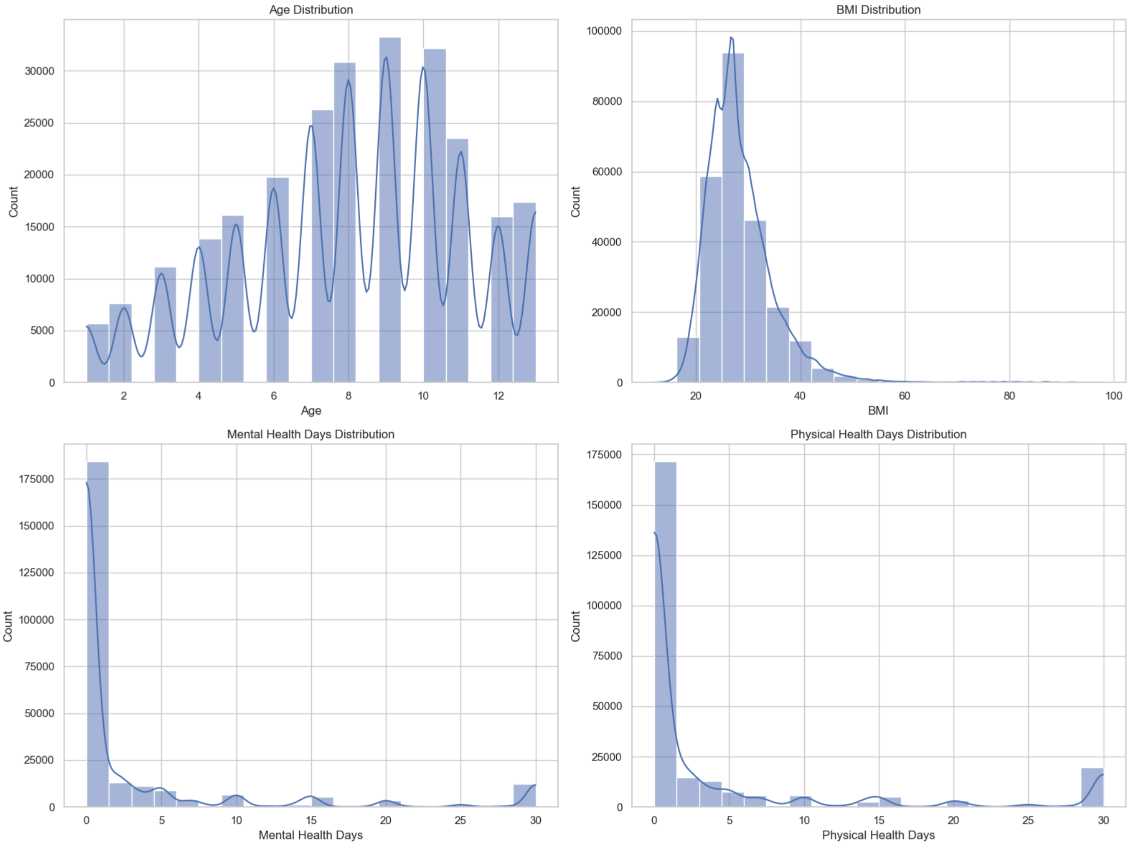
In [1]:

```
import pandas as pd

# Load the dataset
df=pd.read_csv("heart_disease_health_indicators_BRFSS2015 2 (1).csv")
print(df)

# Clean column names
df.columns = df.columns.str.strip()

# Convert categorical columns to category dtype
categorical_columns = [
    'HeartDiseaseorAttack', 'HighBP', 'HighChol', 'CholCheck', 'Smoker',
    'Stroke', 'Diabetes', 'PhysActivity', 'Fruits', 'Veggies', 'HvyAlcoholC
    'AnyHealthcare', 'NoDocbcCost', 'DiffWalk', 'Sex', 'GenHlth', 'Age',
    'Education', 'Income'
]
df[categorical_columns] = df[categorical_columns].astype('category')

# Verify data types
print(df.info())
```

```
        HeartDiseaseorAttack  HighBP  HighChol  CholCheck  BMI  Smoker  \
0                          0       1         1          1   40       1   
1                          0       0         0          0   25       1   
2                          0       1         1          1   28       0   
3                          0       1         0          1   27       0   
4                          0       1         1          1   24       0   
...                      ...     ...       ...        ...  ...     ...   
253675                     0       1         1          1   45       0   
253676                     0       1         1          1   18       0   
253677                     0       0         0          1   28       0   
253678                     0       1         0          1   23       0   
253679                     1       1         1          1   25       0   

        Stroke  Diabetes  PhysActivity  Fruits  ...  AnyHealthcare  \
0            0         0             0       0  ...              1   
1            0         0             1       0  ...              0   
2            0         0             0       1  ...              1   
3            0         0             1       1  ...              1   
4            0         0             1       1  ...              1   
...        ...       ...           ...     ...  ...            ...   
253675       0         0             0       1  ...              1   
253676       0         2             0       0  ...              1   
253677       0         0             1       1  ...              1   
253678       0         0             0       1  ...              1   
253679       0         2             1       1  ...              1   

        NoDocbcCost  GenHlth  MentHlth  PhysHlth  DiffWalk  Sex  Age  \
0                 0        5        18        15         1    0    9   
