In [2]:
```
# Importing Libraries
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

# Importing data and reviewing it

In [3]:
```
df = pd.read_csv(r"C:\Users\arnip\Downloads\heart_disease_health_indicators.csv"
df.head()
```

Out[3]:

| | HeartDiseaseorAttack | HighBP | HighChol | CholCheck | BMI | Smoker | Stroke | Diabete |
|---|---|---|---|---|---|---|---|---|
| **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 | |

5 rows × 22 columns

In [3]:
```
df.tail()
```

Out[3]:

| | HeartDiseaseorAttack | HighBP | HighChol | CholCheck | BMI | Smoker | Stroke | D |
|---|---|---|---|---|---|---|---|---|
| **253675** | 0 | 1 | 1 | 1 | 45 | 0 | 0 | |
| **253676** | 0 | 1 | 1 | 1 | 18 | 0 | 0 | |
| **253677** | 0 | 0 | 0 | 1 | 28 | 0 | 0 | |
| **253678** | 0 | 1 | 0 | 1 | 23 | 0 | 0 | |
| **253679** | 1 | 1 | 1 | 1 | 25 | 0 | 0 | |

5 rows × 22 columns

In [4]:
```
# To identify total number of rows and column
df.shape
```

Out[4]:
```
(253680, 22)
```

In [5]:
```
df.info()
```

```
<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  int64
 1   HighBP                253680 non-null  int64
 2   HighChol              253680 non-null  int64
 3   CholCheck             253680 non-null  int64
 4   BMI                   253680 non-null  int64
 5   Smoker                253680 non-null  int64
 6   Stroke                253680 non-null  int64
 7   Diabetes              253680 non-null  int64
 8   PhysActivity          253680 non-null  int64
 9   Fruits                253680 non-null  int64
 10  Veggies               253680 non-null  int64
 11  HvyAlcoholConsump     253680 non-null  int64
 12  AnyHealthcare         253680 non-null  int64
 13  NoDocbcCost           253680 non-null  int64
 14  GenHlth               253680 non-null  int64
 15  MentHlth              253680 non-null  int64
 16  PhysHlth              253680 non-null  int64
 17  DiffWalk              253680 non-null  int64
 18  Sex                   253680 non-null  int64
 19  Age                   253680 non-null  int64
 20  Education             253680 non-null  int64
 21  Income                253680 non-null  int64
dtypes: int64(22)
memory usage: 42.6 MB
```

In [6]:
```
df.describe()
```

Out[6]:

| | HeartDiseaseorAttack | HighBP | HighChol | CholCheck | BM |
|---|---|---|---|---|---|
| **count** | 253680.000000 | 253680.000000 | 253680.000000 | 253680.000000 | 253680.00000 |
| **mean** | 0.094186 | 0.429001 | 0.424121 | 0.962670 | 28.38236 |
| **std** | 0.292087 | 0.494934 | 0.494210 | 0.189571 | 6.60869 |
| **min** | 0.000000 | 0.000000 | 0.000000 | 0.000000 | 12.00000 |
| **25%** | 0.000000 | 0.000000 | 0.000000 | 1.000000 | 24.00000 |
| **50%** | 0.000000 | 0.000000 | 0.000000 | 1.000000 | 27.00000 |
| **75%** | 0.000000 | 1.000000 | 1.000000 | 1.000000 | 31.00000 |
| **max** | 1.000000 | 1.000000 | 1.000000 | 1.000000 | 98.00000 |

8 rows × 22 columns

# Handling missing value and inconsistency

In [7]:
```
# To check missing values
df.isnull().sum()
```

Out[7]:
```
HeartDiseaseorAttack    0
HighBP                  0
HighChol                0
CholCheck               0
BMI                     0
Smoker                  0
Stroke                  0
Diabetes                0
PhysActivity            0
Fruits                  0
Veggies                 0
HvyAlcoholConsump       0
AnyHealthcare           0
NoDocbcCost             0
GenHlth                 0
MentHlth                0
PhysHlth                0
DiffWalk                0
Sex                     0
Age                     0
Education               0
Income                  0
dtype: int64
```

In [8]:
```
# Removed the duplicates
df.duplicated()
```

Out[8]:
```
0         False
1         False
2         False
3         False
4         False
          ...
