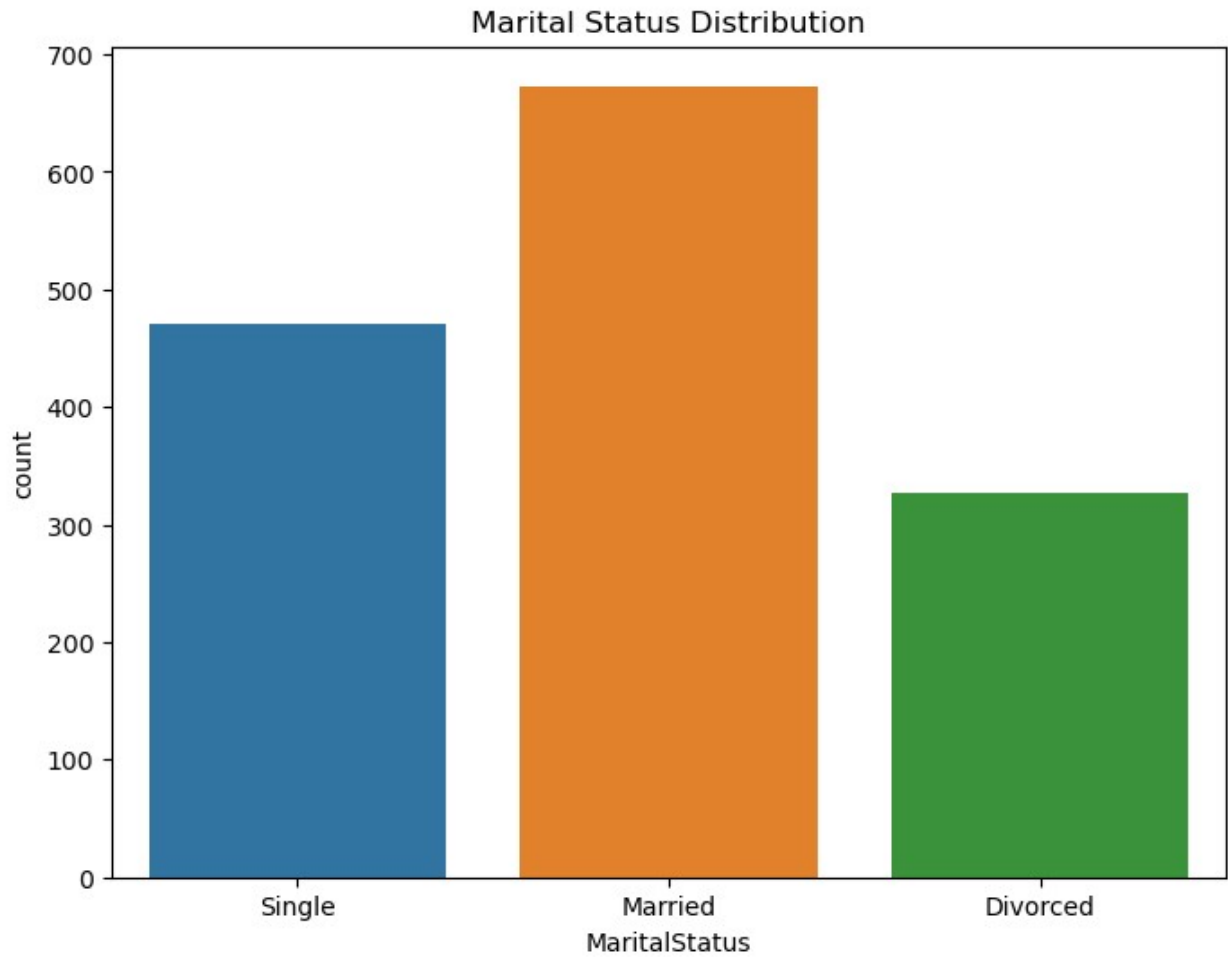
2. Overtime:

```
plt.figure(figsize=(8, 6))
sns.countplot(x='OverTime', data=df)
plt.title('Distribution of Overtime')
plt.show()
```



