

```
In [5]: import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

```
In [18]: file = 'C:/Users/Pratiksha/Downloads/Heart_Disease.csv'
data = pd.read_csv(file)
```

```
In [20]: print(data.head())
```

	age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	\
0	52	1	0	125	212	0	1	168	0	1.0	2	
1	53	1	0	140	203	1	0	155	1	3.1	0	
2	70	1	0	145	174	0	1	125	1	2.6	0	
3	61	1	0	148	203	0	1	161	0	0.0	2	
4	62	0	0	138	294	1	1	106	0	1.9	1	

	ca	thal	target
0	2	3	0
1	0	3	0
2	0	3	0
3	1	3	0
4	3	2	0

```
In [21]: print(data.info())
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1025 entries, 0 to 1024
Data columns (total 14 columns):
#   Column      Non-Null Count  Dtype
---  -
0   age         1025 non-null   int64
1   sex         1025 non-null   int64
2   cp          1025 non-null   int64
3   trestbps    1025 non-null   int64
4   chol        1025 non-null   int64
5   fbs         1025 non-null   int64
6   restecg     1025 non-null   int64
7   thalach     1025 non-null   int64
8   exang       1025 non-null   int64
9   oldpeak     1025 non-null   float64
10  slope       1025 non-null   int64
11  ca          1025 non-null   int64
12  thal        1025 non-null   int64
13  target      1025 non-null   int64
dtypes: float64(1), int64(13)
