In [1]:

```
import pandas as pd
```

In [2]:

```
import numpy as np
```

In [3]:

```
import matplotlib.pyplot as plt
```

In [4]:

```
import seaborn as sns
```

In [6]:

```
file_path = "diabetes_data.csv"
```

In [7]:

```
df = pd.read_csv(r"C:\Users\user\Desktop\DAFINAL PROJECT\archive\diabetes_data.csv")
```

In [8]:

```
print(df.shape)
```

```
(70692, 18)
```

In [9]:

```
print(df.info())
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 70692 entries, 0 to 70691
Data columns (total 18 columns):
 #   Column                Non-Null Count  Dtype  
---  ------                --------------  -----  
 0   Age                   70692 non-null  float64
 1   Sex                   70692 non-null  float64
 2   HighChol              70692 non-null  float64
 3   CholCheck             70692 non-null  float64
 4   BMI                   70692 non-null  float64
 5   Smoker                70692 non-null  float64
 6   HeartDiseaseorAttack  70692 non-null  float64
 7   PhysActivity          70692 non-null  float64
 8   Fruits                70692 non-null  float64
 9   Veggies               70692 non-null  float64
 10  HvyAlcoholConsump     70692 non-null  float64
 11  GenHlth               70692 non-null  float64
 12  MentHlth              70692 non-null  float64
 13  PhysHlth              70692 non-null  float64
 14  DiffWalk              70692 non-null  float64
 15  Stroke                70692 non-null  float64
 16  HighBP                70692 non-null  float64
 17  Diabetes              70692 non-null  float64
dtypes: float64(18)
memory usage: 9.7 MB
None
```

In [10]:

```
print(df.describe())
```

```
                Age           Sex      HighChol     CholCheck           BMI  \
count  70692.000000  70692.000000  70692.000000  70692.000000  70692.000000   
mean       8.584055      0.456997      0.525703      0.975259     29.856985   
std        2.852153      0.498151      0.499342      0.155336      7.113954   
min        1.000000      0.000000      0.000000      0.000000     12.000000   
25%        7.000000      0.000000      0.000000      1.000000     25.000000   
```

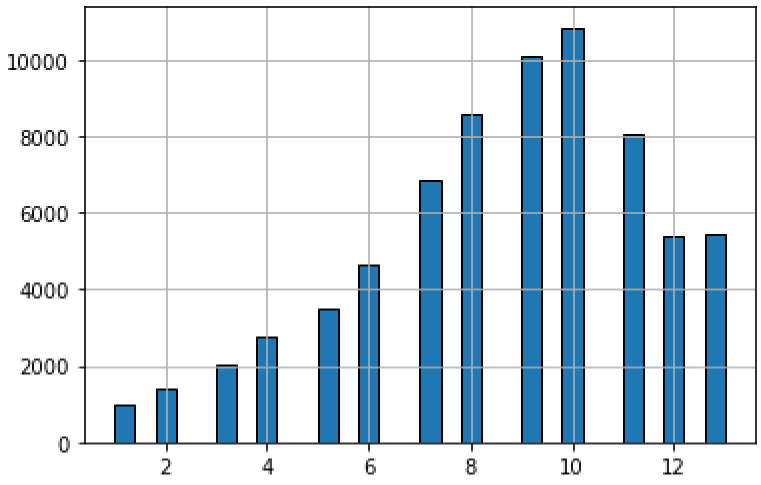
```
50%        9.000000      0.000000      1.000000      1.000000     29.000000
75%       11.000000      1.000000      1.000000      1.000000     33.000000
max       13.000000      1.000000      1.000000      1.000000     98.000000

             Smoker  HeartDiseaseorAttack  PhysActivity        Fruits  \
count  70692.000000          70692.000000  70692.000000  70692.000000
mean       0.475273              0.147810      0.703036      0.611795
std        0.499392              0.354914      0.456924      0.487345
min        0.000000              0.000000      0.000000      0.000000
25%        0.000000              0.000000      0.000000      0.000000
50%        0.000000              0.000000      1.000000      1.000000
75%        1.000000              0.000000      1.000000      1.000000
max        1.000000              1.000000      1.000000      1.000000

            Veggies  HvyAlcoholConsump       GenHlth      MentHlth  \
count  70692.000000       70692.000000  70692.000000  70692.000000
mean       0.788774           0.042721      2.837082      3.752037
std        0.408181           0.202228      1.113565      8.155627
min        0.000000           0.000000      1.000000      0.000000
25%        1.000000           0.000000      2.000000      0.000000
50%        1.000000           0.000000      3.000000      0.000000
75%        1.000000           0.000000      4.000000      2.000000
max        1.000000           1.000000      5.000000     30.000000

           PhysHlth      DiffWalk        Stroke        HighBP      Diabetes
count  70692.000000  70692.000000  70692.000000  70692.000000  70692.000000
mean       5.810417      0.252730      0.062171      0.563458      0.500000
std       10.062261      0.434581      0.241468      0.495960      0.500004
min        0.000000      0.000000      0.000000      0.000000      0.000000
25%        0.000000      0.000000      0.000000      0.000000      0.000000
50%        0.000000      0.000000      0.000000      1.000000      0.500000
75%        6.000000      1.000000      0.000000      1.000000      1.000000
max       30.000000      1.000000      1.000000      1.000000      1.000000
```