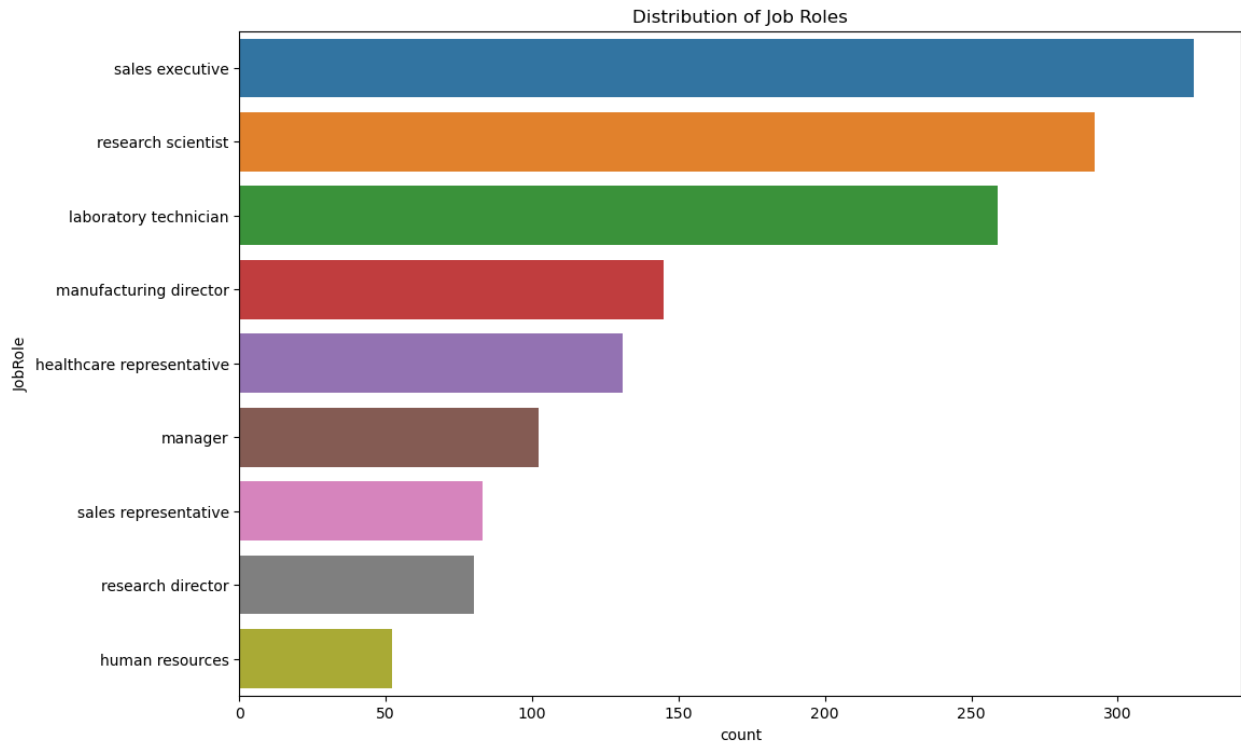
3. Marital Status:

```
plt.figure(figsize=(8, 6))  
sns.countplot(x='MaritalStatus', data=df)  
plt.title('Marital Status Distribution')  
plt.show()
```



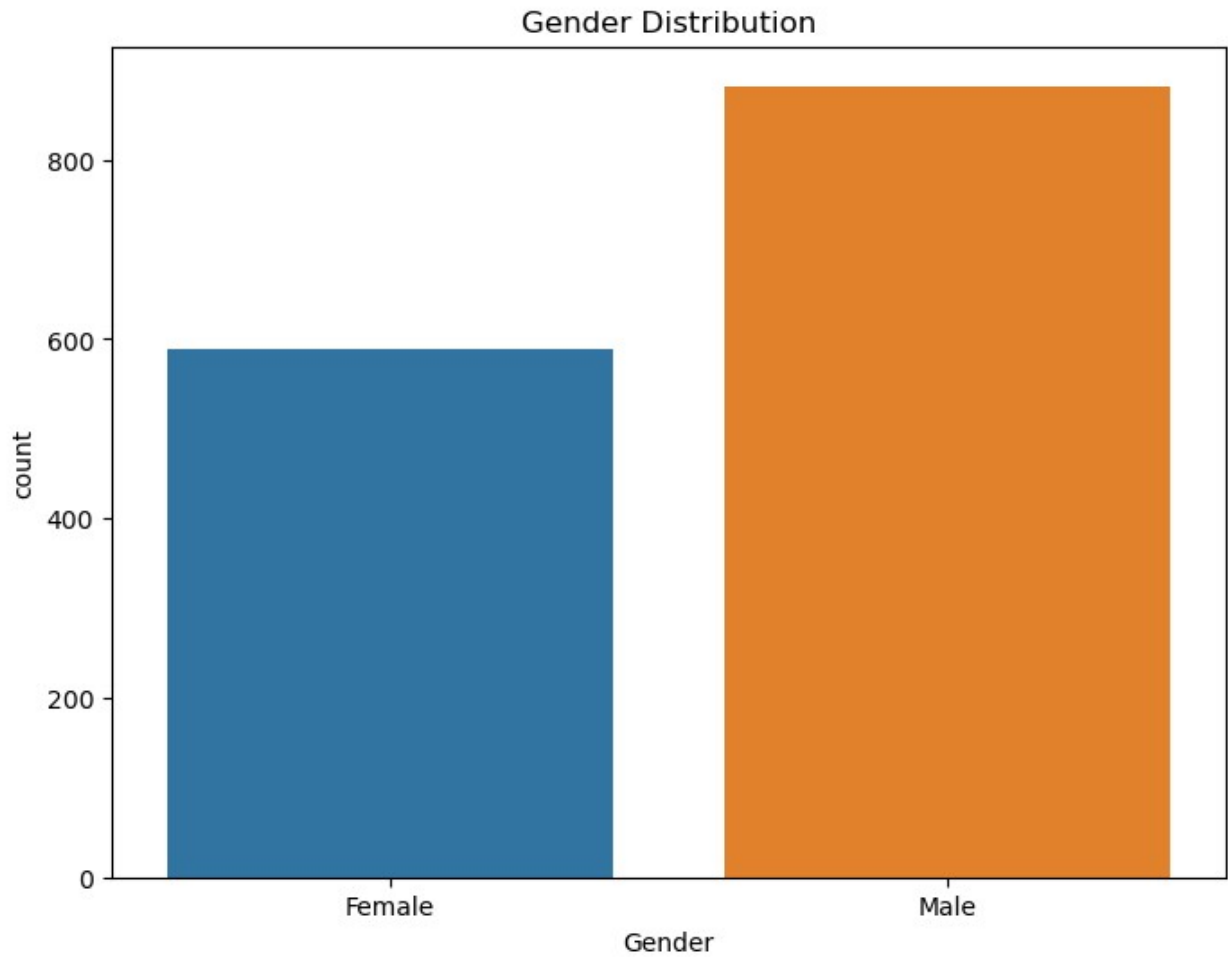
4. Job Role

```
plt.figure(figsize=(12, 8))  
sns.countplot(y='JobRole', data=df)  
plt.title('Distribution of Job Roles')  
plt.show()
```



