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

```
data= pd.read_csv("/content/Attrition data.csv")
```

```
data.head(10)
```

	EmployeeID	Age	Attrition	BusinessTravel	Department	DistanceFromHome	Education	EducationField	EmployeeCount	Gender	...
0	1	51	No	Travel_Rarely	Sales	6	2	Life Sciences	1	Female	...
1	2	31	Yes	Travel_Frequently	Research & Development	10	1	Life Sciences	1	Female	...
2	3	32	No	Travel_Frequently	Research & Development	17	4	Other	1	Male	...
3	4	38	No	Non-Travel	Research & Development	2	5	Life Sciences	1	Male	...
4	5	32	No	Travel_Rarely	Research & Development	10	1	Medical	1	Male	...
5	6	46	No	Travel_Rarely	Research & Development	8	3	Life Sciences	1	Female	...
6	7	28	Yes	Travel_Rarely	Research & Development	11	2	Medical	1	Male	...
7	8	29	No	Travel_Rarely	Research & Development	18	3	Life Sciences	1	Male	...
8	9	31	No	Travel_Rarely	Research & Development	1	3	Life Sciences	1	Male	...
9	10	25	No	Non-Travel	Research & Development	7	4	Medical	1	Female	...

10 rows × 29 columns

```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 4410 entries, 0 to 4409
Data columns (total 29 columns):
 #   Column                   Non-Null Count  Dtype
---  ------                   --------------  -----
 0   EmployeeID               4410 non-null   int64
 1   Age                      4410 non-null   int64
 2   Attrition                4410 non-null   object
 3   BusinessTravel           4410 non-null   object
 4   Department               4410 non-null   object
 5   DistanceFromHome         4410 non-null   int64
 6   Education                4410 non-null   int64
 7   EducationField           4410 non-null   object
 8   EmployeeCount            4410 non-null   int64
 9   Gender                   4410 non-null   object
 10  JobLevel                 4410 non-null   int64
 11  JobRole                  4410 non-null   object
 12  MaritalStatus            4410 non-null   object
 13  MonthlyIncome            4410 non-null   int64
 14  NumCompaniesWorked       4391 non-null   float64
 15  Over18                   4410 non-null   object
 16  PercentSalaryHike        4410 non-null   int64
 17  StandardHours            4410 non-null   int64
 18  StockOptionLevel         4410 non-null   int64
 19  TotalWorkingYears        4401 non-null   float64
 20  TrainingTimesLastYear    4410 non-null   int64
 21  YearsAtCompany           4410 non-null   int64
 22  YearsSinceLastPromotion  4410 non-null   int64
 23  YearsWithCurrManager     4410 non-null   int64
 24  EnvironmentSatisfaction  4385 non-null   float64
 25  JobSatisfaction          4390 non-null   float64
 26  WorkLifeBalance          4372 non-null   float64
 27  JobInvolvement           4410 non-null   int64
 28  PerformanceRating        4410 non-null   int64
dtypes: float64(5), int64(16), object(8)
memory usage: 999.3+ KB
```

```
data.isnull().sum()
```

	0
EmployeeID	0
Age	0
Attrition	0
BusinessTravel	0
Department	0
DistanceFromHome	0
Education	0
EducationField	0
EmployeeCount	0
Gender	0
JobLevel	0
JobRole	0
MaritalStatus	0
MonthlyIncome	0
NumCompaniesWorked	19
Over18	0
PercentSalaryHike	0
StandardHours	0
StockOptionLevel	0
TotalWorkingYears	9
TrainingTimesLastYear	0
YearsAtCompany	0
YearsSinceLastPromotion	0
YearsWithCurrManager	0
EnvironmentSatisfaction	25
JobSatisfaction	20
WorkLifeBalance	38
JobInvolvement	0
PerformanceRating	0

dtype: int64

```
data['NumCompaniesWorked'].fillna(data['NumCompaniesWorked'].median(), inplace=True)
data['TotalWorkingYears'].fillna(data['TotalWorkingYears'].median(), inplace=True)
data['EnvironmentSatisfaction'].fillna(data['EnvironmentSatisfaction'].median(), inplace=True)
data['JobSatisfaction'].fillna(data['JobSatisfaction'].median(), inplace=True)
data['WorkLifeBalance'].fillna(data['WorkLifeBalance'].median(), inplace=True)
```