memory usage: 112.2 KB
None
```

```
In [22]: print(data.describe())
```

	age	sex	cp	trestbps	chol \
count	1025.000000	1025.000000	1025.000000	1025.000000	1025.000000
mean	54.434146	0.695610	0.942439	131.611707	246.000000
std	9.072290	0.460373	1.029641	17.516718	51.59251
min	29.000000	0.000000	0.000000	94.000000	126.000000
25%	48.000000	0.000000	0.000000	120.000000	211.000000
50%	56.000000	1.000000	1.000000	130.000000	240.000000
75%	61.000000	1.000000	2.000000	140.000000	275.000000
max	77.000000	1.000000	3.000000	200.000000	564.000000

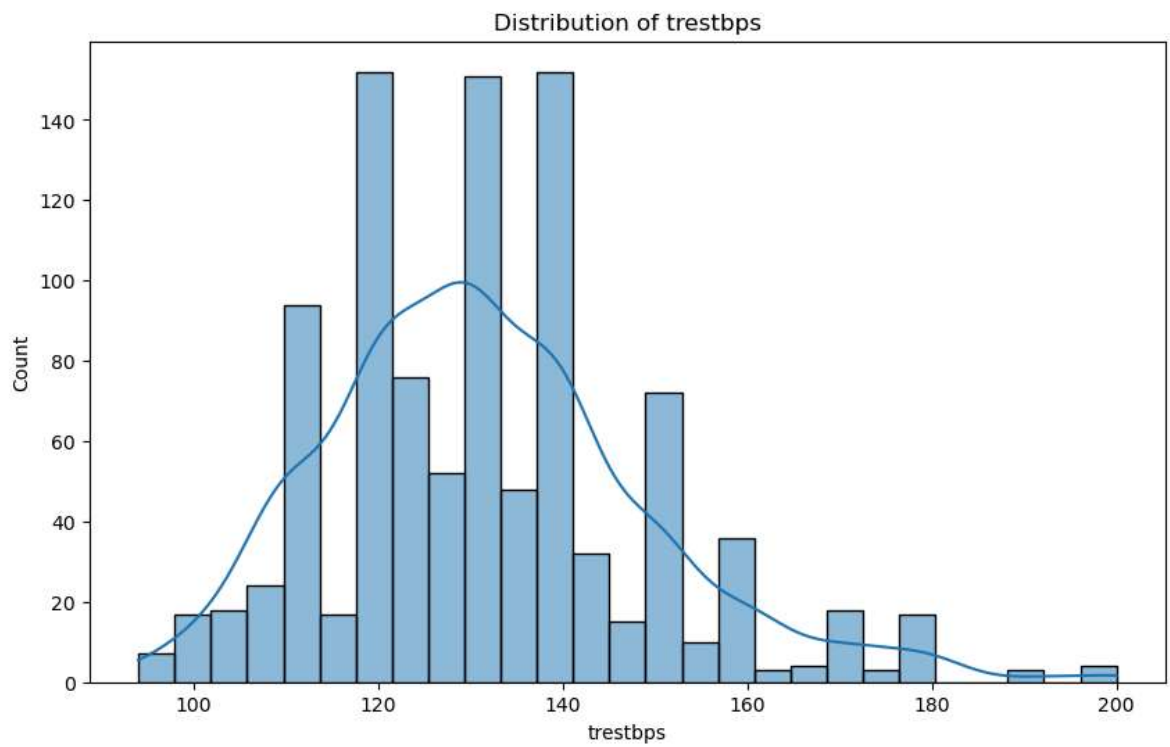
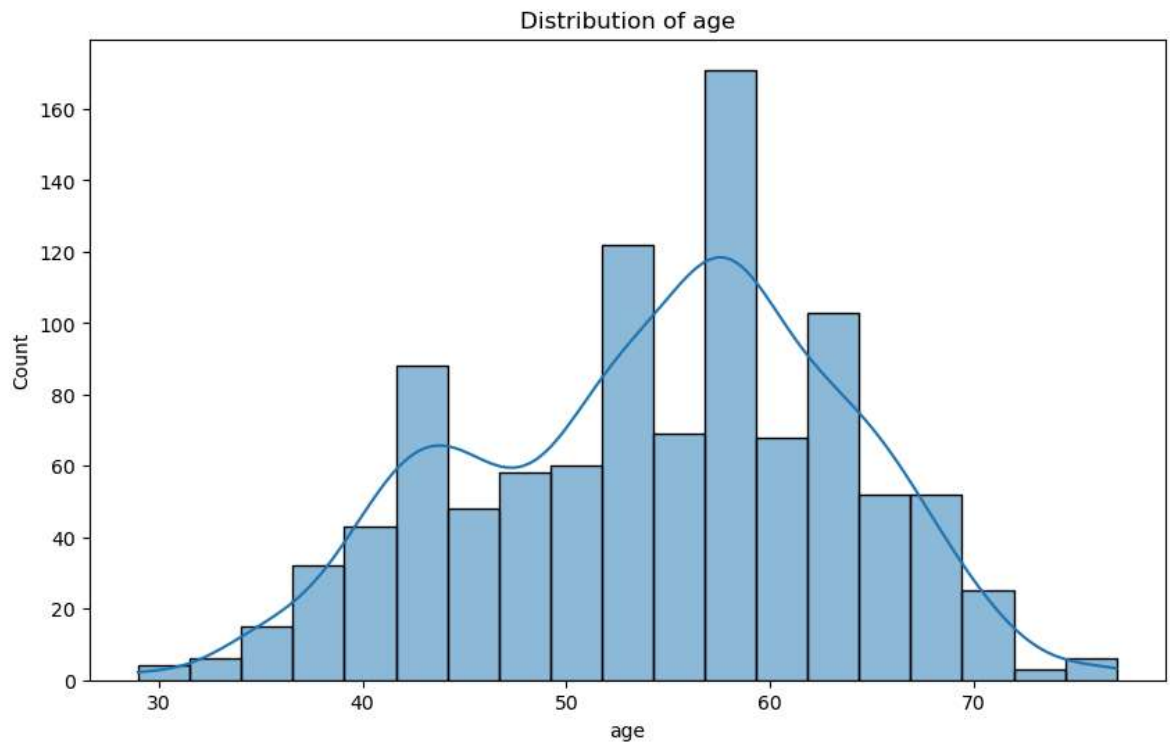