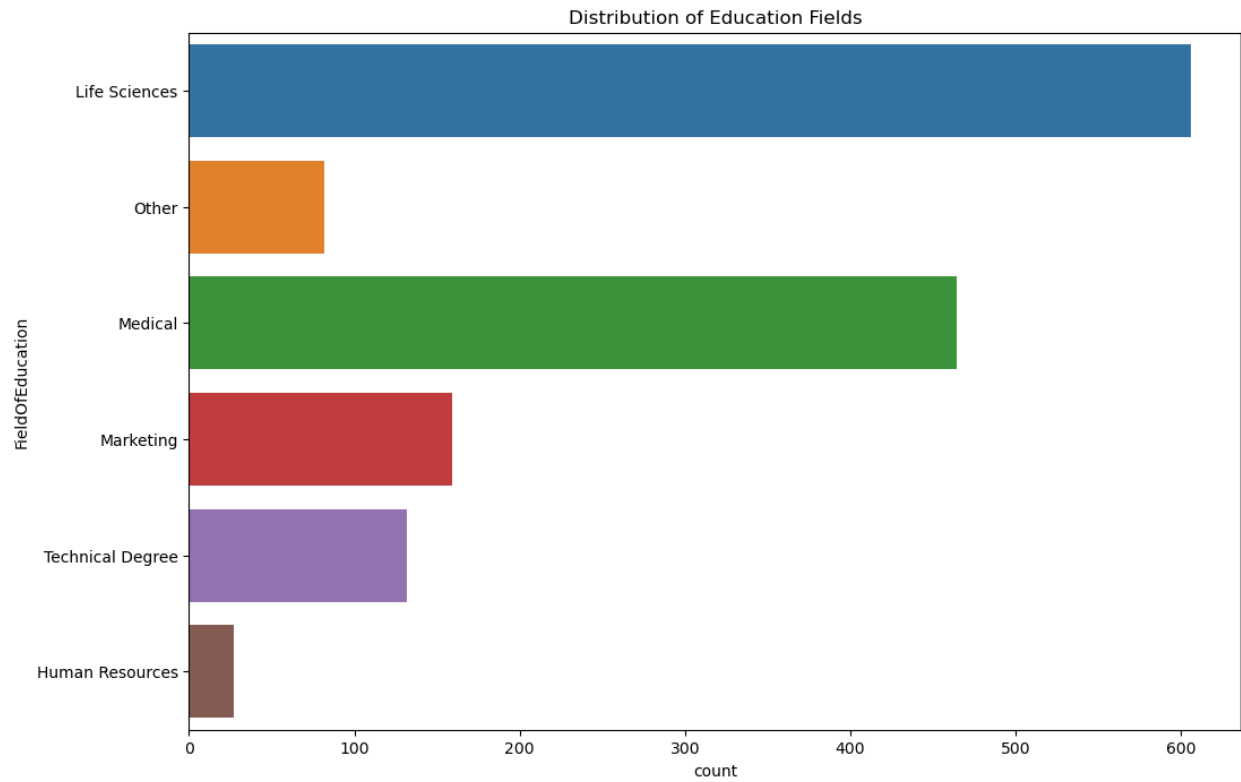
5. Gender

```
plt.figure(figsize=(8, 6))  
sns.countplot(x='Gender', data=df)  
plt.title('Gender Distribution')  
plt.show()
```



