

Employee Data Analysis

Data Source: <https://www.kaggle.com/datasets/tawfikelmetwally/employee-dataset>

Importing primary libraries

```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
import warnings
warnings.filterwarnings('ignore')
```

(1) Data loading

```
emp=pd.read_csv('/content/Employee.csv')
```

```
# first 3 rows
emp.head(3)
```

	Education	JoiningYear	City	PaymentTier	Age	Gender	EverBenched	ExperienceInCurrentDomain	LeaveOrNot
0	Bachelors	2017	Bangalore	3	34	Male	No	0	0
1	Bachelors	2013	Pune	1	28	Female	No	3	1
2	Bachelors	2014	New Delhi	3	38	Female	No	2	0

```
# last 3 rows
emp.tail(3)
```

	Education	JoiningYear	City	PaymentTier	Age	Gender	EverBenched	ExperienceInCurrentDomain	LeaveOrNot
4650	Masters	2018	New Delhi	3	27	Male	No	5	1
4651	Bachelors	2012	Bangalore	3	30	Male	Yes	2	0
4652	Bachelors	2015	Bangalore	3	33	Male	Yes	4	0

(2) Exploring the dataset

Dimensions of the dataset

```
emp.shape

(4653, 9)
```

Checking the data types for each column

```
emp.dtypes

Education          object
JoiningYear        int64
City               object
PaymentTier        int64
Age                int64
Gender             object
EverBenched        object
ExperienceInCurrentDomain  int64
LeaveOrNot          int64
dtype: object
```

Plotting different variables

a. Checking for no.of distinct values in each column in the dataset

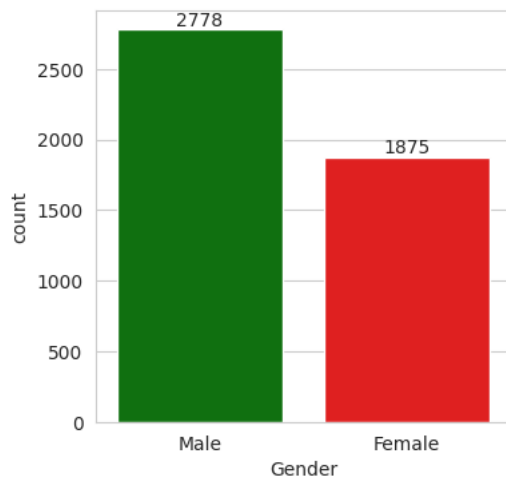
```
emp.nunique()

Education          3
JoiningYear        7
City               3
```

```
PaymentTier      3
Age              20
Gender           2
EverBenched      2
ExperienceInCurrentDomain  8
LeaveOrNot        2
dtype: int64
```

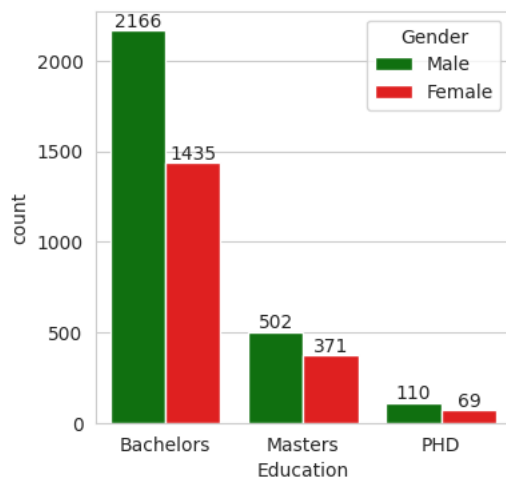
b. Number of employees gender wise

```
plt.figure(figsize=(4,4))
sns.set_style('whitegrid')
col={'Male':'green','Female':'red'}
e=sns.countplot(data=emp,x='Gender',palette=col)
for i in e.containers:
    e.bar_label(i)
```



