In [ ]:
```
# Exploratory Data Analysis (EDA) for Tuberculosis X-Ray Dataset (Synthetic)
```

In [34]:
```
# Step 1: Import Libraries

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

In [ ]:
```
# Set plot style

sns.set(style="whitegrid", palette="pastel")
```

In [ ]:
```
# Step 2: Load the Dataset

df = pd.read_csv("tuberculosis_xray_dataset.csv")
```

In [ ]:
```
# Exploring the data
```

In [54]:
```
df.head()
```

Out[54]:

	Patient_ID	Age	Gender	Chest_Pain	Cough_Severity	Breathlessness	Fatigue	Weight_Loss	Fever	Night_Sweats	Sputum_
0	PID000001	69	Male	Yes	1	2	3	2.37	Moderate	Yes	
1	PID000002	32	Female	Yes	3	0	9	6.09	Moderate	No	
2	PID000003	89	Male	No	7	0	3	2.86	Mild	Yes	
3	PID000004	78	Female	Yes	2	0	6	4.57	Moderate	No	
4	PID000005	38	Male	No	7	2	5	13.86	High	Yes	

In [56]:
```
df.shape
```

Out[56]:
```
(20000, 15)
```

In [90]:
```
Duplicate_values = df.duplicated().sum

print("Duplicate_values:\n",Duplicate_values)
```

```
Duplicate_values:
 <bound method Series.sum of 0        False
1        False
2        False
3        False
4        False
         ...
19995    False
19996    False
19997    False
19998    False
19999    False
Length: 20000, dtype: bool>
```

In [ ]:
```
# Value count for some columns
```

In [64]:
```
df['Gender'].value_counts()
```

Out[64]:
```
Gender
Male      10171
Female     9829
Name: count, dtype: int64
```

In [78]:
```
df['Fever'].value_counts()
```

Out[78]:
```
Fever
Moderate    6713
Mild        6701
High        6586
Name: count, dtype: int64
```

In [80]:
```
df['Class'].value_counts()
```

Out[80]:
```
Class
Normal          14082
Tuberculosis     5918
Name: count, dtype: int64
```

In [ ]:
```
# Step 3: Basic Information
```

In [92]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 20000 entries, 0 to 19999
Data columns (total 15 columns):
 #   Column               Non-Null Count  Dtype  
---  ------               --------------  -----  
 0   Patient_ID           20000 non-null  object 
 1   Age                  20000 non-null  int64  
 2   Gender               20000 non-null  object 
 3   Chest_Pain           20000 non-null  object 
 4   Cough_Severity       20000 non-null  int64  
 5   Breathlessness       20000 non-null  int64  
 6   Fatigue              20000 non-null  int64  
 7   Weight_Loss          20000 non-null  float64
 8   Fever                20000 non-null  object 
 9   Night_Sweats         20000 non-null  object 
 10  Sputum_Production    20000 non-null  object 
 11  Blood_in_Sputum      20000 non-null  object 
 12  Smoking_History      20000 non-null  object 
 13  Previous_TB_History  20000 non-null  object 
 14  Class                20000 non-null  object 
dtypes: float64(1), int64(4), object(10)
memory usage: 2.3+ MB
```

In [11]:
```
print(df.describe())
```

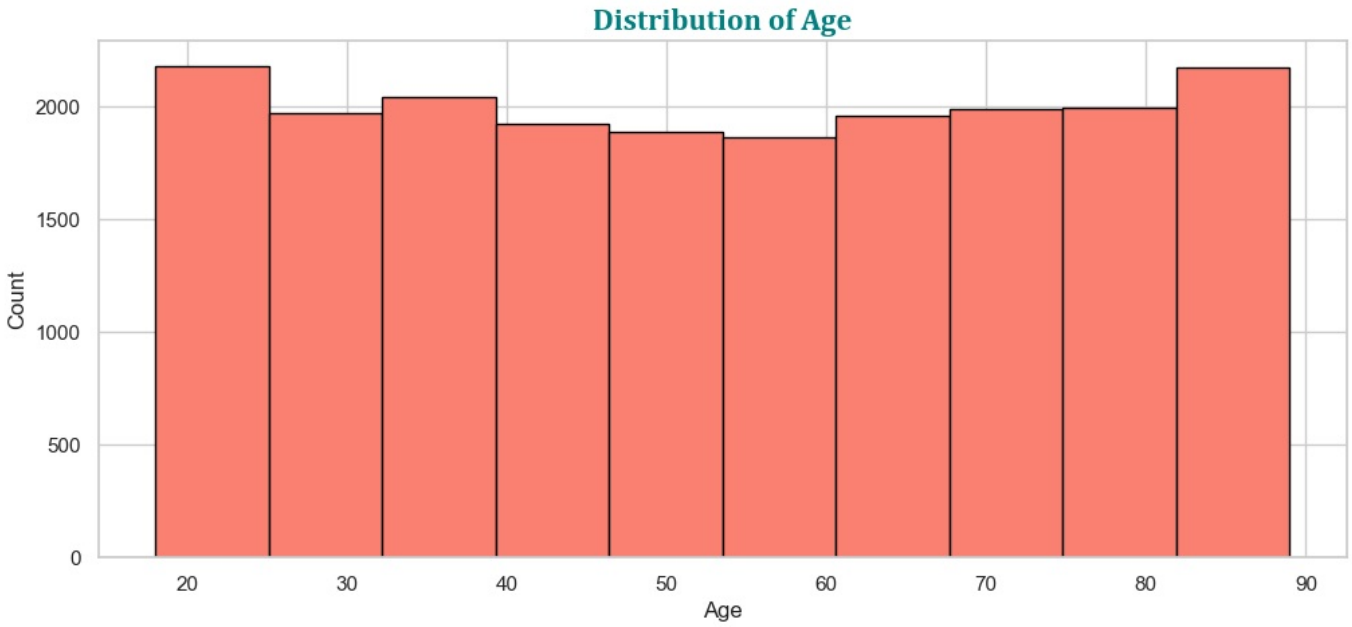
```
                Age  Cough_Severity  Breathlessness       Fatigue  \
count  20000.000000    20000.000000    20000.000000  20000.000000   
mean      53.467450        4.491350        2.003450      4.508450   
std       20.773984        2.864723        1.417123      2.881552   
min       18.000000        0.000000        0.000000      0.000000   
25%       35.000000        2.000000        1.000000      2.000000   
50%       53.000000        4.000000        2.000000      5.000000   
75%       71.000000        7.000000        3.000000      7.000000   
max       89.000000        9.000000        4.000000      9.000000   

        Weight_Loss  
count  20000.000000  
mean       7.455281  
std        4.339864  
min        0.000000  
25%        3.640000  
50%        7.490000  
75%       11.200000  
max       15.000000  
```