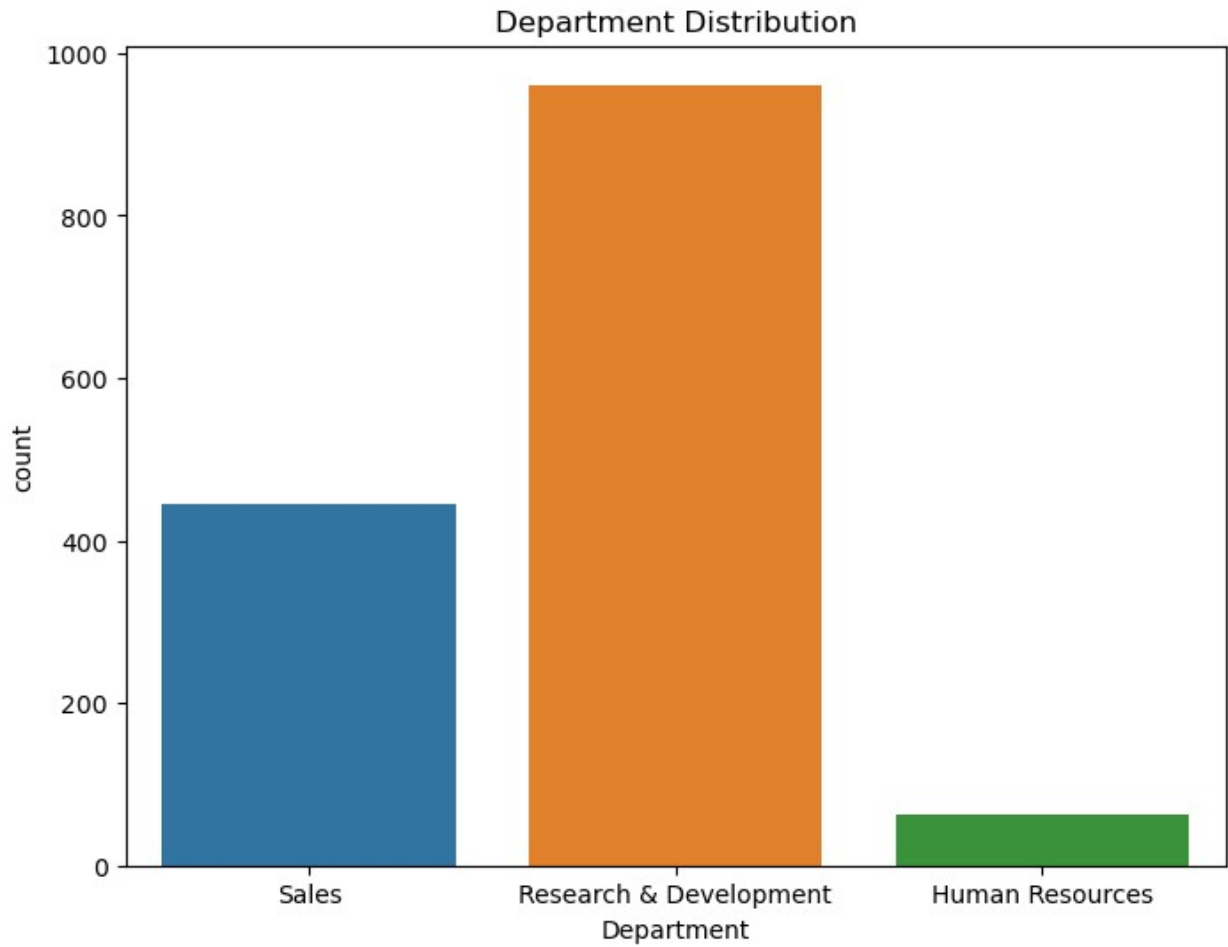
6. Education Field

```
plt.figure(figsize=(12, 8))
sns.countplot(y='FieldOfEducation', data=df)
plt.title('Distribution of Education Fields')
plt.show()
```



