Project-16:HR Absenteeim Data

> Objective-1 : Analysing tha data to get basic information

```
#Importing required libraries
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

```
#Importing Required data into a data frame df variable from the csv file using read_csv() function
df = pd.read_csv('/content/drive/MyDrive/HR Absenteeism data.csv')
```

```
#using head function inorder to display 1st 5 rows of dataframe
df.head()
```

	EmployeeNumber	Surname	GivenName	Gender	City	JobTitle	DepartmentName	StoreLocation	Division	
0	1	Gutierrez	Molly	F	Burnaby	Baker	Bakery	Burnaby	Stores	32.0
1	2	Hardwick	Stephen	M	Courtenay	Baker	Bakery	Nanaimo	Stores	40.3
2	3	Delgado	Chester	M	Richmond	Baker	Bakery	Richmond	Stores	48.8
3	4	Simon	Irene	F	Victoria	Baker	Bakery	Victoria	Stores	44.5
4	5	Delvalle	Edward	M	New Westminster	Baker	Bakery	New Westminster	Stores	35.6

```
#Shape Function used to know number of columns and rows
df.shape
```

```
(8336, 13)
```

```
df.columns
```

```
Index(['EmployeeNumber', 'Surname', 'GivenName', 'Gender', 'City', 'JobTitle',
       'DepartmentName', 'StoreLocation', 'Division', 'Age', 'LengthService',
       'AbsentHours', 'BusinessUnit'],
      dtype='object')
```

```
df.index
```

```
RangeIndex(start=0, stop=8336, step=1)
```

```
df.size
```

```
108368
```

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 8336 entries, 0 to 8335
Data columns (total 13 columns):
 #   Column          Non-Null Count  Dtype  
---  ------          --------------  -----  
 0   EmployeeNumber  8336 non-null   int64  
 1   Surname         8336 non-null   object 
 2   GivenName       8336 non-null   object 
 3   Gender          8336 non-null   object 
 4   City            8336 non-null   object 
 5   JobTitle        8336 non-null   object 
 6   DepartmentName  8336 non-null   object 
 7   StoreLocation   8336 non-null   object 
 8   Division        8336 non-null   object 
 9   Age             8336 non-null   float64
 10  LengthService   8336 non-null   float64
 11  AbsentHours     8336 non-null   float64
 12  BusinessUnit    8336 non-null   object 
dtypes: float64(3), int64(1), object(9)
memory usage: 846.8+ KB
```

```
df.dtypes
```

```
EmployeeNumber      int64
Surname            object
GivenName          object
Gender             object
City               object
```

```
JobTitle          object
DepartmentName    object
StoreLocation     object
Division          object
Age              float64
LengthService    float64
AbsentHours      float64
BusinessUnit      object
dtype: object
```

```
df
```

	EmployeeNumber	Surname	GivenName	Gender	City	JobTitle	DepartmentName	StoreLocation	Division	
0	1	Gutierrez	Molly	F	Burnaby	Baker	Bakery	Burnaby	Stores	3
1	2	Hardwick	Stephen	M	Courtenay	Baker	Bakery	Nanaimo	Stores	4
2	3	Delgado	Chester	M	Richmond	Baker	Bakery	Richmond	Stores	4
3	4	Simon	Irene	F	Victoria	Baker	Bakery	Victoria	Stores	4
4	5	Delvalle	Edward	M	New Westminster	Baker	Bakery	New Westminster	Stores	3
...	...	...	...	...	...	...	...	...	...	
8331	8332	Coniglio	Bianca	F	Langley	Cashier	Customer Service	Langley	Stores	4
8332	8333	Cox	Jimmie	M	Montney	Cashier	Customer Service	Fort St John	Stores	3
8333	8334	Hawkins	Mary	F	West Vancouver	Cashier	Customer Service	West Vancouver	Stores	5
8334	8335	Proctor	Theresa	F	Vancouver	Dairy Person	Dairy	Vancouver	Stores	4
8335	8336	Salter	Charles	M	Vancouver	Dairy Person	Dairy	Vancouver	Stores	4

8336 rows × 13 columns

> Objective-2:DataCleaning and Statisticsl Operations

```
#using any() function to know if the 'Absent hours' contained any value 0
#It returns true if it contains 0 else false
columns_with_zeros = df.columns[(df == 0).any()]

# Print columns with zero values
if len(columns_with_zeros) > 0:
    print("Columns with zero values:")
    for column in columns_with_zeros:
        print(column)
else:
    print("No columns have zero values.")
```

```
Columns with zero values:
AbsentHours
```

```
#using isna() function in order to find if any of the columns have any nulolo values
#isna():returns true if that column have any null values else false
df.isna().any()
```

```
EmployeeNumber    False
Surname           False
GivenName         False
Gender            False
City              False
JobTitle          False
DepartmentName    False
StoreLocation     False
Division          False
Age               False
LengthService     False
AbsentHours       False
BusinessUnit      False
dtype: bool
```

```
print((df['AbsentHours']==0).sum())
```

```
1320
```

```
#using replace() function to replace the all 0 values with mean value of Aabsent hours' columns
Mean_of_absenthrs = df['AbsentHours'].mean()
df['AbsentHours']=df['AbsentHours'].replace(0.,Mean_of_absenthrs)
print((df['AbsentHours']==0).sum())
```

```
0
```

Double-click (or enter) to edit

```
def combine_values(row):
    return f"{row['Surname']}: {row['GivenName']}"

# Apply the custom function to create the 'Combined' column
df['Combined'] = df.apply(combine_values, axis=1)
#print(df['Combined'])
print((df.duplicated(subset = 'Combined',keep = False)).sum())
```

```
232
```