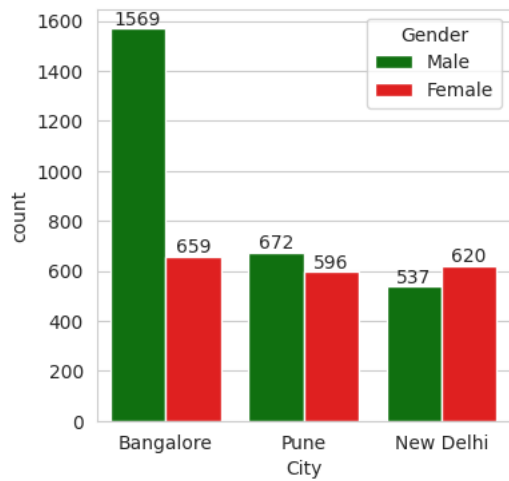
c. Exploring Education level of employees

```
plt.figure(figsize=(4,4))
col={'Male':'green','Female':'red'}
m=sns.countplot(data=emp,x='Education',hue='Gender',palette=col)
for i in m.containers:
    m.bar_label(i)
```



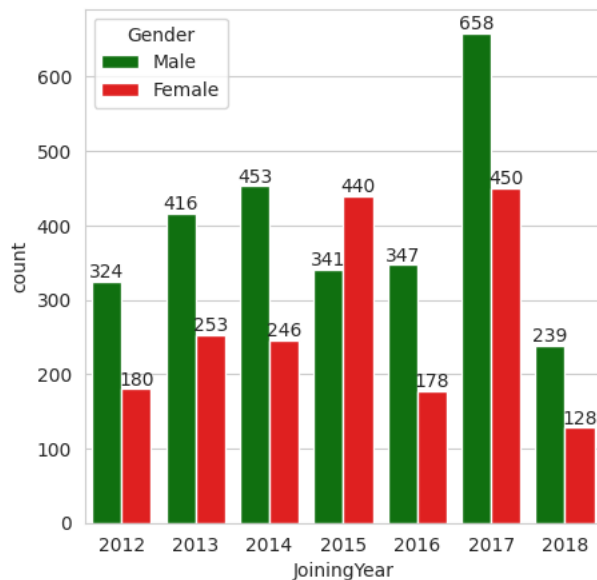
d. Employees from different cities

```
plt.figure(figsize=(4,4))
col={'Male':'green','Female':'red'}
s=sns.countplot(data=emp,x='City',hue='Gender',palette=col)
for i in s.containers:
    s.bar_label(i)
```



e. Year of joining

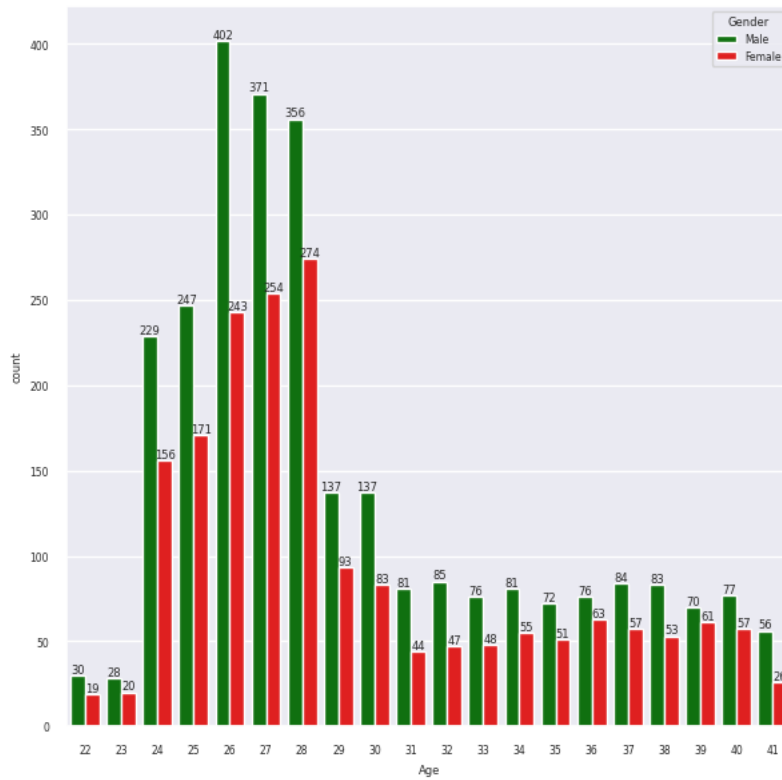
```
plt.figure(figsize=(5,5))
col={'Male':'green','Female':'red'}
s=sns.countplot(data=emp,x='JoiningYear',hue='Gender',palette=col)
for i in s.containers:
    s.bar_label(i)
```