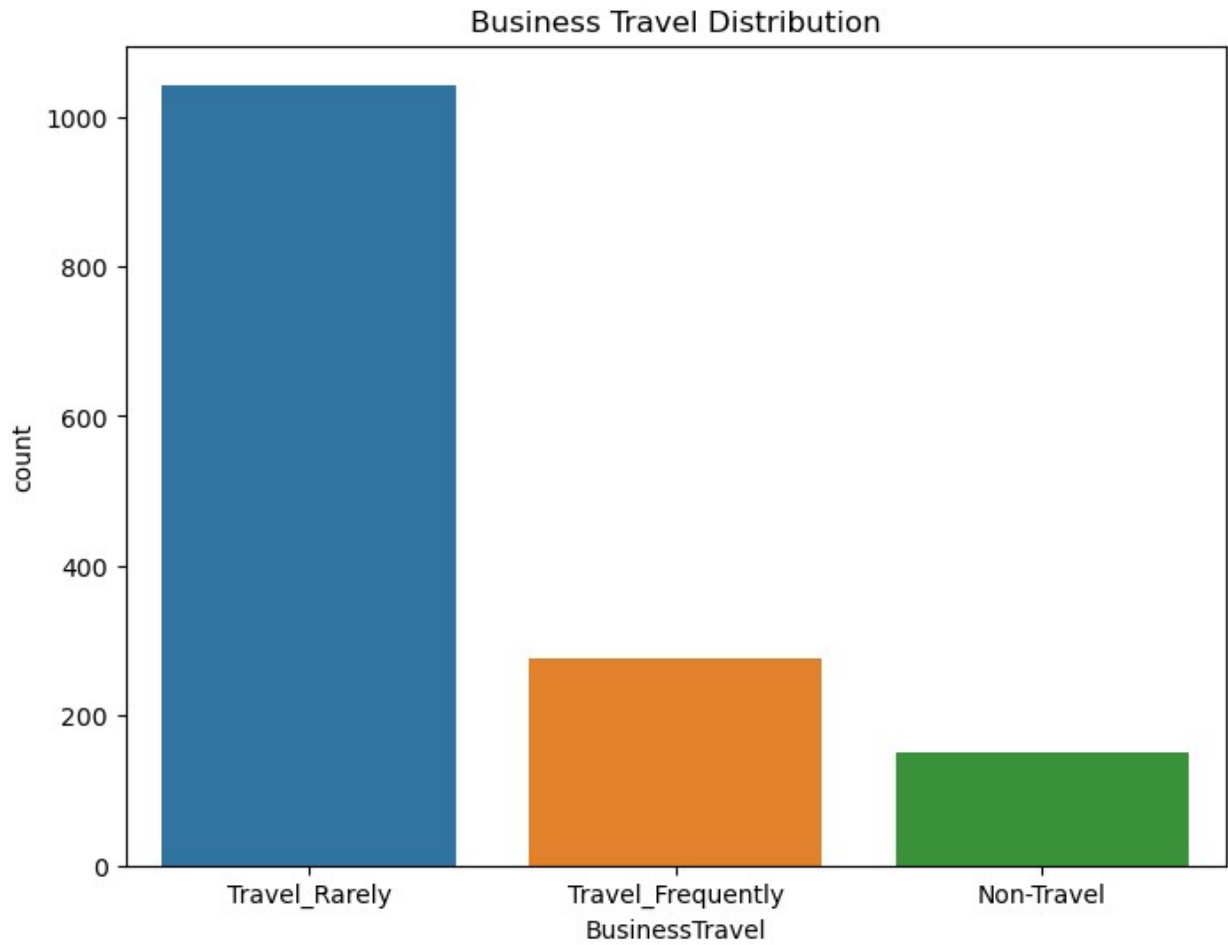
7. Department

```
plt.figure(figsize=(8, 6))  
sns.countplot(x='Department', data=df)  
plt.title('Department Distribution')  
plt.show()
```