I have combined the columns Surname and Given name to know if there are any Duplicate names and they exist (There are 232 Employees who have same surname and Given name) but they have different Employee id and that solves the problem.

```
min_age = df['Age'].min()
max_age = df['Age'].max()
```

```
print(min_age)
print(max_age)
```

```
3.504742504
77.93800302
```

Here the minimum age is 3 yrs which is not possible.To solve this we shall take age value greater than 22.Inoreder to solve this we shall take the Age of employee from Age 25

```
sub2=df[df['Age']>=25]
sub2
```

	EmployeeNumber	Surname	GivenName	Gender	City	JobTitle	DepartmentName	StoreLocation	Division	
0	1	Gutierrez	Molly	F	Burnaby	Baker	Bakery	Burnaby	Stores	3
1	2	Hardwick	Stephen	M	Courtenay	Baker	Bakery	Nanaimo	Stores	4
2	3	Delgado	Chester	M	Richmond	Baker	Bakery	Richmond	Stores	4
3	4	Simon	Irene	F	Victoria	Baker	Bakery	Victoria	Stores	4
4	5	Delvalle	Edward	M	New Westminster	Baker	Bakery	New Westminster	Stores	3
...	...	...	...	...	...	...	...	...	...	
8331	8332	Coniglio	Bianca	F	Langley	Cashier	Customer Service	Langley	Stores	4
8332	8333	Cox	Jimmie	M	Montney	Cashier	Customer Service	Fort St John	Stores	3
8333	8334	Hawkins	Mary	F	West Vancouver	Cashier	Customer Service	West Vancouver	Stores	5
8334	8335	Proctor	Theresa	F	Vancouver	Dairy Person	Dairy	Vancouver	Stores	4
8335	8336	Salter	Charles	M	Vancouver	Dairy Person	Dairy	Vancouver	Stores	4