More employees joined in the year 2017

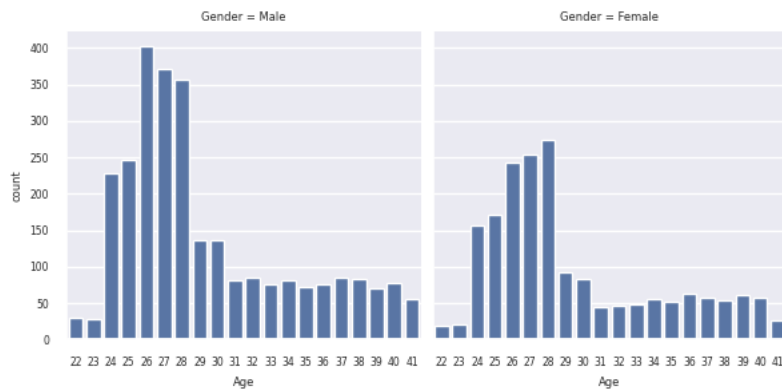
f. Age of employees

```
sns.set(font_scale=0.5)
plt.figure(figsize=(7,7))
col={'Male':'green','Female':'red'}
f=sns.countplot(data=emp,x='Age',hue='Gender',palette=col)
for i in f.containers:
    f.bar_label(i)
```



```
plt.figure(figsize=(7,7))
d=sns.FacetGrid(emp,col='Gender')
d.map(sns.countplot,'Age')
```

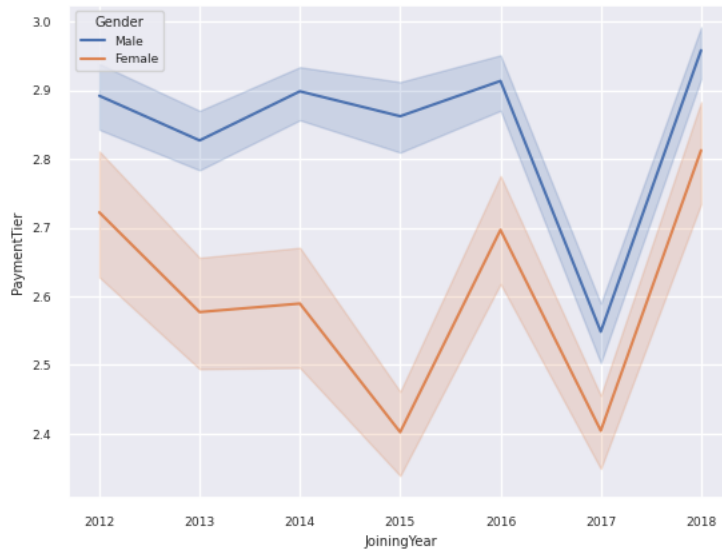
```
<seaborn.axisgrid.FacetGrid at 0x7e632ac57dc0>
<Figure size 700x700 with 0 Axes>
```



g. Payment and joining year

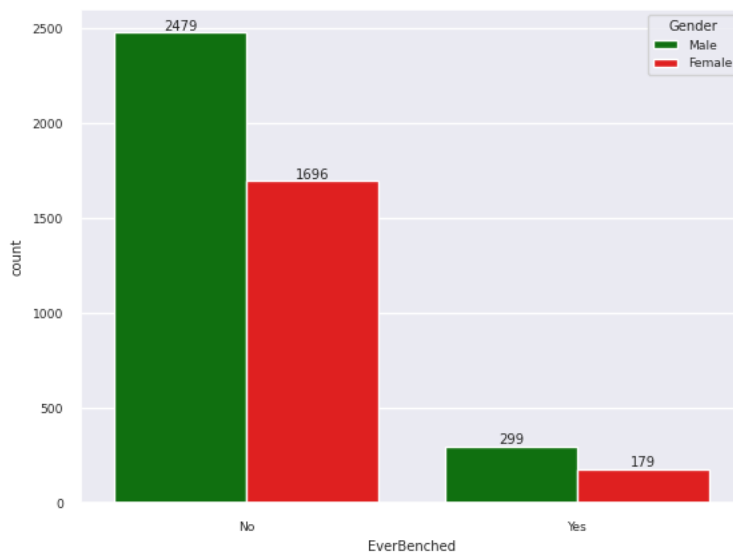
```
sns.set(font_scale=0.6)
sns.lineplot(data=emp,x='JoiningYear',y='PaymentTier',hue='Gender')
```