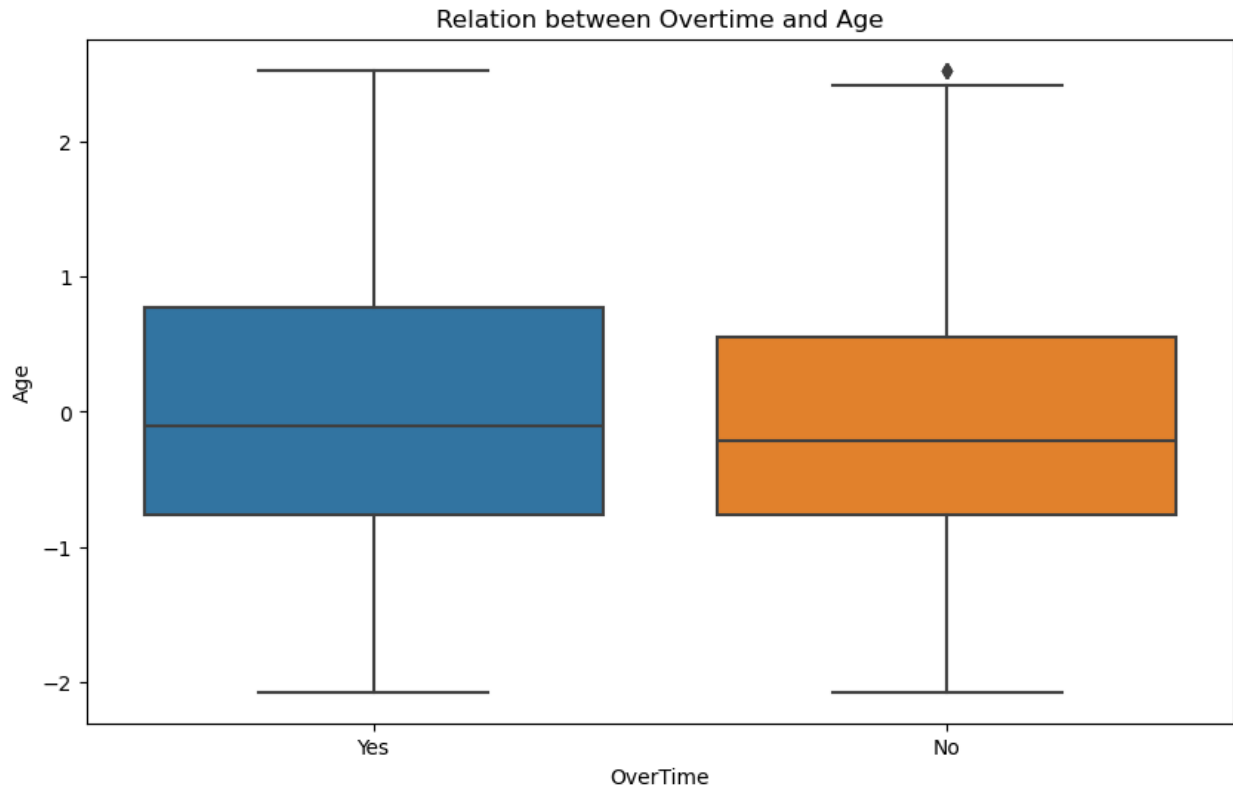
8. Business Travel

```
plt.figure(figsize=(8, 6))
sns.countplot(x='BusinessTravel', data=df)
plt.title('Business Travel Distribution')
plt.show()
```

9. Relation between Overtime and Age

```
plt.figure(figsize=(10, 6))
sns.boxplot(x='OverTime', y='Age', data=df)
plt.title('Relation between Overtime and Age')
plt.show()
```