253675    False
253676    False
253677    False
253678    False
253679    False
Length: 253680, dtype: bool
```

In [9]:
```
# Value Count to check number of patience having heartdiseases Attack
df["HeartDiseaseorAttack"].value_counts()
```

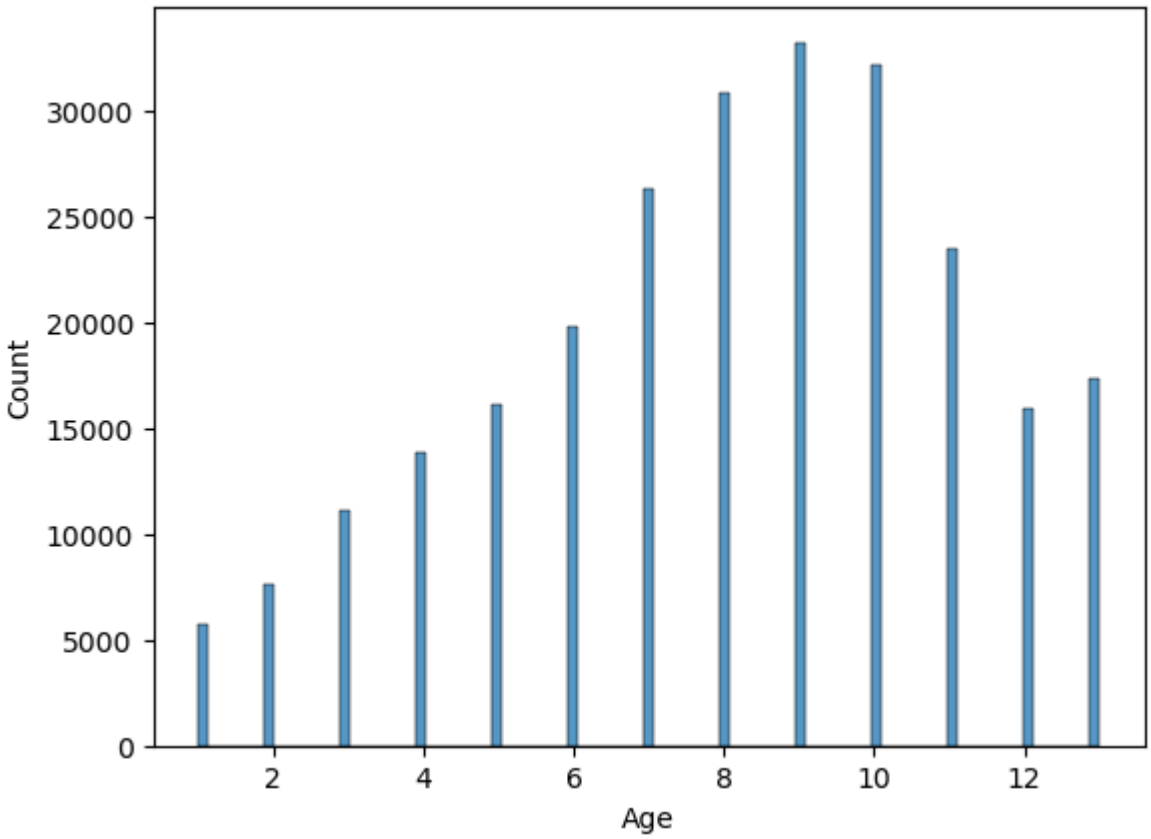