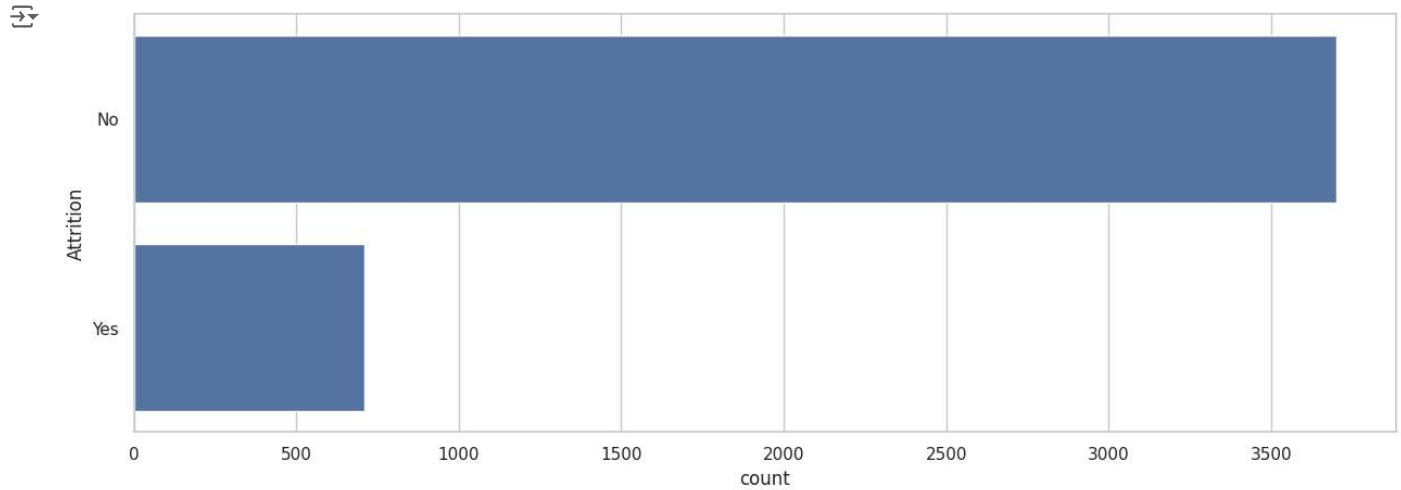
```
# Display the summary statistics
print(data.describe())
```

```
        EmployeeID          Age  DistanceFromHome    Education  EmployeeCount  \
count  4410.000000  4410.000000       4410.000000  4410.000000         4410.0
mean   2205.500000    36.923810          9.192517     2.912925            1.0
std    1273.201673     9.133301          8.105026     1.023933            0.0
min       1.000000    18.000000          1.000000     1.000000            1.0
25%    1103.250000    30.000000          2.000000     2.000000            1.0
50%    2205.500000    36.000000          7.000000     3.000000            1.0
75%    3307.750000    43.000000         14.000000     4.000000            1.0
max    4410.000000    60.000000         29.000000     5.000000            1.0

          JobLevel  MonthlyIncome  NumCompaniesWorked  PercentSalaryHike  \
count  4410.000000    4410.000000         4410.000000        4410.000000
mean      2.063946   65029.312925            2.691837          15.209524
std       1.106689   47068.888559            2.493912           3.659108
min       1.000000   10090.000000            0.000000          11.000000
25%       1.000000   29110.000000            1.000000          12.000000
50%       2.000000   49190.000000            2.000000          14.000000
75%       3.000000   83800.000000            4.000000          18.000000
max       5.000000  199990.000000            9.000000          25.000000
```

```
       StandardHours  ...  TotalWorkingYears  TrainingTimesLastYear  \
count         4410.0  ...        4410.000000            4410.000000
mean             8.0  ...          11.277324               2.799320
std              0.0  ...           7.774490               1.288978
min              8.0  ...           0.000000               0.000000
25%              8.0  ...           6.000000               2.000000
50%              8.0  ...          10.000000               3.000000
75%              8.0  ...          15.000000               3.000000
max              8.0  ...          40.000000               6.000000

       YearsAtCompany  YearsSinceLastPromotion  YearsWithCurrManager  \
count     4410.000000              4410.000000           4410.000000
mean         7.008163                 2.187755              4.123129
std          6.125135                 3.221699              3.567327
min          0.000000                 0.000000              0.000000
25%          3.000000                 0.000000              2.000000
50%          5.000000                 1.000000              3.000000
75%          9.000000                 3.000000              7.000000
max         40.000000                15.000000             17.000000

       EnvironmentSatisfaction  JobSatisfaction  WorkLifeBalance  \
count              4410.000000      4410.000000      4410.000000
mean                  2.725170         2.729478         2.763492
std                   1.089852         1.098904         0.703541
min                   1.000000         1.000000         1.000000
25%                   2.000000         2.000000         2.000000
50%                   3.000000         3.000000         3.000000
75%                   4.000000         4.000000         3.000000
max                   4.000000         4.000000         4.000000

       JobInvolvement  PerformanceRating
count     4410.000000        4410.000000
mean         2.729932           3.153741
std          0.711400           0.360742
min          1.000000           3.000000
25%          2.000000           3.000000
50%          3.000000           3.000000
75%          3.000000           3.000000
```