In [13]:
```
print(df.isnull().sum())
```

```
Patient_ID             0
Age                    0
Gender                 0
Chest_Pain             0
Cough_Severity         0
Breathlessness         0
Fatigue                0
Weight_Loss            0
Fever                  0
Night_Sweats           0
Sputum_Production      0
Blood_in_Sputum        0
Smoking_History        0
Previous_TB_History    0
Class                  0
dtype: int64
```

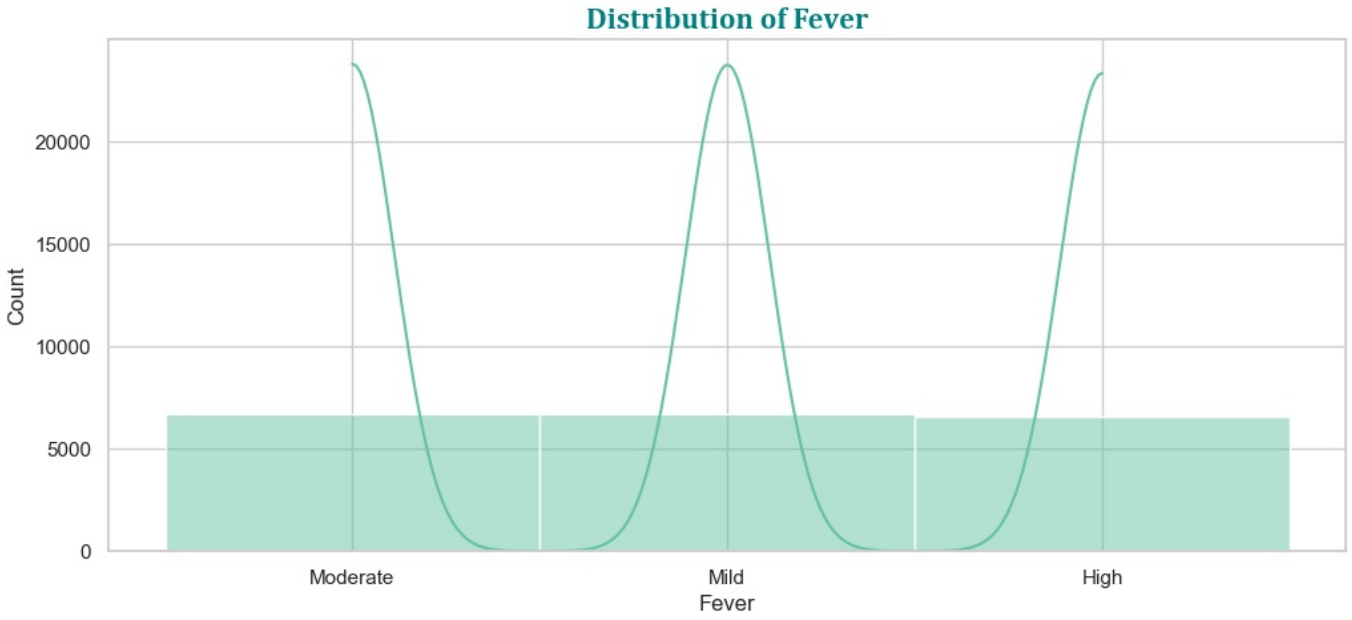
In [ ]:
```
# Step 4: Univariate Analysis
```

In [358…
```
# Numerical Features
# (i) Age distribution

# With increased chart size
plt.figure(figsize= (12, 5))

# Histogram visualization(Matplotlib):
plt.hist(df['Age'], bins = 10, color='salmon', edgecolor = 'black')
plt.title("Distribution of Age",
          fontname='Cambria', fontweight='bold', fontsize=16, color="teal")         # Defining Font

plt.xlabel("Age")
plt.ylabel("Count")

plt.show()
```



In [360…

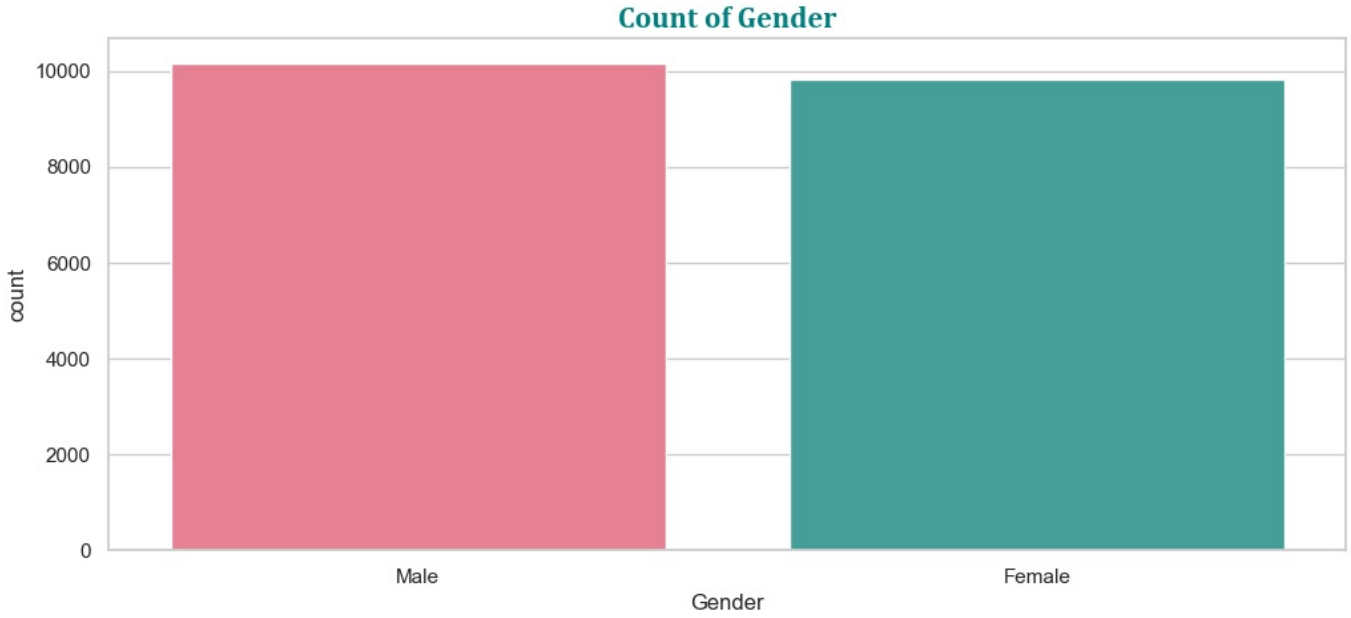
```
# (ii) Fever Distribution
plt.figure(figsize=(12, 5))

# Histogram visualization (Seaborn):
sns.histplot(df['Fever'], kde=True, color=sns.color_palette("Set2")[0])
plt.title("Distribution of Fever",
          fontname='Cambria', fontweight='bold', fontsize=16, color="teal")

plt.show()
```