Out[9]:
```
HeartDiseaseorAttack
0    229787
1     23893
Name: count, dtype: int64
```

# EDA(Exploratory data analysis)

# 1) Univariate Analysis

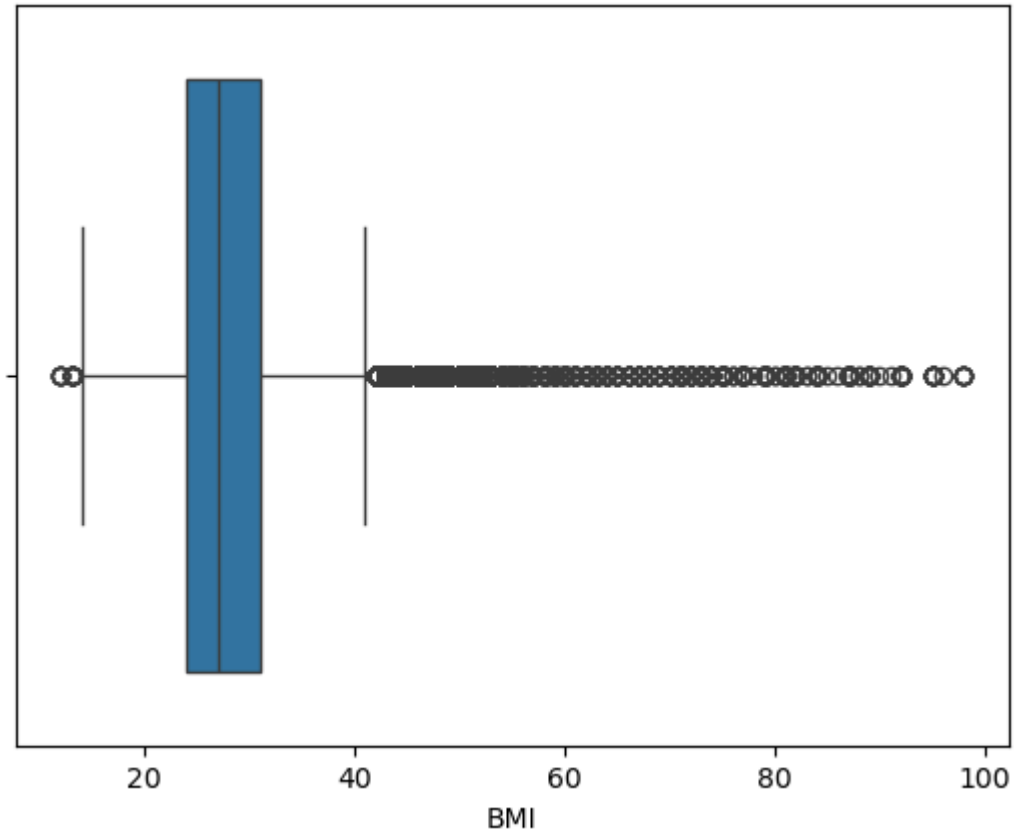
In [10]:
```
sns.histplot(df["Age"])
```

Out[10]:
```
<Axes: xlabel='Age', ylabel='Count'>