1                 1        3         0         0         0    0    7   
2                 1        5        30        30         1    0    9   
3                 0        2         0         0         0    0   11   
4                 0        2         3         0         0    0   11   
...             ...      ...       ...       ...       ...  ...  ...   
253675            0        3         0         5         0    1    5   
253676            0        4         0         0         1    0   11   
253677            0        1         0         0         0    0    2   
253678            0        3         0         0         0    1    7   
253679            0        2         0         0         0    0    9   

        Education  Income  
0               4       3  
1               6       1  
2               4       8  
3               3       6  
4               5       4  
...           ...     ...  
253675          6       7  
253676          2       4  
253677          5       2  
253678          5       1  
253679          6       2  

[253680 rows x 22 columns]
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 253680 entries, 0 to 253679
Data columns (total 22 columns):
 #   Column                Non-Null Count   Dtype   
---  ------                --------------   -----   
 0   HeartDiseaseorAttack  253680 non-null  category
 1   HighBP                253680 non-null  category
 2   HighChol              253680 non-null  category
 3   CholCheck             253680 non-null  category
 4   BMI                   253680 non-null  int64
 5   Smoker                253680 non-null  category
 6   Stroke                253680 non-null  category
 7   Diabetes              253680 non-null  category
 8   PhysActivity          253680 non-null  category
 9   Fruits                253680 non-null  category
 10  Veggies               253680 non-null  category
 11  HvyAlcoholConsump     253680 non-null  category
 12  AnyHealthcare         253680 non-null  category
 13  NoDocbcCost           253680 non-null  category
 14  GenHlth               253680 non-null  category
 15  MentHlth              253680 non-null  int64
 16  PhysHlth              253680 non-null  int64
 17  DiffWalk              253680 non-null  category
 18  Sex                   253680 non-null  category
 19  Age                   253680 non-null  category
 20  Education             253680 non-null  category
 21  Income                253680 non-null  category
dtypes: category(19), int64(3)
memory usage: 10.4 MB
None
```