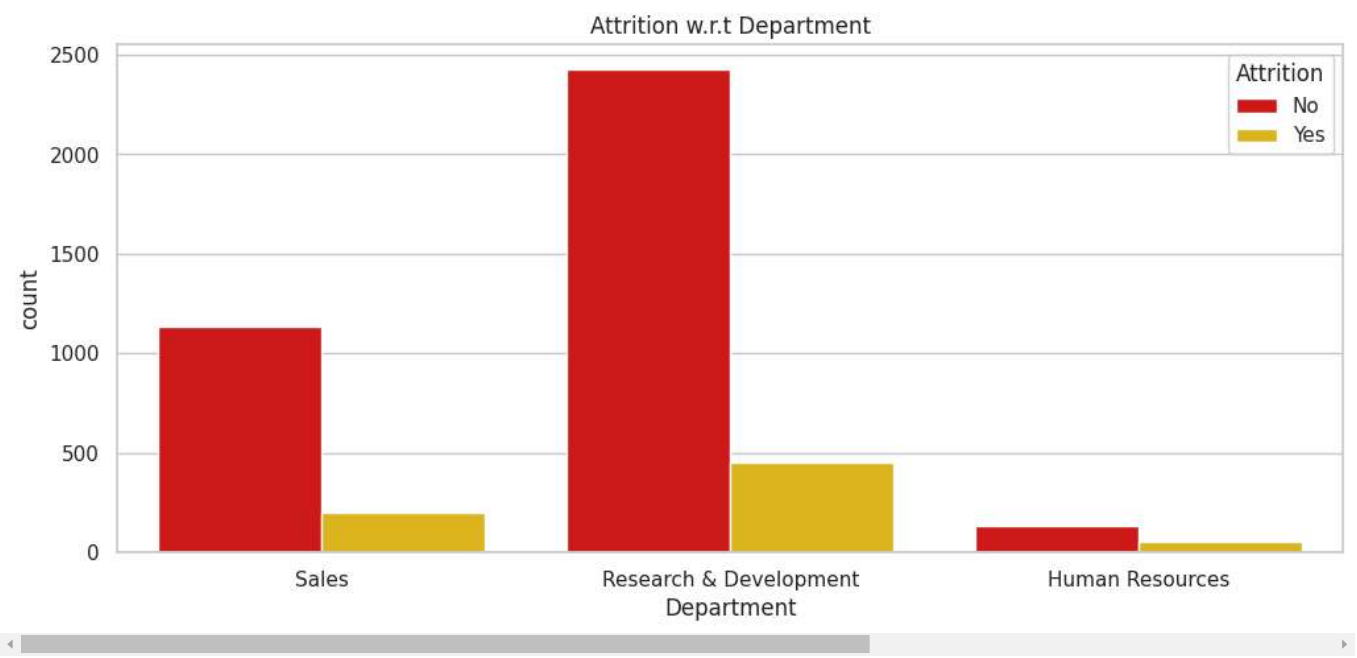
*Target Variable *

```
plt.figure(figsize=(15,5))
plt.rc("font",size=14)
sns.countplot(y='Attrition',data=data)
plt.show()
```

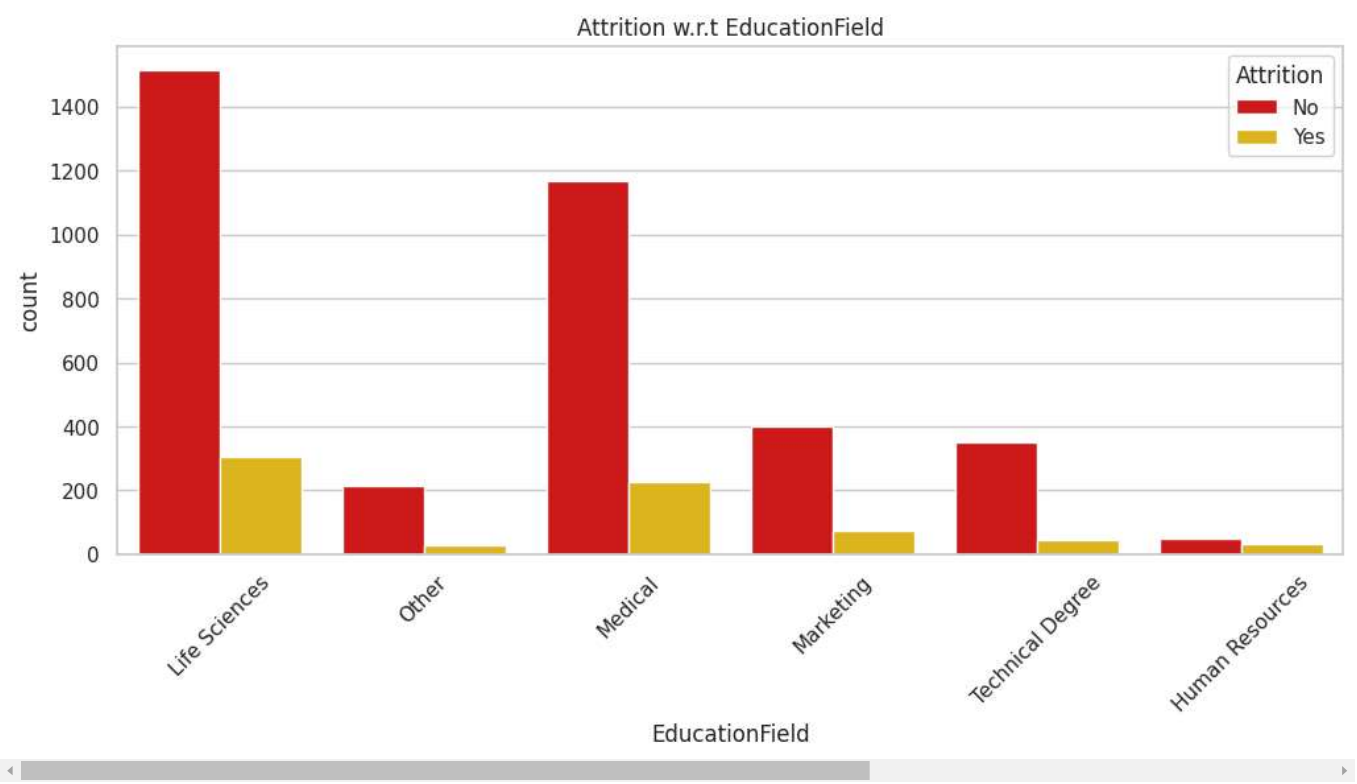


Exploratory Data Analysis

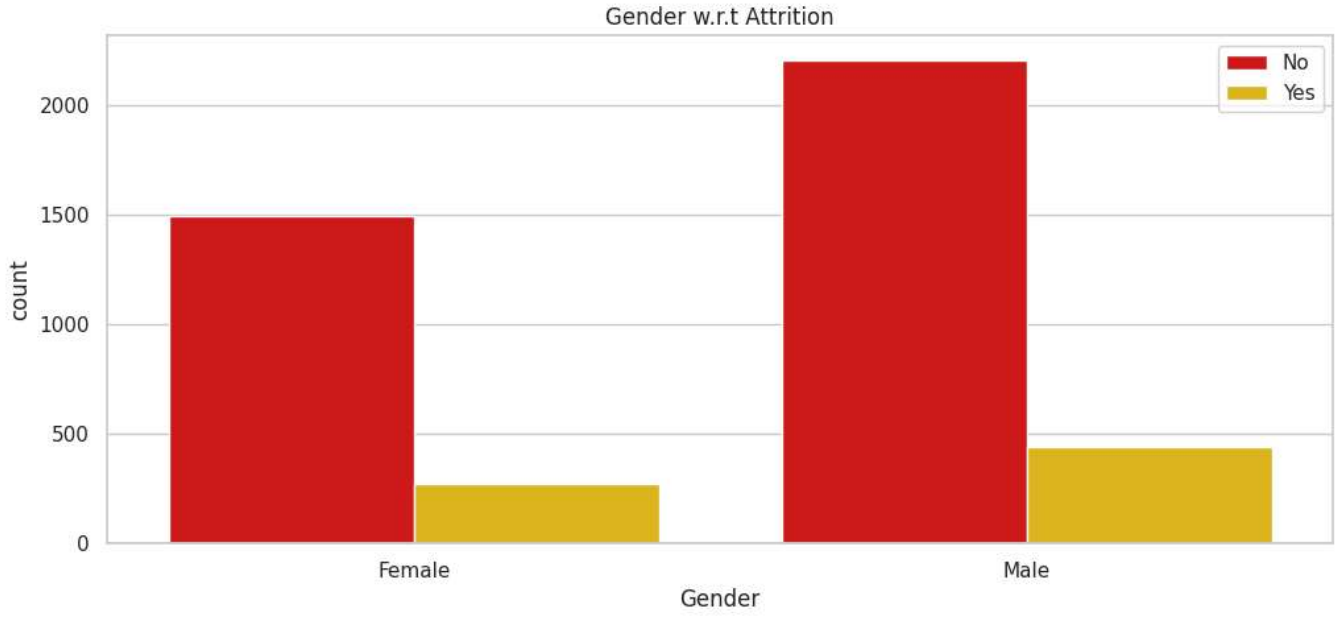
```
# Department wrt Attrition
plt.figure(figsize=(12,5))
sns.countplot(x='Department', hue='Attrition', data=data, palette='hot')
plt.title("Attrition w.r.t Department")
plt.show()
```



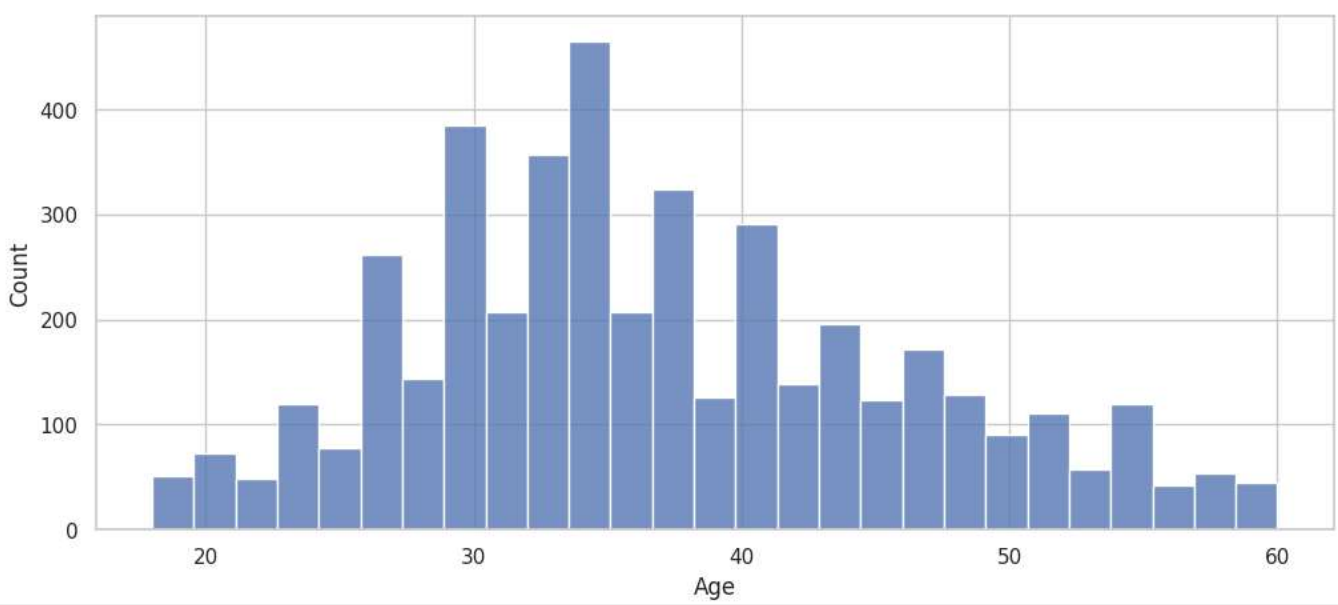

```
# Education wrt Attrition
plt.figure(figsize=(12,5))
sns.countplot(x='EducationField',hue='Attrition',data=data, palette='hot')
plt.title("Attrition w.r.t EducationField")
plt.xticks(rotation=45)
plt.show()
```




```
# most male of female employes Attrition
# department wrt Attrition
plt.figure(figsize=(12,5))
sns.countplot(x='Gender',hue='Attrition',data=data, palette='hot')
plt.title('Gender w.r.t Attrition')
plt.legend(loc='best')
plt.show()
```



