


```
import pandas as pd
data= pd.read_csv("https://raw.githubusercontent.com/Insiya24/datasets/main/WA_Fn-UseC_-HR-Employee-Attrition.csv")
```


data



	Age	Attrition	Department	DistanceFromHome	EducationField	EmployeeCount	EmployeeNumber	EnvironmentSatisfaction	Gender
0	41	Yes	Sales	1	Life Sciences	1.0	1.0	2.0	Female
1	49	No	Research & Development	8	Life Sciences	1.0	2.0	3.0	Male
2	37	Yes	Research & Development	2	Other	1.0	4.0	4.0	Male
3	33	No	Research & Development	3	Life Sciences	1.0	5.0	4.0	Female
4	27	No	Research & Development	2	Medical	NaN	7.0	1.0	Male
...
1465	36	No	Research & Development	23	Medical	1.0	2061.0	3.0	Male
1466	39	No	Research & Development	6	Medical	1.0	2062.0	4.0	Male
1467	27	No	Research & Development	4	Life Sciences	1.0	2064.0	2.0	Male
1468	49	No	Sales	2	Medical	1.0	2065.0	4.0	Male
1469	34	No	Research & Development	8	Medical	1.0	2068.0	2.0	Male


1470 rows × 25 columns

data.axes




```
[RangeIndex(start=0, stop=1470, step=1),
Index(['Age', 'Attrition', 'Department', 'DistanceFromHome', 'EducationField',
      'EmployeeCount', 'EmployeeNumber', 'EnvironmentSatisfaction', 'Gender',
      'HourlyRate', 'JobInvolvement', 'JobLevel', 'JobSatisfaction',
      'MaritalStatus', 'MonthlyIncome', 'MonthlyRate', 'OverTime',
      'PercentSalaryHike', 'PerformanceRating', 'RelationshipSatisfaction',
      'StandardHours', 'TotalWorkingYears', 'WorkLifeBalance',
      'YearsAtCompany', 'YearsSinceLastPromotion'],
      dtype='object')]
```

data.shape



```
(1470, 25)
```

data.columns



```
Index(['Age', 'Attrition', 'Department', 'DistanceFromHome', 'EducationField',
      'EmployeeCount', 'EmployeeNumber', 'EnvironmentSatisfaction', 'Gender',
      'HourlyRate', 'JobInvolvement', 'JobLevel', 'JobSatisfaction',
      'MaritalStatus', 'MonthlyIncome', 'MonthlyRate', 'OverTime',
      'PercentSalaryHike', 'PerformanceRating', 'RelationshipSatisfaction',
      'StandardHours', 'TotalWorkingYears', 'WorkLifeBalance',
      'YearsAtCompany', 'YearsSinceLastPromotion'],
      dtype='object')
```

data.ndim



```
2
```

data.size



```
36750
```

data.dtypes

```
➡ Age int64
Attrition object
Department object
DistanceFromHome int64
EducationField object
EmployeeCount float64
EmployeeNumber float64
EnvironmentSatisfaction float64
Gender object
HourlyRate float64
JobInvolvement float64
JobLevel float64
JobSatisfaction float64
MaritalStatus object
MonthlyIncome int64
MonthlyRate int64
OverTime object
PercentSalaryHike float64
PerformanceRating int64
RelationshipSatisfaction int64
StandardHours int64
TotalWorkingYears int64
WorkLifeBalance int64
YearsAtCompany int64
YearsSinceLastPromotion int64
dtype: object
```

data.isna().sum()

```
➡ Age 0
Attrition 0
Department 2
DistanceFromHome 0
EducationField 19
EmployeeCount 2
EmployeeNumber 2
EnvironmentSatisfaction 1
Gender 12
HourlyRate 3
JobInvolvement 21
JobLevel 14
JobSatisfaction 11
MaritalStatus 0
MonthlyIncome 0
MonthlyRate 0
OverTime 0
PercentSalaryHike 6
PerformanceRating 0
RelationshipSatisfaction 0
StandardHours 0
TotalWorkingYears 0
WorkLifeBalance 0
YearsAtCompany 0
YearsSinceLastPromotion 0
dtype: int64
```

data.isnull().sum()

```
➡ Age 0
Attrition 0
Department 2
DistanceFromHome 0
EducationField 19
EmployeeCount 2
EmployeeNumber 2
EnvironmentSatisfaction 1
Gender 12
HourlyRate 3
JobInvolvement 21
JobLevel 14
JobSatisfaction 11
MaritalStatus 0
MonthlyIncome 0
MonthlyRate 0
OverTime 0
PercentSalaryHike 6
PerformanceRating 0
RelationshipSatisfaction 0
StandardHours 0
TotalWorkingYears 0
WorkLifeBalance 0
YearsAtCompany 0
YearsSinceLastPromotion 0
dtype: int64
```

```
data.count()
```