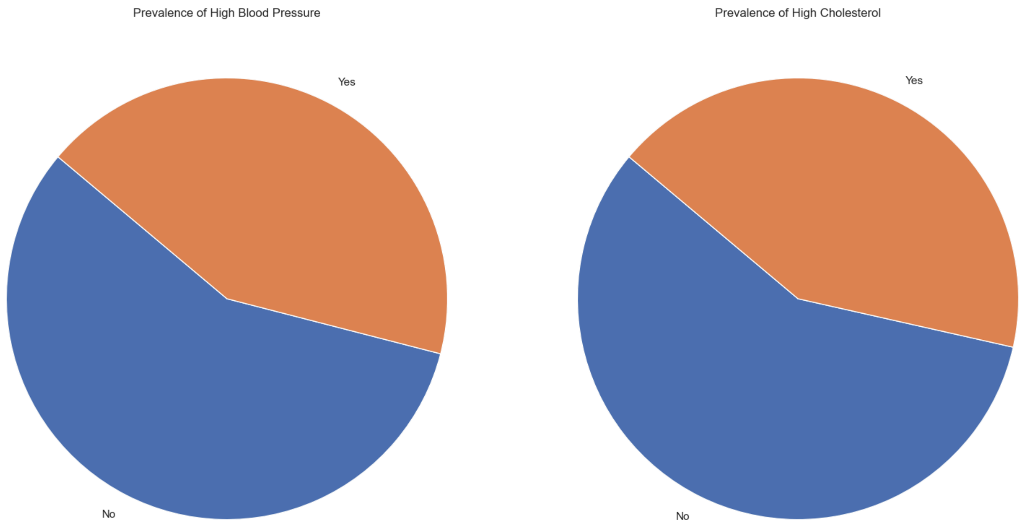
In [10]:

```
#a. Examine Distributions of Individual Variables:
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns


# plotting area
fig, axes = plt.subplots(2, 2, figsize=(16, 12))

# Plot histogram for Age
sns.histplot(df['Age'], bins=20, kde=True, ax=axes[0, 0])
axes[0, 0].set_title('Age Distribution')
axes[0, 0].set_xlabel('Age')
axes[0, 0].set_ylabel('Count')

# Plot histogram for BMI
sns.histplot(df['BMI'], bins=20, kde=True, ax=axes[0, 1])
axes[0, 1].set_title('BMI Distribution')
axes[0, 1].set_xlabel('BMI')
axes[0, 1].set_ylabel('Count')

# Plot histogram for MentHlth
sns.histplot(df['MentHlth'], bins=20, kde=True, ax=axes[1, 0])
axes[1, 0].set_title('Mental Health Days Distribution')
axes[1, 0].set_xlabel('Mental Health Days')
axes[1, 0].set_ylabel('Count')

# Plot histogram for PhysHlth
sns.histplot(df['PhysHlth'], bins=20, kde=True, ax=axes[1, 1])
axes[1, 1].set_title('Physical Health Days Distribution')
axes[1, 1].set_xlabel('Physical Health Days')
axes[1, 1].set_ylabel('Count')

# Adjust layout
plt.tight_layout()
plt.show()
```