```



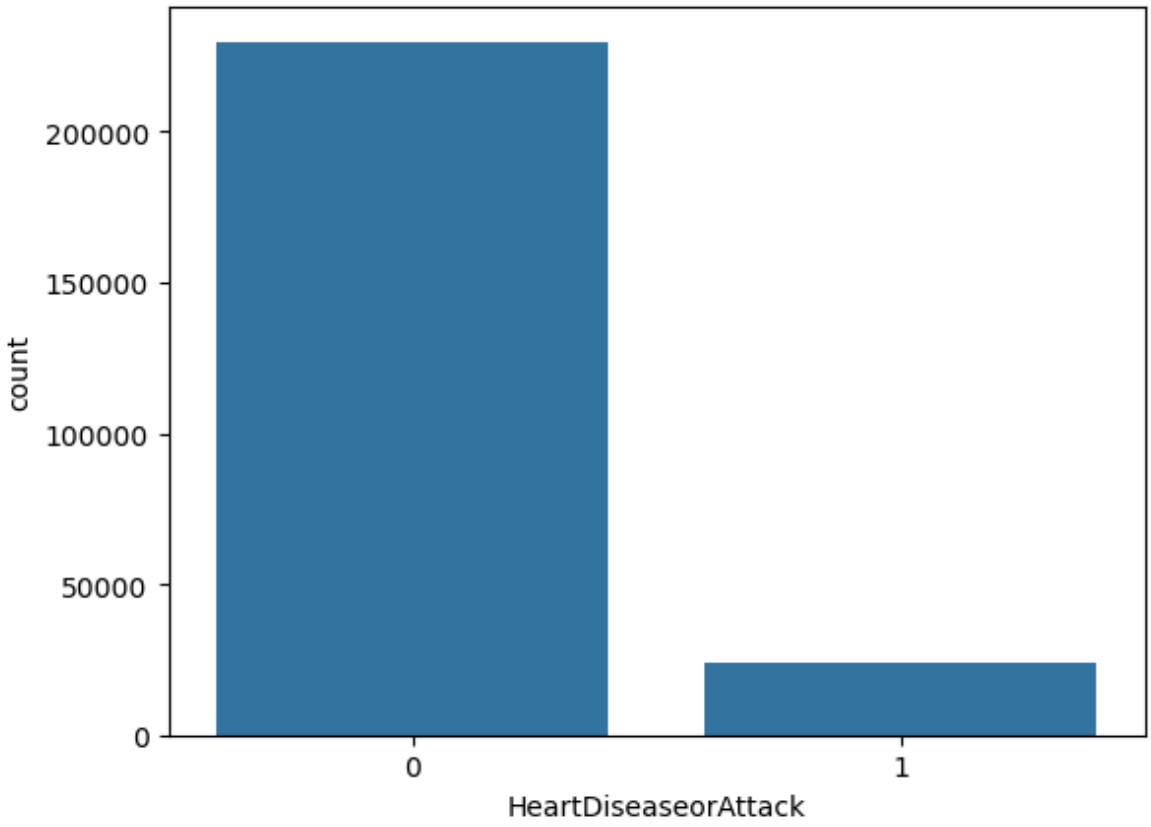

In [11]:
```
sns.boxplot(x=df['BMI'])
```

Out[11]:
```
<Axes: xlabel='BMI'>
```



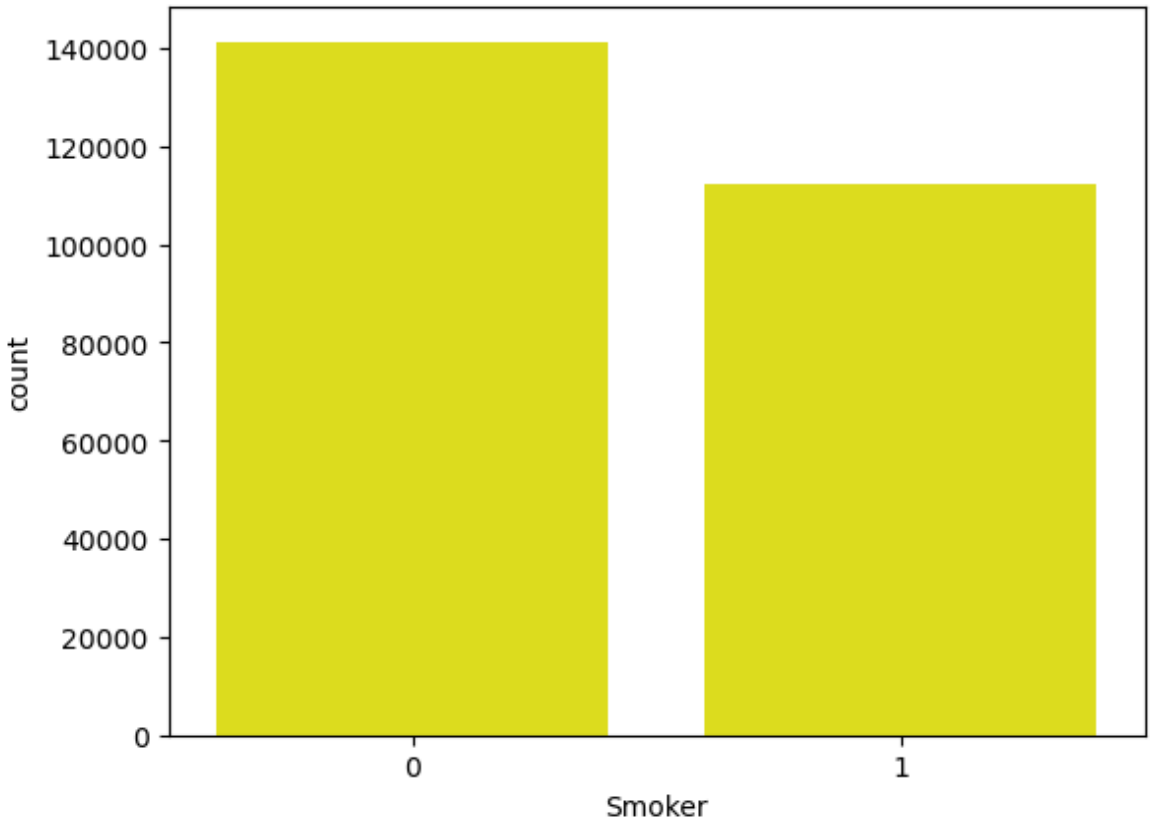

In [12]:
```
sns.countplot(x='HeartDiseaseorAttack', data=df)
```

Out[12]:
```
<Axes: xlabel='HeartDiseaseorAttack', ylabel='count'>