Age	1470
Attrition	1470
Department	1468
DistanceFromHome	1470
EducationField	1451
EmployeeCount	1468
EmployeeNumber	1468
EnvironmentSatisfaction	1469
Gender	1458
HourlyRate	1467
JobInvolvement	1449
JobLevel	1456
JobSatisfaction	1459
MaritalStatus	1470
MonthlyIncome	1470
MonthlyRate	1470
OverTime	1470
PercentSalaryHike	1464
PerformanceRating	1470
RelationshipSatisfaction	1470
StandardHours	1470
TotalWorkingYears	1470
WorkLifeBalance	1470
YearsAtCompany	1470
YearsSinceLastPromotion	1470

dtype: int64

```
data.values
```

array([[41, 'Yes', 'Sales', ..., 1, 6, 0],
[49, 'No', 'Research & Development', ..., 3, 10, 1],
[37, 'Yes', 'Research & Development', ..., 3, 0, 0],
...,
[27, 'No', 'Research & Development', ..., 3, 6, 0],
[49, 'No', 'Sales', ..., 2, 9, 0],
[34, 'No', 'Research & Development', ..., 4, 4, 1]], dtype=object)

```
data.head(7)
```

	Age	Attrition	Department	DistanceFromHome	EducationField	EmployeeCount	Empl
0	41	Yes	Sales	1	Life Sciences	1.0	
1	49	No	Research & Development	8	Life Sciences	1.0	
2	37	Yes	Research & Development	2	Other	1.0	
3	33	No	Research & Development	3	Life Sciences	1.0	
4	27	No	Research & Development	2	Medical	NaN	
5	32	No	Research & Development	2	Life Sciences	1.0	
6	59	No	Research & Development	3	Medical	1.0	

7 rows × 25 columns

```
data.tail(5)
```

	Age	Attrition	Department	DistanceFromHome	EducationField	EmployeeCount	E
1465	36	No	Research & Development	23	Medical	1.0	
1466	39	No	Research & Development	6	Medical	1.0	
1467	27	No	Research & Development	4	Life Sciences	1.0	
1468	49	No	Sales	2	Medical	1.0	
1469	34	No	Research & Development	8	Medical	1.0	
5 rows × 25 columns							

```
data['EmployeeCount'].mean()
```

1.0

```
data['EmployeeCount'].fillna(1.0,inplace=True)
```

```
data['EmployeeNumber'].mean()
```

1026.0517711171663

```
data['EmployeeNumber'].fillna(1.1735,inplace=True)
```

```
data['EnvironmentSatisfaction'].mean()
```

2.7229407760381212

```
data['EnvironmentSatisfaction'].fillna(1.1735,inplace=True)
```

```
data['HourlyRate'].mean()
```

65.88207225630539

```
data['HourlyRate'].fillna(1.1735,inplace=True)
```

```
data['JobInvolvement'].mean()
```

2.732919254658385

```
data['JobInvolvement'].fillna(1.1735,inplace=True)
```

```
data['JobSatisfaction'].mean()
```

2.726525017135024

```
data['JobSatisfaction'].fillna(1.1735,inplace=True)
```

```
data['JobLevel'].mean()
```

2.0611263736263736

```
data['JobLevel'].fillna(1.1735,inplace=True)
```

```
data['PercentSalaryHike'].mean()
```