Age Distribution
Count
0
5000
10000
15000
20000
25000
30000
2
4
6
8
10
12
Age
BMI Distribution
Count
0
20000
40000
60000
80000
100000
20
40
60
80
100
BMI
Mental Health Days Distribution
Count
0
25000
50000
75000
100000
125000
150000
175000
0
5
10
15
20
25
30
Mental Health Days
Physical Health Days Distribution
Count
0
25000
50000
75000
100000
125000
150000
175000
0
5
10
15
20
25
30
Physical Health Days

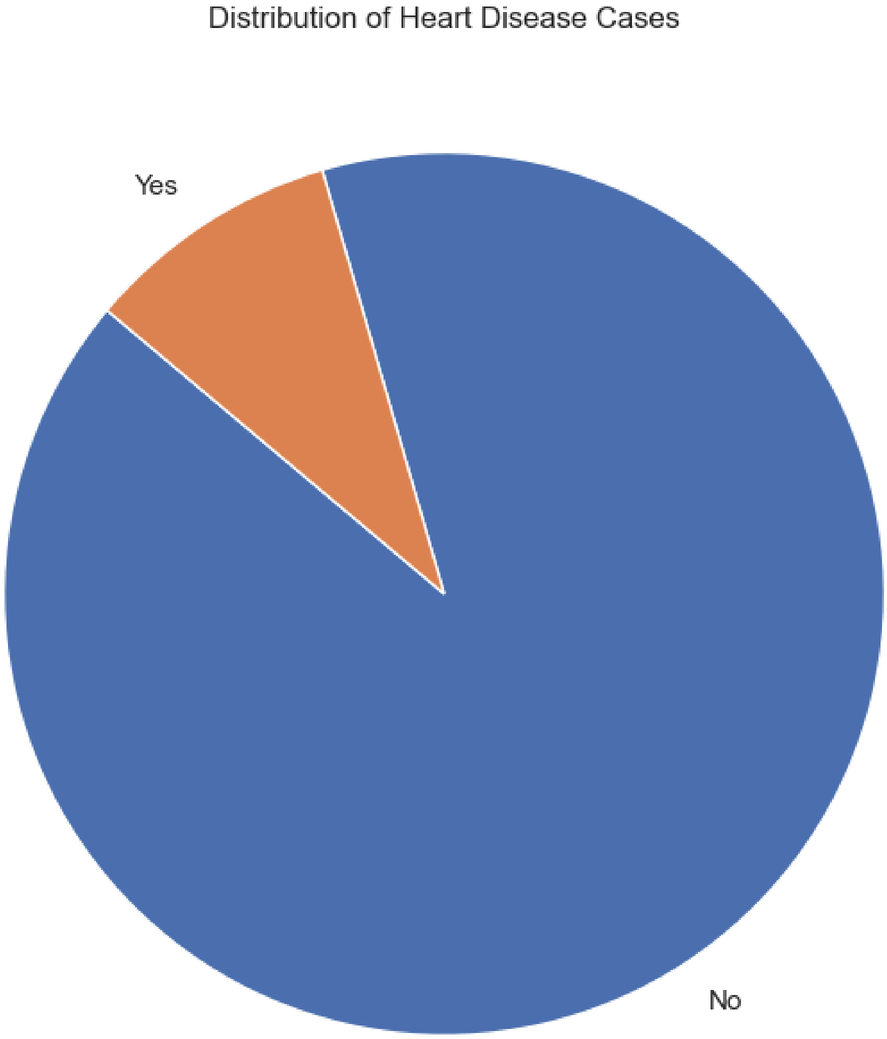
In [8]:

```
#b. Investigate Prevalence of Health Conditions:
import pandas as pd
import matplotlib.pyplot as plt


# Plotting the prevalence of health conditions using pie charts
fig, axes = plt.subplots(1, 2, figsize=(16, 8))

# pie chart for High Blood Pressure
high_bp_counts = df['HighBP'].value_counts()
axes[0].pie(high_bp_counts, labels=['No', 'Yes'], startangle=140)
axes[0].set_title('Prevalence of High Blood Pressure')

# pie chart for High Cholesterol
high_chol_counts = df['HighChol'].value_counts()
axes[1].pie(high_chol_counts, labels=['No', 'Yes'], startangle=140)
axes[1].set_title('Prevalence of High Cholesterol')

# Adjust layout
plt.tight_layout()
plt.show()
```



In [9]:

```
#c. Analyze Distribution of Heart Disease (Target Variable):
import pandas as pd
import matplotlib.pyplot as plt


# pie chart for Heart Disease
heart_disease_counts = df['HeartDiseaseorAttack'].value_counts()
plt.figure(figsize=(8, 8))
plt.pie(heart_disease_counts, labels=['No', 'Yes'], startangle=140)
plt.title('Distribution of Heart Disease Cases')
plt.show()
```