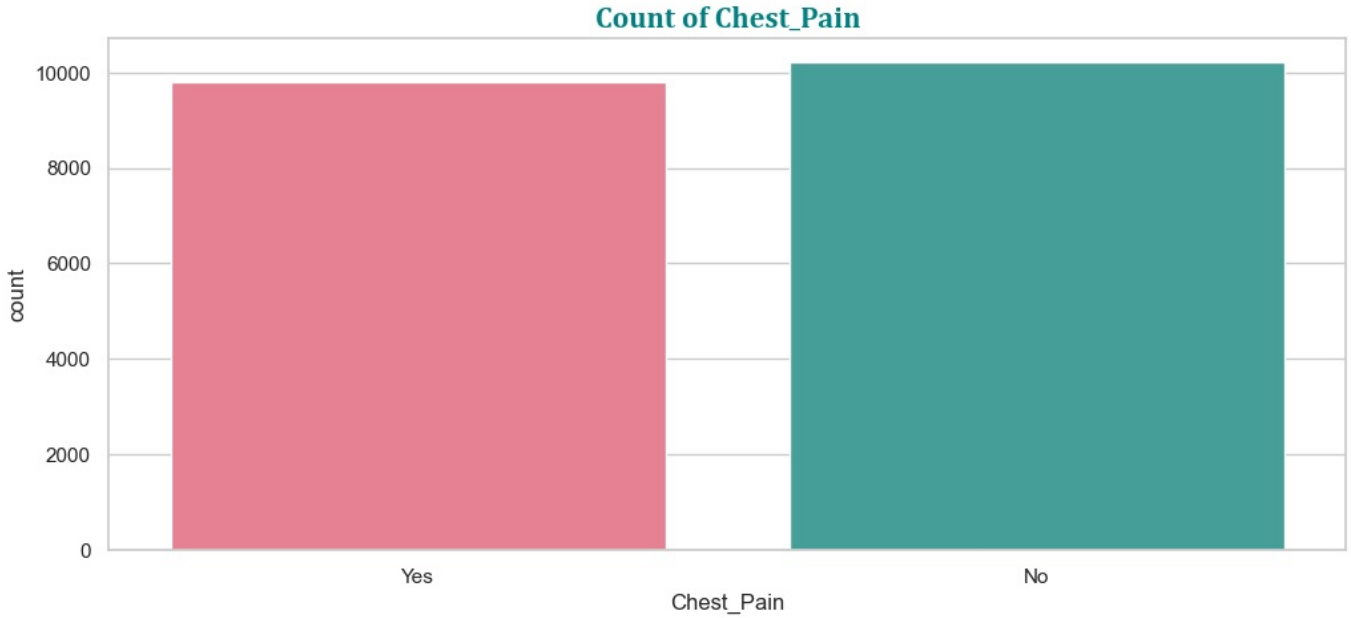
In [364…

```
# Categorical Features

# Storing data in 'cat_col'
cat_cols = ['Gender', 'Chest_Pain', 'Cough_Severity', 'Breathlessness',
            'Night_Sweats', 'Sputum_Production', 'Blood_in_Sputum',
            'Smoking_History', 'Previous_TB_History', 'Class']

for col in cat_cols:
    plt.figure(figsize=(12, 5))

    sns.countplot(x=col, data=df, palette="husl")
    plt.title(f'Count of {col}',
              fontname='Cambria', fontweight='bold', fontsize=16, color="teal")

    # Aligning X-Axis horizontally(rotation = 0)
    plt.xticks(rotation=0)

    plt.show()
```

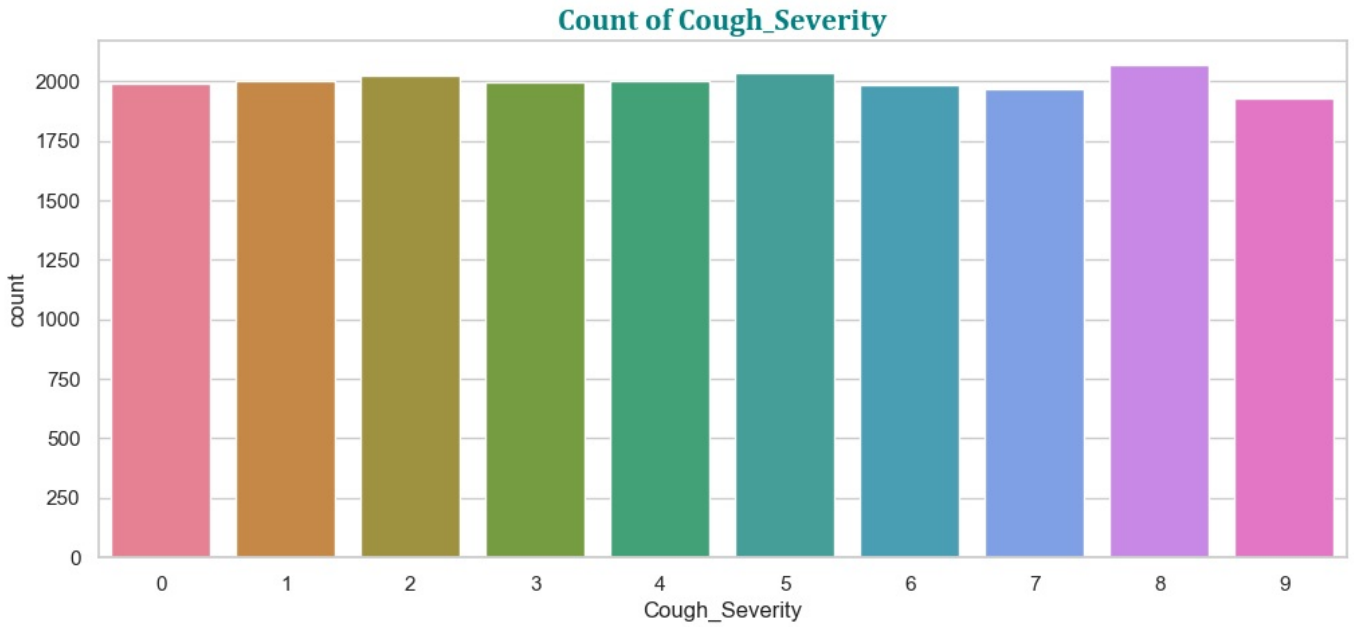
```
C:\Users\damuj\AppData\Local\Temp\ipykernel_23160\757042436.py:11: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v0.14.0. Assign the `x` variable
to `hue` and set `legend=False` for the same effect.

  sns.countplot(x=col, data=df, palette="husl")
```



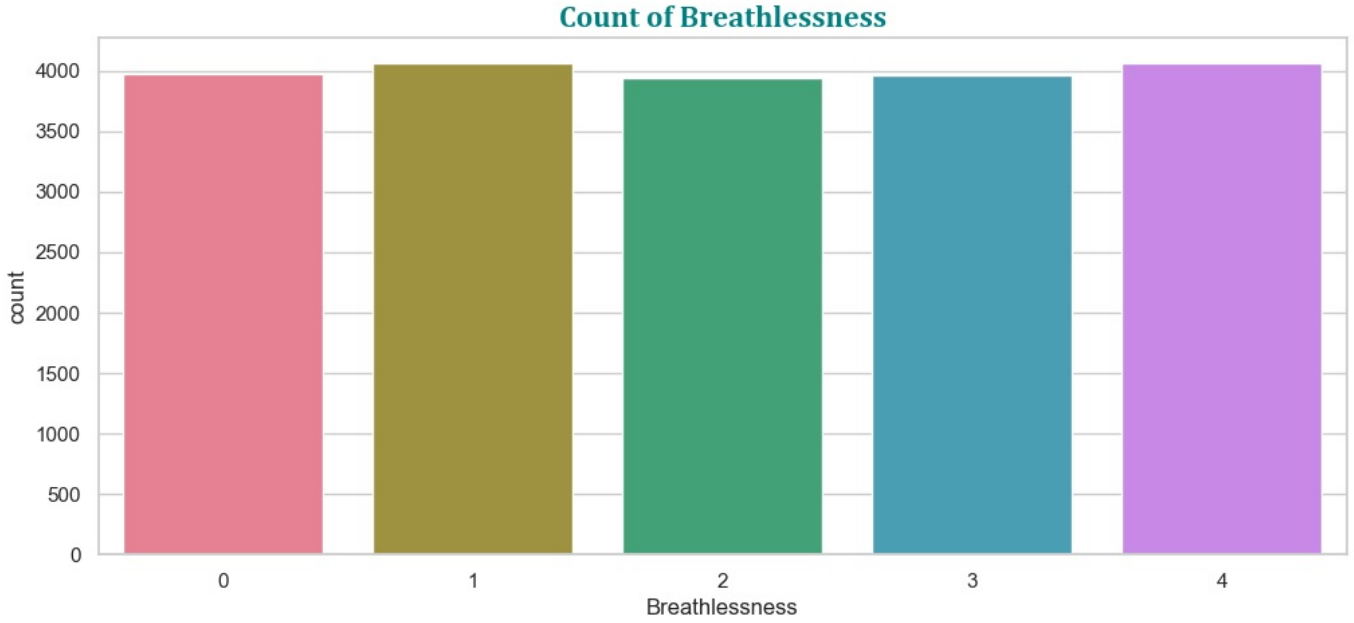

```
C:\Users\damuj\AppData\Local\Temp\ipykernel_23160\757042436.py:11: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v0.14.0. Assign the `x` variable
to `hue` and set `legend=False` for the same effect.

  sns.countplot(x=col, data=df, palette="husl")
```



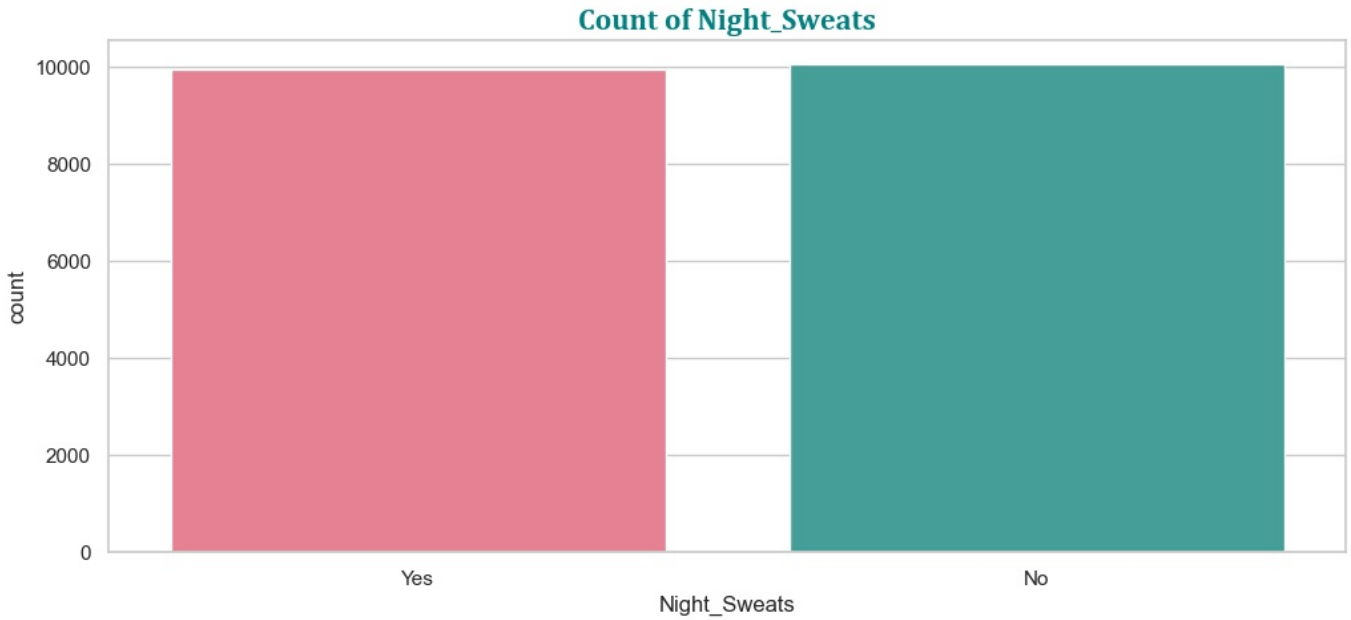

```
C:\Users\damuj\AppData\Local\Temp\ipykernel_23160\757042436.py:11: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v0.14.0. Assign the `x` variable
to `hue` and set `legend=False` for the same effect.

  sns.countplot(x=col, data=df, palette="husl")
```



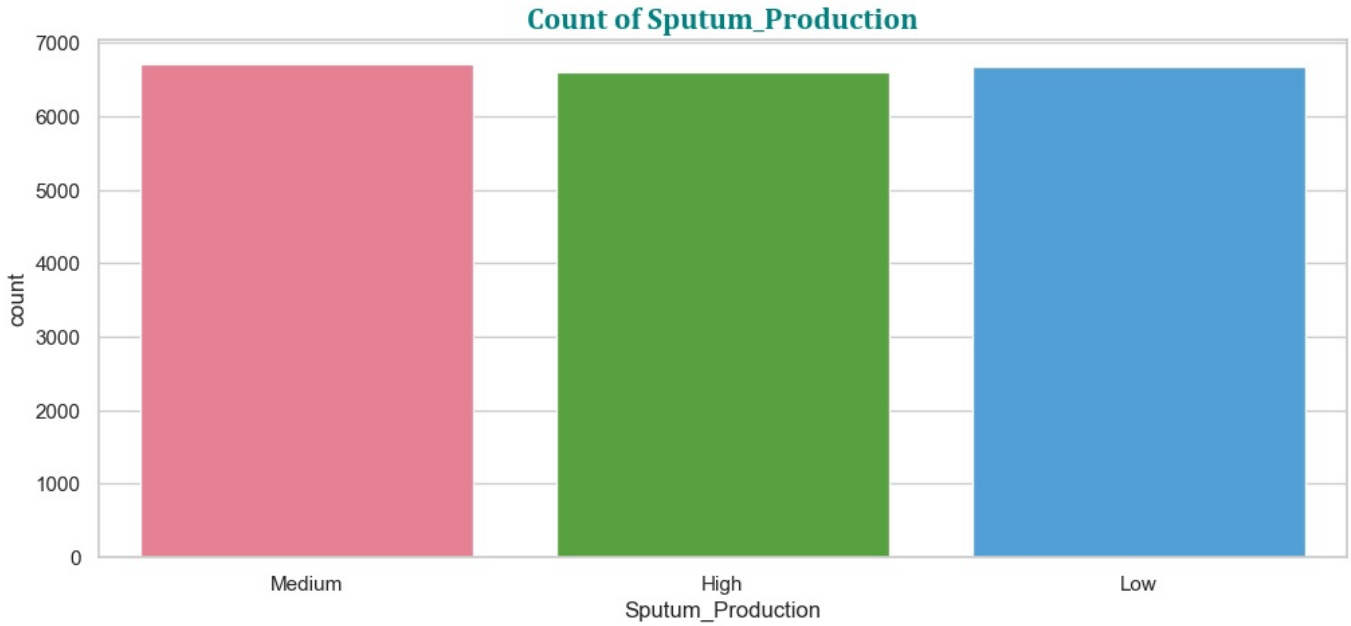

```
C:\Users\damuj\AppData\Local\Temp\ipykernel_23160\757042436.py:11: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v0.14.0. Assign the `x` variable
to `hue` and set `legend=False` for the same effect.

  sns.countplot(x=col, data=df, palette="husl")
```



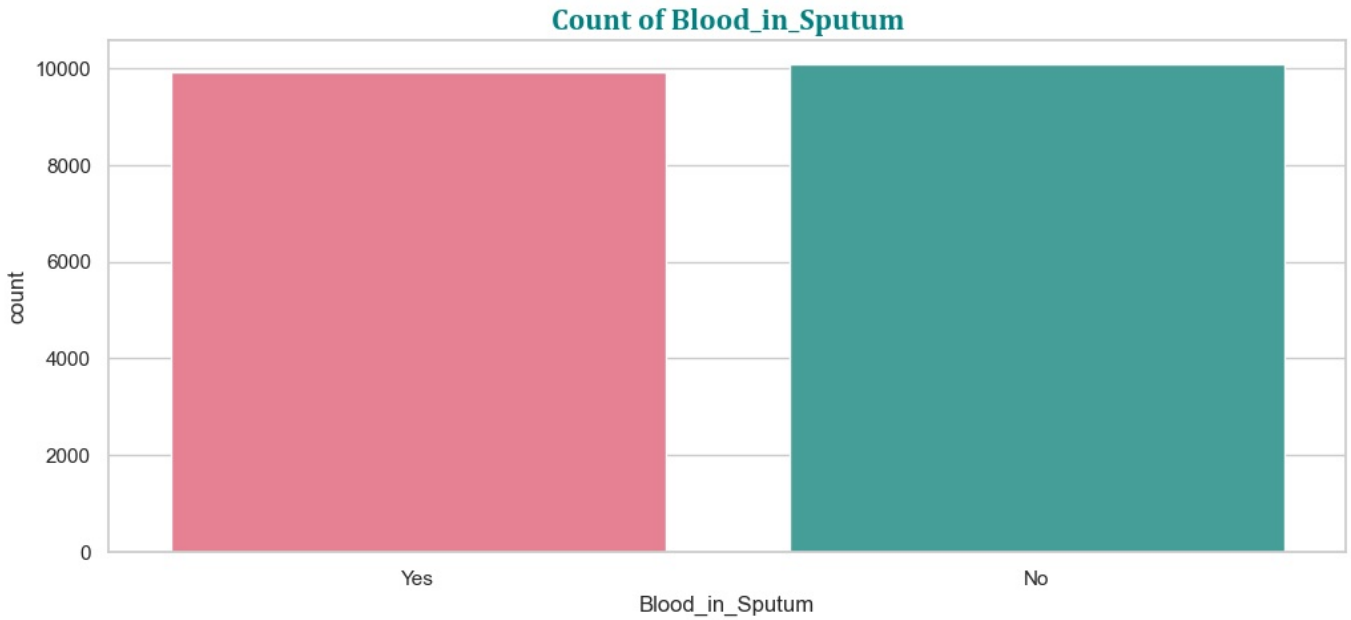

```
C:\Users\damuj\AppData\Local\Temp\ipykernel_23160\757042436.py:11: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v0.14.0. Assign the `x` variable
to `hue` and set `legend=False` for the same effect.

  sns.countplot(x=col, data=df, palette="husl")
```