	fbs	restecg	thalach	exang	oldpeak \
count	1025.000000	1025.000000	1025.000000	1025.000000	1025.000000
mean	0.149268	0.529756	149.114146	0.336585	1.071512
std	0.356527	0.527878	23.005724	0.472772	1.175053
min	0.000000	0.000000	71.000000	0.000000	0.000000
25%	0.000000	0.000000	132.000000	0.000000	0.000000
50%	0.000000	1.000000	152.000000	0.000000	0.800000
75%	0.000000	1.000000	166.000000	1.000000	1.800000
max	1.000000	2.000000	202.000000	1.000000	6.200000

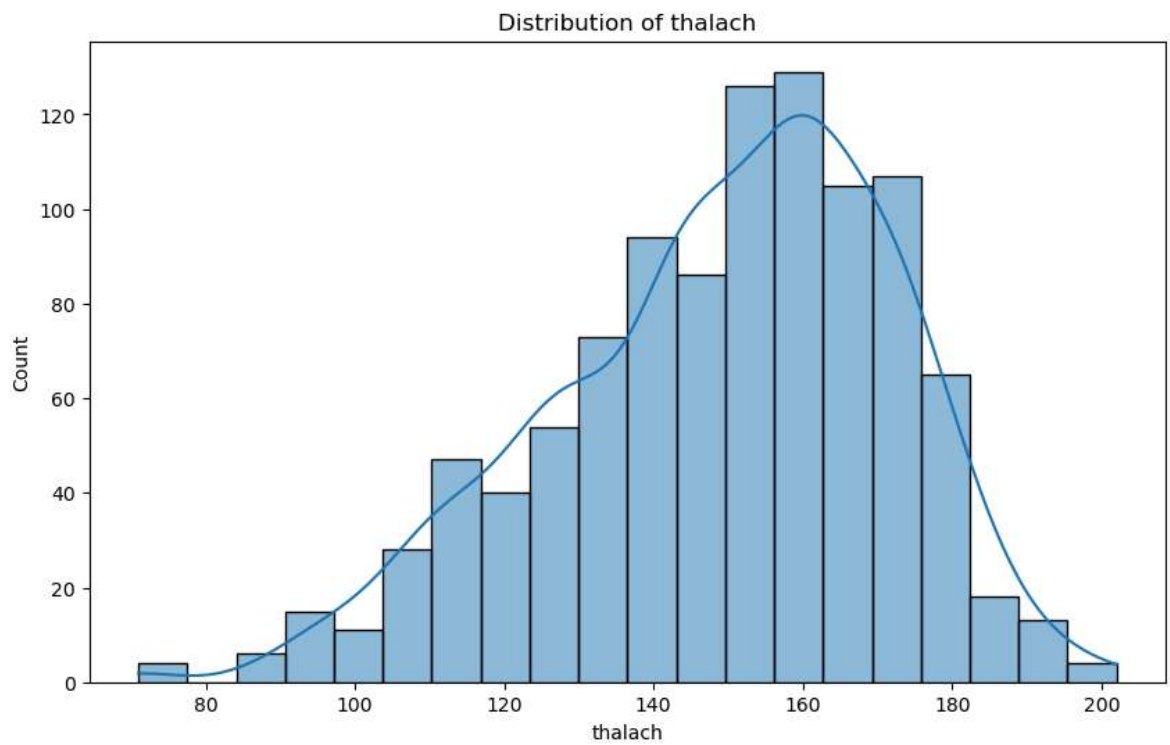
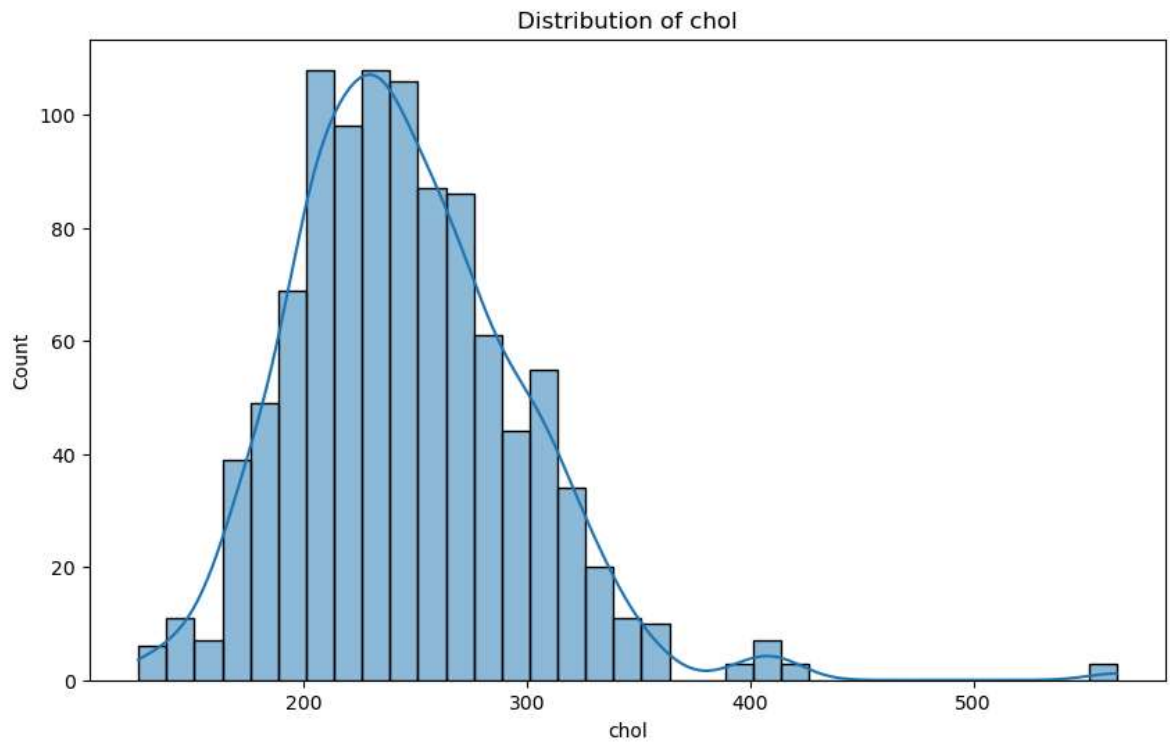
	slope	ca	thal	target
count	1025.000000	1025.000000	1025.000000	1025.000000