7974 rows × 15 columns

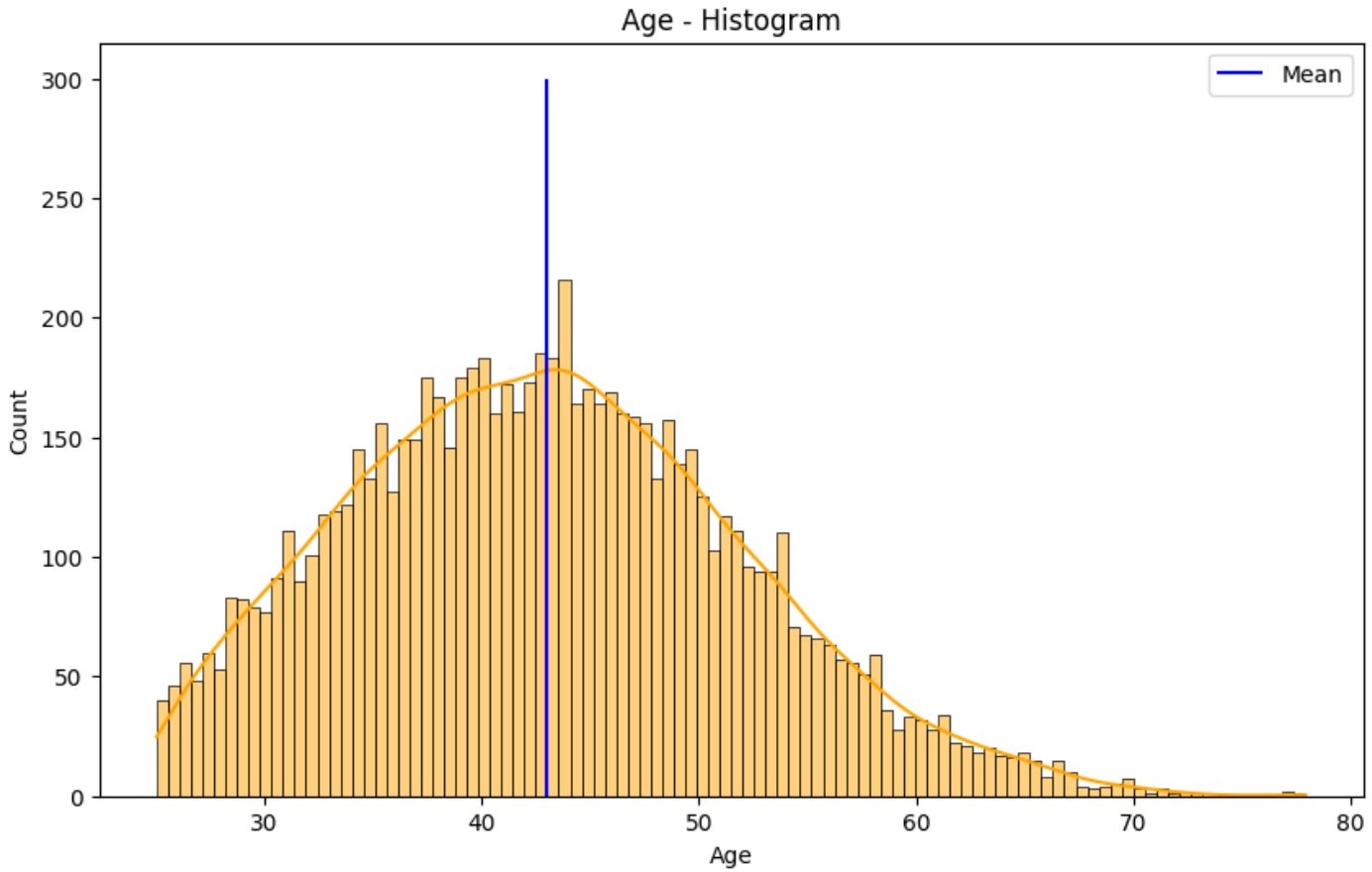
>> >Objective-3 : Data Visualization

1.**Histogram**

```
mean_Age = sub2['Age'].mean()
print(mean_Age)
```

```
42.975422462959614
```

```
plt.figure(figsize=(10,6))
sns.histplot(x='Age',data=sub2,kde=True,color='orange',bins=100)
plt.title('Age - Histogram')
plt.vlines(mean_Age,ymin=0,ymax=300,color='b',label='Mean')
plt.legend()
plt.show()
```



Red line indicates median

```
print(df['LengthService'].median())
print(df['LengthService'].min(),df['LengthService'].max())
```

```
4.600248169
0.012097544 43.735239
```

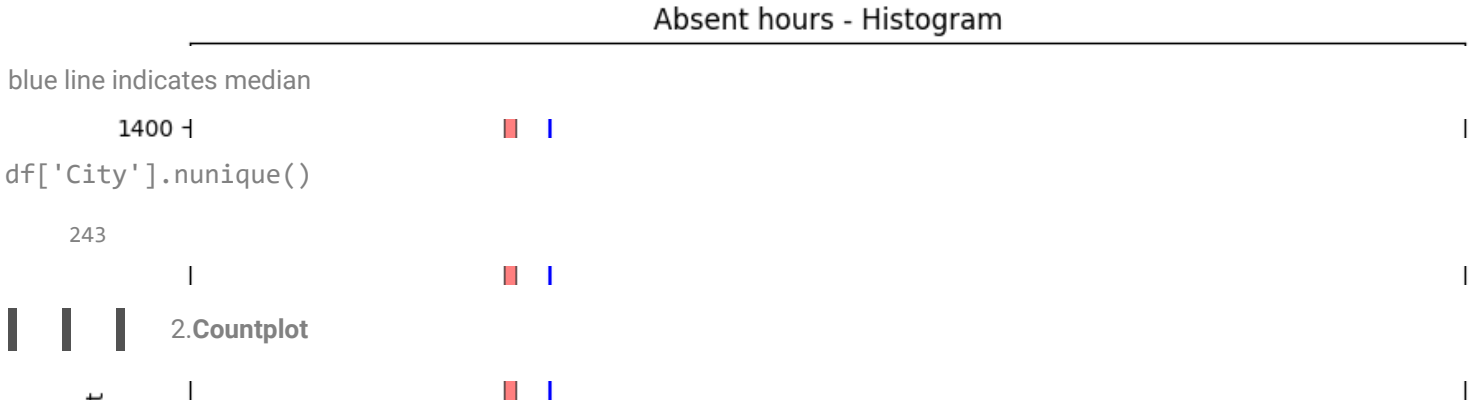
```
plt.figure(figsize=(10,6))
sns.histplot(x='LengthService',data=df,kde=True,color='blue',bins=100)
plt.title('Length of Service - Histogram')
plt.vlines(df['LengthService'].mean(),ymin=0,ymax=1100,color='r',label='Mean')
plt.legend()
plt.show()
```

Red line indicates median

```
df['AbsentHours'].median()
```

```
56.005807515
```


```
plt.figure(figsize=(10,6))
sns.histplot(x='AbsentHours',data=df,kde=True,color='red',bins=100)
plt.title('Absent hours - Histogram')
plt.vlines(df['AbsentHours'].mean(),ymin=0,ymax=1500,color='b',label='Mean')
plt.legend()
plt.show()
```

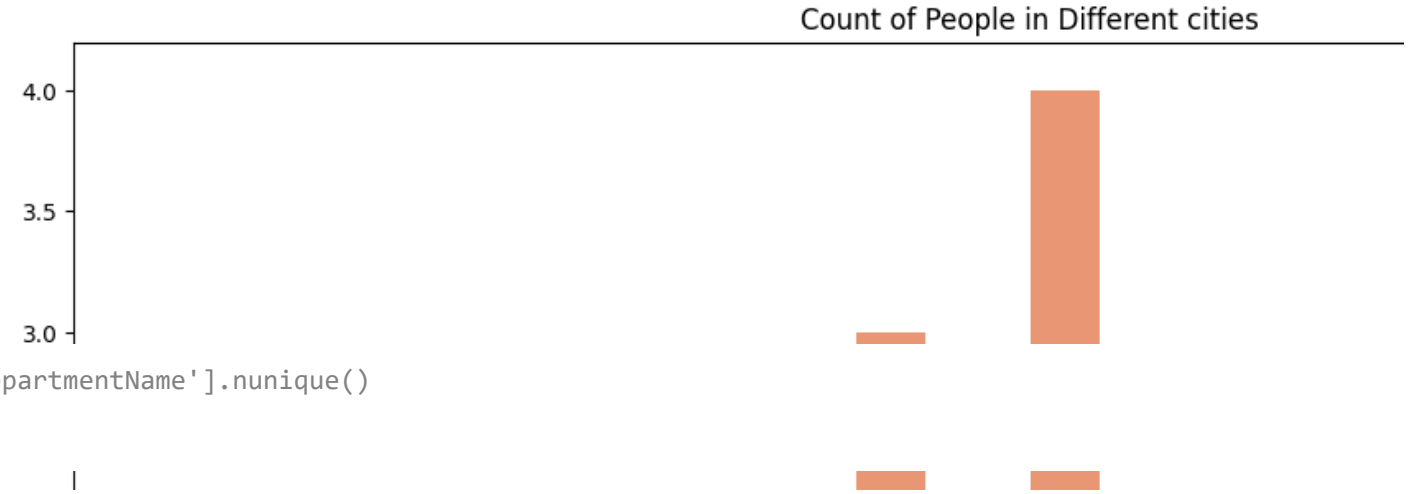



```
df['City'].nunique()
```

```
243
```

2.Countplot

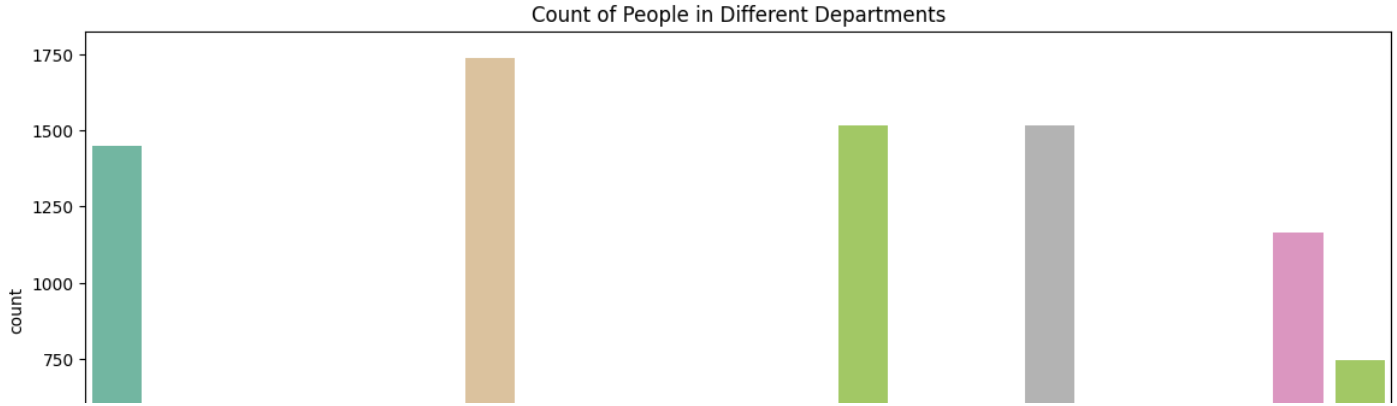
```
plt.figure(figsize=(15,8))
sns.countplot(x='City',data=df[0:20],hue='Gender',palette="Set2")
plt.title('Count of People in Different cities')
plt.show()
```



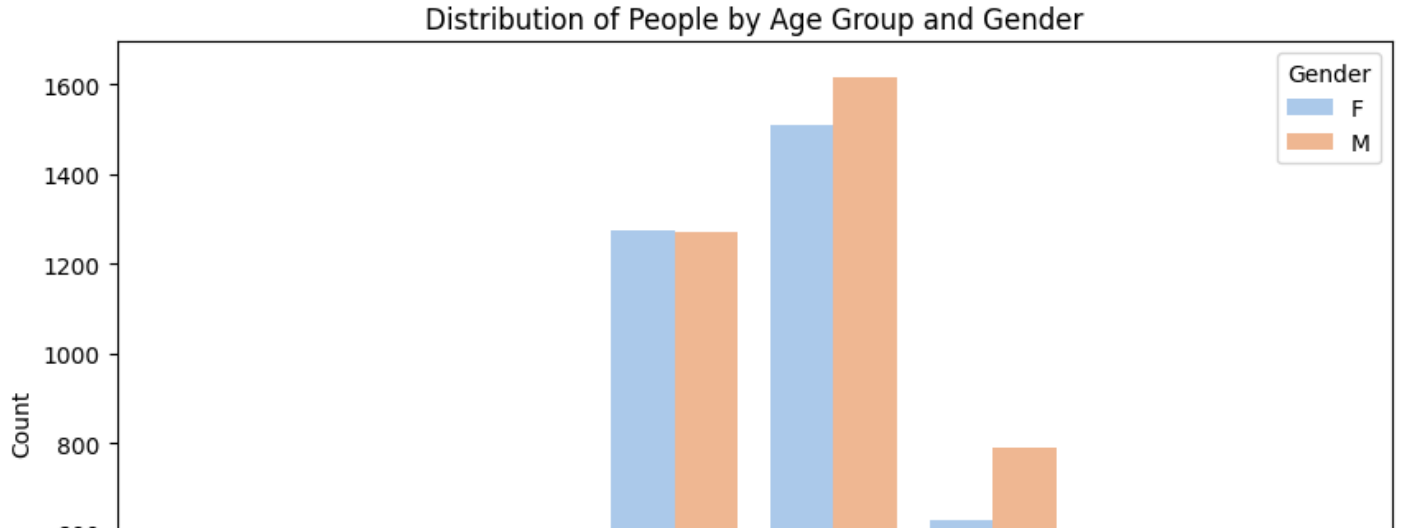

```
df['DepartmentName'].nunique()
```

```
21
```

