In [1]:

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
```

In [2]:

```
df=pd.read_csv("E:\\heart_failure_clinical_records_dataset.csv")
```

In [3]:

```
df
```

Out[3]:

	age	anaemia	creatinine_phosphokinase	diabetes	ejection_fraction	high_blood_pressure
0	75.0	0	582	0	20	1
1	55.0	0	7861	0	38	0
2	65.0	0	146	0	20	0
3	50.0	1	111	0	20	0
4	65.0	1	160	1	20	0
...	...	...	...	...	...	...
294	62.0	0	61	1	38	1
295	55.0	0	1820	0	38	0
296	45.0	0	2060	1	60	0
297	45.0	0	2413	0	38	0
298	50.0	0	196	0	45	0

299 rows × 13 columns

In [5]:

```
df.isnull().sum()
```

Out[5]:

```
age                         0
anaemia                     0
creatinine_phosphokinase    0
diabetes                    0
ejection_fraction           0
high_blood_pressure         0
platelets                   0
serum_creatinine            0
serum_sodium                0
sex                         0
smoking                     0
time                        0
DEATH_EVENT                 0
dtype: int64
```

In [6]:

```
df.info
```

```
296                   0  742000.00                0.8          138    0
297                   0  140000.00                1.4          140    1
298                   0  395000.00                1.6          136    1

     smoking  time  DEATH_EVENT
0          0     4            1
1          0     6            1
2          1     7            1
3          0     7            1
4          0     8            1
..       ...   ...          ...
294        1   270            0
295        0   271            0
296        0   278            0
297        1   280            0
298        1   285            0

[299 rows x 13 columns]>
```

In [14]:

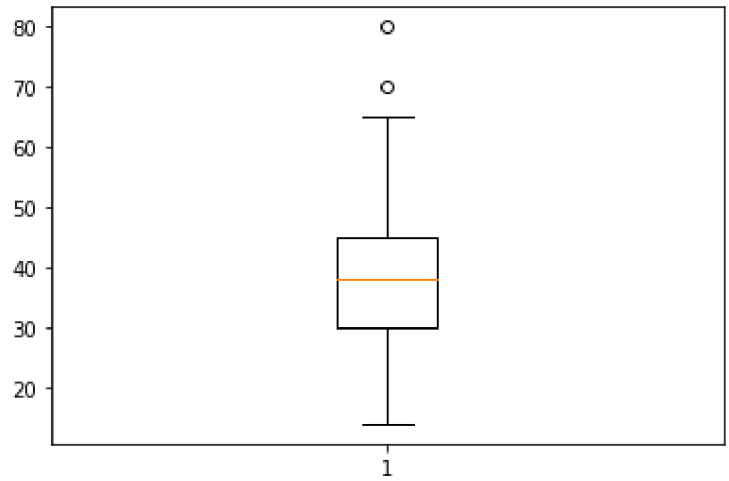
```
df.columns
```

Out[14]:

```
Index(['age', 'anaemia', 'creatinine_phosphokinase', 'diabetes',
       'ejection_fraction', 'high_blood_pressure', 'platelets',
       'serum_creatinine', 'serum_sodium', 'sex', 'smoking', 'time',
       'DEATH_EVENT'],
      dtype='object')
```

In [35]:

```
plt.boxplot(df.ejection_fraction)
plt.show()
```



In [50]:

```
#age count
x=df['age'].count()
x=df['age'].unique().count()
x
```

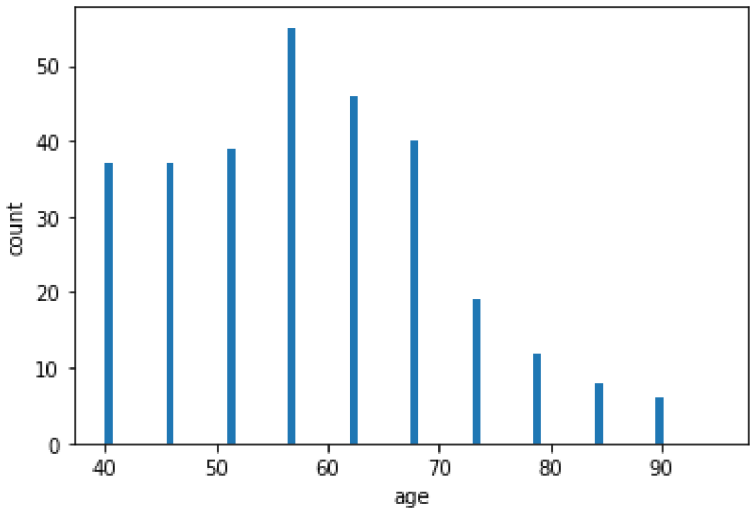
```
---------------------------------------------------------------------------
AttributeError                            Traceback (most recent call last)
<ipython-input-50-f7750eb588ca> in <module>
      1 #age count
      2 x=df['age'].count()
----> 3 x=df['age'].unique().count()
      4 x

AttributeError: 'numpy.ndarray' object has no attribute 'count'
```

In [68]:

```
plt.hist(df['age'],width=0.7)
plt.xlabel('age')
plt.ylabel('count')
```

Out[68]:

```
Text(0, 0.5, 'count')
```



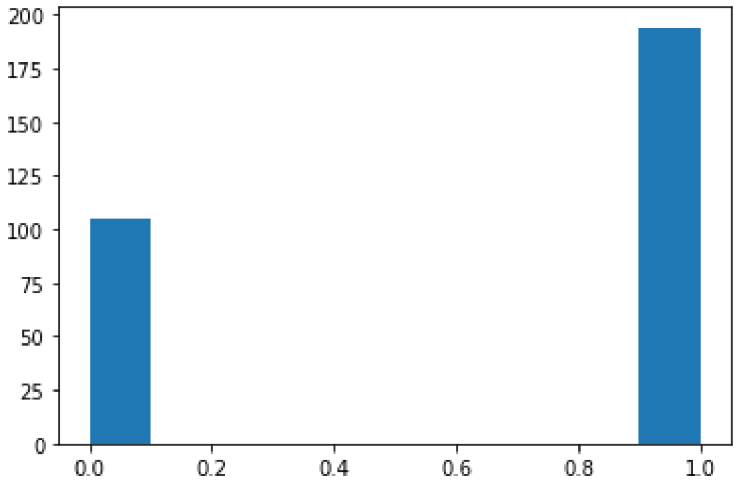
from this we can conclude that more number of patients are in age of 50-60

In [64]:

```
plt.hist(df['sex'])
```

Out[64]:

```
(array([105.,   0.,   0.,   0.,   0.,   0.,   0.,   0.,   0., 194.]),
 array([0. , 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1. ]),
 <BarContainer object of 10 artists>)
```



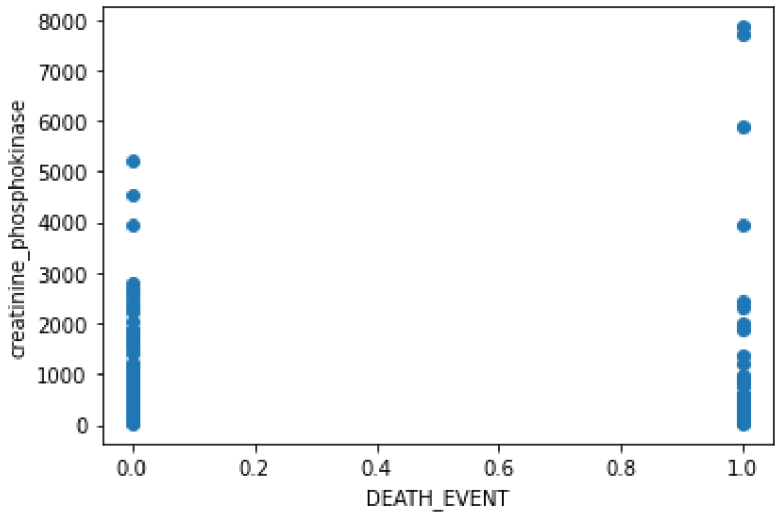

```
from research increased creatinine_phosphokinase is usually detected in someones heart
who is going to have heart attack in 4-6 hours
```

In [79]:

```
#visualize at how much rate increased creatinine_phosphokinase affect in death due to hear
plt.scatter(df['DEATH_EVENT'],df['creatinine_phosphokinase'])
plt.xlabel('DEATH_EVENT')
plt.ylabel('creatinine_phosphokinase')
```

Out[79]:

```
Text(0, 0.5, 'creatinine_phosphokinase')
```

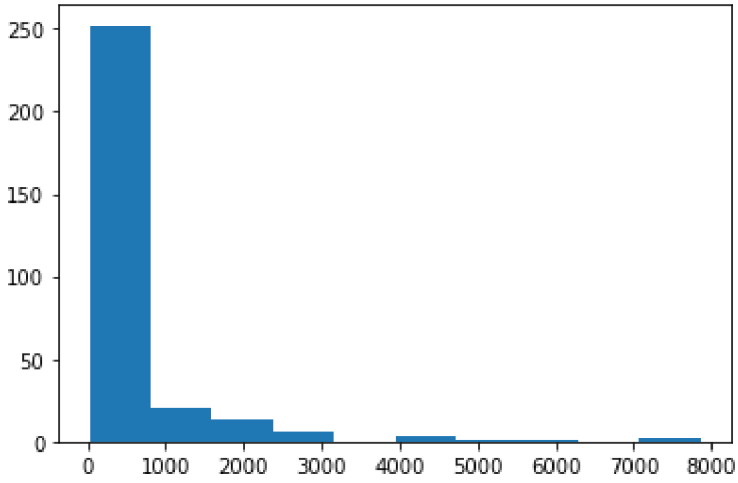


In [84]:

```
plt.hist(df['creatinine_phosphokinase'])
```

Out[84]:

```
(array([252.,  21.,  13.,   6.,   0.,   3.,   1.,   1.,   0.,   2.]),
 array([  23. ,  806.8, 1590.6, 2374.4, 3158.2, 3942. , 4725.8, 5509.6,
        6293.4, 7077.2, 7861. ]),
 <BarContainer object of 10 artists>)
```

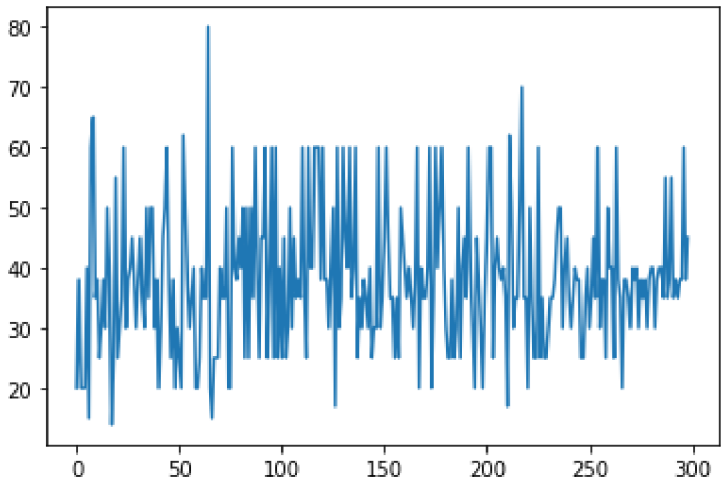


In [85]:

```
plt.plot(df['ejection_fraction'])
```

Out[85]:

```
[<matplotlib.lines.Line2D at 0xae293d0>]
```



In [90]:

```
x=df['ejection_fraction'].unique()
```

Out[90]:

```
array([20, 38, 40, 15, 60, 65, 35, 25, 30, 50, 14, 55, 45, 62, 80, 17, 70],
      dtype=int64)
```

In [95]:

```
plt.hist(x,width=1.0)
plt.xlabel('ejection_fraction')
plt.ylabel('count')
```

Out[95]:

```
Text(0, 0.5, 'count')
```