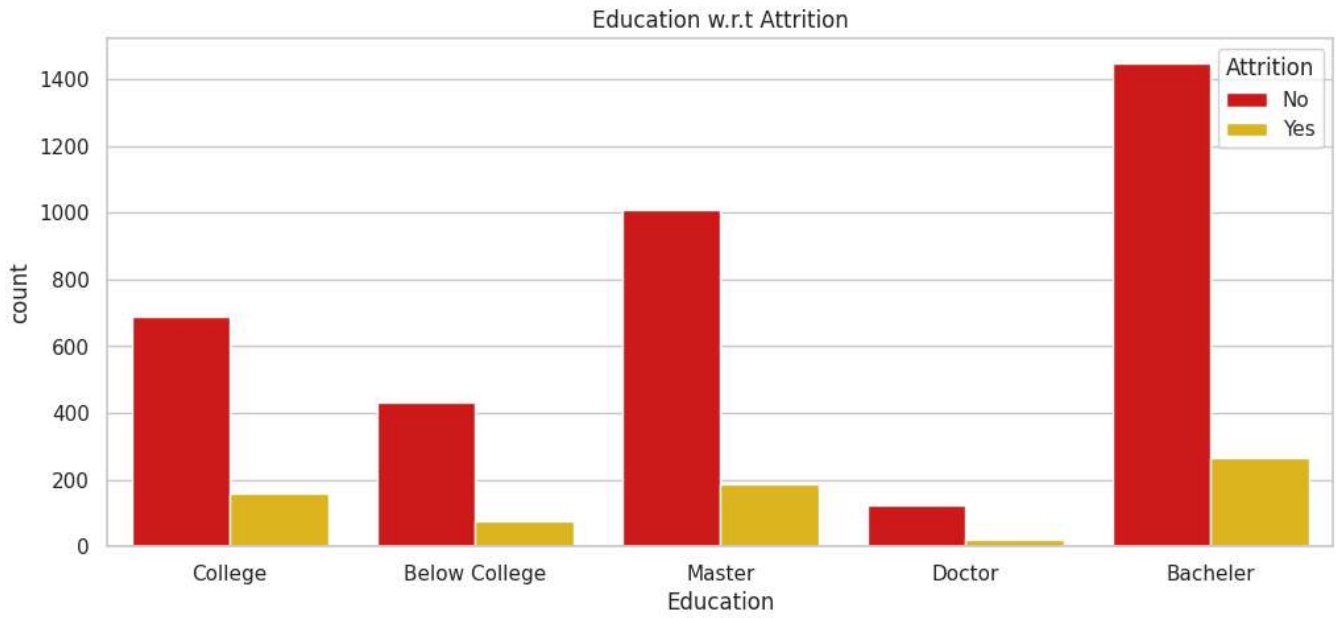

```
# Distribution of age
plt.figure(figsize=(12,5))
sns.histplot(data['Age'])
plt.show()
```




```
ordinal_features = ['Education','EnvironmentSatisfaction','JobInvolvement','JobSatisfaction','PerformanceRating','WorkLifeBalance']
data[ordinal_features].head()
```