mean	1.385366	0.754146	2.323902	0.513171
std	0.617755	1.030798	0.620660	0.500070
min	0.000000	0.000000	0.000000	0.000000
25%	1.000000	0.000000	2.000000	0.000000
50%	1.000000	0.000000	2.000000	1.000000
75%	2.000000	1.000000	3.000000	1.000000
max	2.000000	4.000000	3.000000	1.000000

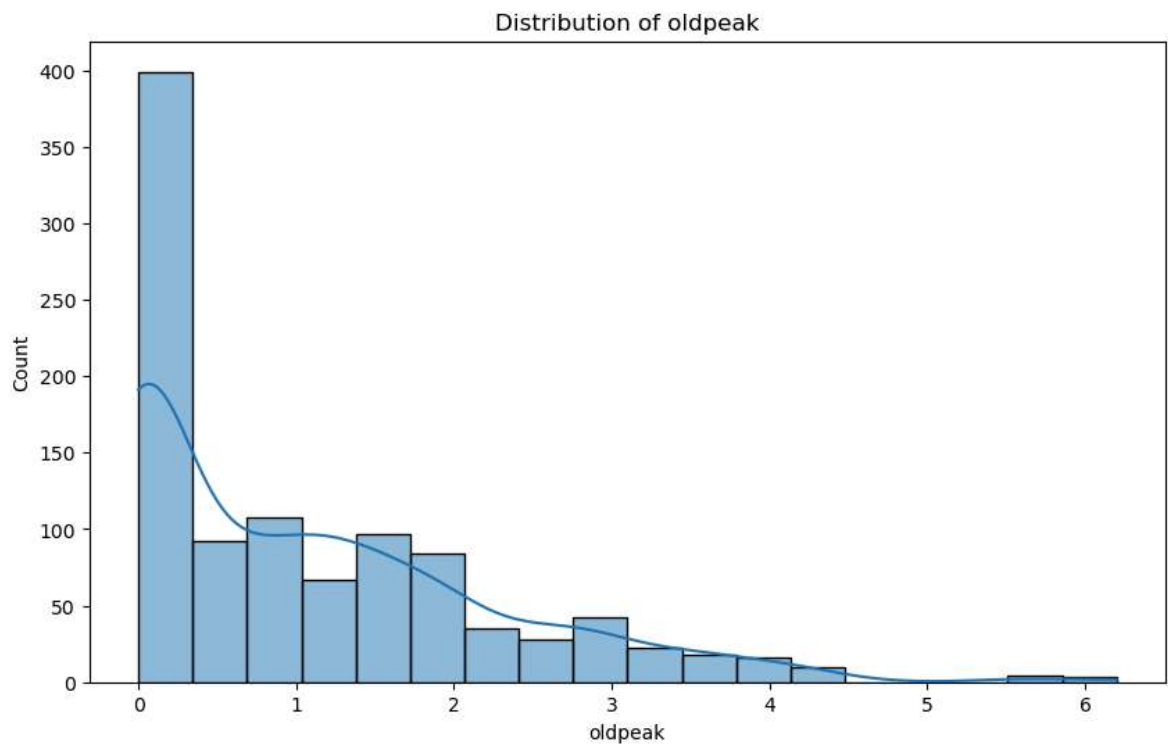
```
In [23]: print(data.isnull().sum())
```

```
age      0
sex      0
cp       0
trestbps 0
chol     0
fbs      0
restecg  0
thalach  0
exang    0
oldpeak  0
slope    0
ca       0
thal     0
target   0
dtype: int64
```

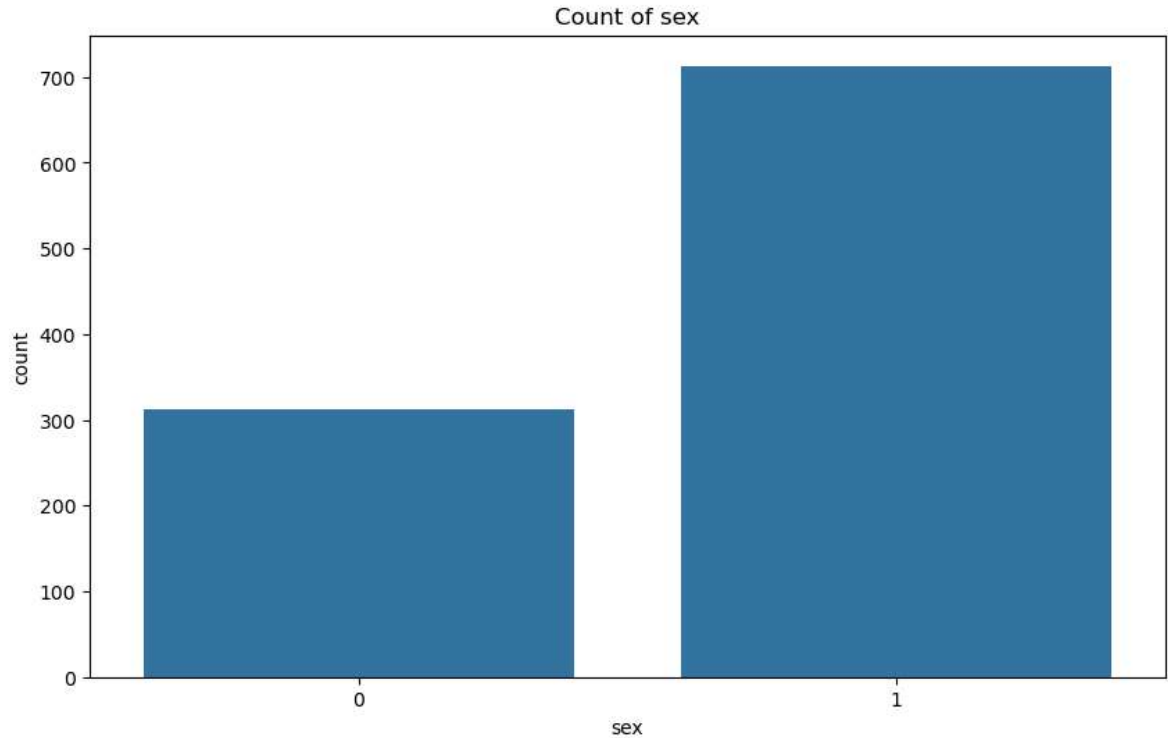
```
In [24]: features = ['age', 'trestbps', 'chol', 'thalach', 'oldpeak']
for i in features:
    plt.figure(figsize=(10, 6))
    sns.histplot(df[i], kde=True)
    plt.title(f'Distribution of {i}')
    plt.show()
```

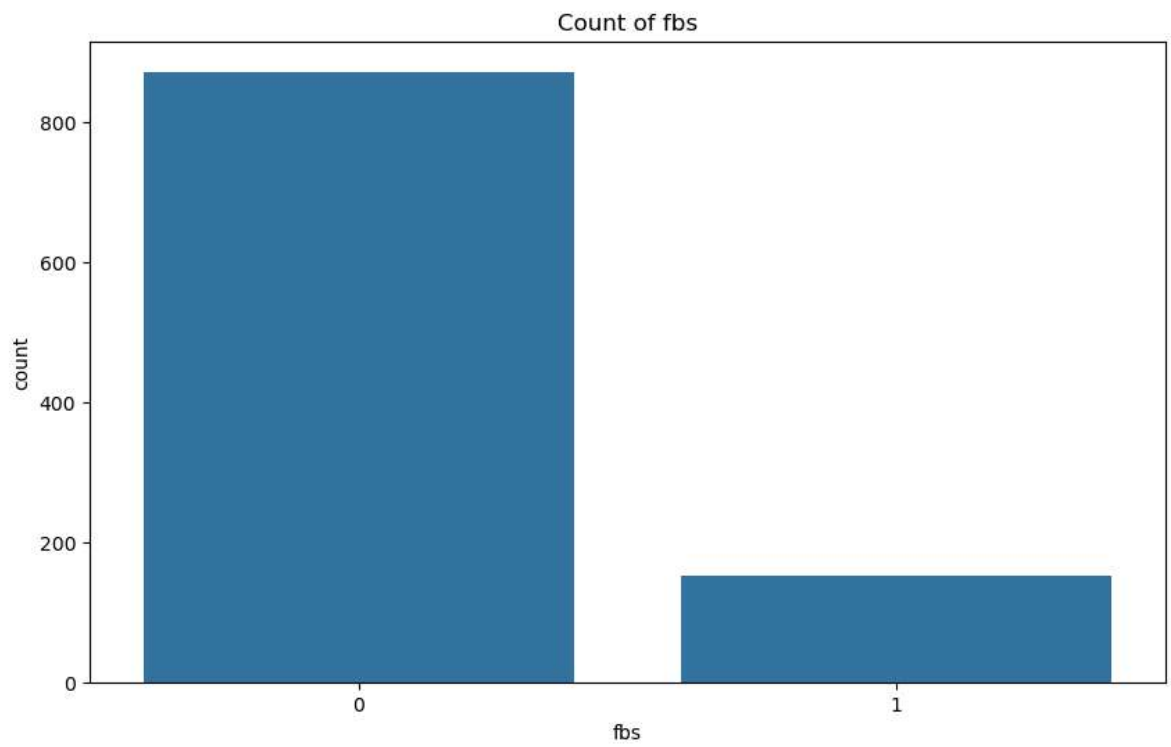
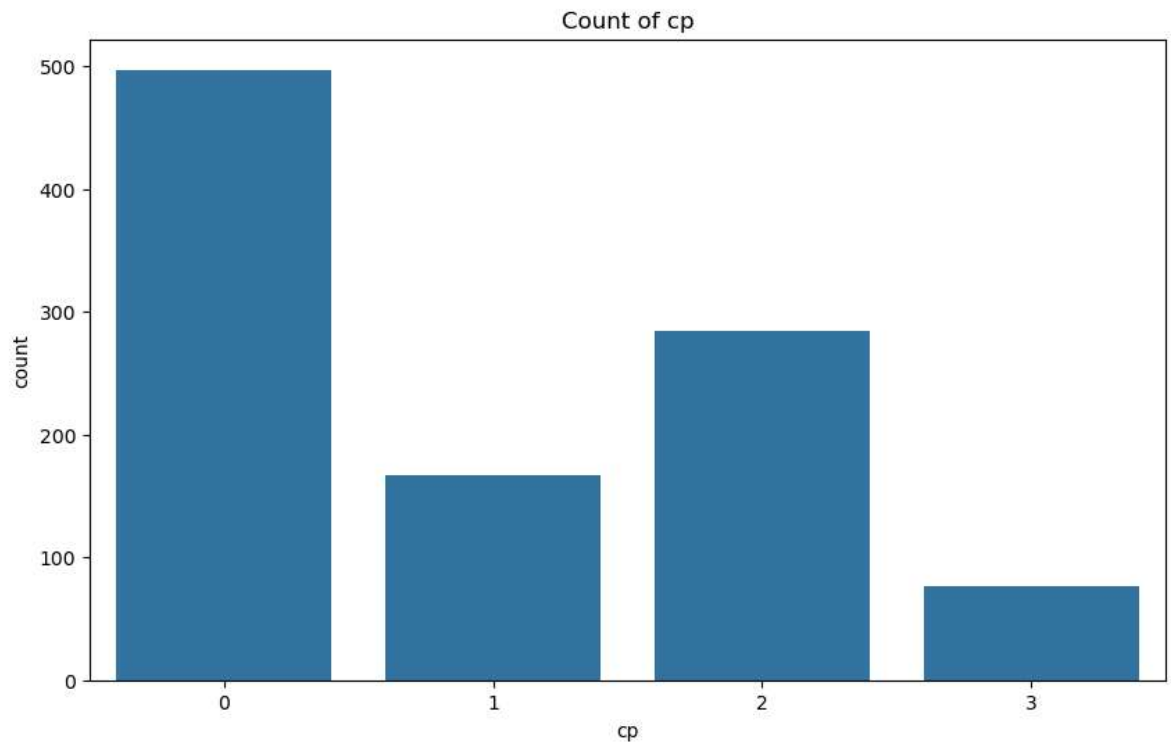


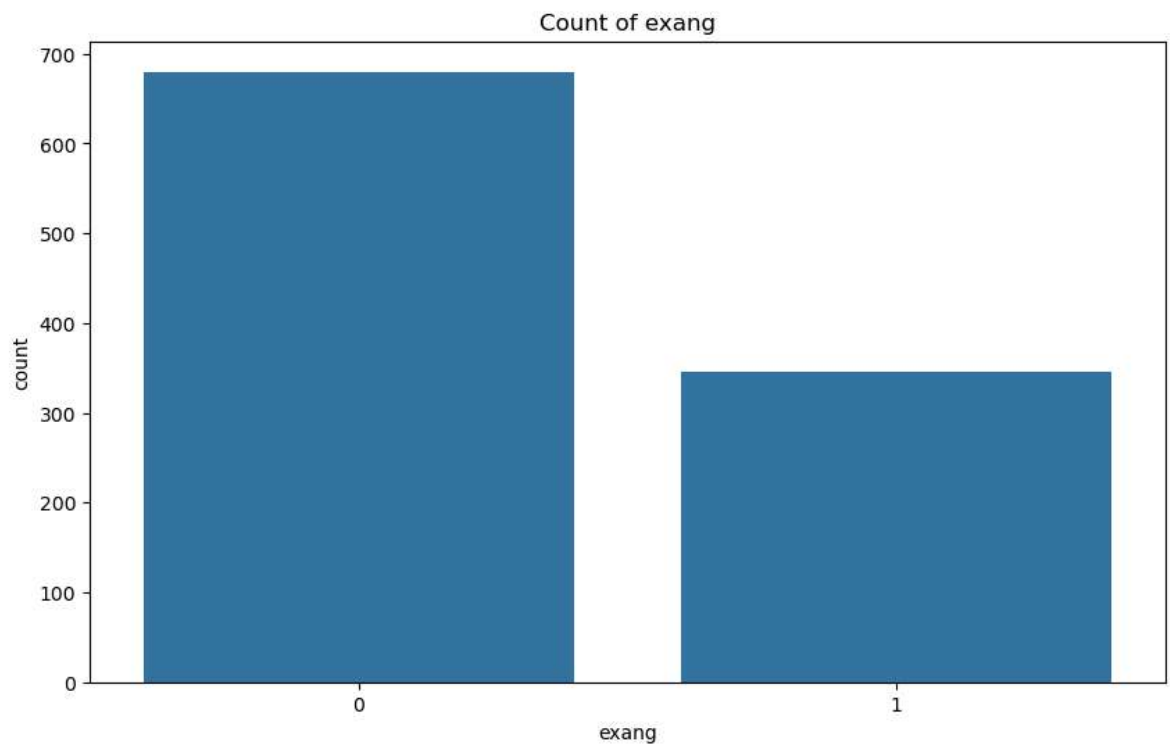
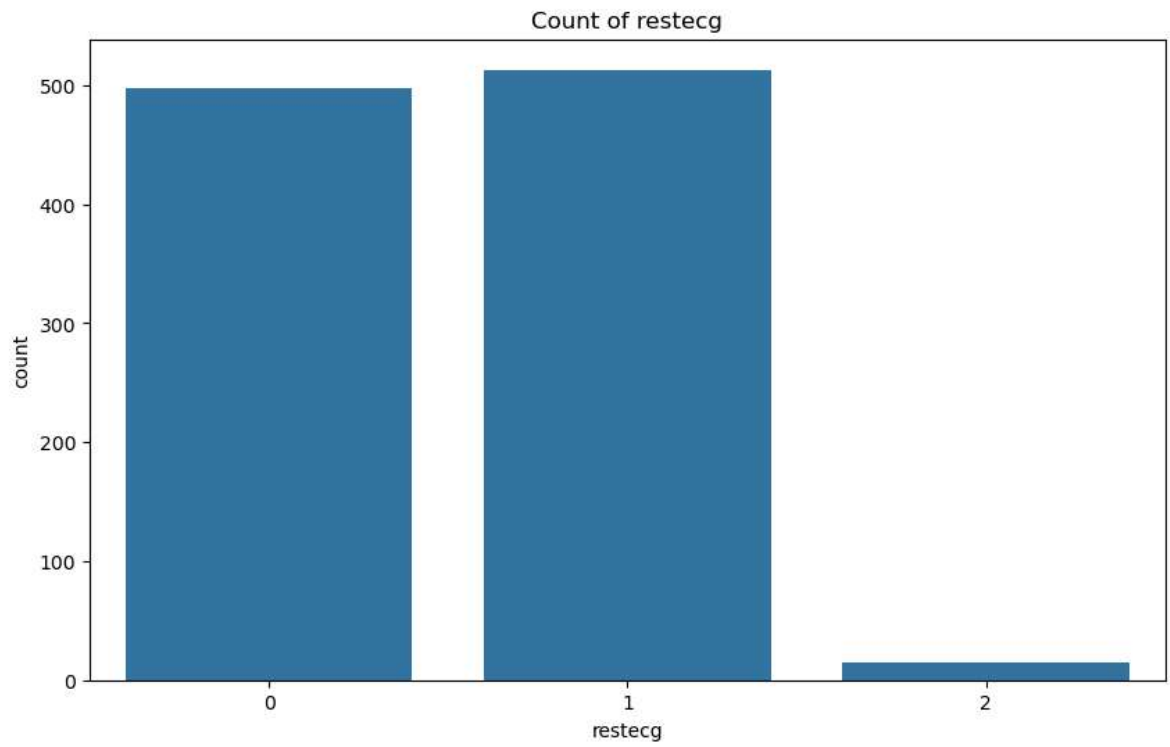


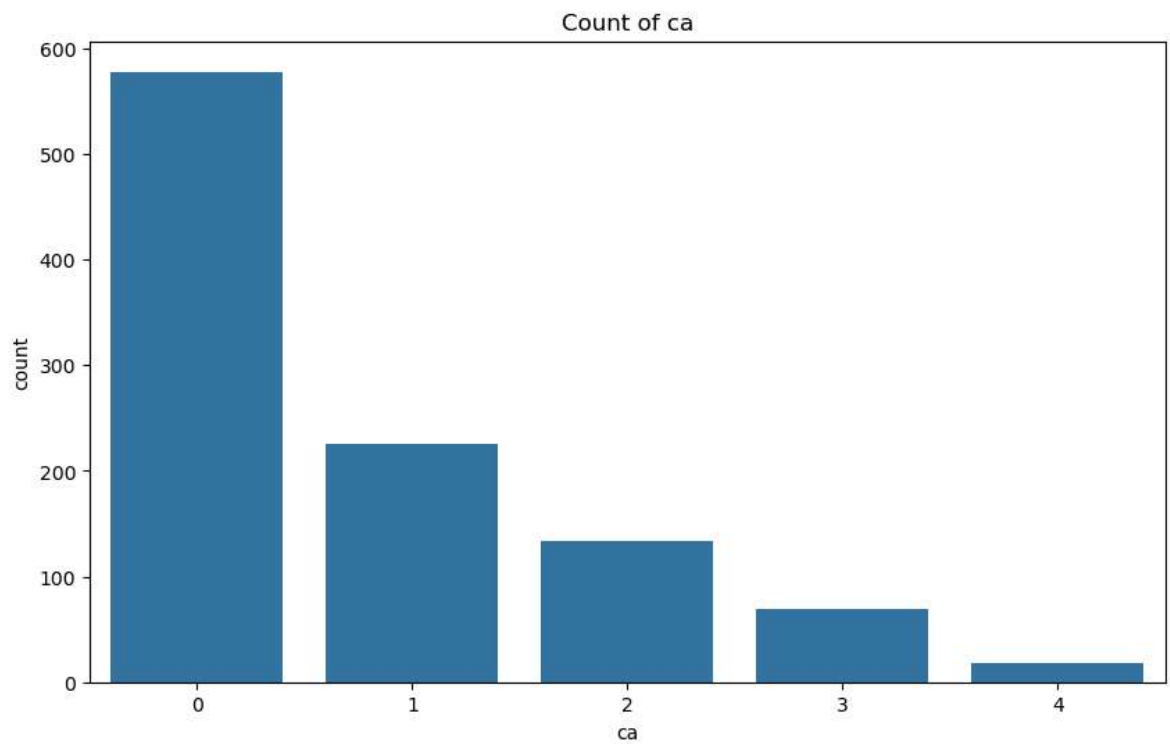
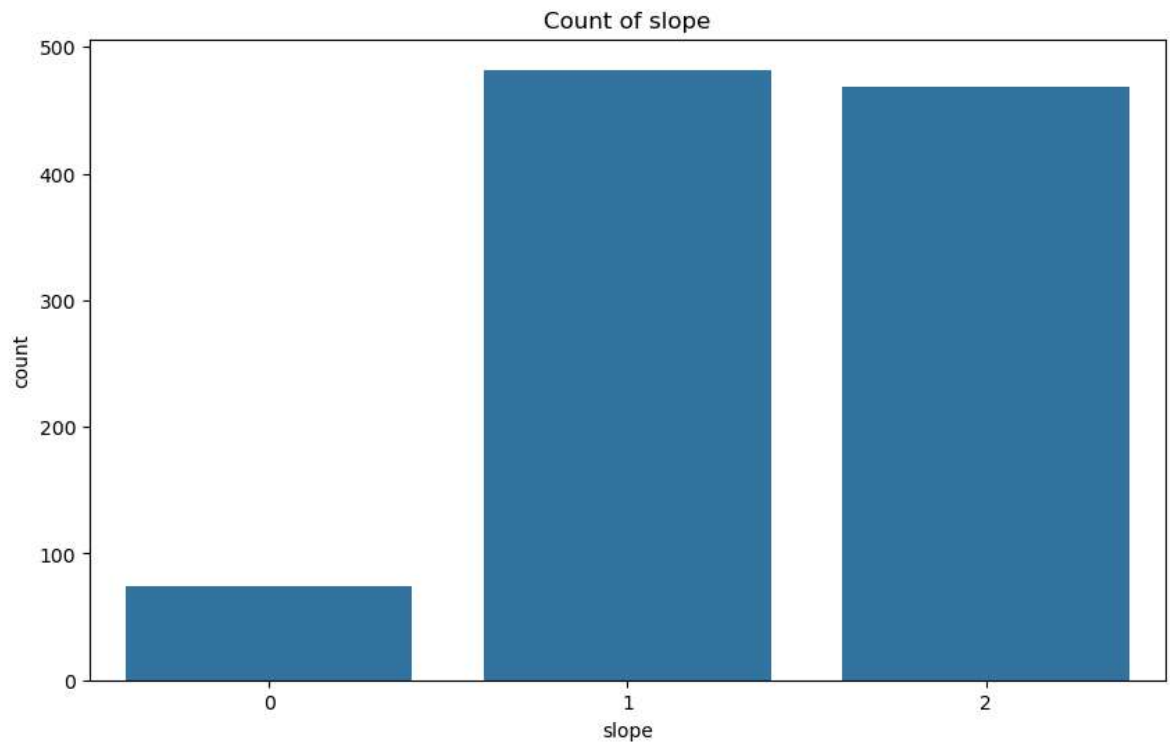


```
In [26]: extra_features = ['sex', 'cp', 'fbs', 'restecg', 'exang', 'slope', 'ca', 'thal',  
for i in extra_features:  
    plt.figure(figsize=(10, 6))  