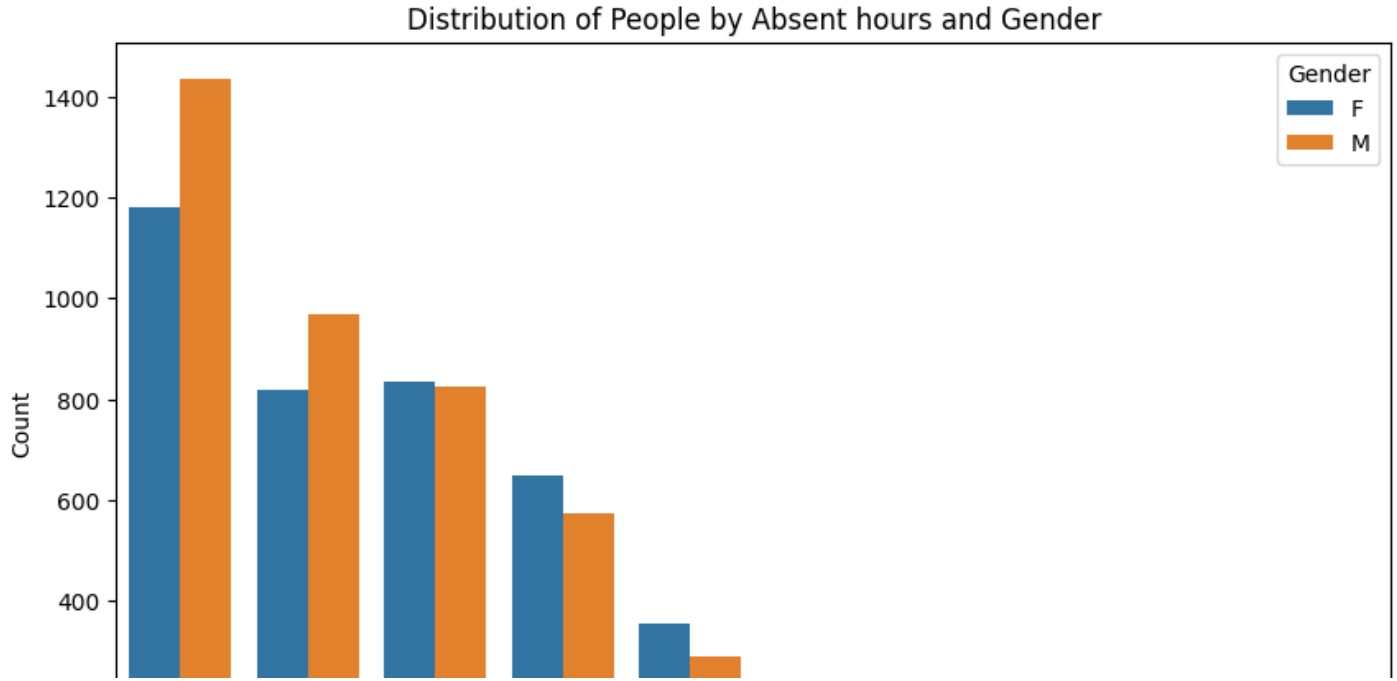
```
plt.figure(figsize=(14,6))
sns.countplot(x='DepartmentName',data=df,palette="Set2")
plt.title('Count of People in Different Departments')
plt.xticks(rotation = 45)
plt.show()
```




```
age_bins = [0, 10, 20, 30, 40, 50, 60, 70,80]
age_labels = ['0-10', '11-20', '21-30', '31-40', '41-50', '51-60', '61-70','71-80']

# Assign age groups to the dataframe using pd.cut()
df['Age Group'] = pd.cut(df['Age'], bins=age_bins, labels=age_labels, right=False)

# Create a stacked bar plot
plt.figure(figsize=(10, 6))
sns.countplot(data=df, x='Age Group', hue='Gender', palette='pastel')
plt.xlabel('Age Group')
plt.ylabel('Count')
plt.title('Distribution of People by Age Group and Gender')
plt.legend(title='Gender', loc='upper right')
plt.xticks(rotation=45)
plt.show()
```




```
age_bins = [0, 30, 60,90,120,150,180, 210, 240,270,300]
age_labels = ['0-30', '31-60', '61-90', '91-120', '121-150', '151-180', '181-210','211-240','241-270','271-300']

# Assign age groups to the dataframe using pd.cut()
df['Age Group'] = pd.cut(df['AbsentHours'], bins=age_bins, labels=age_labels, right=False)

# Create a stacked bar plot
plt.figure(figsize=(10, 6))
sns.countplot(data=df, x='Age Group', hue='Gender', palette=['#1f77b4', '#ff7f0e', '#2ca02c', '#d62728', '#9467bd', '#8c564b', '#e377c
plt.xlabel('Age Group')
plt.ylabel('Count')
plt.title('Distribution of People by Absent hours and Gender')
plt.legend(title='Gender', loc='upper right')
plt.xticks(rotation=45)
plt.show()
```