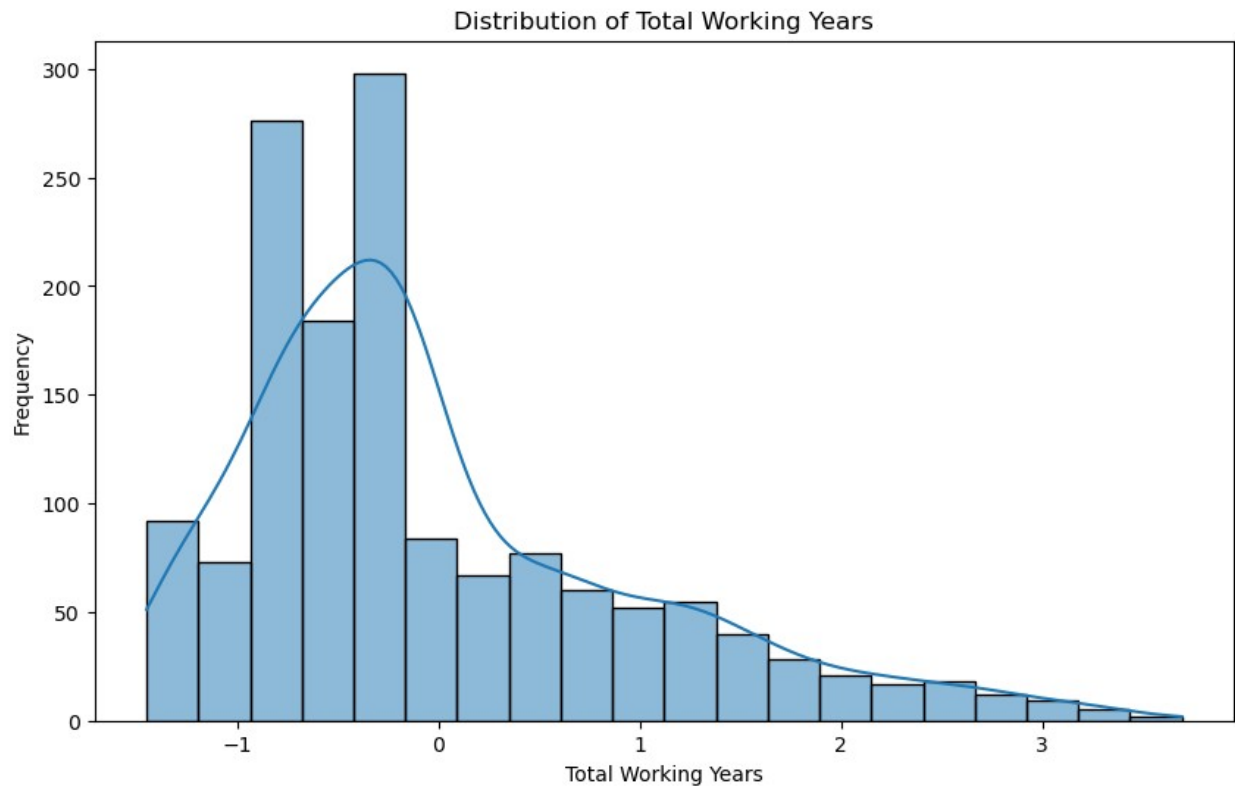


10. Total Working Years

```
plt.figure(figsize=(10, 6))
sns.histplot(df['TotalWorkingYears'], bins=20, kde=True)
plt.title('Distribution of Total Working Years')
plt.xlabel('Total Working Years')
plt.ylabel('Frequency')
plt.show()
```

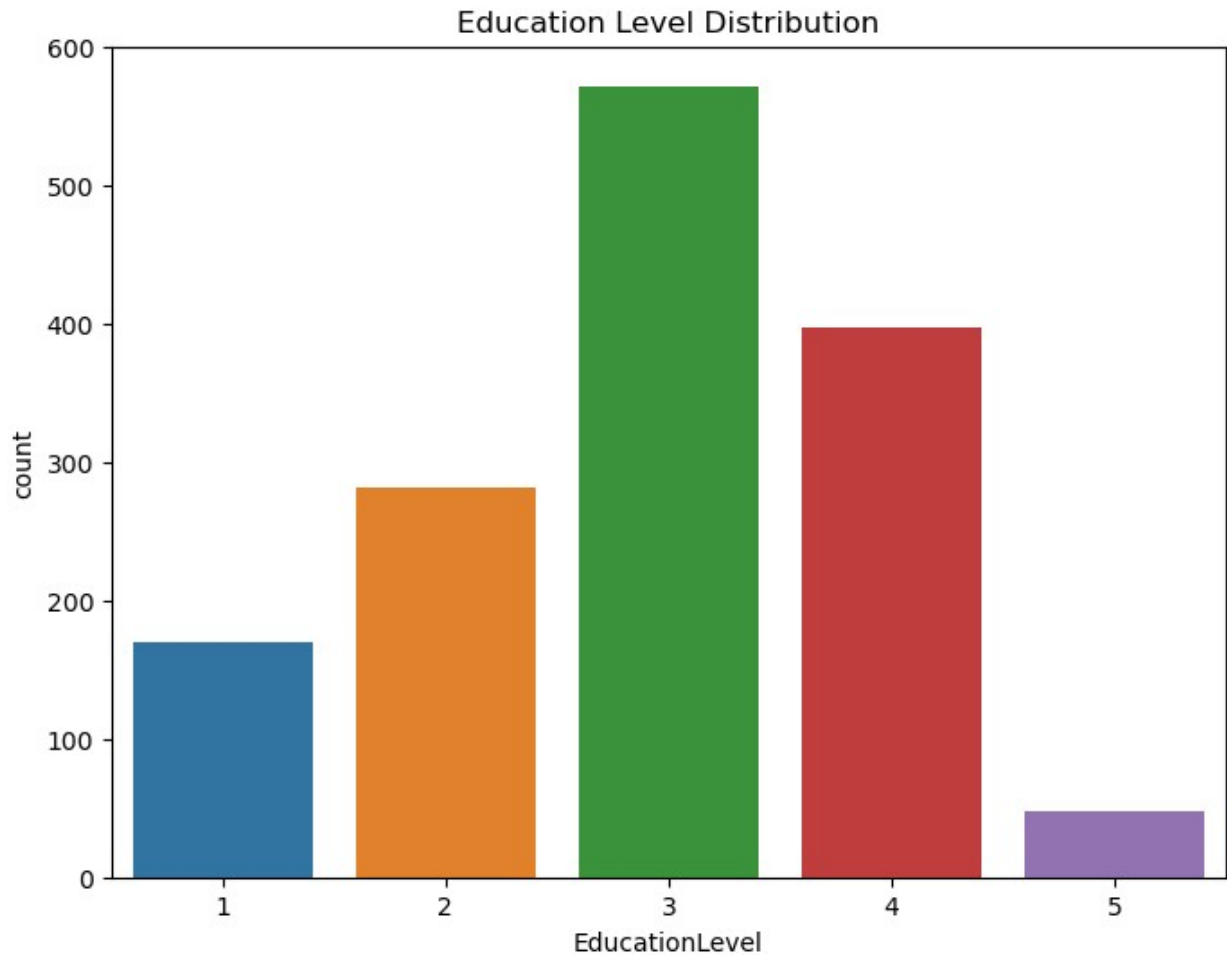
C:\Users\MSI\anaconda3\Lib\site-packages\seaborn_oldcore.py:1119: FutureWarning: use_inf_as_na option is deprecated and will be removed in a future version. Convert inf values to NaN before operating instead.

```
with pd.option_context('mode.use_inf_as_na', True):
```