    sns.countplot(x=i, data=data)  
    plt.title(f'Count of {i}')  
    plt.show()
```

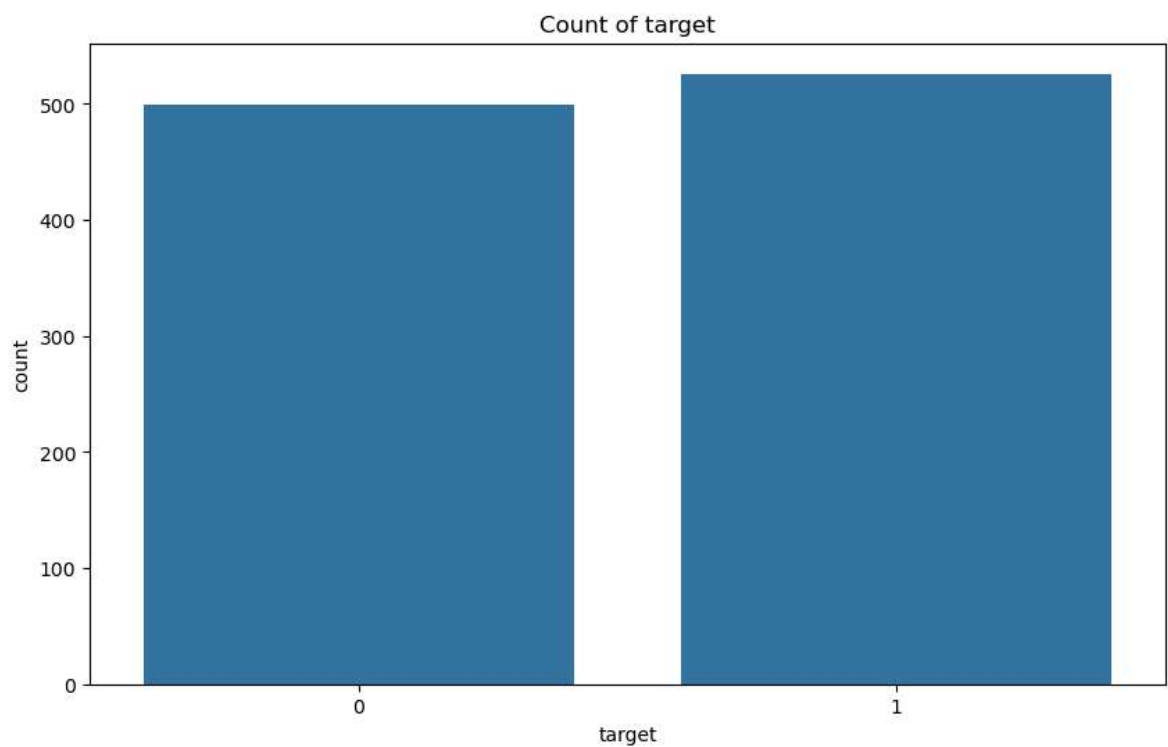
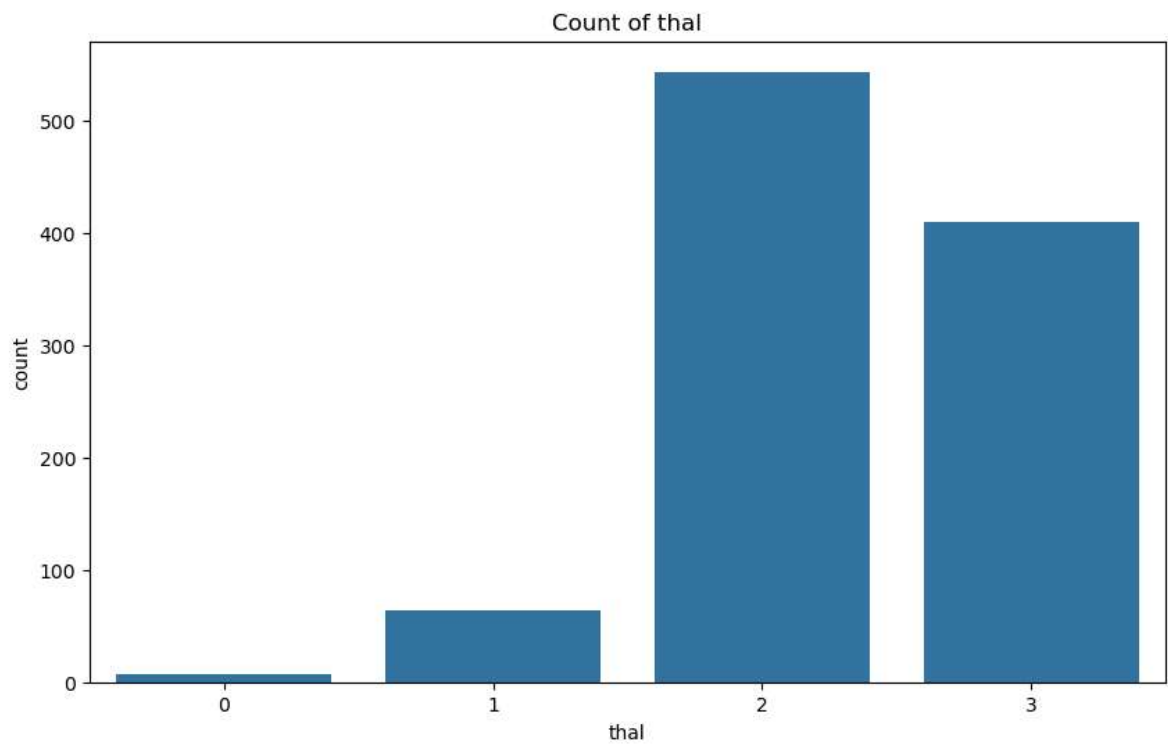




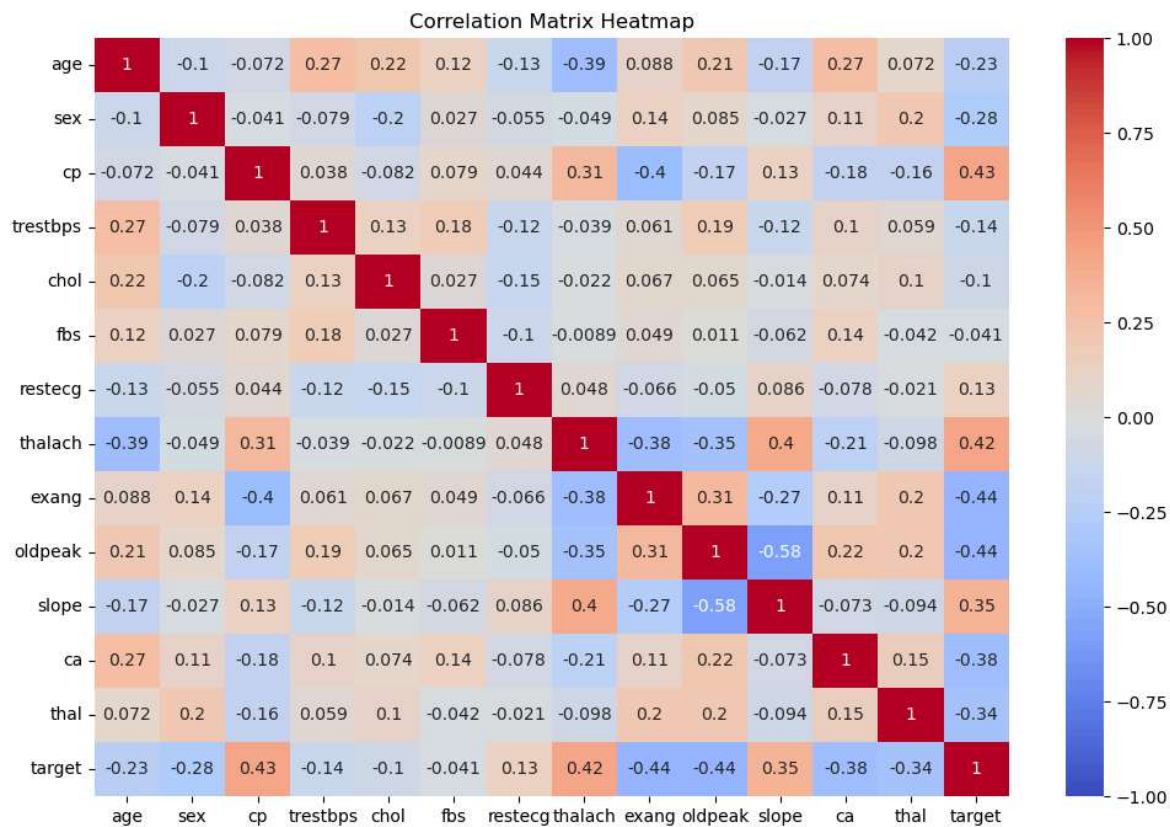






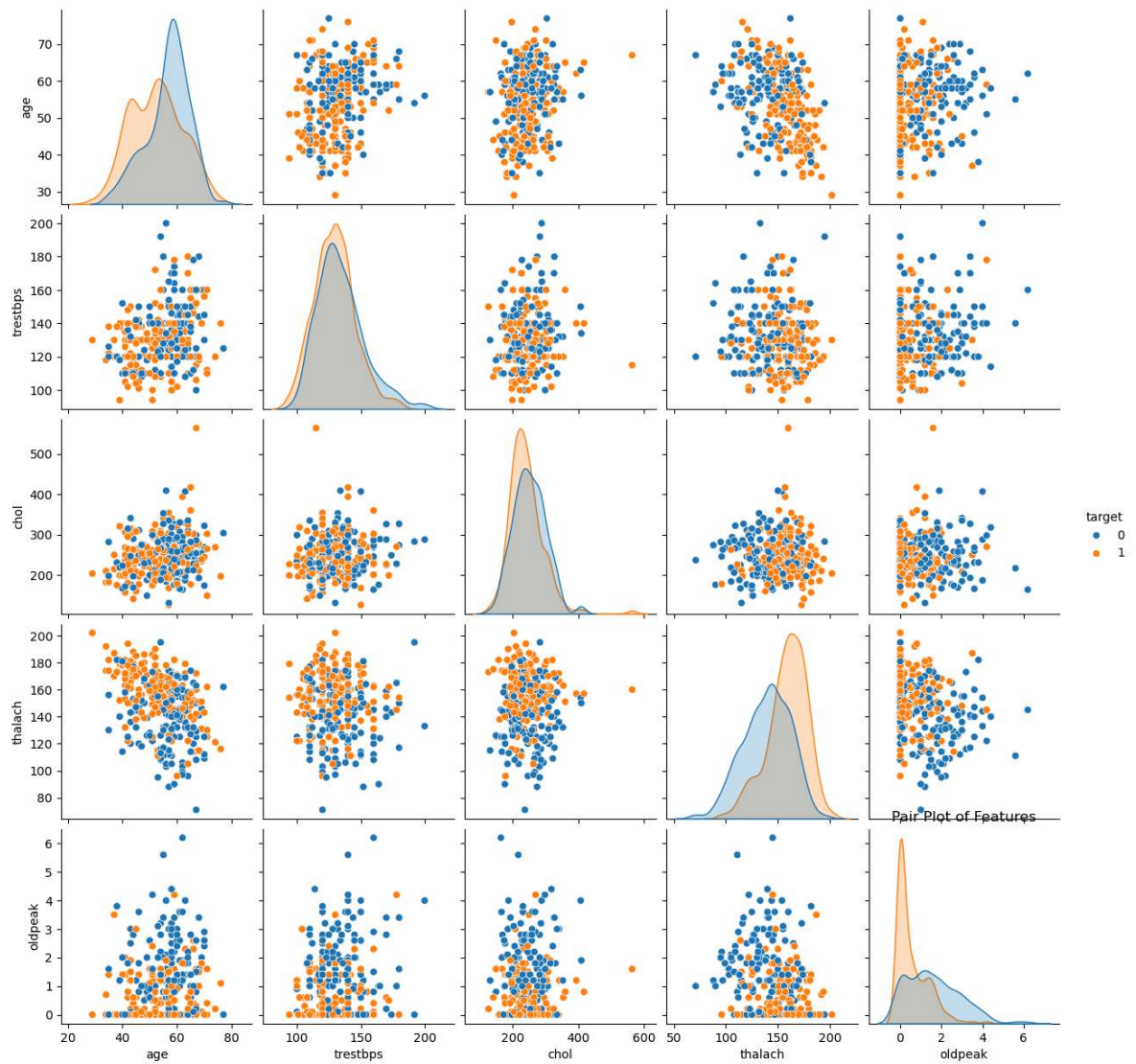


```
In [27]: correlation_matrix = data.corr()
plt.figure(figsize=(12, 8))
sns.heatmap(correlation_matrix, annot=True, cmap='coolwarm', vmin=-1, vmax=1)
plt.title('Correlation Matrix Heatmap')
plt.show()
```

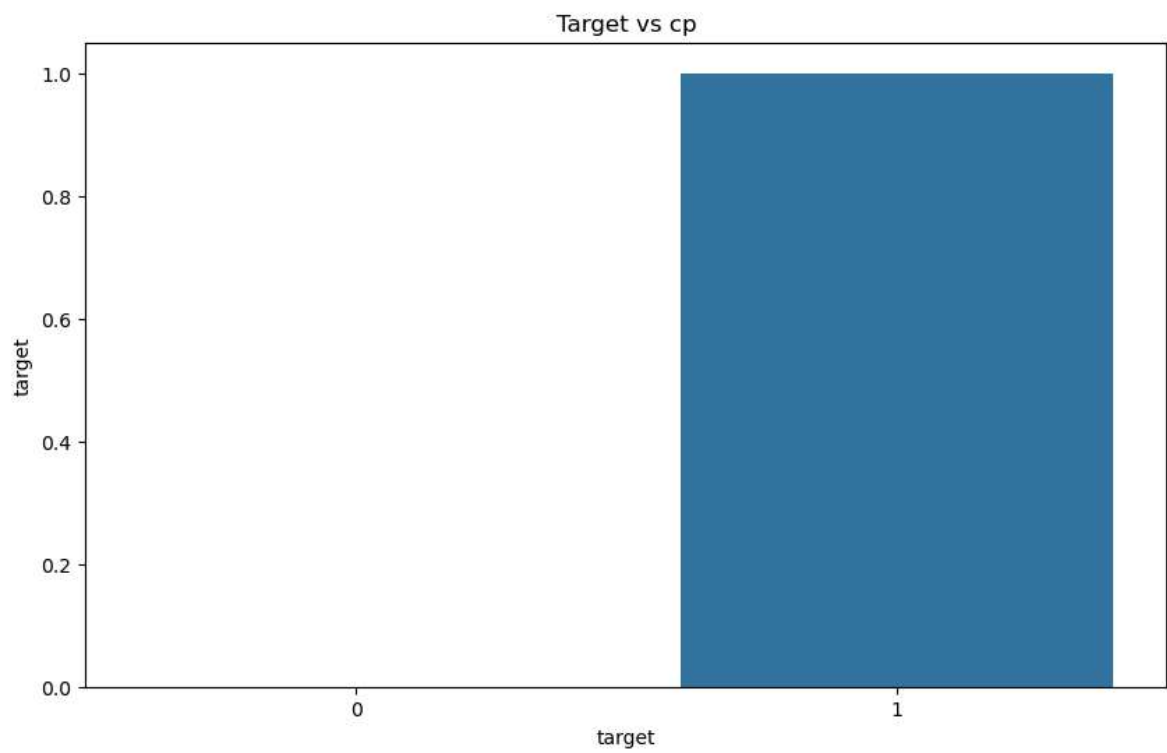
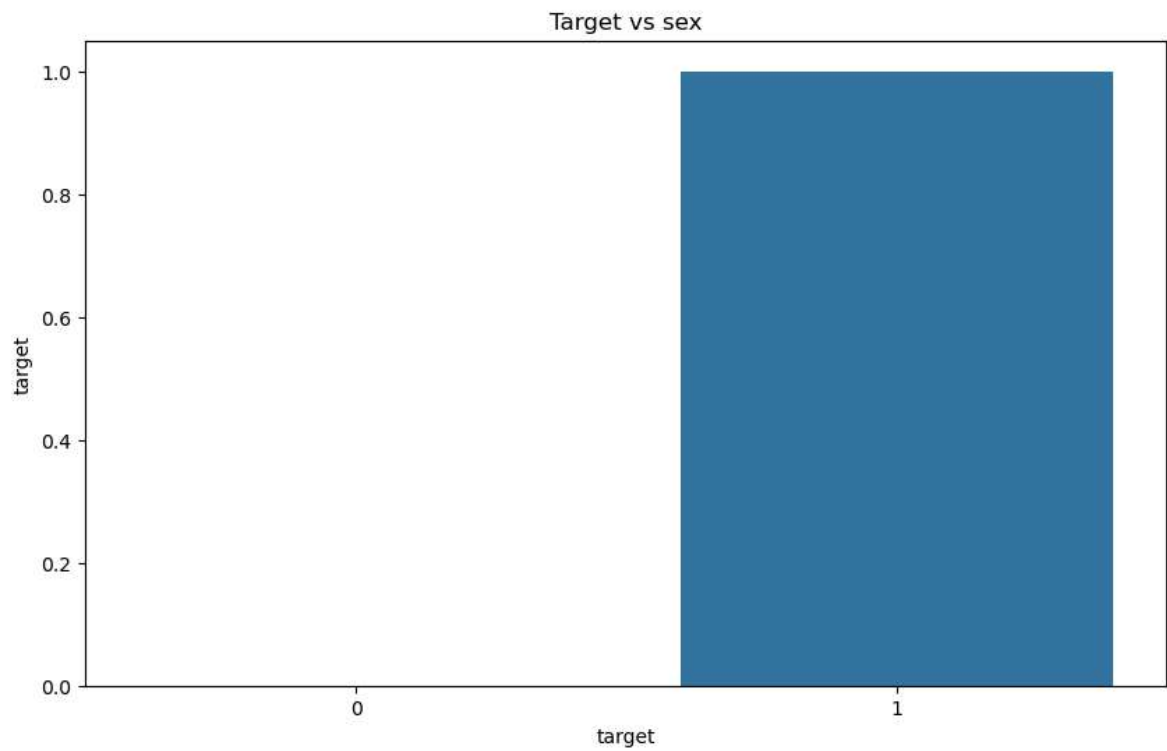


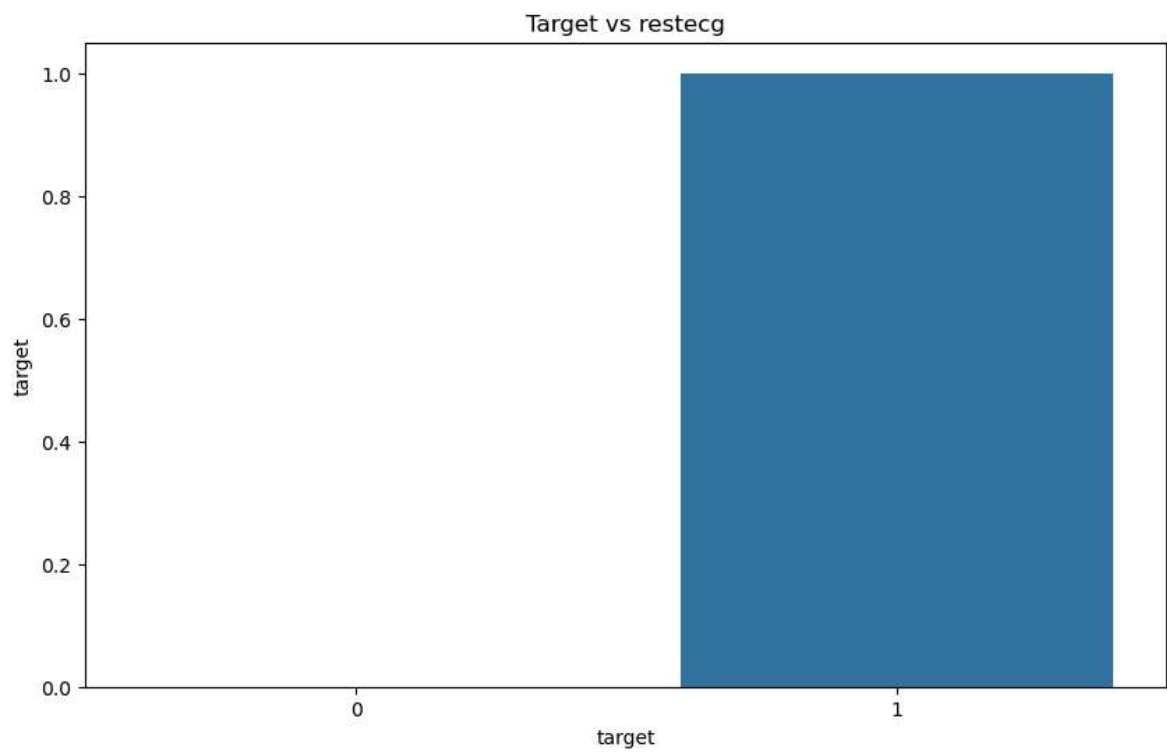
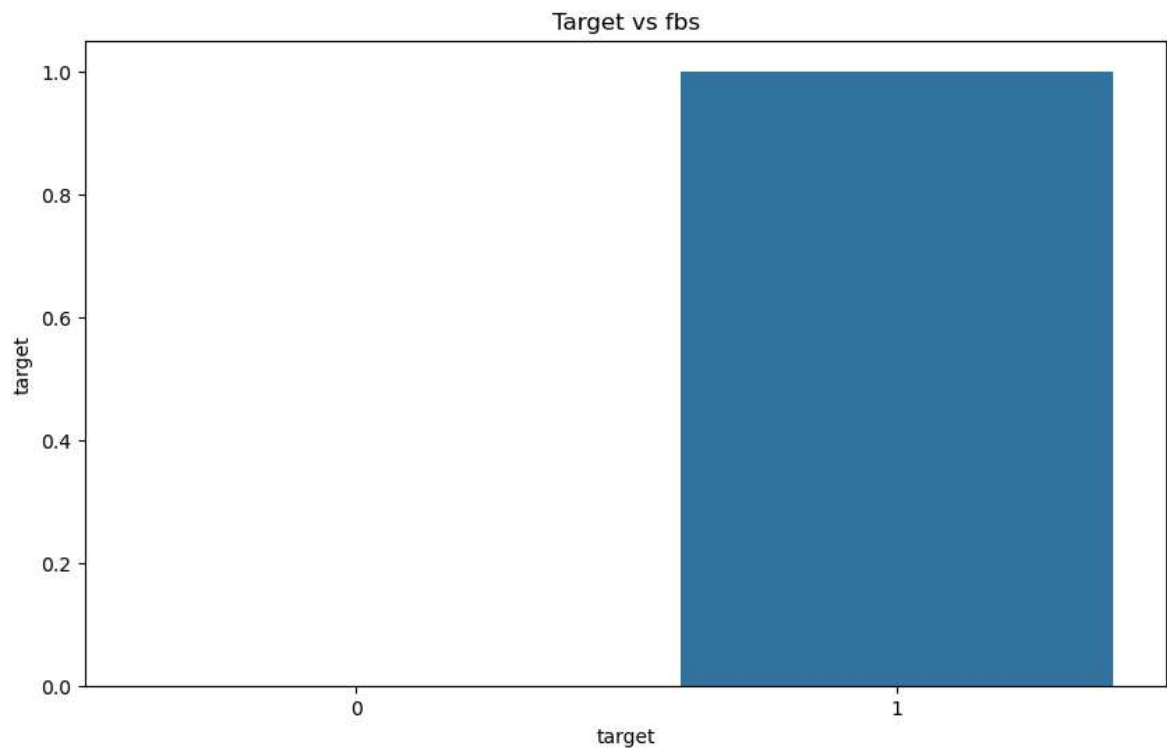