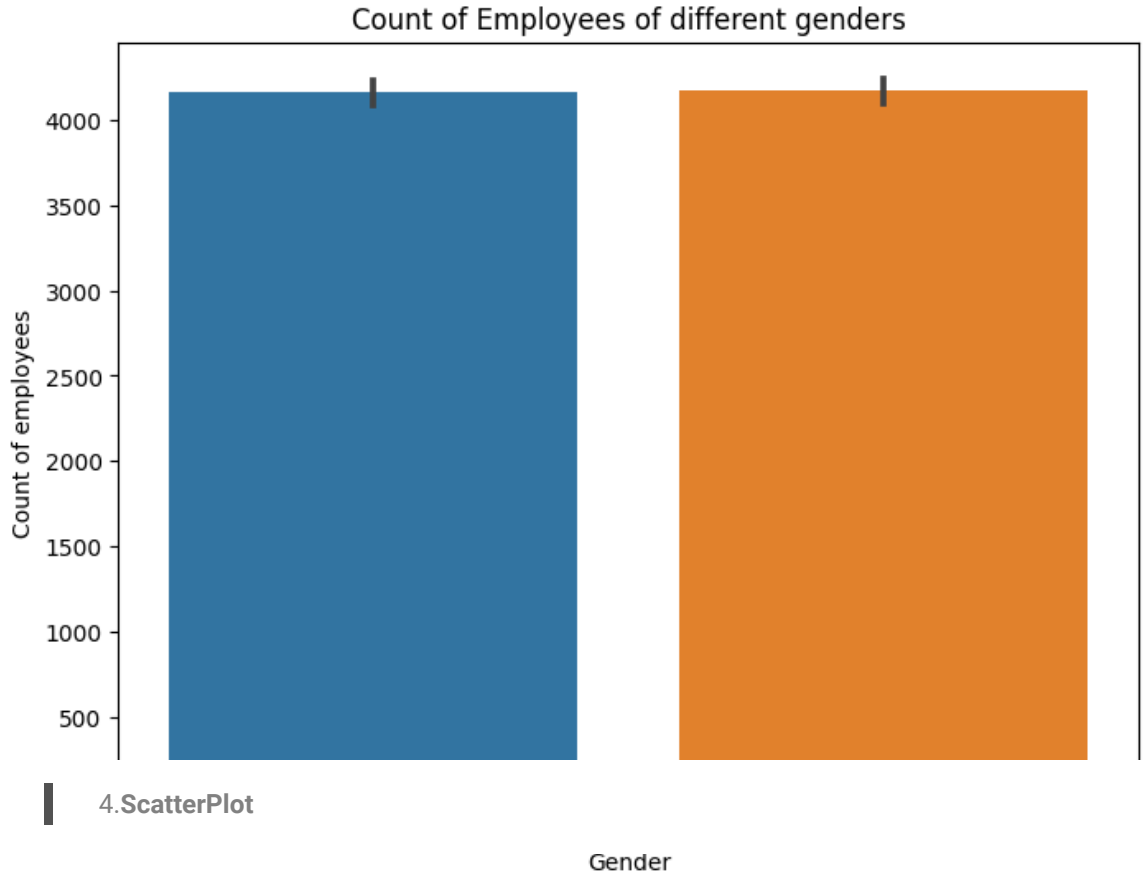


3.Barplot

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

```
M    4216
F    4120
Name: Gender, dtype: int64
```

```
plt.figure(figsize=(8,6))
sns.barplot(data=df,y='EmployeeNumber',x='Gender')
plt.ylabel("Count of employees")
plt.xlabel('Gender')
plt.title('Count of Employees of different genders')
plt.show()
```



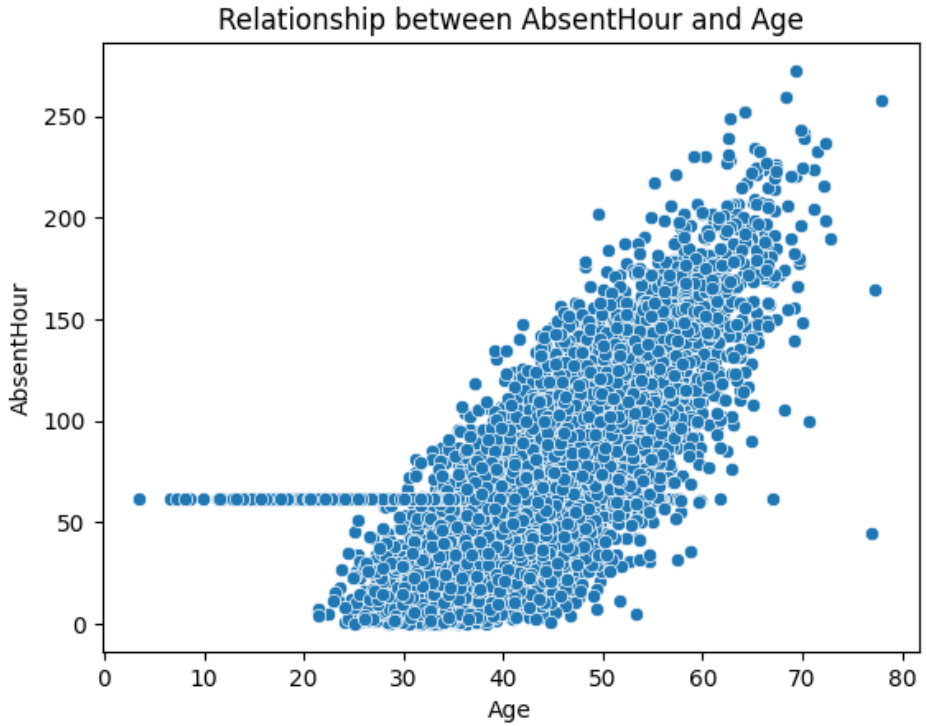
4.**ScatterPlot**

```
cor = df['Age'].corr(df['AbsentHours'])
print(cor)
```

```
0.6790913056408363
```

correlation = 0.6790913056408363 indicates a very weak Positive correlation between Age of Employees and Absent hours. Specifically, it suggests that there is a slight tendency for one variable to decrease slightly as the other variable increases, but the relationship is not strong.

```
sns.scatterplot(data=df,x='Age',y='AbsentHours')
plt.xlabel('Age')
plt.ylabel('AbsentHour')
plt.title('Relationship between AbsentHour and Age')
plt.show()
```