<Axes: xlabel='JoiningYear', ylabel='PaymentTier'>



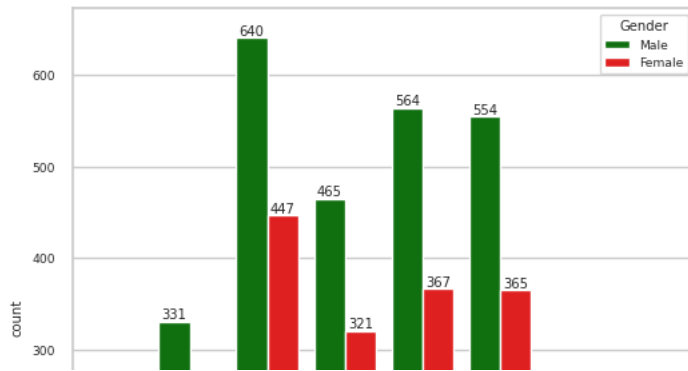
h. Number of employees benched

```
sns.set(font_scale=0.6)
col={'Male':'green','Female':'red'}
s=sns.countplot(data=emp,x='EverBenched',hue='Gender',palette=col)
for i in s.containers:
    s.bar_label(i)
```



i. Experience

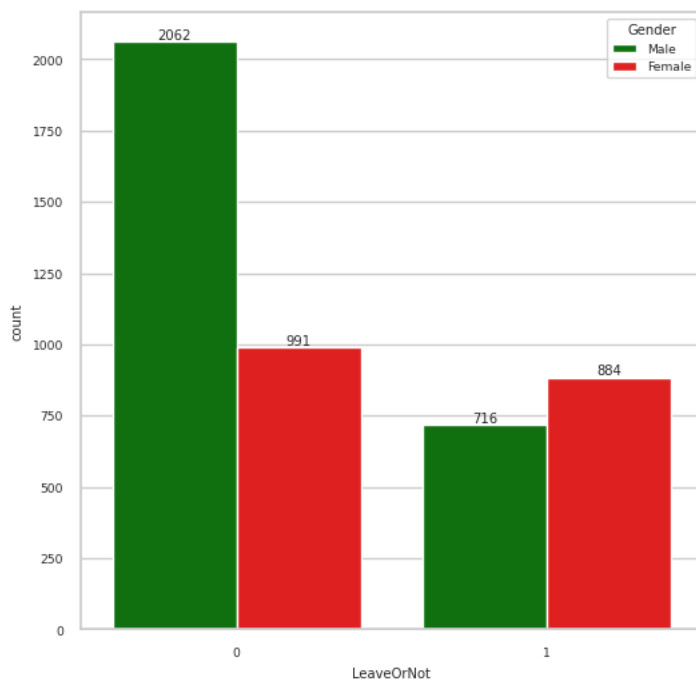
```
sns.set(font_scale=0.6)
plt.figure(figsize=(6,6))
sns.set_style('whitegrid')
col={'Male':'green','Female':'red'}
s=sns.countplot(data=emp,x='ExperienceInCurrentDomain',hue='Gender',palette=col)
for i in s.containers:
    s.bar_label(i)
```



j. Leave or not



```
sns.set(font_scale=0.6)
plt.figure(figsize=(6,6))
sns.set_style('whitegrid')
col={'Male':'green','Female':'red'}
s=sns.countplot(data=emp,x='LeaveOrNot',hue='Gender',palette=col)
for i in s.containers:
    s.bar_label(i)
```



k. Correlation

```
sns.set(font_scale=0.8)
corr=emp.corr()
sns.heatmap(corr,annot=True,fmt='.2f')
```

<Axes: >