15.220628415300547

```
data['PercentSalaryHike'].fillna(1.1735,inplace=True)
```

```
data.isna().sum()
```

```
➡ Age                0
  Attrition           0
  Department          2
  DistanceFromHome    0
  EducationField      19
  EmployeeCount       0
  EmployeeNumber      0
  EnvironmentSatisfaction  0
  Gender             12
  HourlyRate          0
  JobInvolvement      0
  JobLevel            0
  JobSatisfaction     0
  MaritalStatus       0
  MonthlyIncome       0
  MonthlyRate         0
  OverTime            0
  PercentSalaryHike   0
  PerformanceRating   0
  RelationshipSatisfaction  0
  StandardHours       0
  TotalWorkingYears   0
  WorkLifeBalance     0
  YearsAtCompany      0
  YearsSinceLastPromotion  0
dtype: int64
```

```
data['Gender'].mode()
```

```
➡ 0    Male
   Name: Gender, dtype: object
```

```
data['Gender'].fillna("Male",inplace=True)
```

```
data['EducationField'].mode()
```

```
➡ 0    Life Sciences
   Name: EducationField, dtype: object
```

```
data['EducationField'].fillna("Life Sciences",inplace=True)
```

```
data['Department'].mode()
```

```
➡ 0    Research & Development
   Name: Department, dtype: object
```

```
data['Department'].fillna("Research & Development",inplace=True)
```

```
data.isna().sum()
```

```
➡ Age                0
  Attrition           0
  Department          0
  DistanceFromHome    0
  EducationField      0
  EmployeeCount       0
  EmployeeNumber      0
  EnvironmentSatisfaction  0
  Gender             0
  HourlyRate          0
  JobInvolvement      0
  JobLevel            0
  JobSatisfaction     0
  MaritalStatus       0
  MonthlyIncome       0
  MonthlyRate         0
  OverTime            0
  PercentSalaryHike   0
  PerformanceRating   0
  RelationshipSatisfaction  0
  StandardHours       0
  TotalWorkingYears   0
  WorkLifeBalance     0
  YearsAtCompany      0
  YearsSinceLastPromotion  0
dtype: int64
```

data



	Age	Attrition	Department	DistanceFromHome	EducationField	EmployeeCount	E
0	41	Yes	Sales	1	Life Sciences	1.0	
1	49	No	Research & Development	8	Life Sciences	1.0	
2	37	Yes	Research & Development	2	Other	1.0	
3	33	No	Research & Development	3	Life Sciences	1.0	
4	27	No	Research & Development	2	Medical	1.0	
...	
1465	36	No	Research & Development	23	Medical	1.0	
1466	39	No	Research & Development	6	Medical	1.0	
1467	27	No	Research & Development	4	Life Sciences	1.0	
1468	49	No	Sales	2	Medical	1.0	
1469	34	No	Research & Development	8	Medical	1.0	

1470 rows × 25 columns

```
from sklearn import preprocessing
```

```
le=preprocessing.LabelEncoder()
```

```
data['Gender'].value_counts()
```



```
Gender
Male      886
Female    584
Name: count, dtype: int64
```

```
data["Gender"] = data["Gender"].astype(str)
```

```
data["Gender"] = le.fit_transform(data["Gender"])
```

```
data['Gender'].value_counts()
```



```
Gender
1      886
0      584
Name: count, dtype: int64
```

```
data['EducationField'].value_counts()
```



```
EducationField
Life Sciences      619
Medical            454
Marketing           159
Technical Degree   131
Other               81
Human Resources     26
Name: count, dtype: int64
```

```
data["EducationField"] = data["EducationField"].astype(str)
```

```
data["EducationField"]=le.fit_transform(data["EducationField"])
```

```
data['EducationField'].value_counts()
```



```
EducationField
1      619
3      454
2      159
5      131
```

```
4      81
0      26
Name: count, dtype: int64
```

```
data['Department'].value_counts()
```

```
↗ Department
Research & Development    963
Sales                    444
Human Resources           63
Name: count, dtype: int64
```

```
data["Department"] = data["Department"].astype(str)
```

```
data["Department"]=le.fit_transform(data["Department"])
```

```
data['Department'].value_counts()
```

```
↗ Department
1      963
2      444
0       63
Name: count, dtype: int64
```

```
data["Attrition"] = data["Attrition"].astype(str)
```

```
data["Attrition"]=le.fit_transform(data["Attrition"])
```

```
data["MaritalStatus"] = data["MaritalStatus"].astype(str)
```

```
data["MaritalStatus"]=le.fit_transform(data["MaritalStatus"])
```

```
data["OverTime"] = data["OverTime"].astype(str)
```

```
data["OverTime"]=le.fit_transform(data["OverTime"])
```

```
data.head()
```

```
↗
```

	Age	Attrition	Department	DistanceFromHome	EducationField	EmployeeCount	Emplo
0	41	1	2	1	1	1.0	
1	49	0	1	8	1	1.0	
2	37	1	1	2	4	1.0	
3	33	0	1	3	1	1.0	
4	27	0	1	2	3	1.0	

5 rows × 25 columns

```
import seaborn
import matplotlib.pyplot as plt
plt.figure(figsize=(9,9))
correlation = data.corr ()
heatmap = seaborn.heatmap(correlation, annot = True)

plt.show()
```