```
In [29]: plt.figure(figsize=(12, 8))
sns.pairplot(data, hue='target', vars=features)
plt.title('Pair Plot of Features')
plt.show()
```

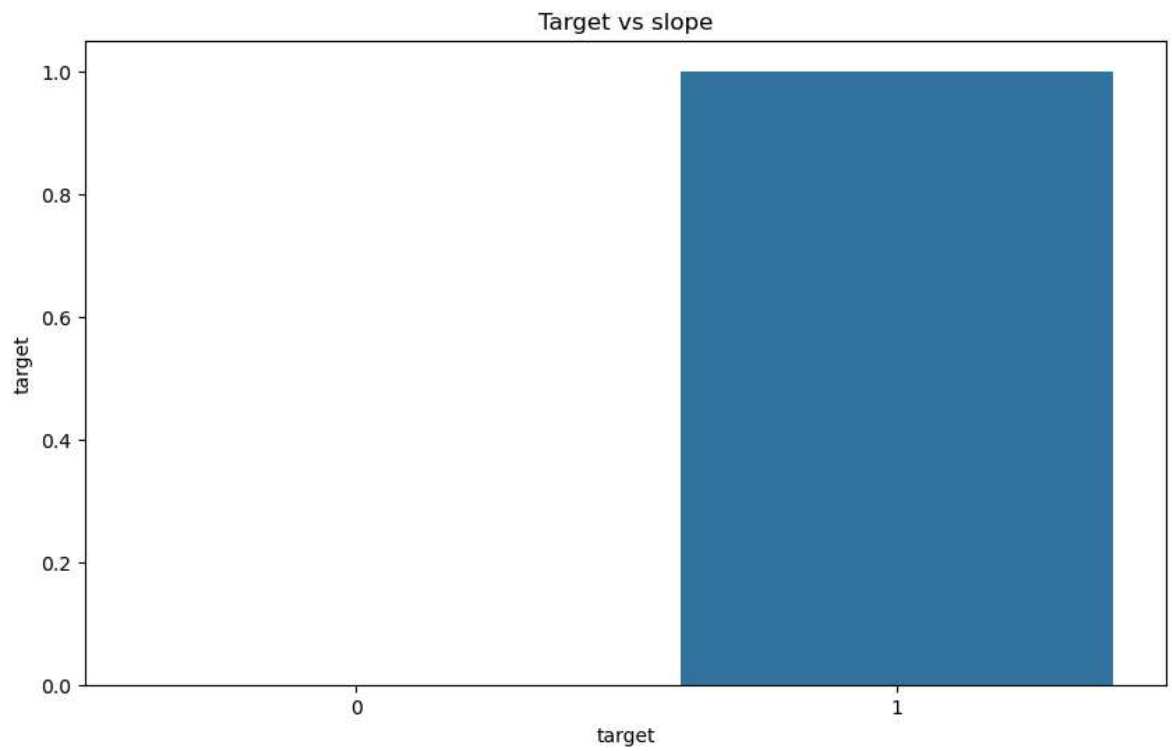
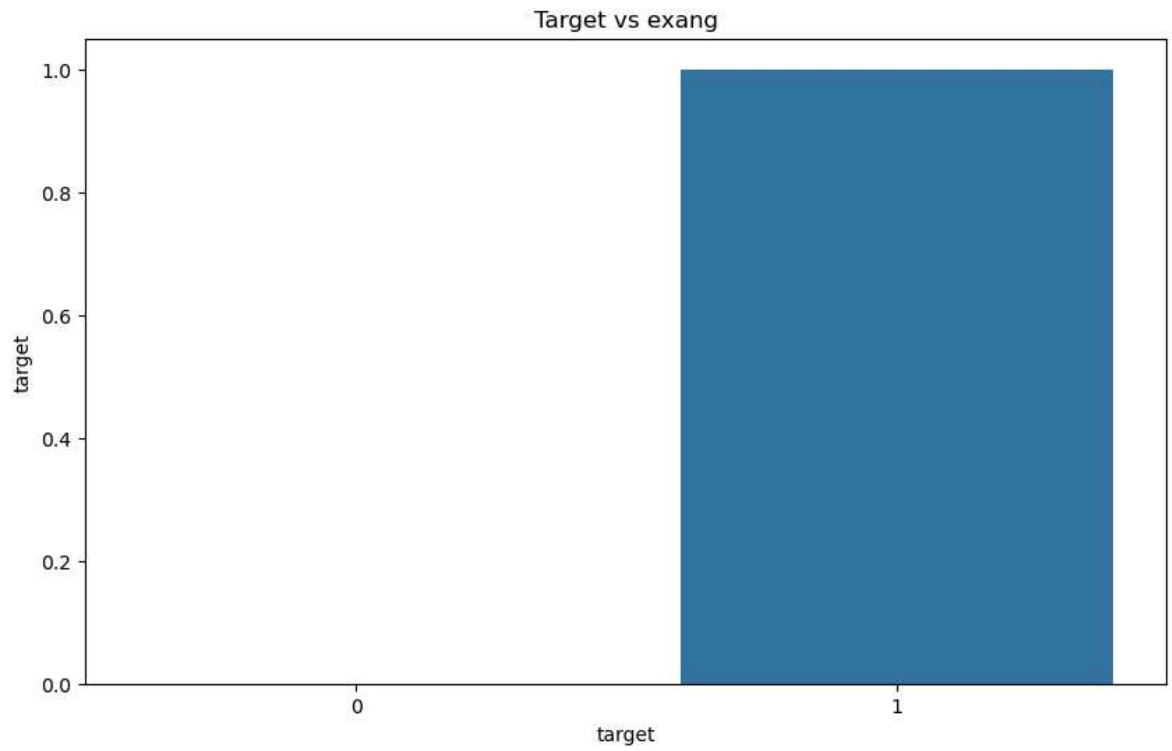
<Figure size 1200x800 with 0 Axes>

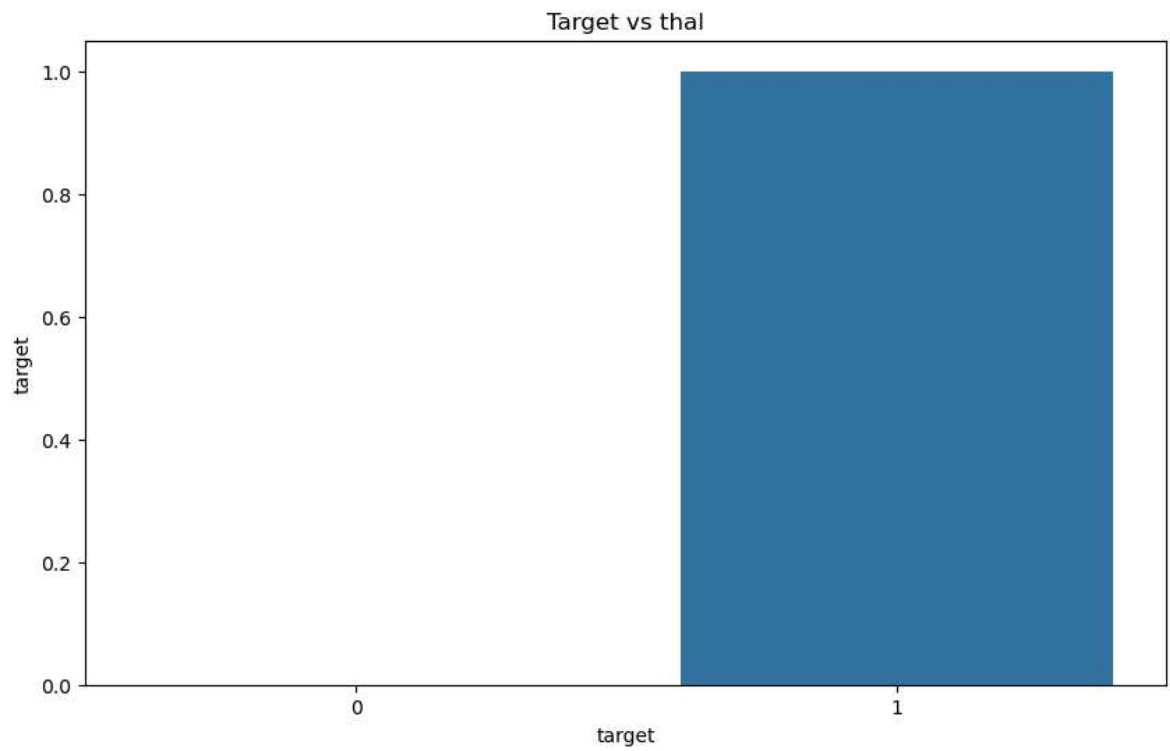
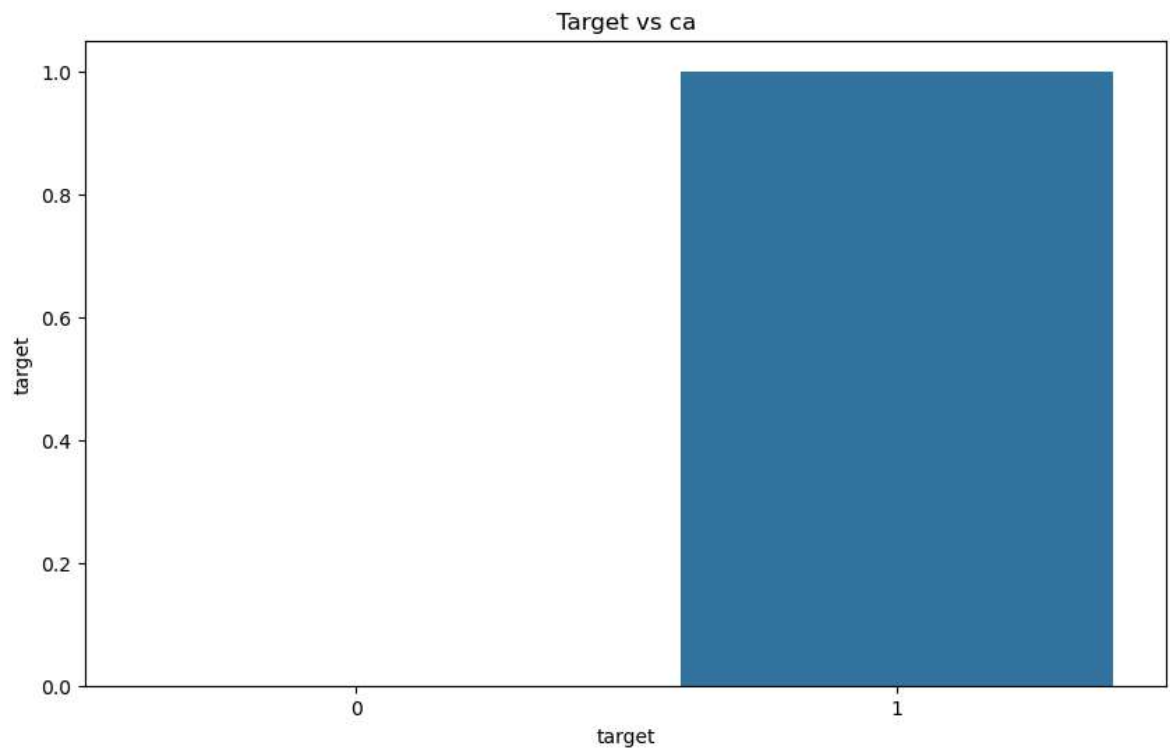


```
In [32]: for i in extra_features:
          if i != 'target':
              plt.figure(figsize=(10, 6))
              sns.barplot(x=feature, y='target', data=data, errorbar=None)
              plt.title(f'Target vs {i}')
              plt.show()
```









In [ ]: