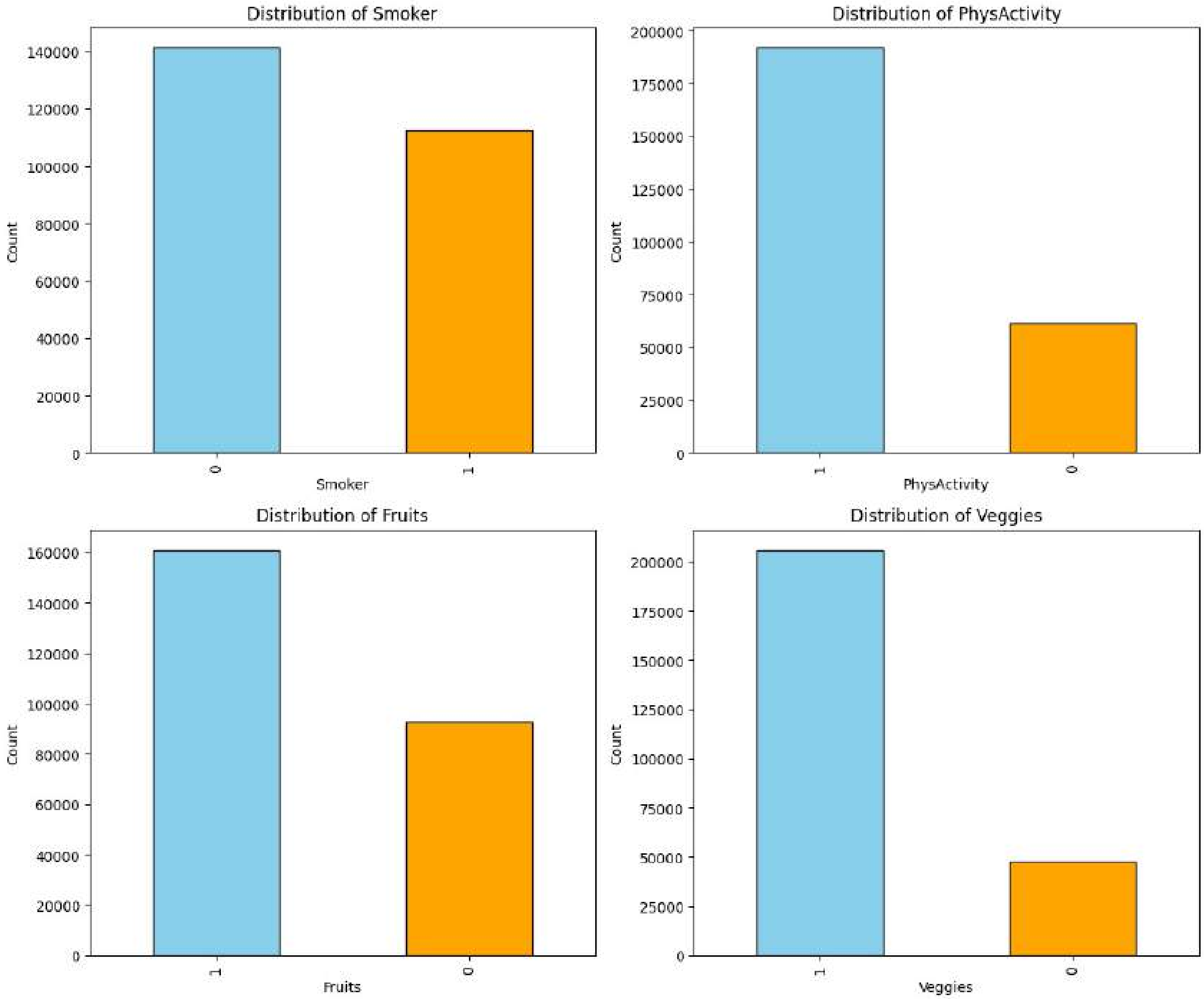
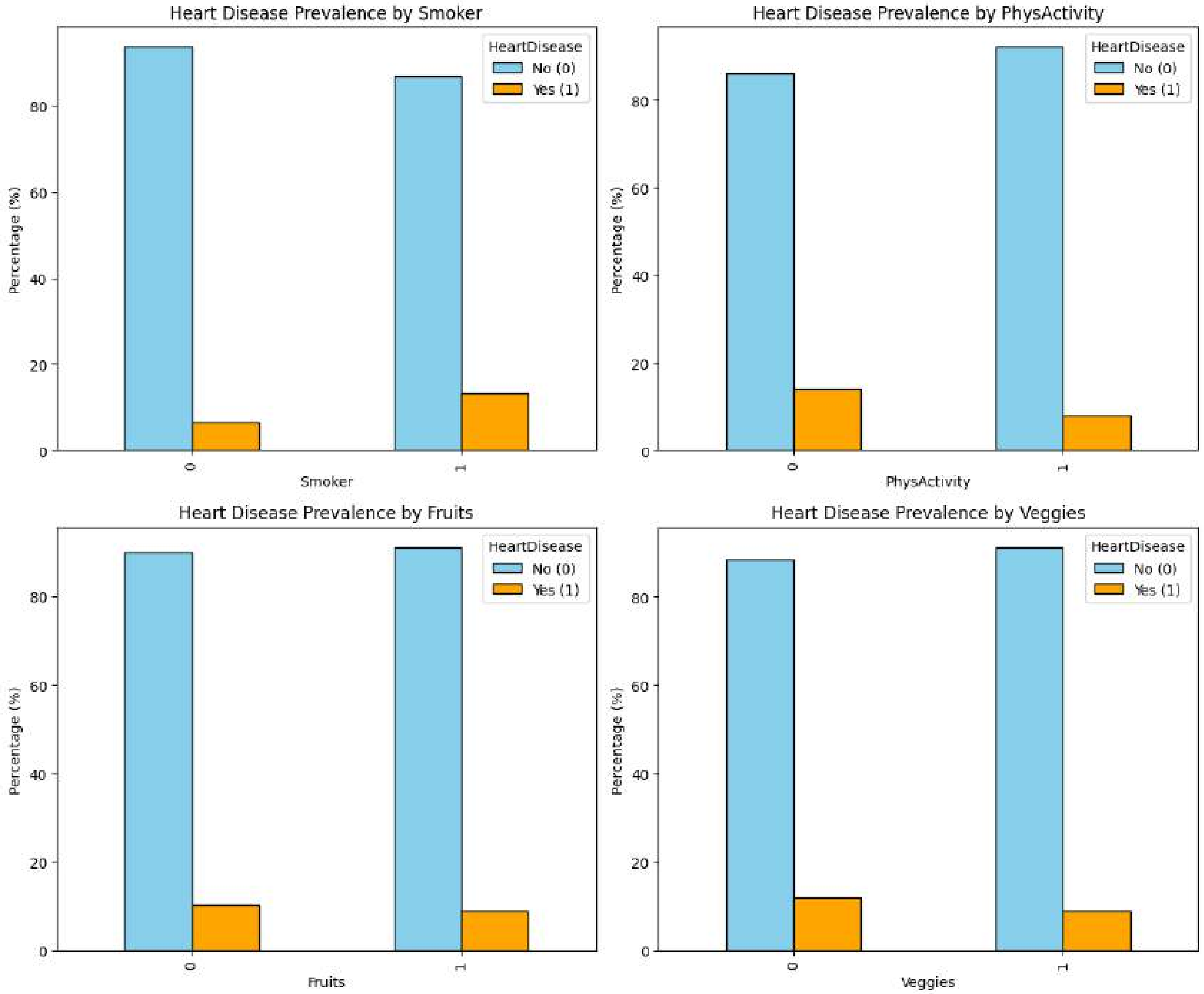
```
!pip install matplotlib seaborn pandas
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

data = pd.read_csv("healthcare_data_set.csv")

data.head()

# Categorical variables
categorical_vars = ['Smoker', 'PhysActivity', 'Fruits', 'Veggies']

# --- Bar Charts (2x2 grid) ---
fig, axes = plt.subplots(2, 2, figsize=(12, 10))

for ax, var in zip(axes.flatten(), categorical_vars):
    counts = data[var].value_counts()
    counts.plot(kind='bar', ax=ax, edgecolor='black',
color=['skyblue', 'orange'])
    ax.set_title(f"Distribution of {var}")
    ax.set_xlabel(var)
    ax.set_ylabel("Count")

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

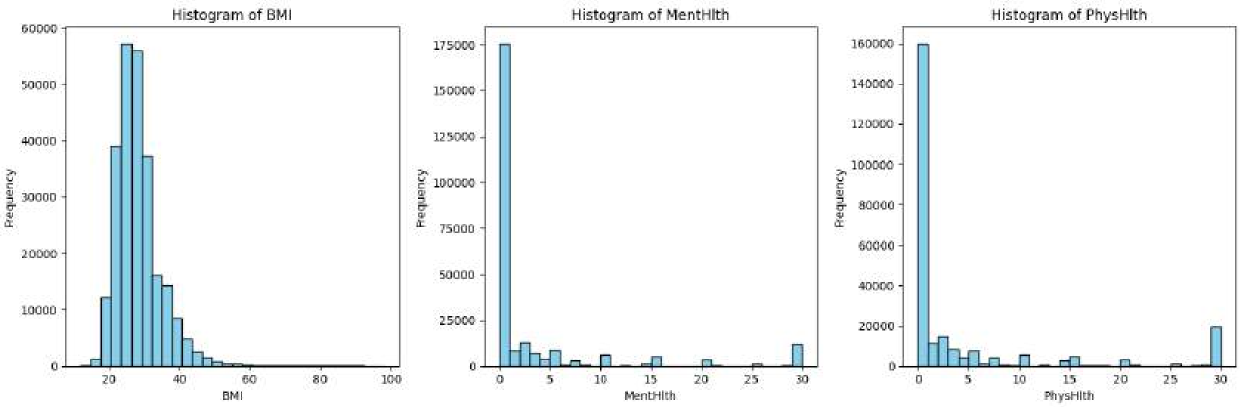


```
# categorical variables

target = 'HeartDiseaseorAttack'
categorical_vars = ['Smoker', 'PhysActivity', 'Fruits', 'Veggies']

# Plot grouped bar charts
fig, axes = plt.subplots(2, 2, figsize=(12, 10))

for ax, var in zip(axes.flatten(), categorical_vars):
    # Create crosstab: counts of HeartDisease by category
    ct = pd.crosstab(data[var], data[target], normalize='index') * 100
# percentage

    # Plot grouped bar chart
    ct.plot(
        kind='bar',
        stacked=False,
        ax=ax,
        edgecolor='black',
        color=['skyblue', 'orange']
    )

    ax.set_title(f"Heart Disease Prevalence by {var}")
    ax.set_xlabel(var)
    ax.set_ylabel("Percentage (%)")
    ax.legend(title='HeartDisease', labels=['No (0)', 'Yes (1)'])

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

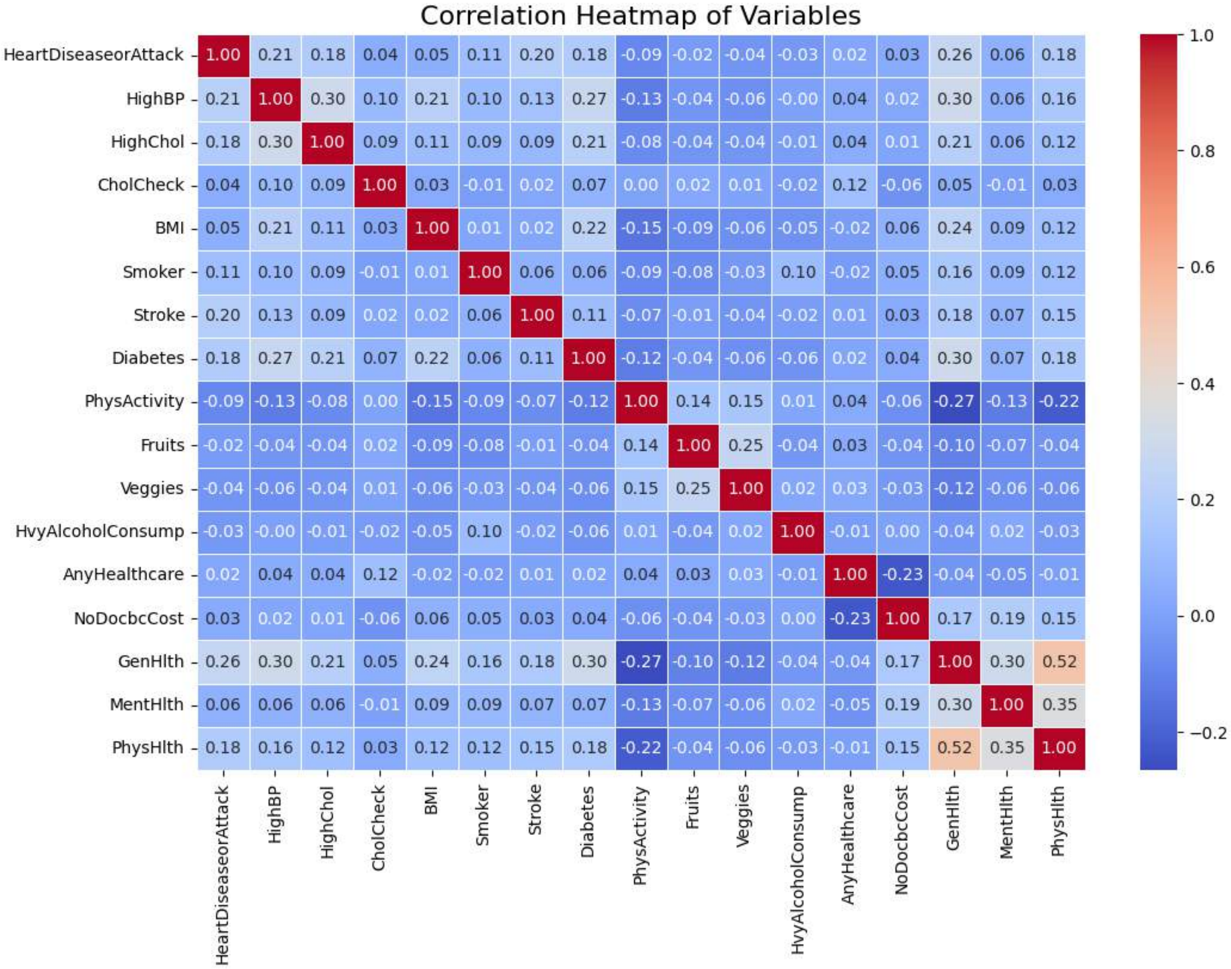


```
##########  a.Display Distributions of Continuous Variables

# Continuous variables
continuous_vars = ['BMI', 'MentHlth', 'PhysHlth']

# --- Histograms (1 row) ---
fig, axes = plt.subplots(1, 3, figsize=(15, 5))
```

```
for ax, var in zip(axes, continuous_vars):
    ax.hist(data[var].dropna(), bins=30, color='skyblue',
edgecolor='black')
    ax.set_title(f"Histogram of {var}")
    ax.set_xlabel(var)
    ax.set_ylabel("Frequency")

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



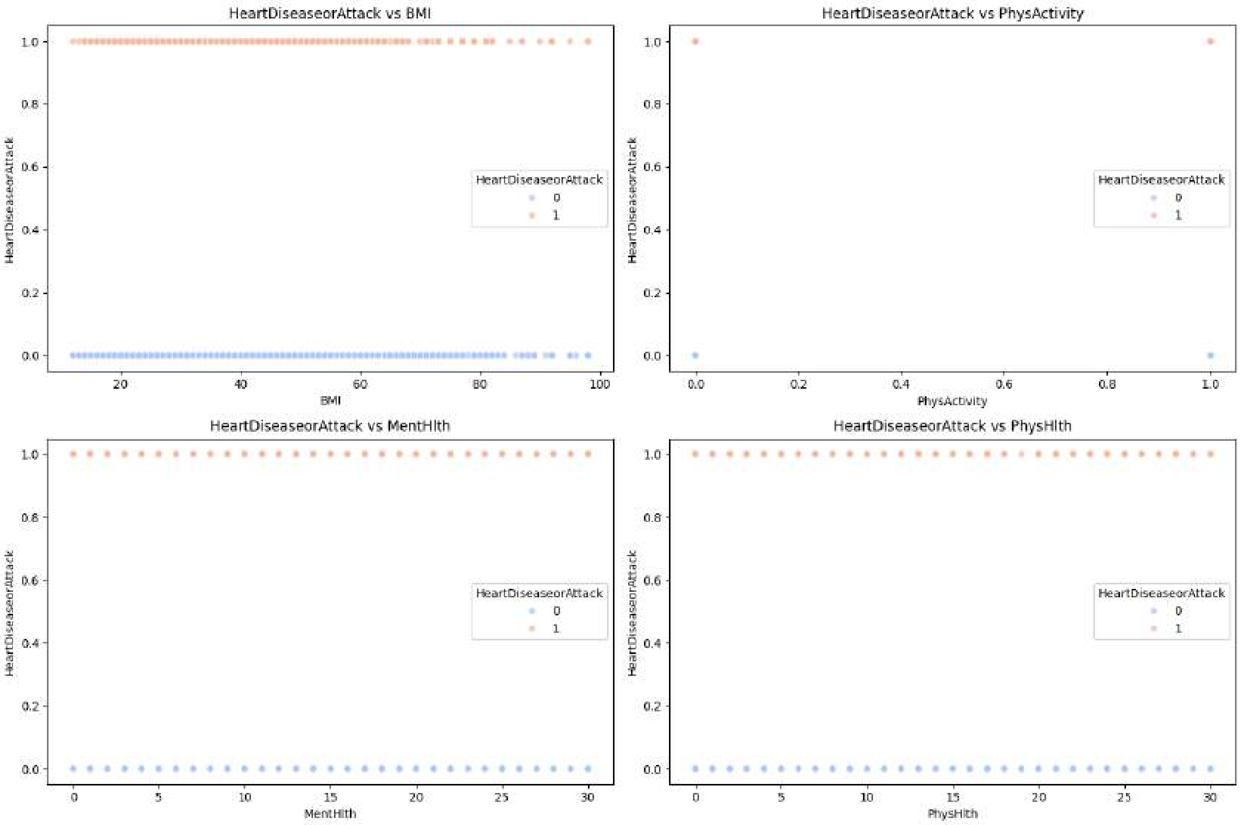
```
###########  b. Heatmaps and Correlation Plots: Explore Variable
Relationships

numeric_vars = ['HeartDiseaseorAttack', 'HighBP', 'HighChol',
'CholCheck',
                'BMI', 'Smoker', 'Stroke', 'Diabetes', 'PhysActivity',

                'Fruits', 'Veggies', 'HvyAlcoholConsump',
'AnyHealthcare',
                'NoDocbcCost', 'GenHlth', 'MentHlth', 'PhysHlth']

data_numeric = data[numeric_vars]

corr_matrix = data_numeric.corr()

plt.figure(figsize=(12, 8))
sns.heatmap(corr_matrix, annot=True, fmt=".2f", cmap="coolwarm",
linewidths=0.5)
plt.title("Correlation Heatmap of Variables", fontsize=16)
plt.show()
```



Correlation Heatmap of Variables

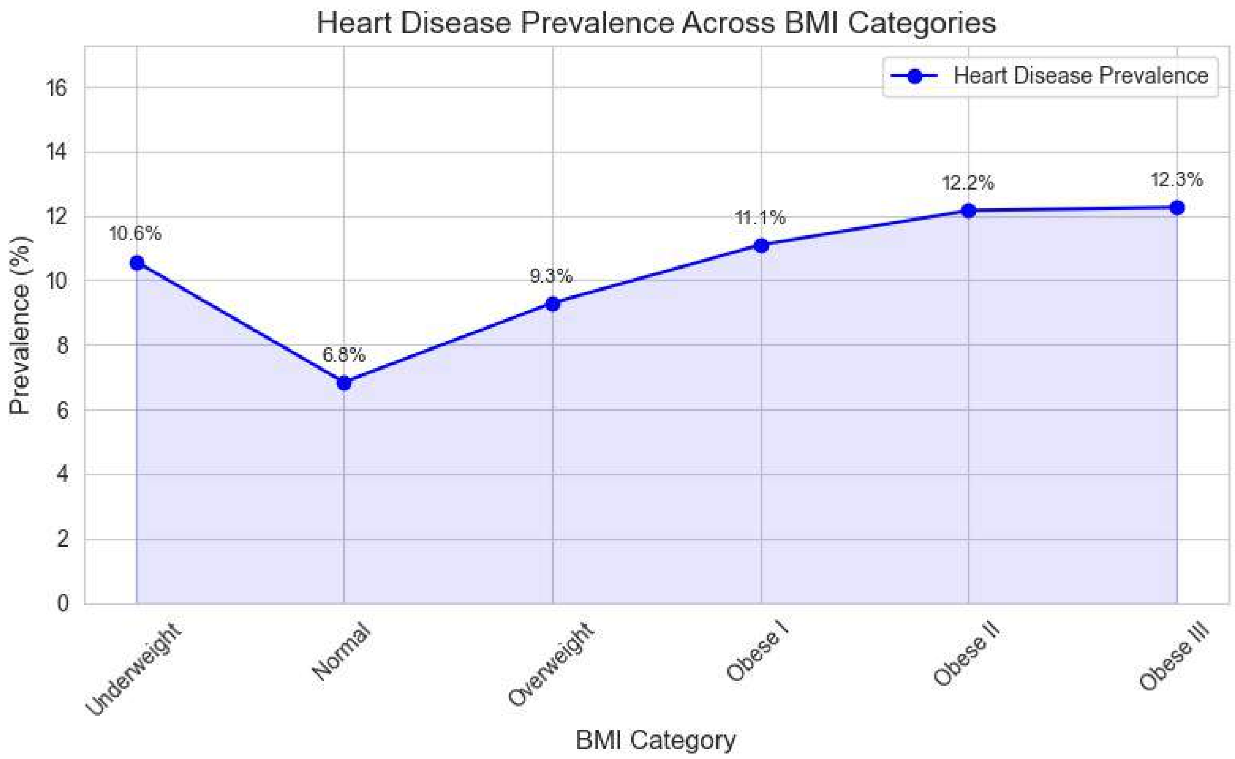
```
!pip install matplotlib seaborn pandas
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

data = pd.read_csv("healthcare_data_set.csv")

data.head()

   HeartDiseaseorAttack  HighBP  HighChol  CholCheck  BMI  Smoker
Stroke  \
0                     0       1         1          1   40       1
0
1                     0       0         0          0   25       1
0
2                     0       1         1          1   28       0
0
3                     0       1         0          1   27       0
0
4                     0       1         1          1   24       0
0

   Diabetes  PhysActivity  Fruits  ...  AnyHealthcare  NoDocbcCost
GenHlth  \
0         0             0       0  ...              1            0
5
1         0             1       0  ...              0            1
3
2         0             0       1  ...              1            1
5
3         0             1       1  ...              1            0
2
4         0             1       1  ...              1            0
2

   MentHlth  PhysHlth  DiffWalk  Sex  Age  Education  Income
0        18        15         1    0    9          4       3
1         0         0         0    0    7          6       1
2        30        30         1    0    9          4       8
3         0         0         0    0   11          3       6
4         3         0         0    0   11          5       4

[5 rows x 22 columns]

############  c. Scatter Plots: Investigate Relationships Between
Variables

risk_factors = ['BMI', 'PhysActivity', 'MentHlth', 'PhysHlth']  #
independent variables
target = 'HeartDiseaseorAttack'  # dependent variable
```

```
missing = [col for col in risk_factors + [target] if col not in
data.columns]
if missing:
    raise ValueError(f"Missing columns in dataset: {missing}")

##########Scatter plots for each risk factor vs. heart disease


plt.figure(figsize=(15, 10))

for i, var in enumerate(risk_factors):
    plt.subplot(2, 2, i+1)
    sns.scatterplot(
        data=data, x=var, y=target,
        hue=target, palette="coolwarm", alpha=0.7
    )
    plt.title(f"{target} vs {var}")
    plt.xlabel(var)
    plt.ylabel(target)

plt.tight_layout()
plt.show()

sns.pairplot(data, vars=risk_factors + [target], hue=target,
palette="coolwarm")
plt.show()
```



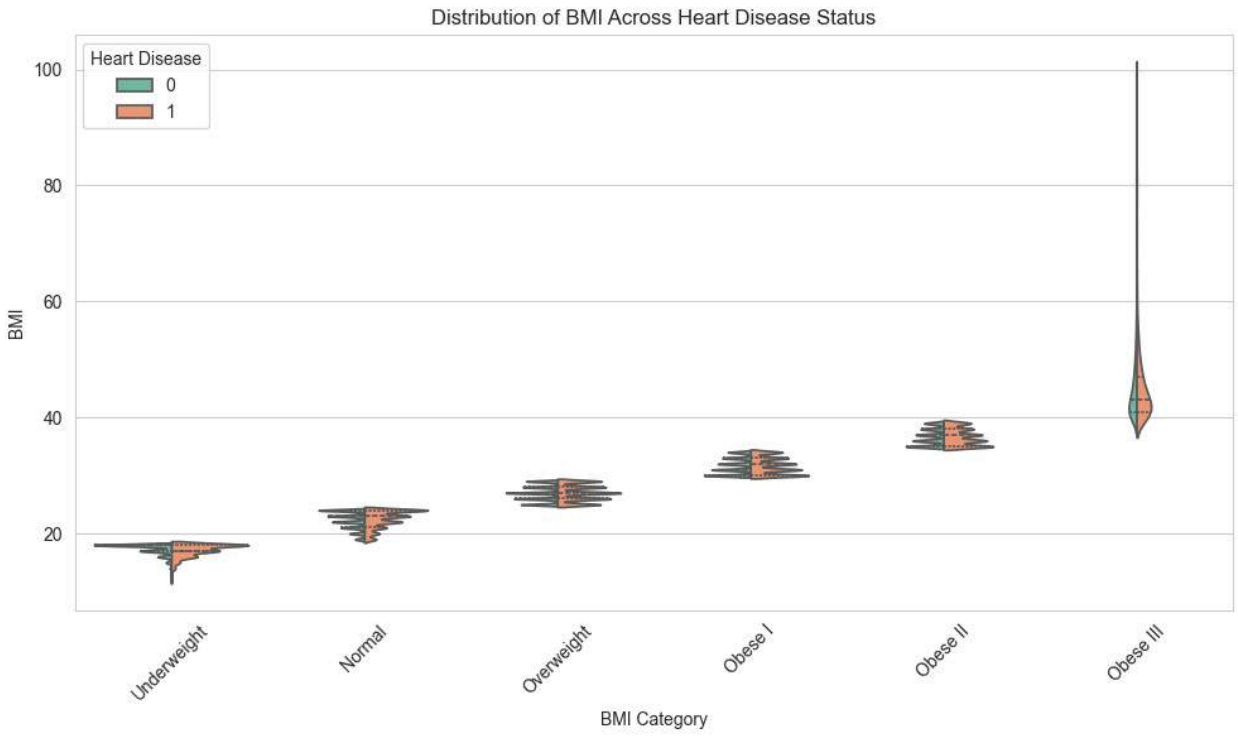
```
#######################d. Line Plots: Display trends over time or
across ordered categories.

target = 'HeartDiseaseorAttack'

# Create BMI categories
bins = [0, 18.5, 25, 30, 35, 40, 100]
labels = ['Underweight', 'Normal', 'Overweight', 'Obese I', 'Obese
II', 'Obese III']
data['BMI_Category'] = pd.cut(data['BMI'], bins=bins, labels=labels,
right=False)

# Calculate prevalence of heart disease by BMI category
prevalence = data.groupby('BMI_Category')[target].mean() * 100

# Line plot with seaborn style
plt.figure(figsize=(8, 5))
sns.set_style("whitegrid")
plt.plot(prevalence.index, prevalence.values, marker='o',
linestyle='-', color='blue', label='Heart Disease Prevalence')

# Add data labels on top of each point
for i, value in enumerate(prevalence.values):
    plt.text(i, value + 0.5, f"{value:.1f}%", ha='center',
va='bottom', fontsize=9)

# Optional: Add a shaded region for visual emphasis
plt.fill_between(prevalence.index, 0, prevalence.values, color='blue',
alpha=0.1)

# Titles and labels
plt.xticks(rotation=45)
plt.title("Heart Disease Prevalence Across BMI Categories",
fontsize=14)
plt.xlabel("BMI Category", fontsize=12)
plt.ylabel("Prevalence (%)", fontsize=12)
plt.ylim(0, prevalence.max() + 5)
plt.legend()
plt.tight_layout()
plt.show()
```

```
C:\Users\seeth\AppData\Local\Temp\ipykernel_24032\2932911419.py:9:
FutureWarning: The default of observed=False is deprecated and will be
changed to True in a future version of pandas. Pass observed=False to
retain current behavior or observed=True to adopt the future default
and silence this warning.
  prevalence = data.groupby('BMI_Category')[target].mean() * 100
```



```
####################e. Violin Plots: Combine aspects of box plots and
density plots to show the distribution of data.
```

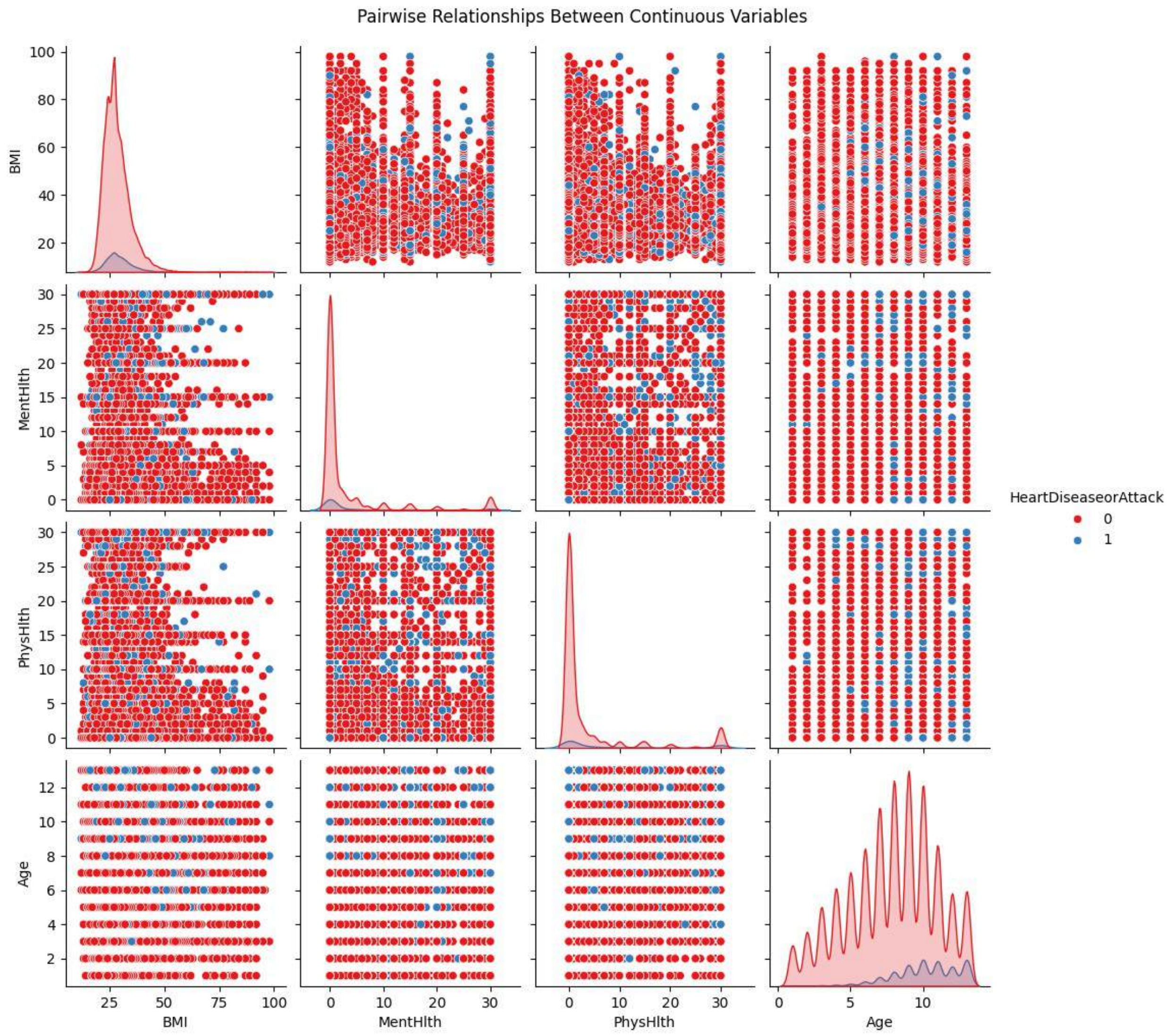
```
# Create BMI categories (ordered)
bins = [0, 18.5, 25, 30, 35, 40, 100]
labels = ['Underweight', 'Normal', 'Overweight', 'Obese I', 'Obese
II', 'Obese III']
data['BMI_Category'] = pd.cut(data['BMI'], bins=bins, labels=labels,
right=False)

# Violin plot: Distribution of BMI by Heart Disease status
plt.figure(figsize=(10, 6))
sns.violinplot(x='BMI_Category', y='BMI', hue='HeartDiseaseorAttack',
data=data,
               split=True, inner='quartile', palette='Set2')

plt.title("Distribution of BMI Across Heart Disease Status")
plt.xlabel("BMI Category")
plt.ylabel("BMI")
plt.xticks(rotation=45)
plt.legend(title='Heart Disease', loc='upper left')
plt.tight_layout()
plt.show()
```



```
################  f. Pair Plots: Plot pairwise relationships across
multiple variables
%matplotlib inline
```

```
continuous_vars = ['BMI', 'MentHlth', 'PhysHlth', 'Age']
target = 'HeartDiseaseorAttack'

# Display first few rows to check
print(data[continuous_vars + [target]].head())

# Pair plot
sns.pairplot(data[continuous_vars + [target]], hue=target,
diag_kind='kde', palette='Set1')
plt.suptitle("Pairwise Relationships Between Continuous Variables",
y=1.02)
plt.show()
```

```
   BMI  MentHlth  PhysHlth  Age  HeartDiseaseorAttack
0   40        18        15    9                     0
1   25         0         0    7                     0
2   28        30        30    9                     0
3   27         0         0   11                     0
4   24         3         0   11                     0
```



Pairwise Relationships Between Continuous Variables