In [11]:

```
df['Age'].hist(bins=30, edgecolor='black')
```

Out[11]:

```
<AxesSubplot:>
```

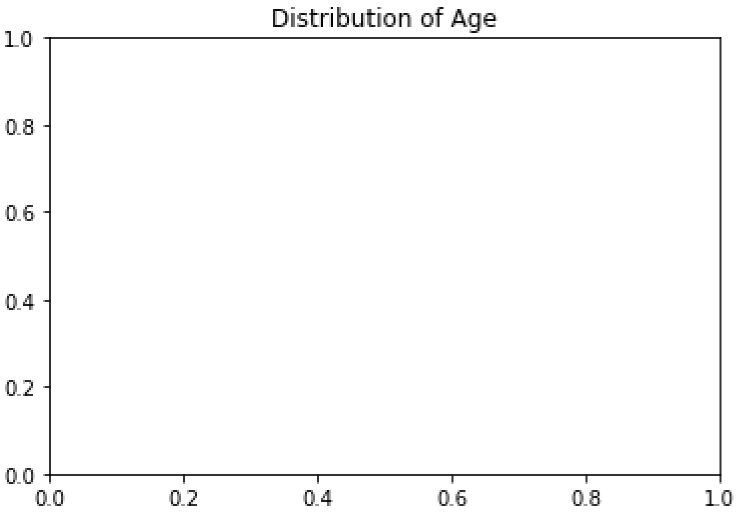


In [12]:

```
plt.title('Distribution of Age')
```

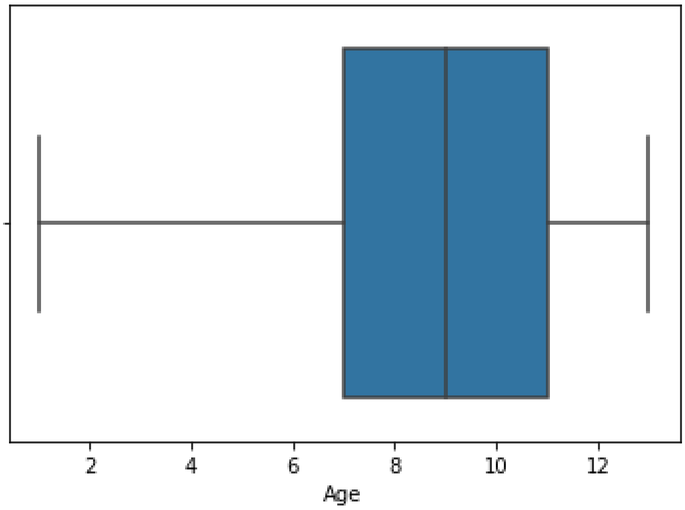
Out[12]:

```
Text(0.5, 1.0, 'Distribution of Age')
```



In [13]:
```
plt.show()
```

In [14]:
```
sns.boxplot(x=df['Age'])
plt.show()
```

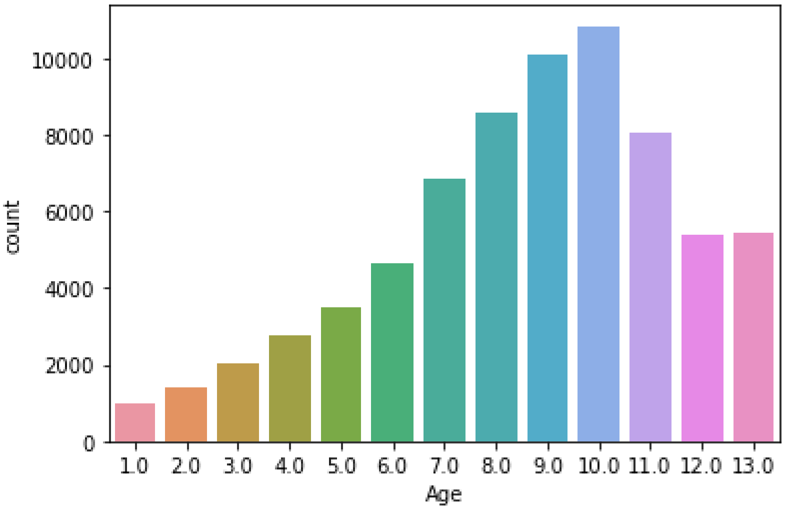


In [15]:
```
print(df['Age'].describe())
```

```
count    70692.000000
mean         8.584055
std          2.852153
min          1.000000
25%          7.000000
50%          9.000000
75%         11.000000
max         13.000000
Name: Age, dtype: float64
```

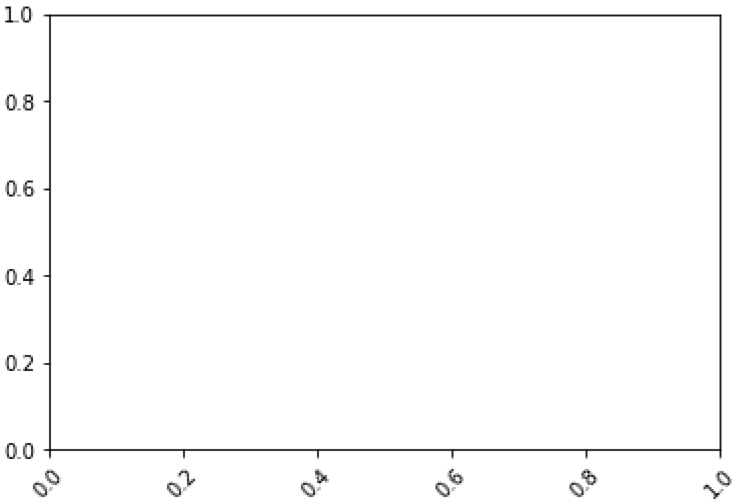
In [16]:
```
sns.countplot(x='Age', data=df)
```

Out[16]:
```
<AxesSubplot:xlabel='Age', ylabel='count'>
```



In [17]:

```
plt.xticks(rotation=45)
plt.show()
```



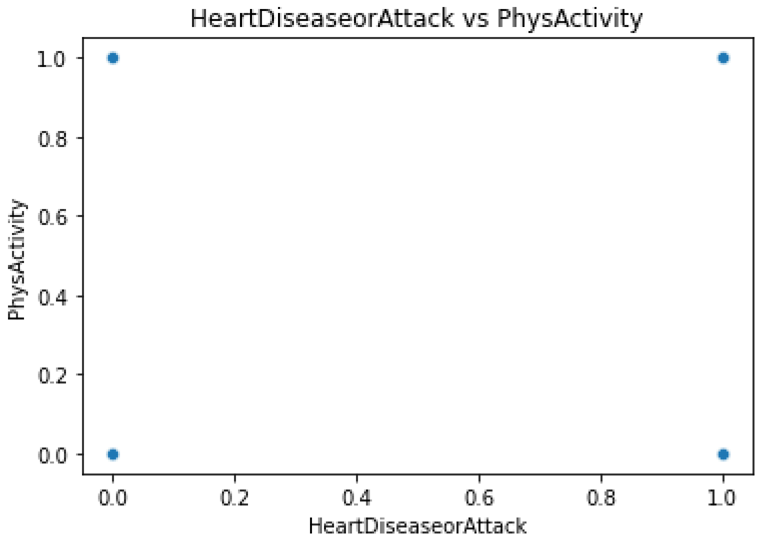
In [18]:

```
print(df['Age'].value_counts(normalize=True) * 100)
```

```
10.0    15.356759
9.0     14.304306
8.0     12.169694
11.0    11.378940
7.0      9.721043
13.0     7.675550
12.0     7.630283
6.0      6.575001
5.0      4.979347
4.0      3.950942
3.0      2.898489
2.0      1.974764
1.0      1.384881
Name: Age, dtype: float64
```

In [20]:

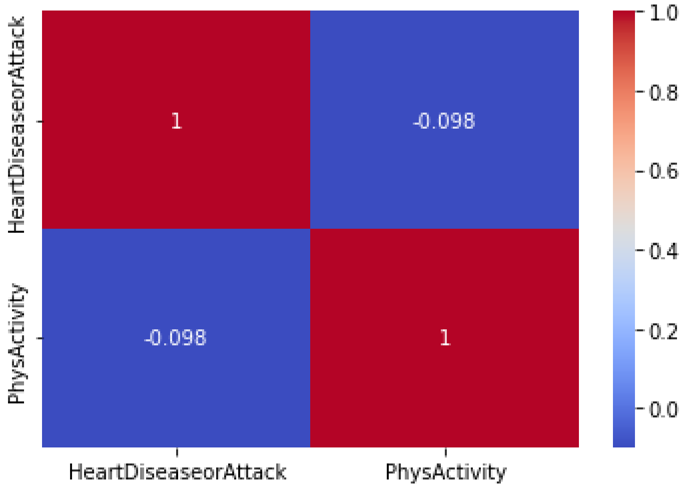
```
sns.scatterplot(x='HeartDiseaseorAttack', y='PhysActivity', data=df)
plt.title('HeartDiseaseorAttack vs PhysActivity')
plt.show()
```



In [21]:

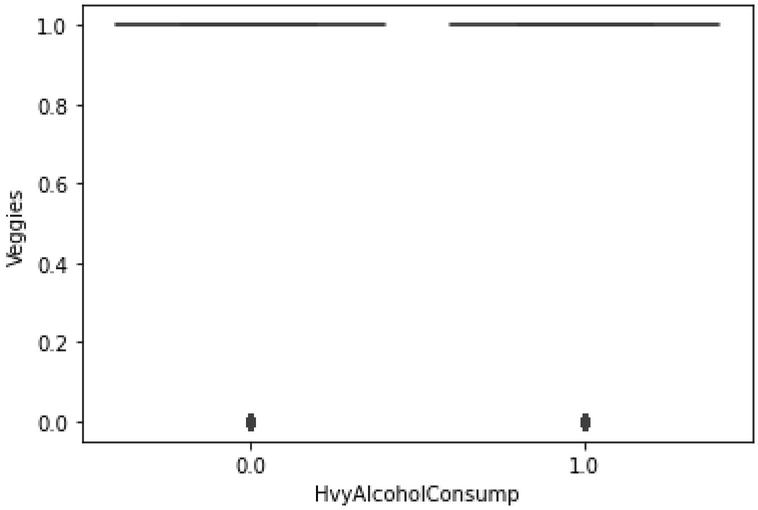
```
print(df[['HeartDiseaseorAttack', 'PhysActivity']].corr())
sns.heatmap(df[['HeartDiseaseorAttack', 'PhysActivity']].corr(), annot=True, cmap='c
plt.show()
```

```
                      HeartDiseaseorAttack  PhysActivity
HeartDiseaseorAttack              1.000000     -0.098223
PhysActivity                     -0.098223      1.000000
```

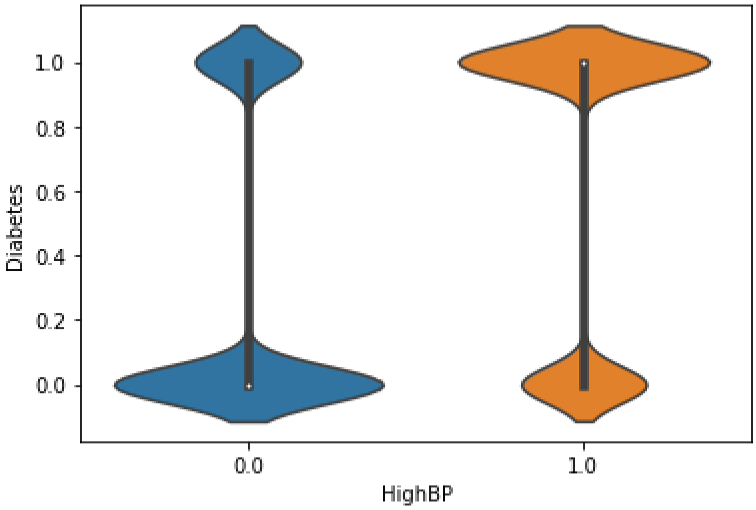


In [23]:

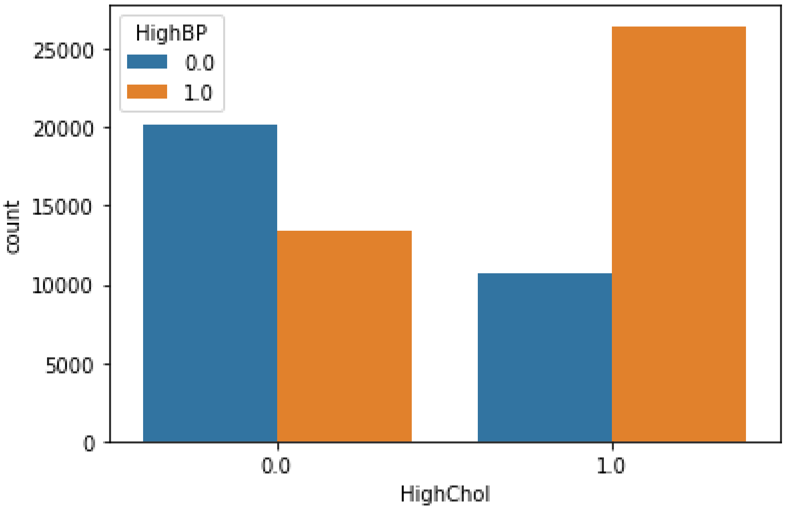
```
sns.boxplot(x='HvyAlcoholConsump', y='Veggies', data=df)
plt.show()
```



```
In [26]:
sns.violinplot(x='HighBP', y='Diabetes', data=df)
plt.show()
```



```
In [28]:
pd.crosstab(df['HighChol'], df['HighBP'], normalize='index') * 100
sns.countplot(x='HighChol', hue='HighBP', data=df)
plt.show()
```



```
In [30]:
df.groupby(['HighChol','HighBP'])['HighBP'].mean().unstack()
```

```
Out[30]:
  HighBP   0.0   1.0
HighChol
     0.0   0.0   1.0
     1.0   0.0   1.0
```

```
In [ ]:
```