11. Education Level

```
plt.figure(figsize=(8, 6))
sns.countplot(x='EducationLevel', data=df)
plt.title('Education Level Distribution')
plt.show()
```

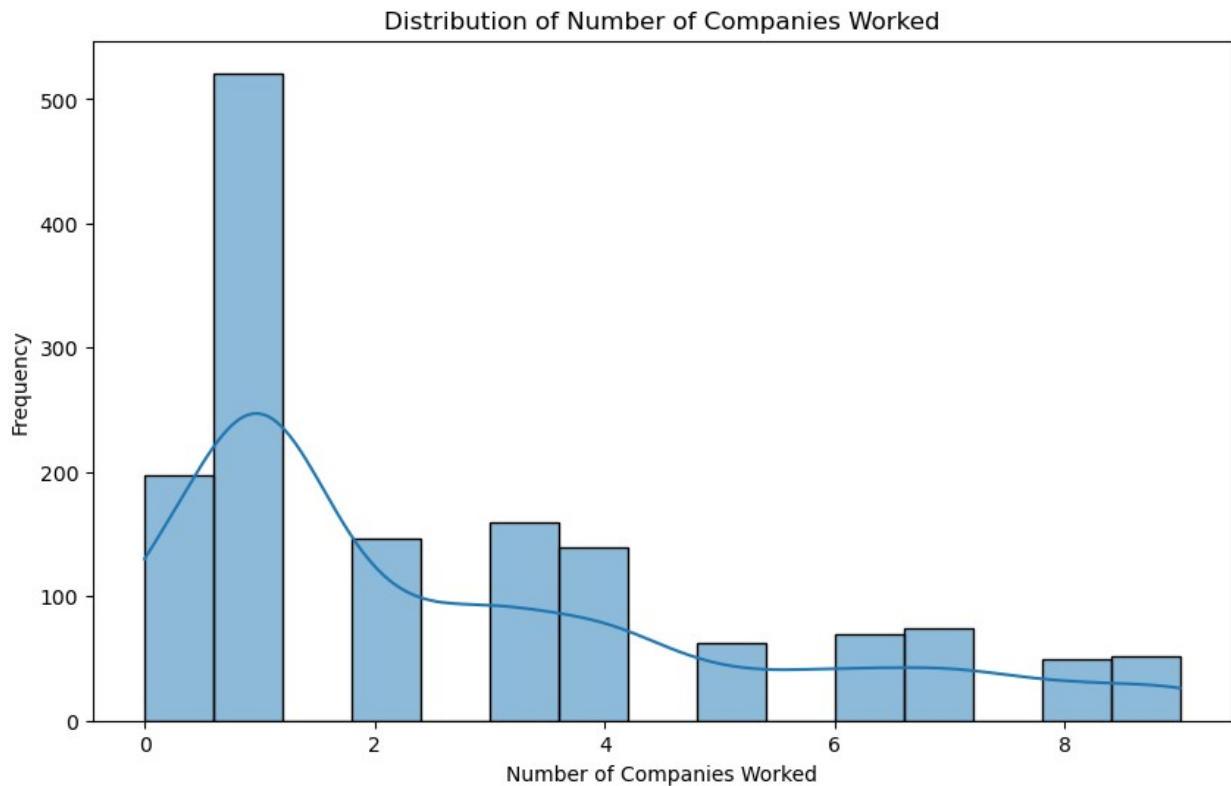


12. Number of Companies Worked

```
plt.figure(figsize=(10, 6))
sns.histplot(df['NumCompaniesWorked'], bins=15, kde=True)
plt.title('Distribution of Number of Companies Worked')
plt.xlabel('Number of Companies Worked')
plt.ylabel('Frequency')
plt.show()
```

C:\Users\MSI\anaconda3\Lib\site-packages\seaborn_oldcore.py:1119:
FutureWarning: use_inf_as_na option is deprecated and will be removed
in a future version. Convert inf values to NaN before operating
instead.

```
with pd.option_context('mode.use_inf_as_na', True):
```



13. Distance from Home

```
plt.figure(figsize=(10, 6))
sns.histplot(df['DistanceFromHome'], bins=20, kde=True)
plt.title('Distribution of Distance from Home')
plt.xlabel('Distance from Home')
plt.ylabel('Frequency')
plt.show()
```

C:\Users\MSI\anaconda3\Lib\site-packages\seaborn_oldcore.py:1119:
FutureWarning: use_inf_as_na option is deprecated and will be removed
in a future version. Convert inf values to NaN before operating
instead.

```
with pd.option_context('mode.use_inf_as_na', True):
```