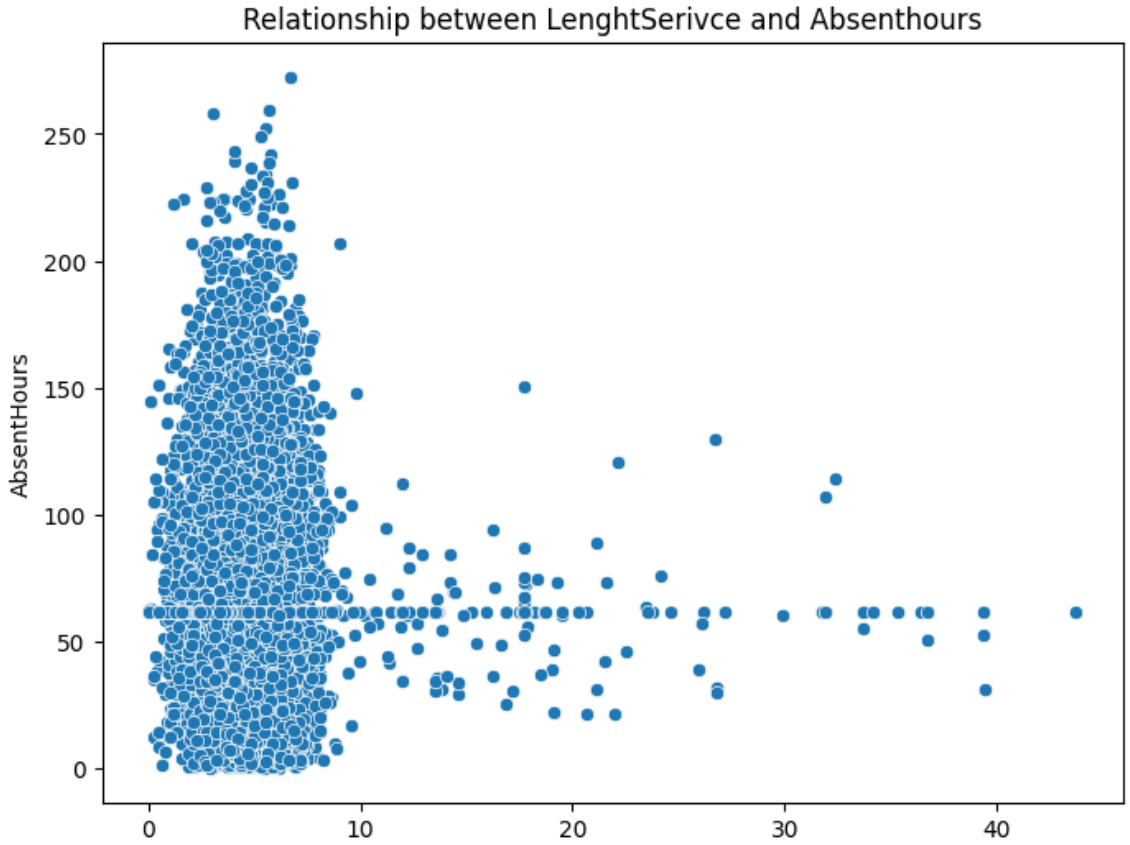



```
cor = df['LengthService'].corr(df['AbsentHours'])
print(cor)
```

```
-0.02352242282285967
```

correlation = -0.02352242282285967 indicates a very weak negative correlation between Lenght of service and Absent hours. Specifically, it suggests that there is a slight tendency for one variable to decrease slightly as the other variable increases, but the relationship is not strong.

```
plt.figure(figsize=(8,6))
sns.scatterplot(data=df,x='LengthService',y='AbsentHours')
plt.xlabel('Lenght of service')
plt.ylabel('AbsentHours')
plt.title('Relationship between LenghtSerivce and Absenthours')
plt.show()
```



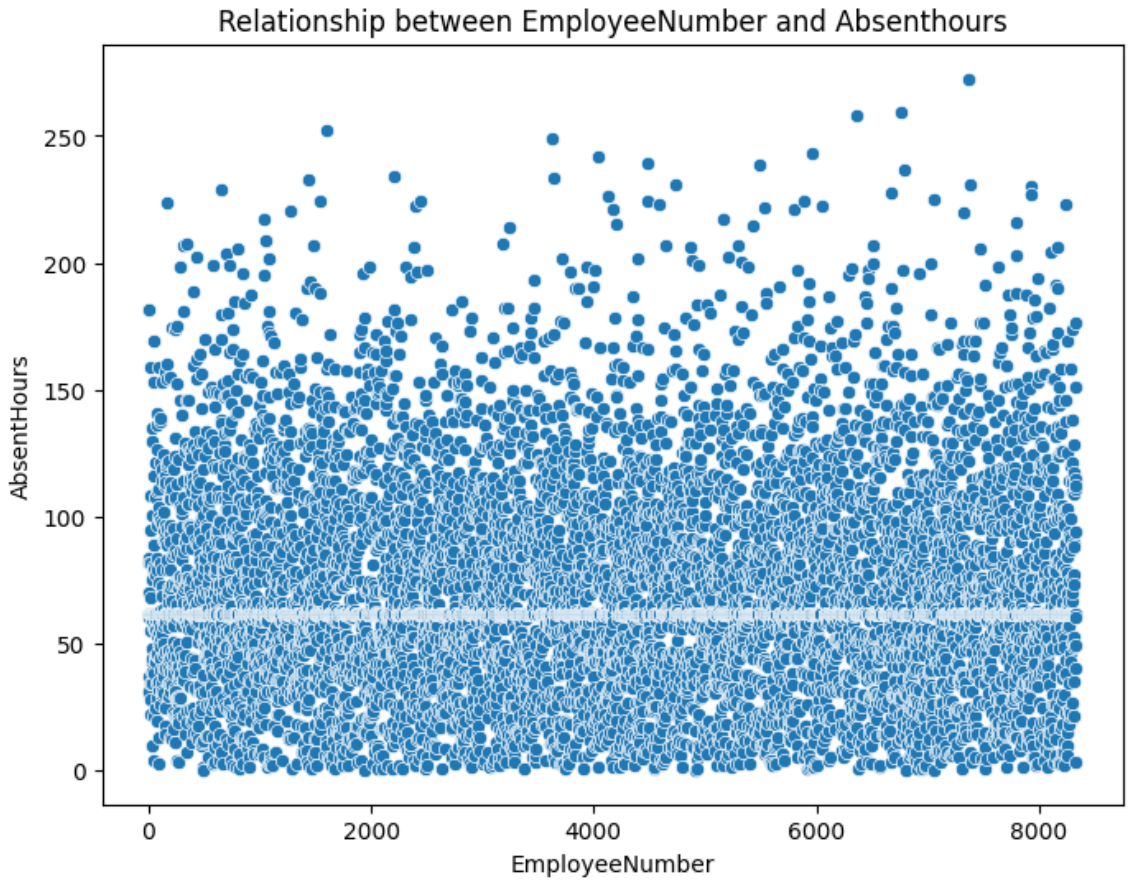

```
cor = df['EmployeeNumber'].corr(df['AbsentHours'])
print(cor)
```

```
0.006319114964287553
```

A correlation coefficient of approximately 0.0063 (rounded) indicates a very weak positive correlation between Employee Number and Absent Hours. Specifically, it suggests that there is a slight tendency for one variable to increase slightly as the other variable increases, but the relationship is extremely weak

```
plt.figure(figsize=(8,6))
sns.scatterplot(data=df,x='EmployeeNumber',y='AbsentHours')
plt.xlabel('EmployeeNumber')
plt.ylabel('AbsentHours')
```

```
plt.title('Relationship between EmployeeNumber and Absenthours')
plt.show()
```

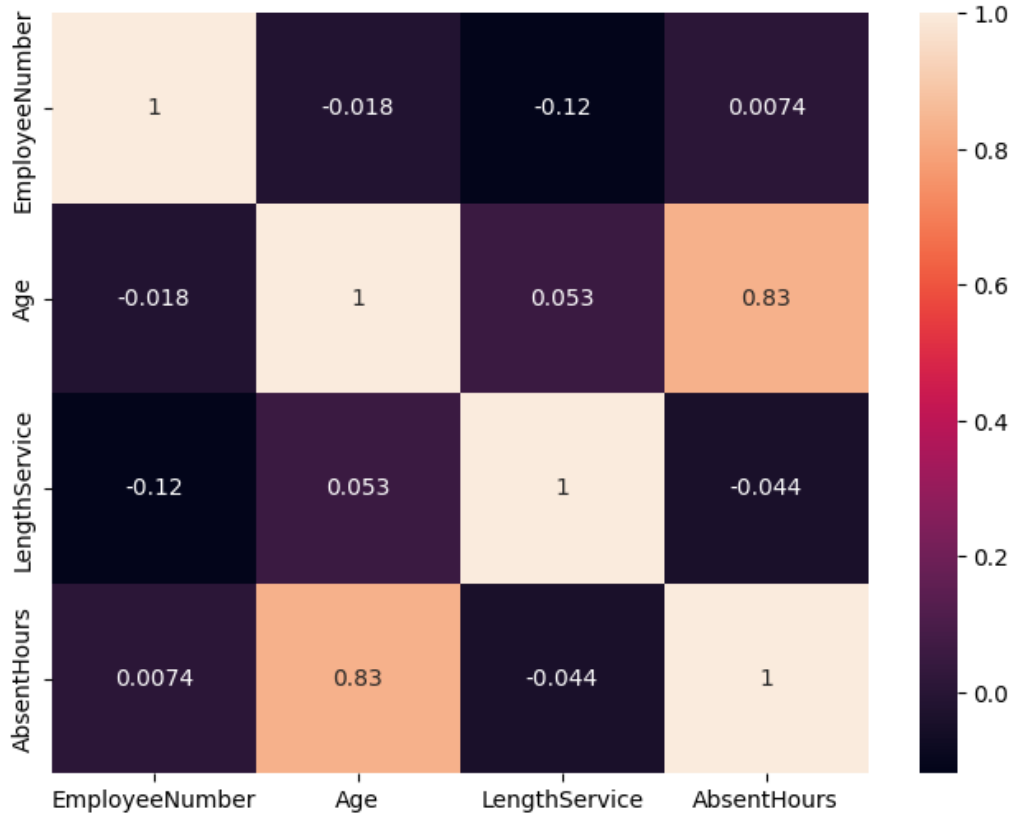


>>>5.HeatMap

This graph shows the corelation between different numeric variables in dataframe

```
plt.figure(figsize=(8,6))
sns.heatmap(df.corr(),annot=True)
plt.show()
```

```
<ipython-input-22-b50ddf15ecaa>:2: FutureWarning: The default value of numeric_only in DataFrame.corr is deprecated. In a futur
  sns.heatmap(df.corr(),annot=True)
```



Colab paid products - Cancel contracts here

✓ 0s completed at 6:25 PM