	Education	EnvironmentSatisfaction	JobInvolvement	JobSatisfaction	PerformanceRating	WorkLifeBalance
0	2	3.0	3	4.0	3	2.0
1	1	3.0	2	2.0	4	4.0
2	4	2.0	3	2.0	3	1.0
3	5	4.0	2	4.0	3	3.0
4	1	4.0	3	1.0	3	3.0

```
edu_map={1:'Below College',2:'College',3:'Bacheler',4:'Master',5:'Doctor'}
plt.figure(figsize=(12,5))
sns.countplot(x=data['Education'].map(edu_map),hue='Attrition',data=data, palette='hot')
plt.title("Education w.r.t Attrition")
plt.show()
```



Label Encoding

```
#Target variable(Attrition)
data['Attrition']=data['Attrition'].replace({'No':0,'Yes':1})


#Encode binary vaiable
data['Over18']=data['Over18'].map({'No':0,'Y':1})
data['Gender']=data['Gender'].map({'Male':0,'Female':1})

data['Over18']
```

	Over18
0	1
1	1
2	1
3	1
4	1
...	...
4405	1
4406	1
4407	1
4408	1
4409	1

4410 rows × 1 columns

dtype: int64

```
# Encode categorical column which are ordinal, use labelencoding
#apply Label encoder to df_categorical
from sklearn.preprocessing import LabelEncoder
encoding_cols=['BusinessTravel','Department','EducationField','JobRole','MaritalStatus']
label_encoders={}
for column in encoding_cols:
    label_encoders[column]=LabelEncoder()
    data[column]=label_encoders[column].fit_transform(data[column])


data.head()# look at the final data
```

	EmployeeID	Age	Attrition	BusinessTravel	Department	DistanceFromHome	Education	EducationField	EmployeeCount	Gender	...	1
0	1	51	0	2	2	6	2	1	1	1	...	
1	2	31	1	1	1	10	1	1	1	1	...	
2	3	32	0	1	1	17	4	4	1	0	...	
3	4	38	0	0	1	2	5	1	1	0	...	
4	5	32	0	2	1	10	1	3	1	0	...	

5 rows × 30 columns

```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 4410 entries, 0 to 4409
Data columns (total 30 columns):
 #   Column                   Non-Null Count  Dtype  
---  ------                   --------------  -----  
 0   EmployeeID               4410 non-null   int64  
 1   Age                      4410 non-null   int64  
 2   Attrition                4410 non-null   int64  
 3   BusinessTravel           4410 non-null   int64  
 4   Department               4410 non-null   int64  
 5   DistanceFromHome         4410 non-null   int64  
 6   Education                4410 non-null   int64  
 7   EducationField           4410 non-null   int64  
 8   EmployeeCount            4410 non-null   int64  
 9   Gender                   4410 non-null   int64  
 10  JobLevel                 4410 non-null   int64  
 11  JobRole                  4410 non-null   int64  
 12  MaritalStatus            4410 non-null   int64  
 13  MonthlyIncome            4410 non-null   int64  
 14  NumCompaniesWorked       4410 non-null   float64
 15  Over18                   4410 non-null   int64  
 16  PercentSalaryHike        4410 non-null   int64  
 17  StandardHours            4410 non-null   int64  
 18  StockOptionLevel         4410 non-null   int64  
 19  TotalWorkingYears        4410 non-null   float64
 20  TrainingTimesLastYear    4410 non-null   int64  
 21  YearsAtCompany           4410 non-null   int64  
 22  YearsSinceLastPromotion  4410 non-null   int64  
 23  YearsWithCurrManager     4410 non-null   int64  
 24  EnvironmentSatisfaction  4410 non-null   float64
 25  JobSatisfaction          4410 non-null   float64
 26  WorkLifeBalance          4410 non-null   float64
 27  JobInvolvement           4410 non-null   int64  
 28  PerformanceRating        4410 non-null   int64  
 29  Attrition_Numeric        4410 non-null   int64  
dtypes: float64(5), int64(25)
memory usage: 1.0 MB
```