In [ ]:
```
import pandas as pd
import numpy as np
import seaborn as sns
from sklearn.metrics import classification_report
from sklearn.metrics import confusion_matrix
from sklearn.neighbors import KNeighborsClassifier
from sklearn.model_selection import train_test_split
import matplotlib.pyplot as plt

%matplotlib inline
```

In [ ]:
```
heart_data=pd.read_csv('Heart.csv')
heart_data=heart_data.sort_values(by='Unnamed: 0')
heart_data=heart_data.iloc[:,:]
heart_data
```

Out[ ]:

	Unnamed: 0	Age	Sex	ChestPain	RestBP	Chol	Fbs	RestECG	MaxHR	ExAng	Oldpea
0	1	63	1	typical	145	233	1	2	150	0	2.
1	2	67	1	asymptomatic	160	286	0	2	108	1	1.
2	3	67	1	asymptomatic	120	229	0	2	129	1	2.
3	4	37	1	nonanginal	130	250	0	0	187	0	3.
4	5	41	0	nontypical	130	204	0	2	172	0	1.
...	...	...	...	...	...	...	...	...	...	...	.
298	299	45	1	typical	110	264	0	0	132	0	1.
299	300	68	1	asymptomatic	144	193	1	0	141	0	3.
300	301	57	1	asymptomatic	130	131	0	0	115	1	1.
301	302	57	0	nontypical	130	236	0	2	174	0	0.
302	303	38	1	nonanginal	138	175	0	0	173	0	0.

303 rows × 15 columns

In [ ]:
```
heart_data.shape
```

Out[ ]:
```
(303, 15)
```

In [ ]:
```
heart_data.isnull().sum()
```

Out[ ]:
```
Unnamed: 0    0
Age           0
Sex           0
ChestPain     0
RestBP        0
Chol          0
Fbs           0
RestECG       0
MaxHR         0
ExAng         0
Oldpeak       0
Slope         0
Ca            4
Thal          2
AHD           0
dtype: int64
```

In [ ]:
```
heart_data.isnull().sum().sum()
```

Out[ ]:
```
6
```

In [ ]:
```
heart_data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 303 entries, 0 to 302
Data columns (total 15 columns):
 #   Column      Non-Null Count  Dtype  
---  ------      --------------  -----  
 0   Unnamed: 0  303 non-null    int64  
 1   Age         303 non-null    int64  
 2   Sex         303 non-null    int64  
 3   ChestPain   303 non-null    object 
 4   RestBP      303 non-null    int64  
 5   Chol        303 non-null    int64  
 6   Fbs         303 non-null    int64  
 7   RestECG     303 non-null    int64  
 8   MaxHR       303 non-null    int64  
 9   ExAng       303 non-null    int64  
 10  Oldpeak     303 non-null    float64
 11  Slope       303 non-null    int64  
 12  Ca          299 non-null    float64
 13  Thal        301 non-null    object 
 14  AHD         303 non-null    object 
dtypes: float64(2), int64(10), object(3)
memory usage: 37.9+ KB
```

In [ ]:
```
(heart_data==0).sum()
```

Out[ ]:
```
Unnamed: 0      0
Age             0
Sex            97
ChestPain       0
RestBP          0
Chol            0
Fbs           258
RestECG       151
MaxHR           0
ExAng         204
Oldpeak        99
Slope           0
Ca            176
Thal            0
AHD             0
dtype: int64
```

In [ ]:
```
mean_ca=heart_data['Ca'].mean()
print(mean_ca)
heart_data['Ca'].fillna(mean_ca,inplace=True)
```

```
0.6722408026755853
```

In [ ]:
```
heart_data
```

Out[ ]:

	Unnamed: 0	Age	Sex	ChestPain	RestBP	Chol	Fbs	RestECG	MaxHR	ExAng	Oldpea
0	1	63	1	typical	145	233	1	2	150	0	2.
1	2	67	1	asymptomatic	160	286	0	2	108	1	1.
2	3	67	1	asymptomatic	120	229	0	2	129	1	2.
3	4	37	1	nonanginal	130	250	0	0	187	0	3.
4	5	41	0	nontypical	130	204	0	2	172	0	1.
...	...	...	...	...	...	...	...	...	...	...	.
298	299	45	1	typical	110	264	0	0	132	0	1.
299	300	68	1	asymptomatic	144	193	1	0	141	0	3.
300	301	57	1	asymptomatic	130	131	0	0	115	1	1.
301	302	57	0	nontypical	130	236	0	2	174	0	0.
302	303	38	1	nonanginal	138	175	0	0	173	0	0.

303 rows × 15 columns

In [ ]:
```
mean_age=heart_data['Age'].mean()
print(mean_age)
```

```
54.43894389438944
```

In [ ]:
```
x_values=heart_data.filter(['Age','Sex','ChestPain','RestBP','Chol'])
y_values=heart_data.filter(['AHD'])
x_values,y_values
```

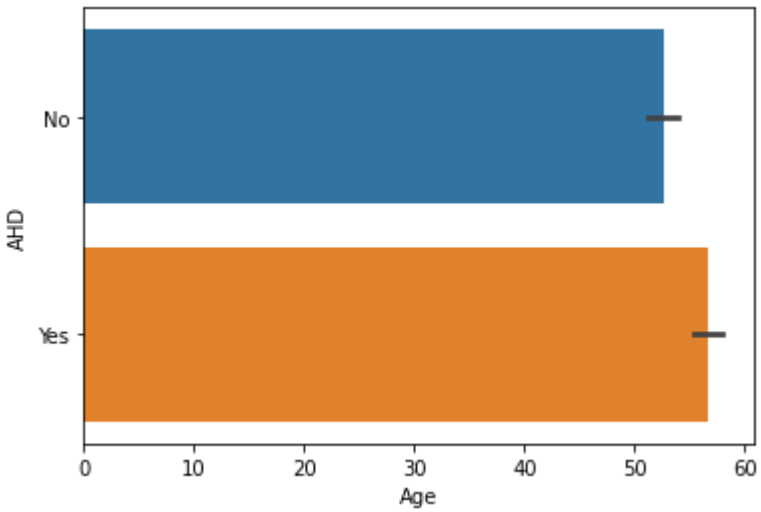
Out[ ]:
```
(     Age  Sex     ChestPain  RestBP  Chol
 0     63    1       typical     145   233
 1     67    1  asymptomatic     160   286
 2     67    1  asymptomatic     120   229
 3     37    1    nonanginal     130   250
 4     41    0    nontypical     130   204
 ..   ...  ...           ...     ...   ...
 298   45    1       typical     110   264
 299   68    1  asymptomatic     144   193
 300   57    1  asymptomatic     130   131
 301   57    0    nontypical     130   236
 302   38    1    nonanginal     138   175

 [303 rows x 5 columns],
      AHD
 0     No
 1    Yes
 2    Yes
 3     No
 4     No
 ..   ...
 298  Yes
 299  Yes
 300  Yes
 301  Yes
 302   No

 [303 rows x 1 columns])
```

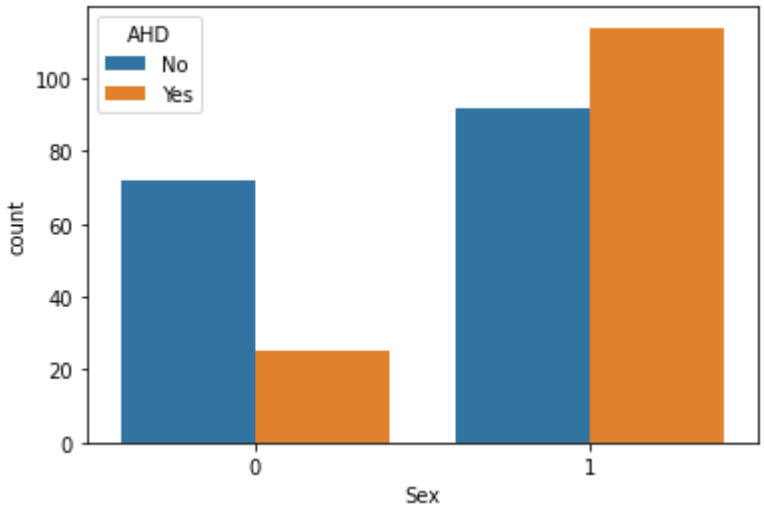
In [ ]:
```
sns.barplot(data=heart_data,x='Age',y='AHD')
```

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



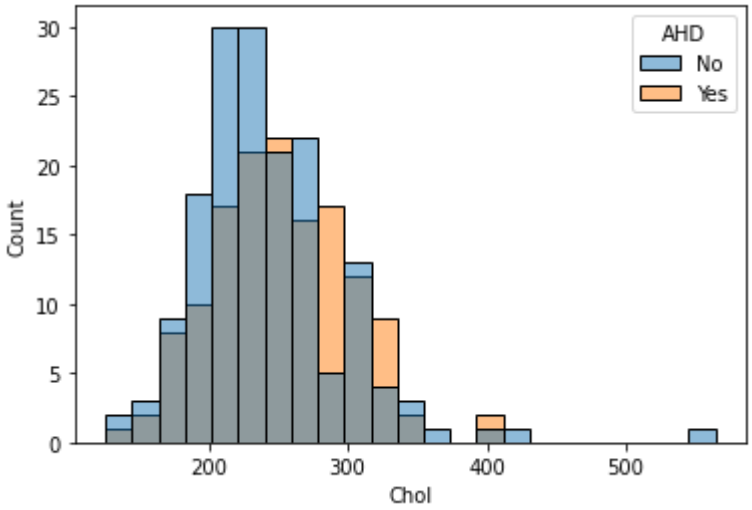
In [ ]:
```
sns.countplot(data=heart_data,x='Sex',hue='AHD')
```

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



In [ ]:
```
sns.histplot(data=heart_data,x='Chol',hue='AHD')
```

Out[ ]:
```
<AxesSubplot: xlabel='Chol', ylabel='Count'>
```



In [ ]:
```
heart_data
```

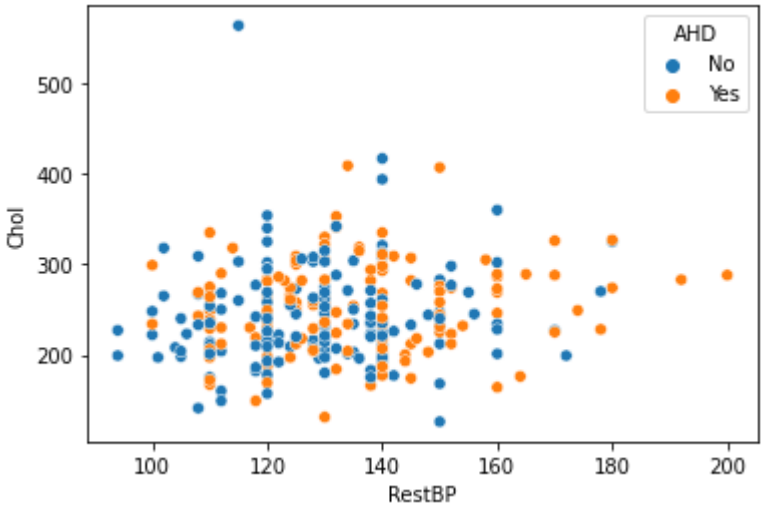
Out[ ]:

	Unnamed: 0	Age	Sex	ChestPain	RestBP	Chol	Fbs	RestECG	MaxHR	ExAng	Oldpea
0	1	63	1	typical	145	233	1	2	150	0	2.
1	2	67	1	asymptomatic	160	286	0	2	108	1	1.
2	3	67	1	asymptomatic	120	229	0	2	129	1	2.
3	4	37	1	nonanginal	130	250	0	0	187	0	3.
4	5	41	0	nontypical	130	204	0	2	172	0	1.
...	...	...	...	...	...	...	...	...	...	...	.
298	299	45	1	typical	110	264	0	0	132	0	1.
299	300	68	1	asymptomatic	144	193	1	0	141	0	3.
300	301	57	1	asymptomatic	130	131	0	0	115	1	1.
301	302	57	0	nontypical	130	236	0	2	174	0	0.
302	303	38	1	nonanginal	138	175	0	0	173	0	0.

303 rows × 15 columns

In [ ]:
```
sns.scatterplot(data=heart_data,x='RestBP',y='Chol',hue='AHD')
```

Out[ ]:
```
<AxesSubplot: xlabel='RestBP', ylabel='Chol'>
```



In [ ]:
```
sns.pairplot(data=heart_data,hue='AHD')
```

Out[ ]:
```
<seaborn.axisgrid.PairGrid at 0x7f9dfaeda8b0>
```

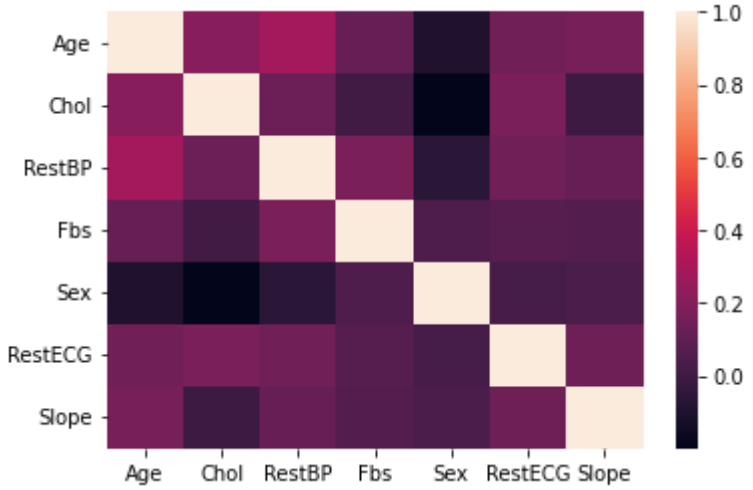
In [ ]:
```
heart_data.filter(['Age','Chol','RestBP','Fbs','Sex','RestECG','Slope','OldP
```

Out[ ]:

	Age	Chol	RestBP	Fbs	Sex	RestECG	Slope
Age	1.000000	0.208950	0.284946	0.118530	-0.097542	0.148868	0.161770
Chol	0.208950	1.000000	0.130120	0.009841	-0.199915	0.171043	-0.004062
RestBP	0.284946	0.130120	1.000000	0.175340	-0.064456	0.146560	0.117382
Fbs	0.118530	0.009841	0.175340	1.000000	0.047862	0.069564	0.059894
Sex	-0.097542	-0.199915	-0.064456	0.047862	1.000000	0.021647	0.037533
RestECG	0.148868	0.171043	0.146560	0.069564	0.021647	1.000000	0.133946
Slope	0.161770	-0.004062	0.117382	0.059894	0.037533	0.133946	1.000000

In [ ]:
```
sns.heatmap(heart_data.filter(['Age','Chol','RestBP','Fbs','Sex','RestECG','
```

Out[ ]:
```
<AxesSubplot: >
```



In [ ]:
```
x_train,y_train,x_test,y_test=train_test_split(x_values,y_values,test_size=0
x_train,y_train,x_test,y_test
```

Out[ ]:
```
(     Age  Sex     ChestPain  RestBP  Chol
 287   58    1    nontypical     125   220
 282   55    0  asymptomatic     128   205
 197   45    0  asymptomatic     138   236
 158   60    1  asymptomatic     140   293
 164   48    1    nonanginal     124   255
 ..    ...  ...           ...     ...   ...
 188   54    1    nontypical     192   283
 71    67    1  asymptomatic     125   254
 106   59    1  asymptomatic     140   177
 270   61    1  asymptomatic     140   207
 102   57    0  asymptomatic     128   303

 [227 rows x 5 columns],
      Age  Sex     ChestPain  RestBP  Chol
 179   53    1    nonanginal     130   246
 228   54    1  asymptomatic     110   206
 111   56    1  asymptomatic     125   249
 246   58    1  asymptomatic     100   234
 60    51    0  asymptomatic     130   305
 ..    ...  ...           ...     ...   ...
 22    58    1    nontypical     120   284
 258   70    1    nontypical     156   245
 56    50    1    nonanginal     140   233
 242   49    0  asymptomatic     130   269
 114   62    0    nonanginal     130   263

 [76 rows x 5 columns],
      AHD
 287   No
 282  Yes
 197   No
 158  Yes
 164   No
 ..   ...
 188  Yes
 71   Yes
 106  Yes
 270  Yes
 102   No

 [227 rows x 1 columns],
      AHD
 179   No
 228  Yes
 111  Yes
 246  Yes
 60   Yes
 ..   ...
 22   Yes
 258   No
 56   Yes
 242   No
 114  Yes

 [76 rows x 1 columns])
```

In [ ]:
```
actual_data=np.array(['positive']*50+['negative']*450)

predicted_data=np.array(['positive']*45+['negative']*5+['positive']*55+['neg
```

```
cm=confusion_matrix(actual_data,predicted_data,labels=['positive','negative'
```

In [ ]:

```
sns.heatmap(cm,
            annot=True,
            fmt='g',
            xticklabels=['Positive','Negative'],
            yticklabels=['Positive','Negative'])
plt.ylabel('Actual', fontsize=13)
plt.title('Confusion Matrix', fontsize=17, pad=20)
plt.gca().xaxis.set_label_position('top')
plt.xlabel('Prediction', fontsize=13)
plt.gca().xaxis.tick_top()

print(classification_report(actual_data,predicted_data,labels=["positive",'r
```

```
              precision    recall  f1-score   support

    positive       0.45      0.90      0.60        50
    negative       0.99      0.88      0.93       450

    accuracy                           0.88       500
   macro avg       0.72      0.89      0.76       500
weighted avg       0.93      0.88      0.90       500
```