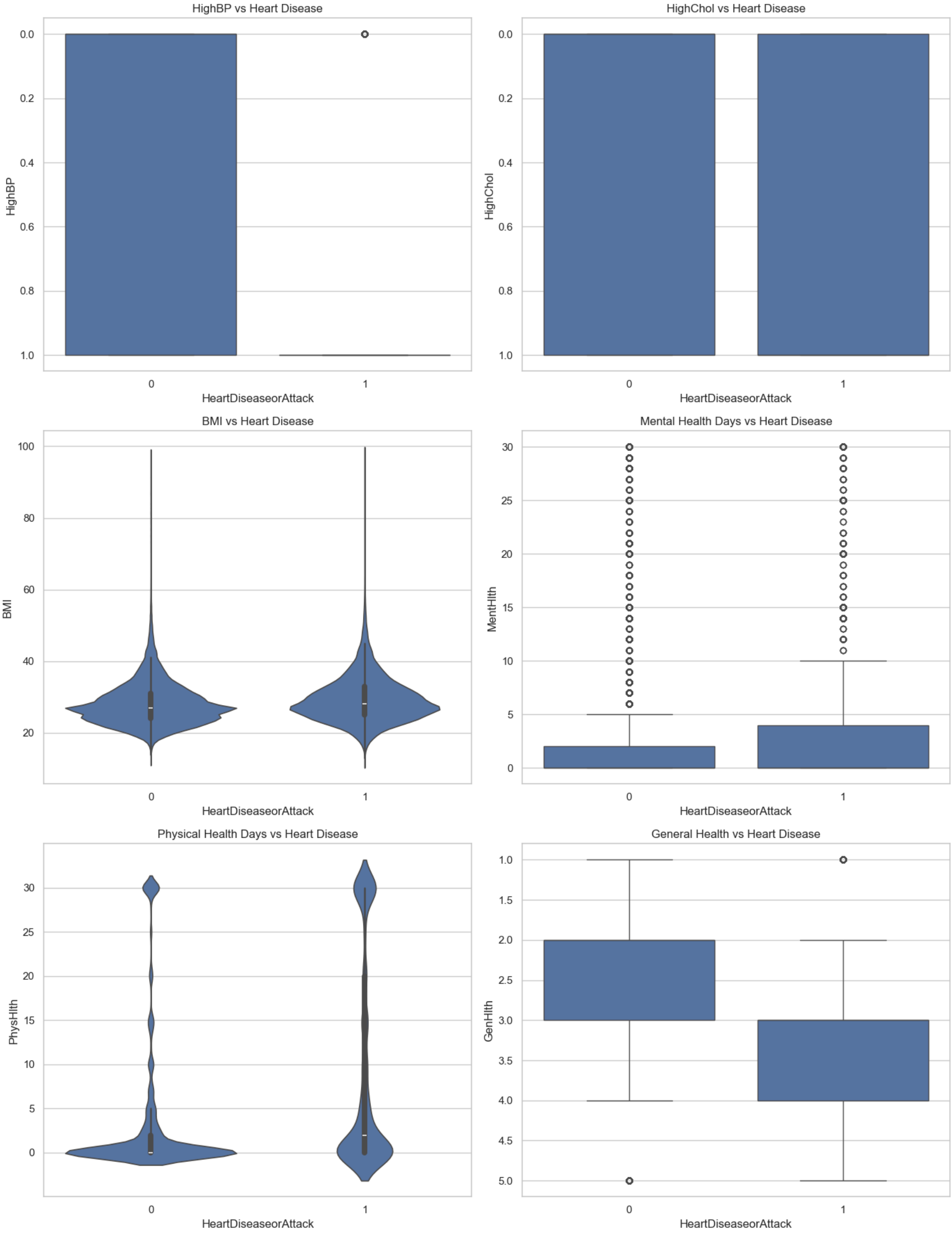


In [ ]:

```
#Bivariate analysis
```

In [6]:

```
#Explore Relationships Heart Disease:
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt

# Load the dataset
df = pd.read_csv("heart_disease_health_indicators_BRFSS2015 2 (1).csv")

# Convert categorical columns to 'category' type
categorical_columns = [
    'HeartDiseaseorAttack', 'HighBP', 'HighChol', 'CholCheck', 'Smoker',
    'Stroke', 'Diabetes', 'PhysActivity', 'Fruits', 'Veggies', 'HvyAlcoholC
    'AnyHealthcare', 'NoDocbcCost', 'DiffWalk', 'Sex', 'GenHlth', 'Age',
    'Education', 'Income'
]
df[categorical_columns] = df[categorical_columns].astype('category')

# Visualize relationships with Heart Disease
fig, axs = plt.subplots(3, 2, figsize=(14, 18))

# HighBP vs Heart Disease
sns.boxplot(x='HeartDiseaseorAttack', y='HighBP', data=df, ax=axs[0, 0])
axs[0, 0].set_title('HighBP vs Heart Disease')

# HighChol vs Heart Disease
sns.boxplot(x='HeartDiseaseorAttack', y='HighChol', data=df, ax=axs[0, 1])
axs[0, 1].set_title('HighChol vs Heart Disease')

# BMI vs Heart Disease
sns.violinplot(x='HeartDiseaseorAttack', y='BMI', data=df, ax=axs[1, 0])
axs[1, 0].set_title('BMI vs Heart Disease')

# MentHlth vs Heart Disease
sns.boxplot(x='HeartDiseaseorAttack', y='MentHlth', data=df, ax=axs[1, 1])
axs[1, 1].set_title('Mental Health Days vs Heart Disease')

# PhysHlth vs Heart Disease
sns.violinplot(x='HeartDiseaseorAttack', y='PhysHlth', data=df, ax=axs[2, 0
axs[2, 0].set_title('Physical Health Days vs Heart Disease')

# GenHlth vs Heart Disease
sns.boxplot(x='HeartDiseaseorAttack', y='GenHlth', data=df, ax=axs[2, 1])
axs[2, 1].set_title('General Health vs Heart Disease')

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



HighBP vs Heart Disease

HighChol vs Heart Disease

BMI vs Heart Disease

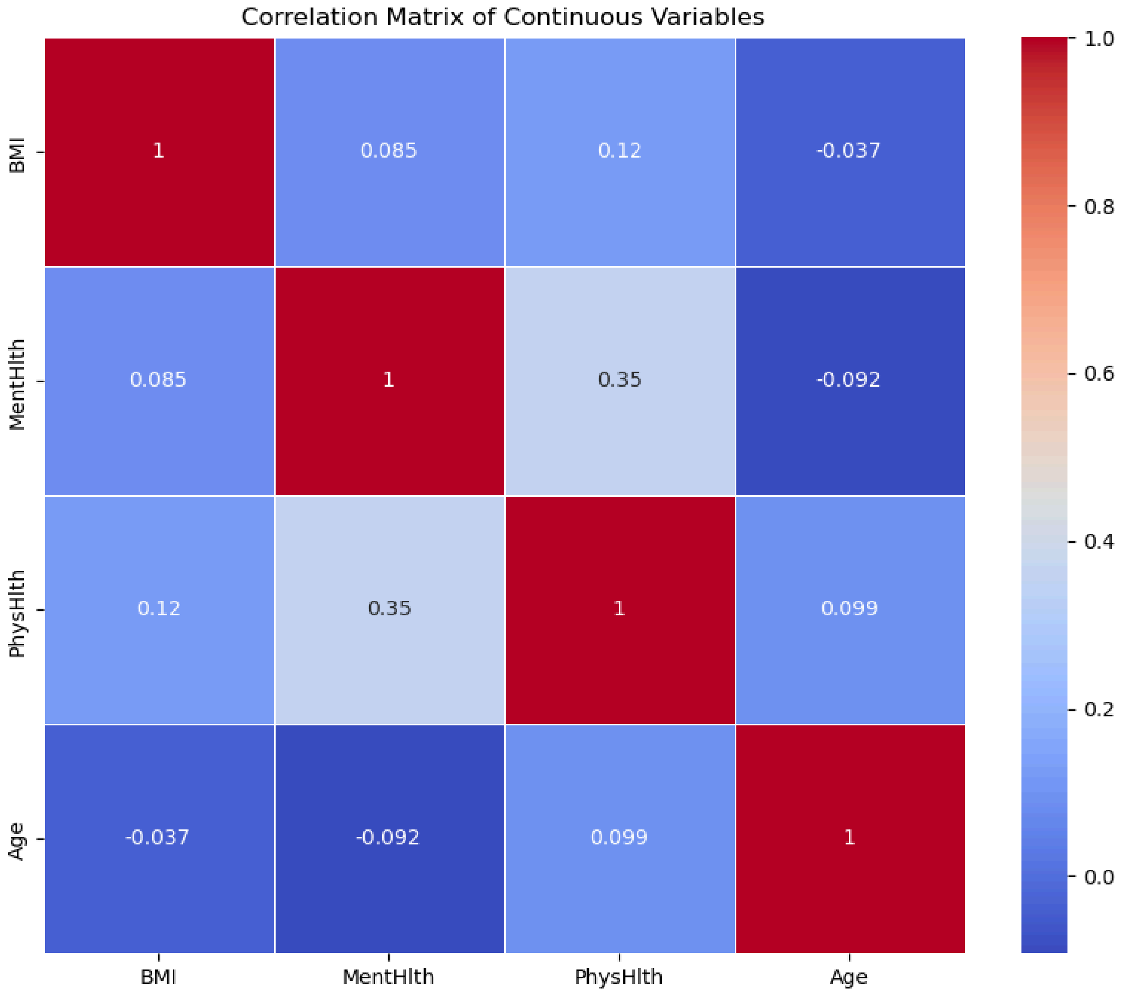
Mental Health Days vs Heart Disease

Physical Health Days vs Heart Disease

General Health vs Heart Disease

In [6]:

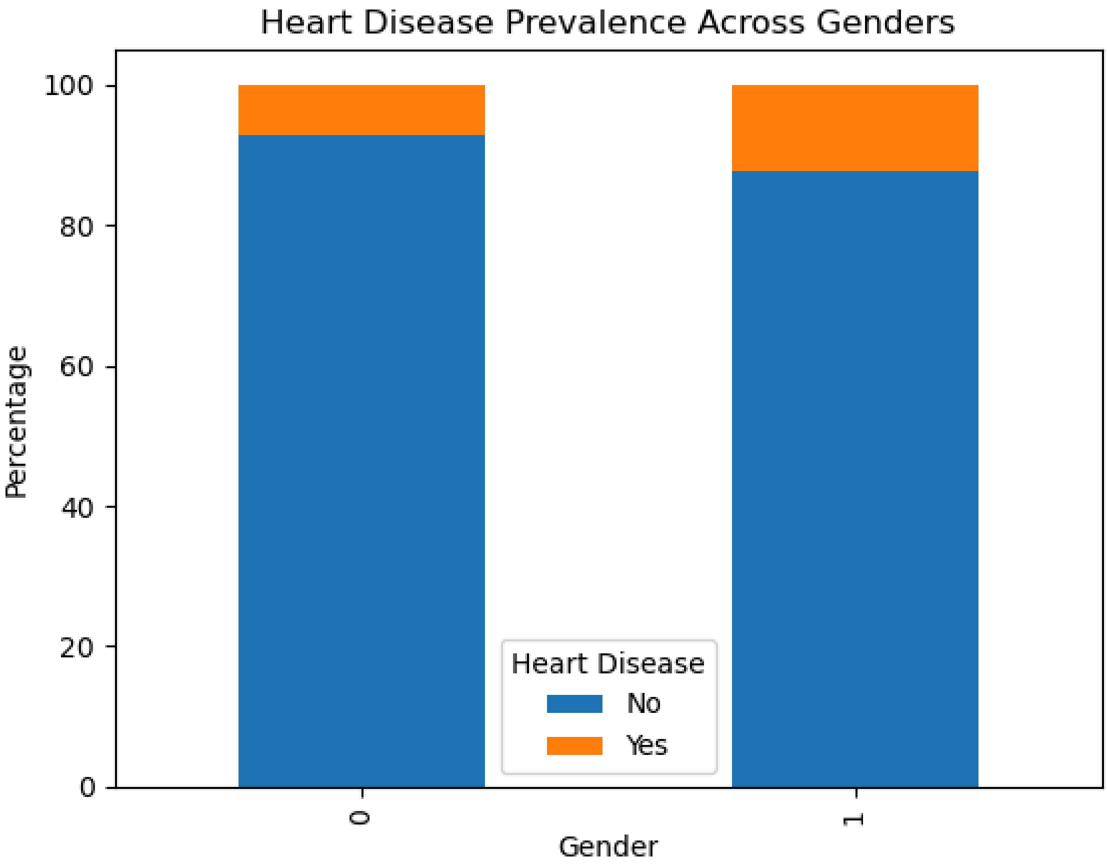
```
#b. Visualize Correlations Between Variables:
import matplotlib.pyplot as plt
import seaborn as sns

# Select continuous variables
continuous_vars = ['BMI', 'MentHlth', 'PhysHlth', 'Age']

# Calculate the correlation matrix
corr_matrix = df[continuous_vars].corr()

# Plot the heatmap
plt.figure(figsize=(10, 8))
sns.heatmap(corr_matrix, annot=True, cmap='coolwarm', linewidths=0.5)
plt.title('Correlation Matrix of Continuous Variables')
plt.show()
```



In [12]:

```
#c. Compare Heart Disease Across Demographic Groups:
# Calculate the percentage of individuals with and without heart disease wi
gender_heart_disease = df.groupby('Sex')['HeartDiseaseorAttack'].value_cour

# Bar plot for Gender vs Heart Disease
plt.figure(figsize=(12, 6))
gender_heart_disease.plot(kind='bar', stacked=True)
plt.title('Heart Disease Prevalence Across Genders')
plt.xlabel('Gender')
plt.ylabel('Percentage')
plt.legend(title='Heart Disease', labels=['No', 'Yes'])
plt.show()
```

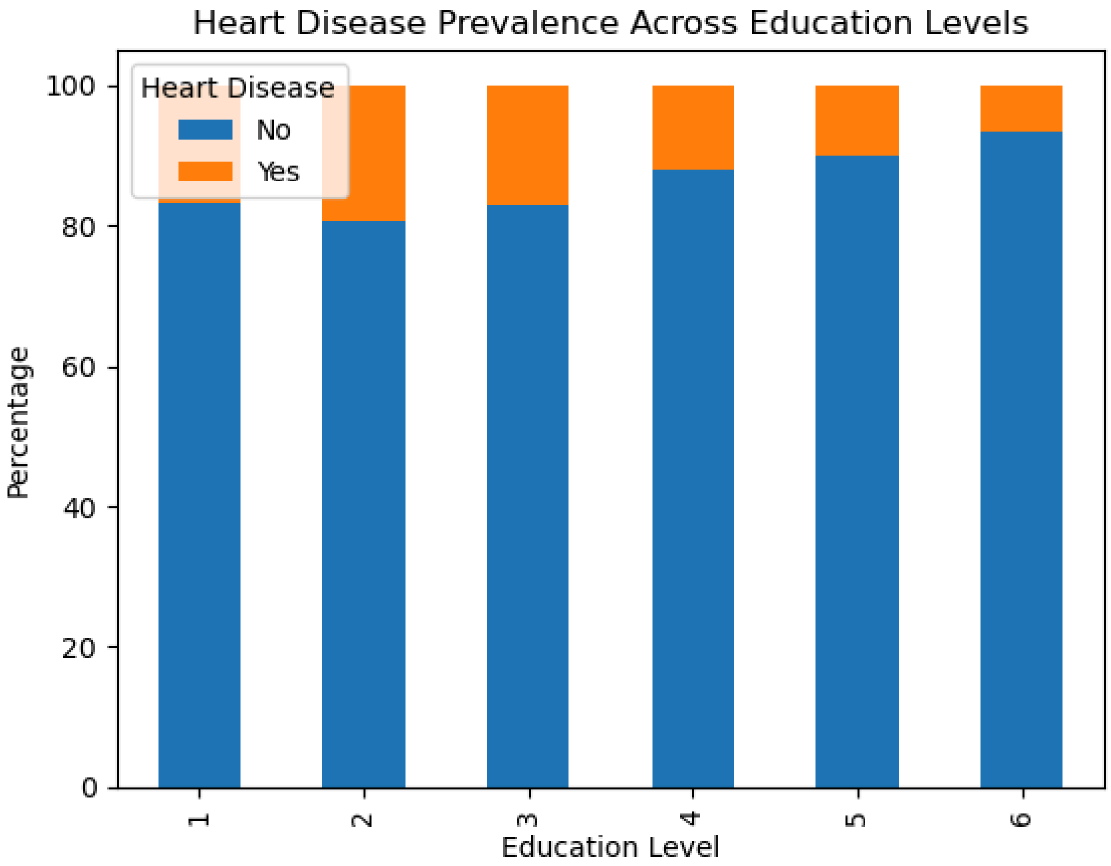
```
<Figure size 1200x600 with 0 Axes>
```



In [13]:

```
# Calculate the percentage of individuals with and without heart disease wi
education_heart_disease = df.groupby('Education')['HeartDiseaseorAttack'].v

# Bar plot for Education vs Heart Disease
plt.figure(figsize=(12, 6))
education_heart_disease.plot(kind='bar', stacked=True)
plt.title('Heart Disease Prevalence Across Education Levels')
plt.xlabel('Education Level')
plt.ylabel('Percentage')
plt.legend(title='Heart Disease', labels=['No', 'Yes'])
plt.show()
```

```
<Figure size 1200x600 with 0 Axes>
```



In [ ]:

```
```