```



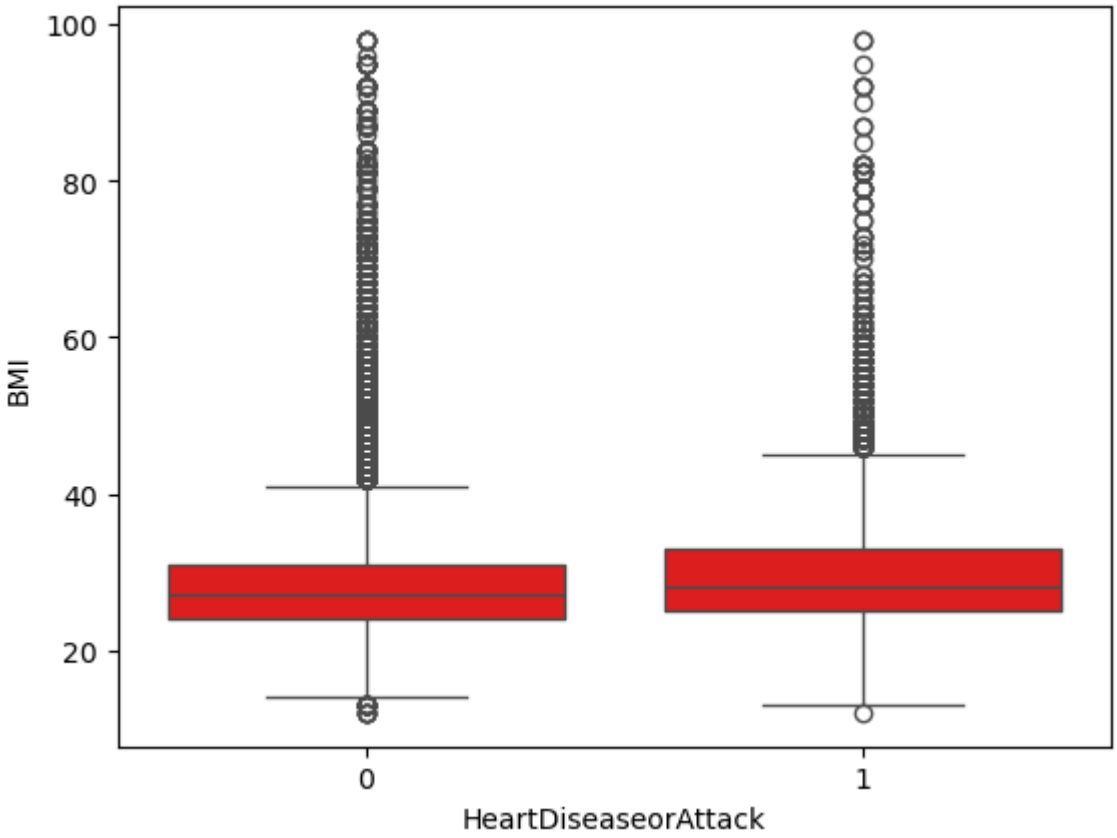

In [24]:
```
sns.countplot(x='Smoker', data=df, color='yellow')
```

Out[24]:
```
<Axes: xlabel='Smoker', ylabel='count'>
```

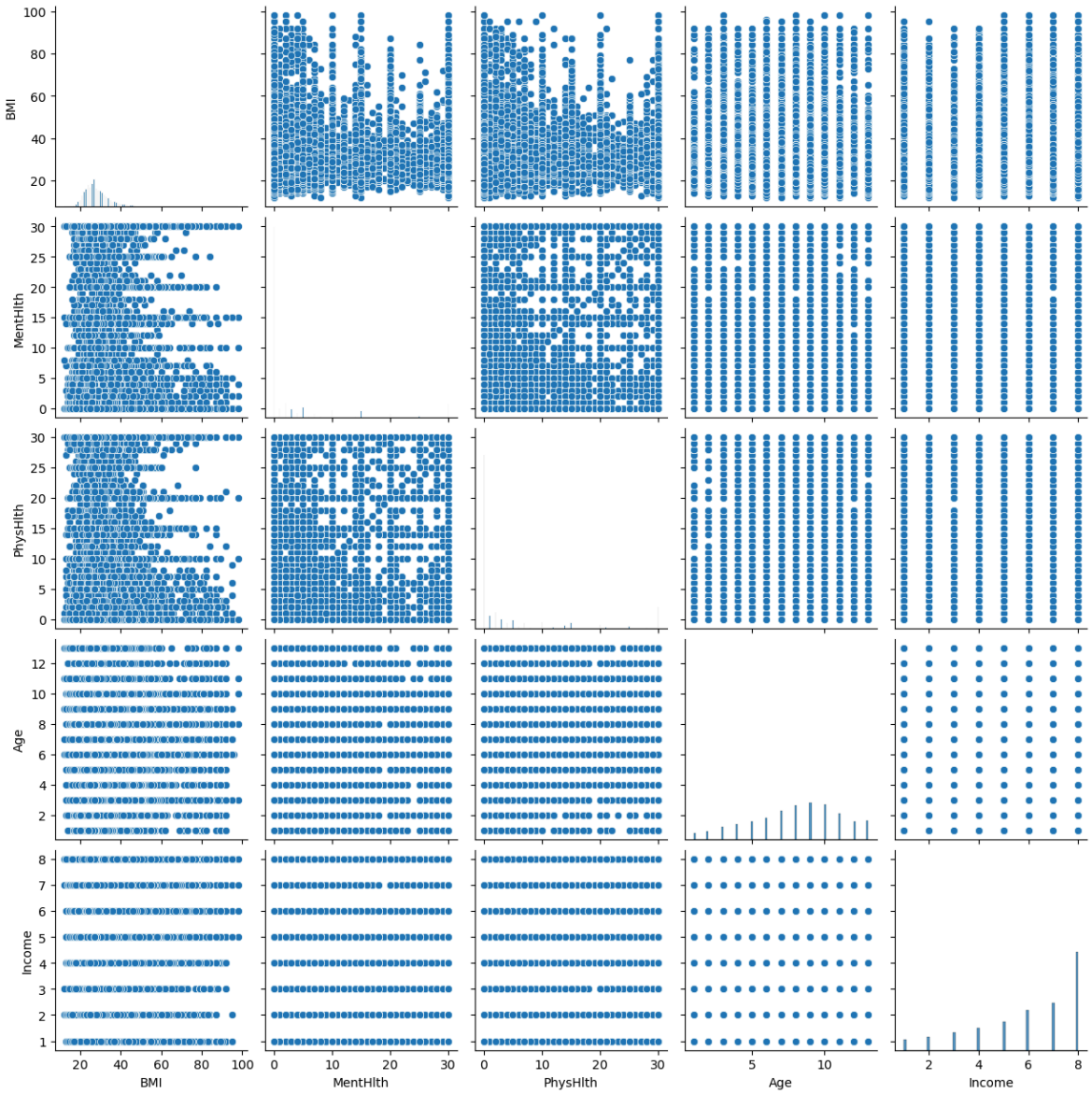



# 2) Multivariate Analysis

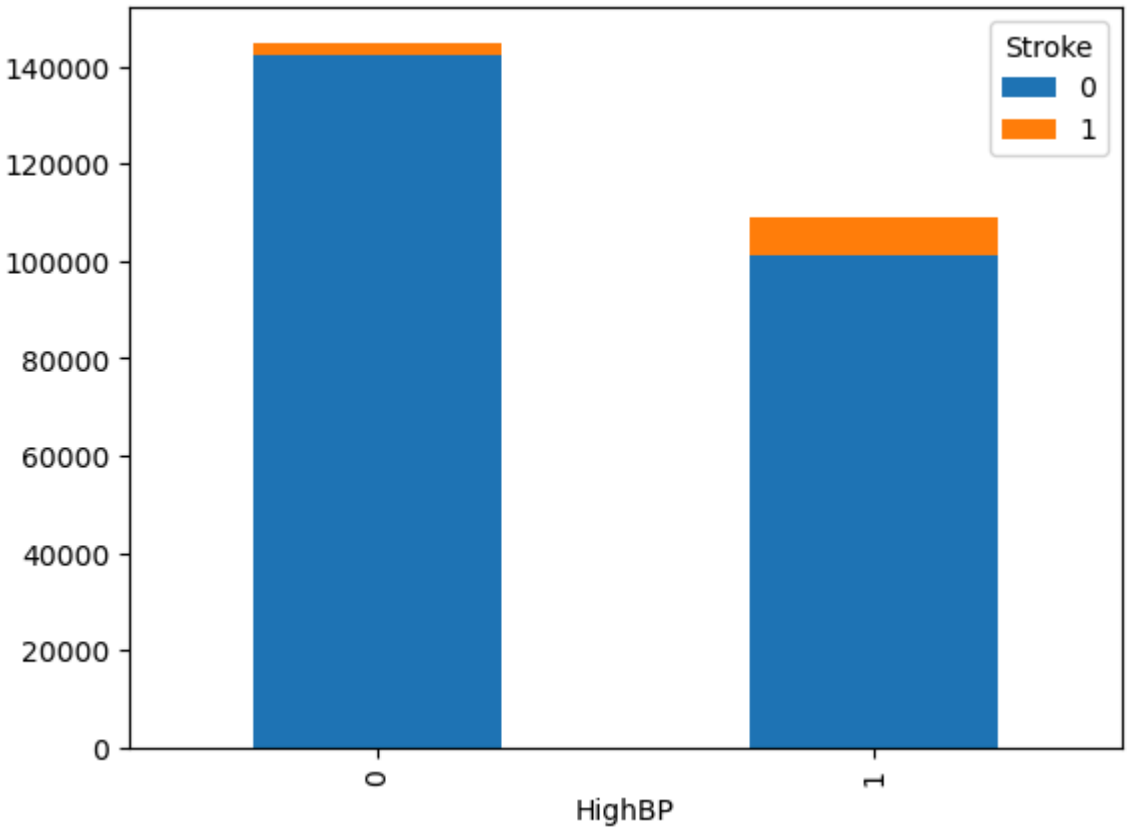
In [22]:
```
sns.boxplot(x='HeartDiseaseorAttack', y='BMI', data=df, color='red')
plt.show()
```



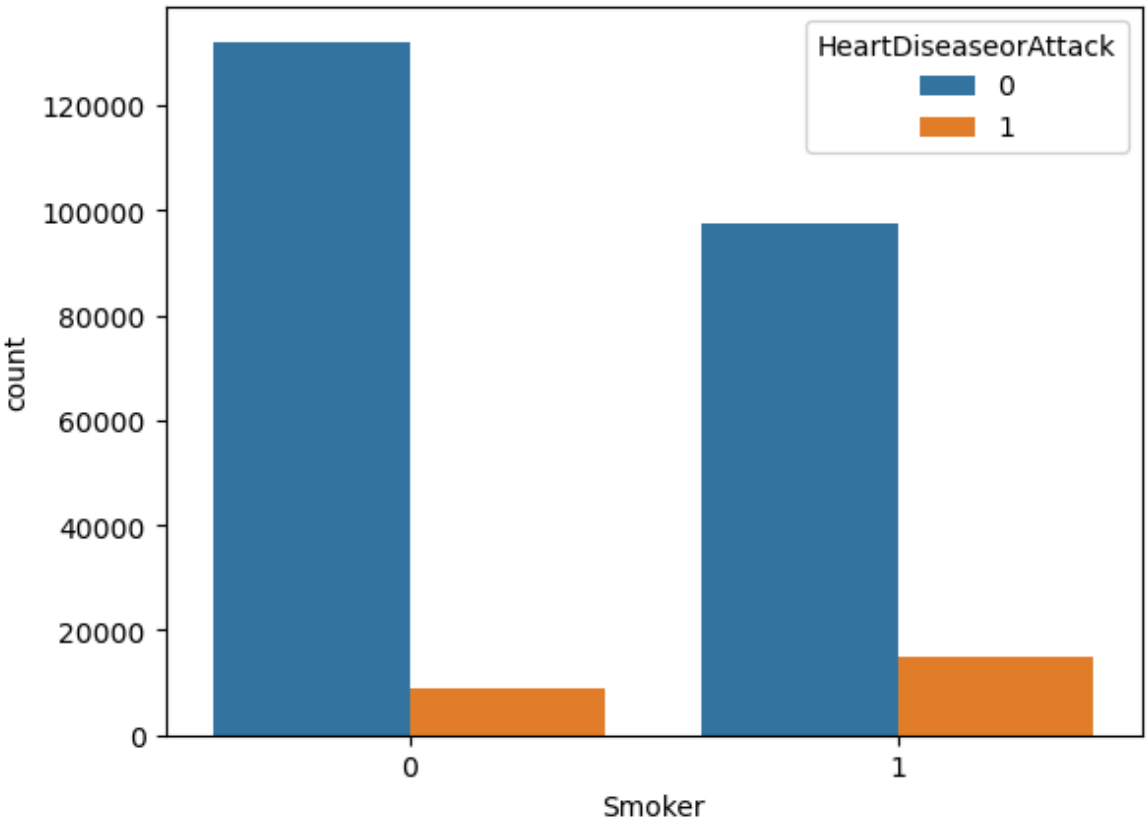

In [16]:
```
selected_cols = ['BMI', 'MentHlth', 'PhysHlth', 'Age', 'Income']
sns.pairplot(df[selected_cols])
plt.show()
```



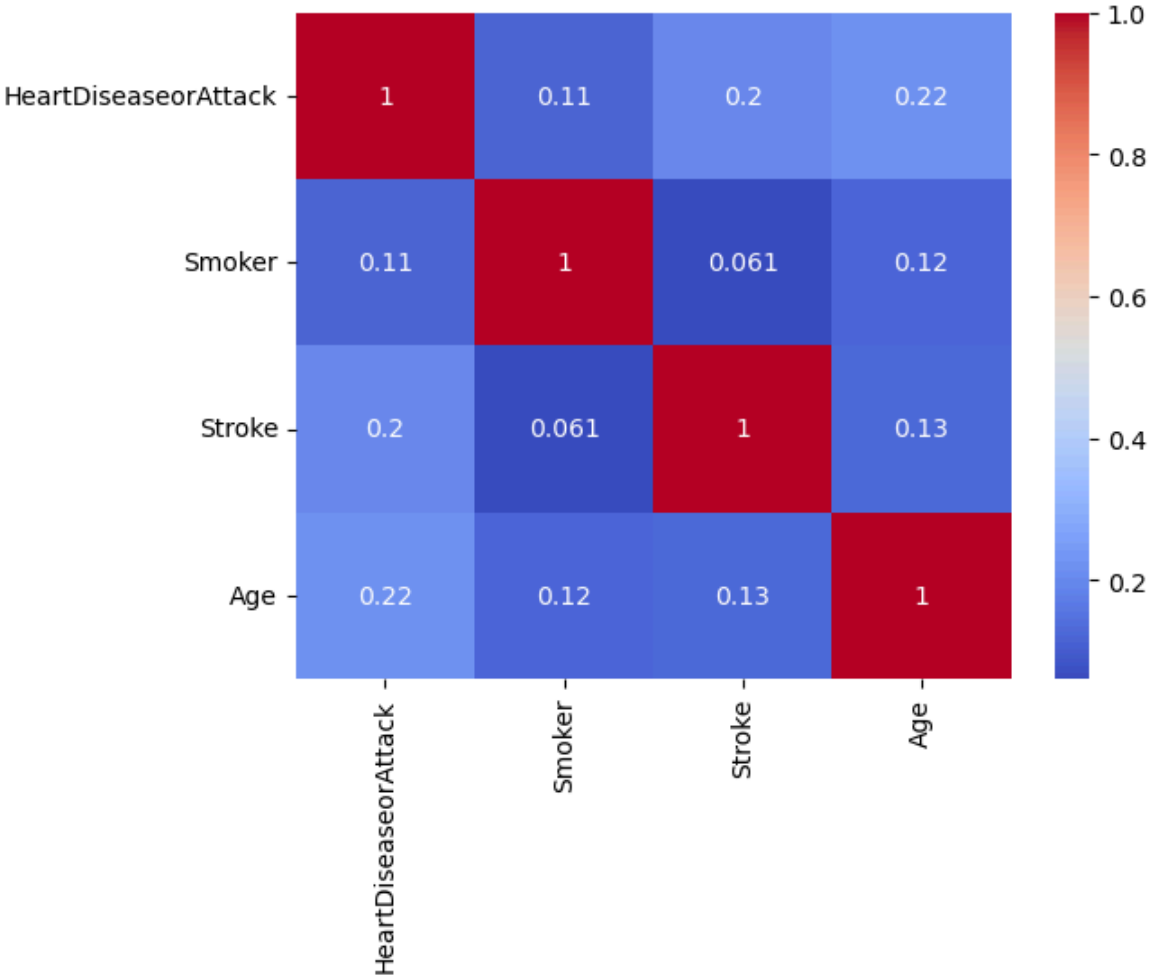

In [20]:
```
pd.crosstab(df['HighBP'], df['Stroke']).plot(kind='bar', stacked=True)
plt.show()
```




In [21]:
```
sns.countplot(x='Smoker', hue='HeartDiseaseorAttack', data=df)
plt.show()
```




In [29]:
```
corr = df[['HeartDiseaseorAttack', 'Smoker', 'Stroke', 'Age']]
sns.heatmap(corr.corr(), annot=True , cmap='coolwarm')
plt.show()
```


HeartDiseaseorAttack
Smoker
Stroke
Age
1
0.11
0.2
0.22
0.11
1
0.061
0.12
0.2
0.061
1
0.13
0.22
0.12
0.13
1
HeartDiseaseorAttack
Smoker
Stroke
Age
1.0
0.8
0.6
0.4
0.2