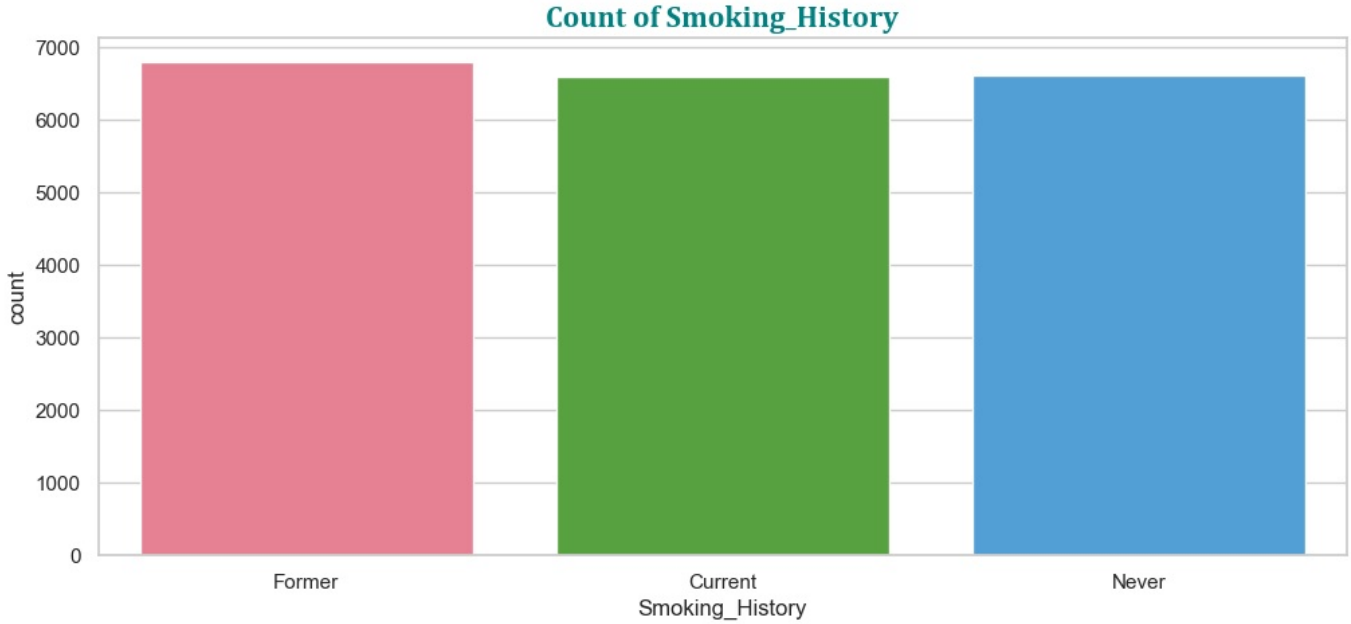

```
C:\Users\damuj\AppData\Local\Temp\ipykernel_23160\757042436.py:11: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v0.14.0. Assign the `x` variable
to `hue` and set `legend=False` for the same effect.

  sns.countplot(x=col, data=df, palette="husl")
```



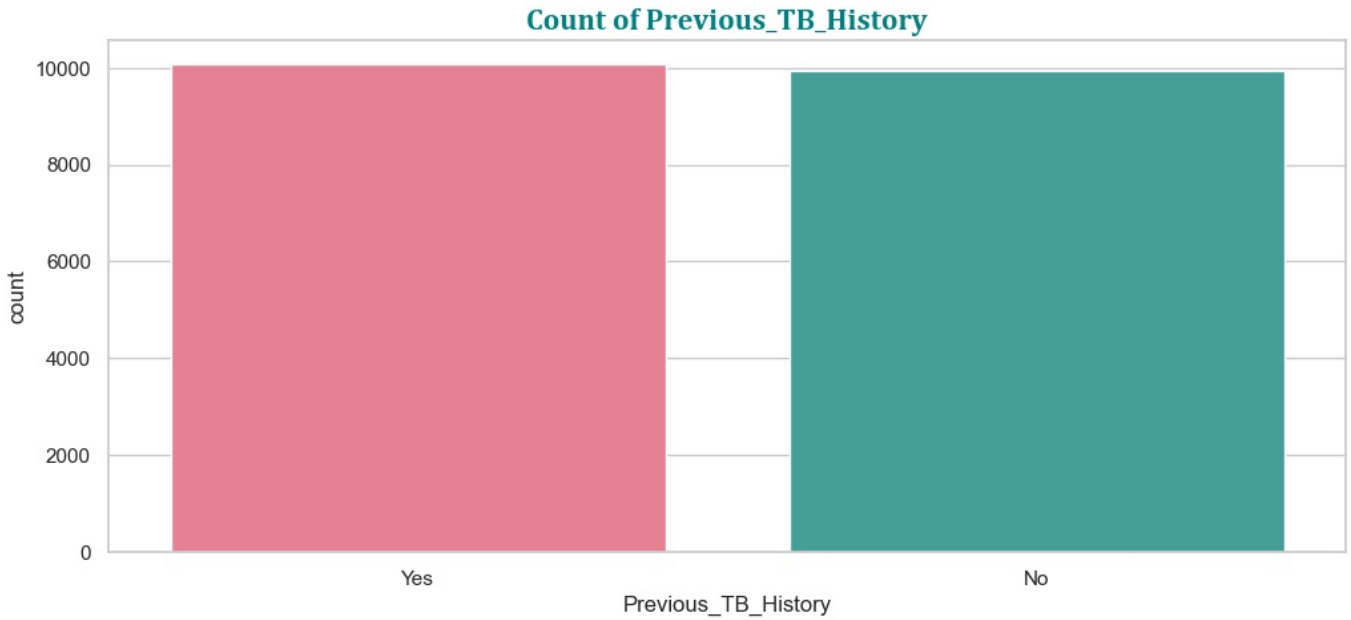

```
C:\Users\damuj\AppData\Local\Temp\ipykernel_23160\757042436.py:11: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v0.14.0. Assign the `x` variable
to `hue` and set `legend=False` for the same effect.

  sns.countplot(x=col, data=df, palette="husl")
```




```
C:\Users\damuj\AppData\Local\Temp\ipykernel_23160\757042436.py:11: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v0.14.0. Assign the `x` variable
to `hue` and set `legend=False` for the same effect.

  sns.countplot(x=col, data=df, palette="husl")
```



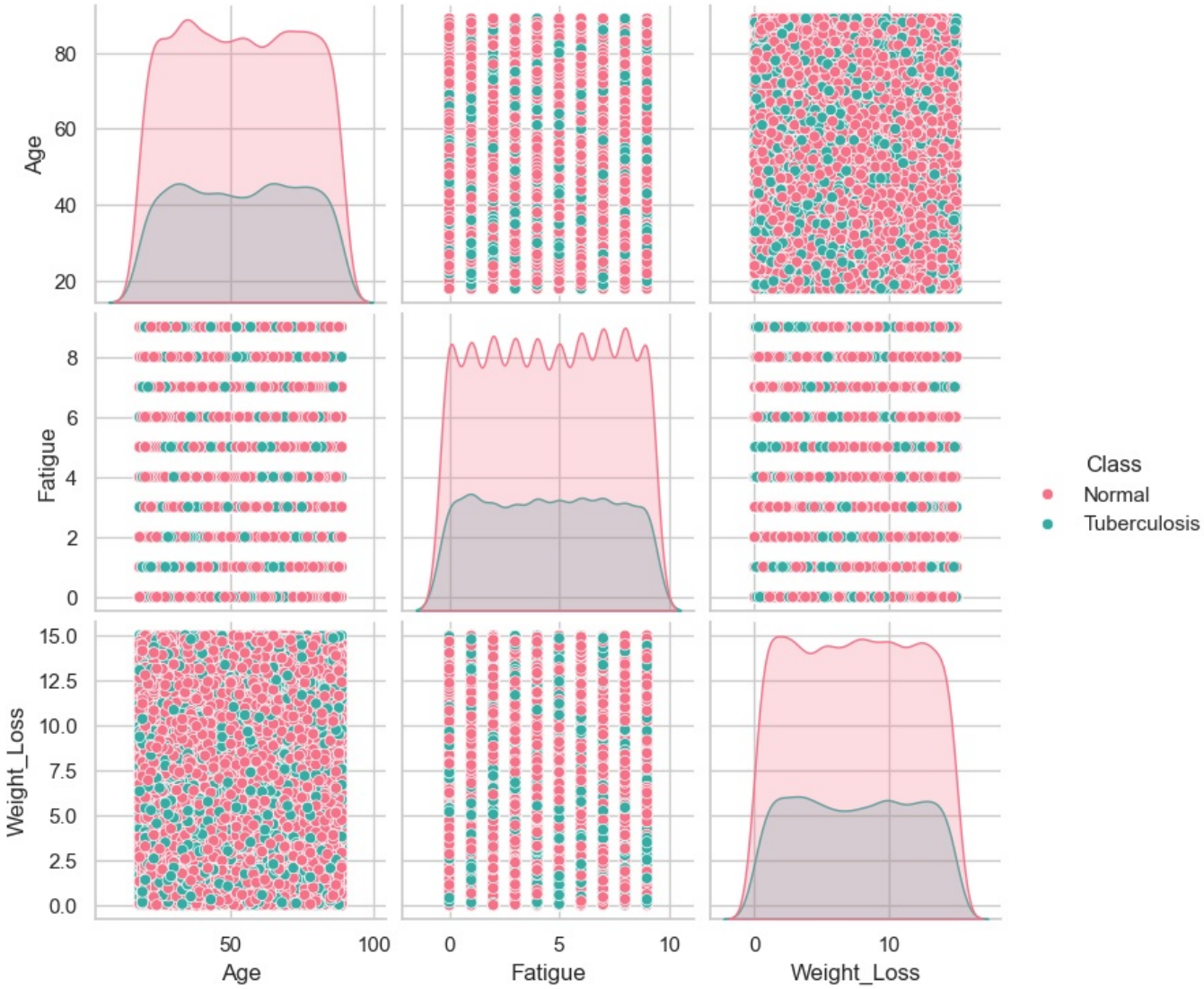

```
C:\Users\damuj\AppData\Local\Temp\ipykernel_23160\757042436.py:11: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v0.14.0. Assign the `x` variable
to `hue` and set `legend=False` for the same effect.

  sns.countplot(x=col, data=df, palette="husl")
```




```
C:\Users\damuj\AppData\Local\Temp\ipykernel_23160\757042436.py:11: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v0.14.0. Assign the `x` variable
to `hue` and set `legend=False` for the same effect.

  sns.countplot(x=col, data=df, palette="husl")
```

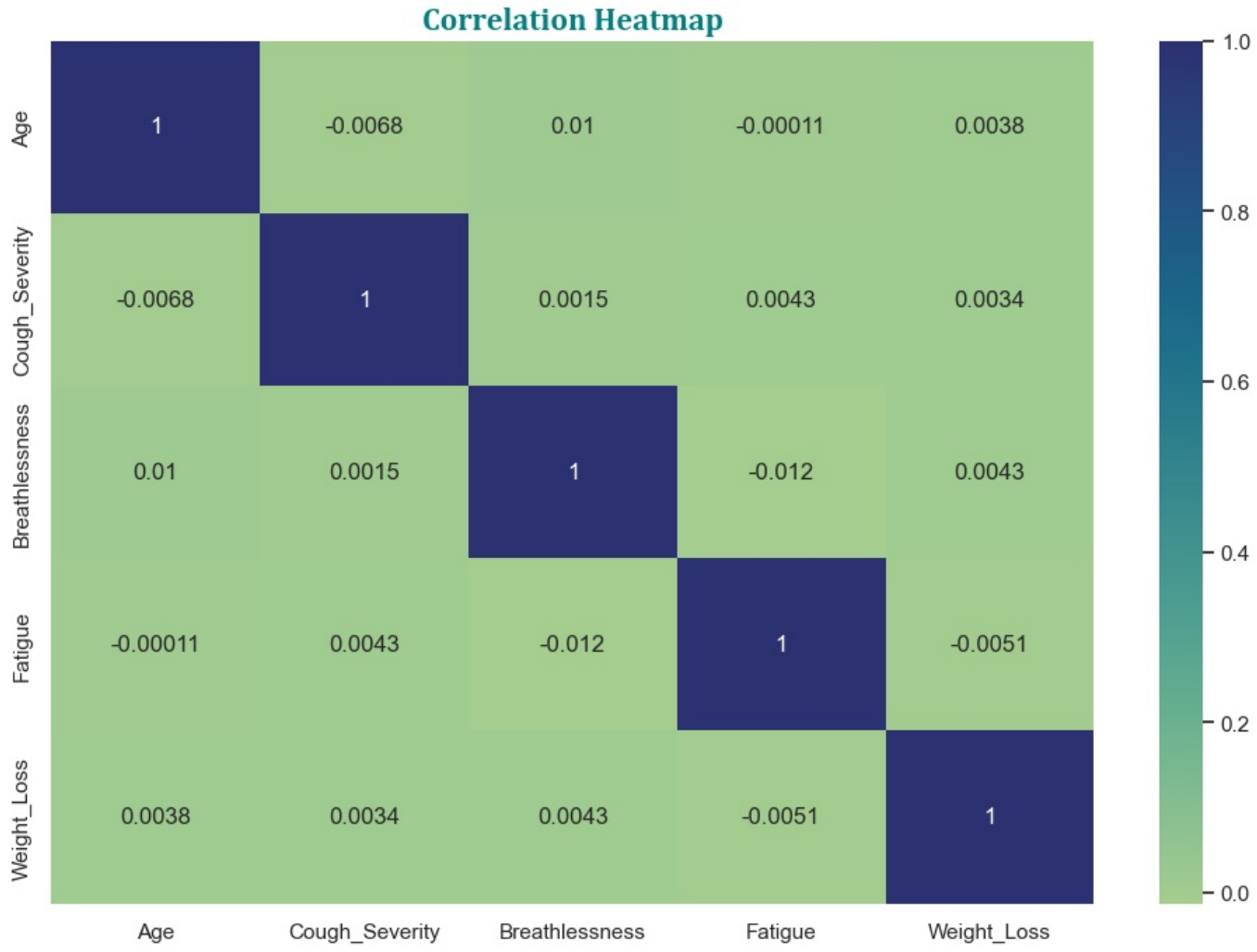
In [272…

```
# Pairplot for key numerical features by Class

sns.pairplot(df[['Age', 'Fever', 'Fatigue', 'Weight_Loss', 'Class']], hue='Class', palette='husl')
plt.show()
```



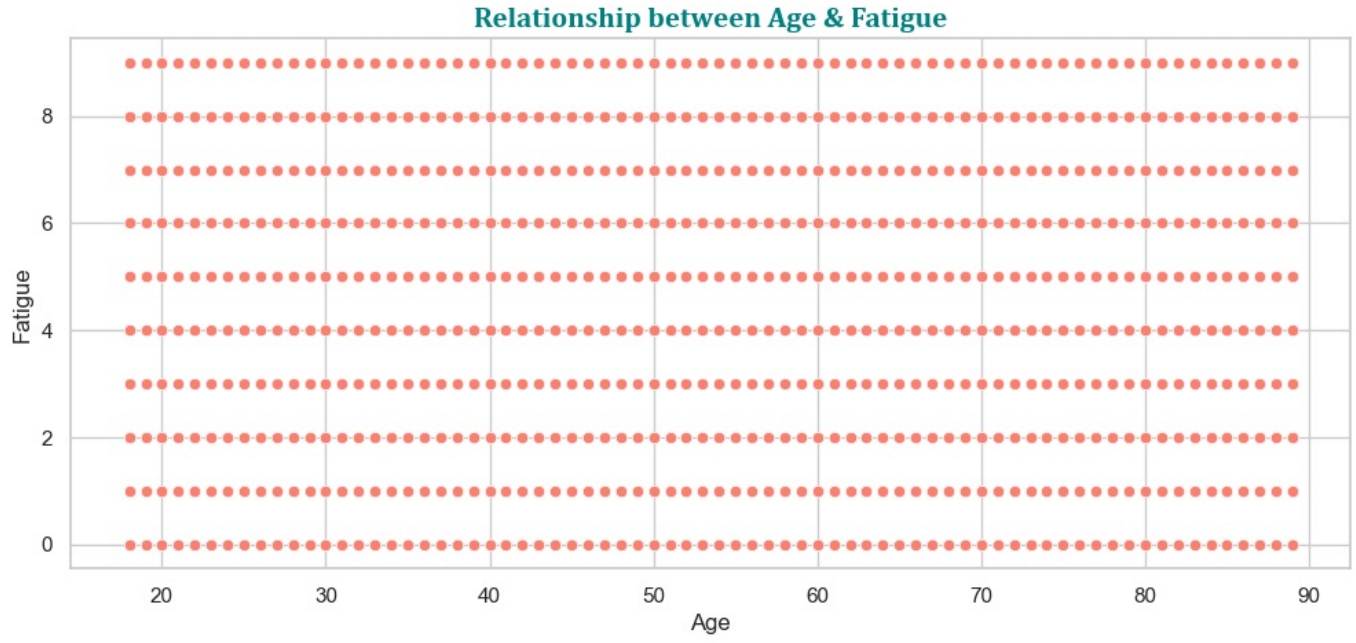
In [366…

```
# Heatmap of Correlation (numeric columns only)

# Storing data in 'numeric_df' for numerical values
numeric_df = df.select_dtypes(include=['number'])

plt.figure(figsize=(12, 8))

# Visualizing Heatmap:
sns.heatmap(numeric_df.corr(), annot=True, cmap= 'crest')
plt.title('Correlation Heatmap',
          fontname='Cambria', fontweight='bold', fontsize=16, color="teal")

plt.show()
```



In [370…

```
# Scatterplot Visualization
plt.figure(figsize=(12, 5))

sns.scatterplot(x="Age", y="Fatigue", data=df, color = 'salmon')
plt.title("Relationship between Age & Fatigue",
          fontname='Cambria', fontweight='bold', fontsize=15, color="teal")

plt.xlabel("Age")
plt.ylabel("Fatigue")

plt.show()
```

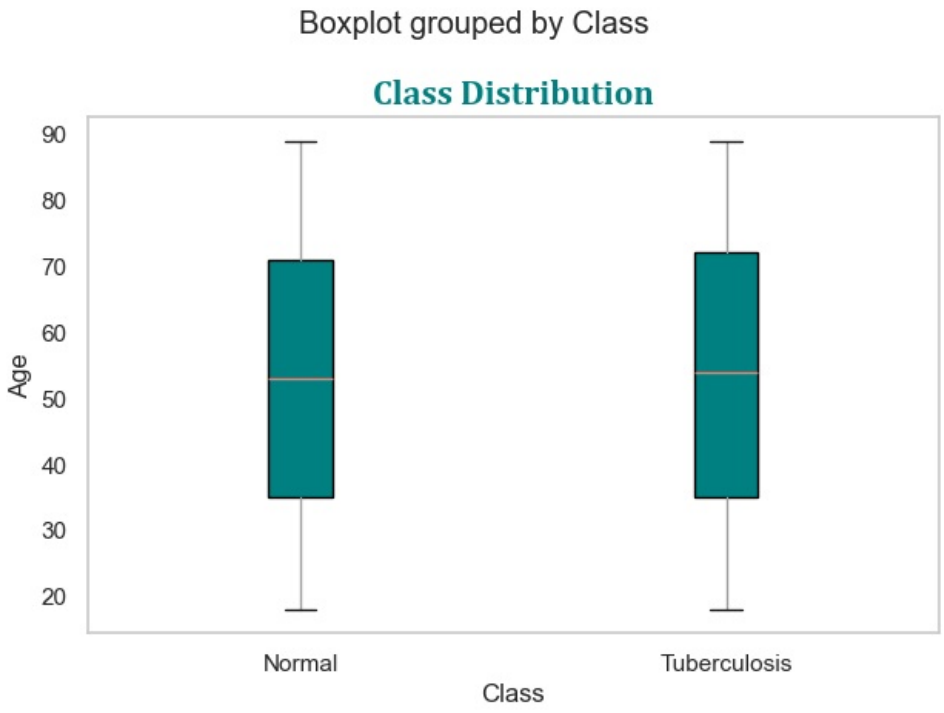


In [380…

```
# Box plots for age by TB class

plt.figure(figsize=(12, 5))

df.boxplot(column='Age', by='Class', grid=False, patch_artist=True,
```

```
            boxprops=dict(facecolor='teal'),
            medianprops=dict(color='salmon'))

plt.title('Class Distribution',
          fontname='Cambria', fontweight='bold', fontsize=16, color="teal")

plt.xlabel('Class')
plt.ylabel('Age')

plt.tight_layout()
plt.show()
```

```
<Figure size 1200x500 with 0 Axes>
```



In [ ]: