In [1]:

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

In [2]:

```
hf_raw_df = pd.read_csv("heart_failure_clinical_records_dataset.csv")
hf_raw_df.head()
```

Out[2]:

	age	anaemia	creatinine_phosphokinase	diabetes	ejection_fraction	high_blood_pressure	platelets	serum_creatinin
0	75.0	0	582	0	20	1	265000.00	1.
1	55.0	0	7861	0	38	0	263358.03	1.
2	65.0	0	146	0	20	0	162000.00	1.
3	50.0	1	111	0	20	0	210000.00	1.
4	65.0	1	160	1	20	0	327000.00	2.

In [3]:

```
heart_failure_df = hf_raw_df.copy()
heart_failure_df.head()
```

Out[3]:

	age	anaemia	creatinine_phosphokinase	diabetes	ejection_fraction	high_blood_pressure	platelets	serum_creatinin
0	75.0	0	582	0	20	1	265000.00	1.
1	55.0	0	7861	0	38	0	263358.03	1.
2	65.0	0	146	0	20	0	162000.00	1.
3	50.0	1	111	0	20	0	210000.00	1.
4	65.0	1	160	1	20	0	327000.00	2.

In [4]:

```
heart_failure_df.shape
```

Out[4]:

```
(299, 13)
```

In [5]:

```
heart_failure_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 299 entries, 0 to 298
Data columns (total 13 columns):
 #   Column                    Non-Null Count  Dtype  
---  ------                    --------------  -----  
 0   age                       299 non-null    float64
 1   anaemia                   299 non-null    int64  
 2   creatinine_phosphokinase  299 non-null    int64  
 3   diabetes                  299 non-null    int64  
 4   ejection_fraction         299 non-null    int64  
 5   high_blood_pressure       299 non-null    int64  
 6   platelets                 299 non-null    float64
 7   serum_creatinine          299 non-null    float64
 8   serum_sodium              299 non-null    int64  
 9   sex                       299 non-null    int64  
 10  smoking                   299 non-null    int64  
 11  time                      299 non-null    int64  
 12  DEATH_EVENT               299 non-null    int64  
dtypes: float64(3), int64(10)
memory usage: 30.5 KB
```

In [6]:

```
heart_failure_df.drop_duplicates().any()
```

Out[6]:

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

In [7]:

```
## Renaming the columns
heart_failure_df.rename(columns={"DEATH_EVENT": "patient_dead"},inplace=True)
```

In [8]:

```
heart_failure_df.head(1)
```

Out[8]:

	age	anaemia	creatinine_phosphokinase	diabetes	ejection_fraction	high_blood_pressure	platelets	serum_creatinine
0	75.0	0	582	0	20	1	265000.0	1.9

In [9]:

```
heart_failure_df.drop(['time','creatinine_phosphokinase'],axis=1,inplace=True)
```

In [10]:

```
heart_failure_df.shape
```

Out[10]:

```
(299, 11)
```

In [11]:

```
heart_failure_df.head()
```

Out[11]:

	age	anaemia	diabetes	ejection_fraction	high_blood_pressure	platelets	serum_creatinine	serum_sodium	sex	sm
0	75.0	0	0	20	1	265000.00	1.9	130	1	
1	55.0	0	0	38	0	263358.03	1.1	136	1	
2	65.0	0	0	20	0	162000.00	1.3	129	1	
3	50.0	1	0	20	0	210000.00	1.9	137	1	
4	65.0	1	1	20	0	327000.00	2.7	116	0	

In [12]:

```
## FLOAT TO INT
heart_failure_df.age = heart_failure_df.age.astype(int)
```

In [13]:
```
heart_failure_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 299 entries, 0 to 298
Data columns (total 11 columns):
 #   Column              Non-Null Count  Dtype  
---  ------              --------------  -----  
 0   age                 299 non-null    int32  
 1   anaemia             299 non-null    int64  
 2   diabetes            299 non-null    int64  
 3   ejection_fraction   299 non-null    int64  
 4   high_blood_pressure  299 non-null    int64  
 5   platelets           299 non-null    float64
 6   serum_creatinine    299 non-null    float64
 7   serum_sodium        299 non-null    int64  
 8   sex                 299 non-null    int64  
 9   smoking             299 non-null    int64  
 10  patient_dead        299 non-null    int64  
dtypes: float64(2), int32(1), int64(8)
memory usage: 24.7 KB
```

In [14]:
```
# Each type of integer has a different range of storage capacity

#    Type      Capacity

#    Int16 -- (-32,768 to +32,767)

#    Int32 -- (-2,147,483,648 to +2,147,483,647)

#    Int64 -- (-9,223,372,036,854,775,808 to +9,223,372,036,854,775,807)
```

In [15]:
```
### Convert Int32 to boolean only "0 & 1 " columns

heart_failure_df[['anaemia','diabetes','high_blood_pressure','smoking','patient_dead']] = heart_f
```

In [16]:
```
heart_failure_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 299 entries, 0 to 298
Data columns (total 11 columns):
 #   Column              Non-Null Count  Dtype  
---  ------              --------------  -----  
 0   age                 299 non-null    int32  
 1   anaemia             299 non-null    bool   
 2   diabetes            299 non-null    bool   
 3   ejection_fraction   299 non-null    int64  
 4   high_blood_pressure  299 non-null    bool   
 5   platelets           299 non-null    float64
 6   serum_creatinine    299 non-null    float64
 7   serum_sodium        299 non-null    int64  
 8   sex                 299 non-null    int64  
 9   smoking             299 non-null    bool   
 10  patient_dead        299 non-null    bool   
dtypes: bool(5), float64(2), int32(1), int64(3)
memory usage: 14.4 KB
```

In [17]:
```
heart_failure_df['sex'] = np.where(heart_failure_df['sex'] == 1,"Male","Female")
```

In [18]:
```
heart_failure_df.head()
```

Out[18]:

	age	anaemia	diabetes	ejection_fraction	high_blood_pressure	platelets	serum_creatinine	serum_sodium	sex
0	75	False	False	20	True	265000.00	1.9	130	Male
1	55	False	False	38	False	263358.03	1.1	136	Male
2	65	False	False	20	False	162000.00	1.3	129	Male
3	50	True	False	20	False	210000.00	1.9	137	Male
4	65	True	True	20	False	327000.00	2.7	116	Female

In [19]:
```
heart_failure_df['platelets'] = (heart_failure_df.platelets/1000).astype(int)
```

In [20]:
```
heart_failure_df.head()
```

Out[20]:

	age	anaemia	diabetes	ejection_fraction	high_blood_pressure	platelets	serum_creatinine	serum_sodium	sex	s
0	75	False	False	20	True	265	1.9	130	Male	
1	55	False	False	38	False	263	1.1	136	Male	
2	65	False	False	20	False	162	1.3	129	Male	
3	50	True	False	20	False	210	1.9	137	Male	
4	65	True	True	20	False	327	2.7	116	Female	

In [21]:
```
## Check the null values
heart_failure_df.isnull().sum()
# heart_failure_df.isnull().any()
```

Out[21]:
```
age                    0
anaemia                0
diabetes               0
ejection_fraction      0
high_blood_pressure    0
platelets              0
serum_creatinine       0
serum_sodium           0
sex                    0
smoking                0
patient_dead           0
dtype: int64
```

In [22]:
```
len(heart_failure_df.columns)
```

Out[22]:
```
11
```

In [23]:
```
!pip install lxml
```

```
Requirement already satisfied: lxml in c:\users\dhruv\appdata\local\programs\python\python38\lib
\site-packages (4.9.3)
```

In [24]:
```
column_deatils_df = pd.read_html("https://bmcmedinformdecismak.biomedcentral.com/articles/10.1186/
```

In [25]:
```
column_deatils_df
```

Out[25]:

	Feature	Explanation	Measurement	Range
0	Age	Age of the patient	Years	[40,..., 95]
1	Anaemia	Decrease of red blood cells or hemoglobin	Boolean	0, 1
2	High blood pressure	If a patient has hypertension	Boolean	0, 1
3	Creatinine phosphokinase	Level of the CPK enzyme in the blood	mcg/L	[23,..., 7861]
4	(CPK)	NaN	NaN	NaN
5	Diabetes	If the patient has diabetes	Boolean	0, 1
6	Ejection fraction	Percentage of blood leaving	Percentage	[14,..., 80]
7	NaN	the heart at each contraction	NaN	NaN
8	Sex	Woman or man	Binary	0, 1
9	Platelets	Platelets in the blood	kiloplatelets/mL	[25.01,..., 850.00]
10	Serum creatinine	Level of creatinine in the blood	mg/dL	[0.50,..., 9.40]
11	Serum sodium	Level of sodium in the blood	mEq/L	[114,..., 148]
12	Smoking	If the patient smokes	Boolean	0, 1
13	Time	Follow-up period	Days	[4,...,285]
14	(target) death event	If the patient died during the follow-up period	Boolean	0, 1

In [26]:
```
column_deatils_df.drop('Range',axis=1,inplace=True)
```

In [27]:
```
column_deatils_df.drop([3,4,7,13],axis=0,inplace=True)
```

In [28]:
```
column_deatils_df.columns = ['feature','explanation','measurement_unit']
```

In [29]:
```
column_deatils_df
```

Out[29]:

	feature	explanation	measurement_unit
0	Age	Age of the patient	Years
1	Anaemia	Decrease of red blood cells or hemoglobin	Boolean
2	High blood pressure	If a patient has hypertension	Boolean
5	Diabetes	If the patient has diabetes	Boolean
6	Ejection fraction	Percentage of blood leaving	Percentage
8	Sex	Woman or man	Binary
9	Platelets	Platelets in the blood	kiloplatelets/mL
10	Serum creatinine	Level of creatinine in the blood	mg/dL
11	Serum sodium	Level of sodium in the blood	mEq/L
12	Smoking	If the patient smokes	Boolean
14	(target) death event	If the patient died during the follow-up period	Boolean

In [30]:
```
heart_failure_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 299 entries, 0 to 298
Data columns (total 11 columns):
 #   Column               Non-Null Count  Dtype  
---  ------               --------------  -----  
 0   age                  299 non-null    int32  
 1   anaemia              299 non-null    bool   
 2   diabetes             299 non-null    bool   
 3   ejection_fraction    299 non-null    int64  
 4   high_blood_pressure  299 non-null    bool   
 5   platelets            299 non-null    int32  
 6   serum_creatinine     299 non-null    float64
 7   serum_sodium         299 non-null    int64  
 8   sex                  299 non-null    object 
 9   smoking              299 non-null    bool   
 10  patient_dead         299 non-null    bool   
dtypes: bool(5), float64(1), int32(2), int64(2), object(1)
memory usage: 13.3+ KB
```

In [31]:
```
column_deatils_df = column_deatils_df.reindex([0,1,5,6,2,9,10,11,8,12,14])
```

In [32]:
```
column_deatils_df
```

Out[32]:

	feature	explanation	measurement_unit
0	Age	Age of the patient	Years
1	Anaemia	Decrease of red blood cells or hemoglobin	Boolean
5	Diabetes	If the patient has diabetes	Boolean
6	Ejection fraction	Percentage of blood leaving	Percentage
2	High blood pressure	If a patient has hypertension	Boolean
9	Platelets	Platelets in the blood	kiloplatelets/mL
10	Serum creatinine	Level of creatinine in the blood	mg/dL
11	Serum sodium	Level of sodium in the blood	mEq/L
8	Sex	Woman or man	Binary
12	Smoking	If the patient smokes	Boolean
14	(target) death event	If the patient died during the follow-up period	Boolean

In [33]:
```
column_deatils_df.feature= heart_failure_df.columns
```

In [34]:
```
column_deatils_df.feature
```

Out[34]:
```
0                     age
1                 anaemia
5                diabetes
6       ejection_fraction
2     high_blood_pressure
9               platelets
10       serum_creatinine
11           serum_sodium
8                     sex
12                smoking
14           patient_dead
Name: feature, dtype: object
```

In [40]:
```
column_deatils_df.explanation
```

Out[40]:
```
0                                   Age of the patient
1           Decrease of red blood cells or hemoglobin
5                         If the patient has diabetes
6                         Percentage of blood leaving
2                       If a patient has hypertension
9                              Platelets in the blood
10                   Level of creatinine in the blood
11                       Level of sodium in the blood
8                                        Woman or man
12                              If the patient smokes
14    If the patient died during the follow-up period
Name: explanation, dtype: object
```

In [41]:
```
column_deatils_df
```

Out[41]:

	feature	explanation	measurement_unit
0	age	Age of the patient	Years
1	anaemia	Decrease of red blood cells or hemoglobin	Boolean
5	diabetes	If the patient has diabetes	Boolean
6	ejection_fraction	Percentage of blood leaving	Percentage
2	high_blood_pressure	If a patient has hypertension	Boolean
9	platelets	Platelets in the blood	kiloplatelets/mL
10	serum_creatinine	Level of creatinine in the blood	mg/dL
11	serum_sodium	Level of sodium in the blood	mEq/L
8	sex	Woman or man	Binary
12	smoking	If the patient smokes	Boolean
14	patient_dead	If the patient died during the follow-up period	Boolean

In [36]:
```
################################################
```

In [42]:
```
#to set the feature column as index for our convenience
column_deatils_df.set_index(['feature'], inplace =True)
```

In [43]:
```
column_deatils_df
```

Out[43]:

	explanation	measurement_unit
feature		
age	Age of the patient	Years
anaemia	Decrease of red blood cells or hemoglobin	Boolean
diabetes	If the patient has diabetes	Boolean
ejection_fraction	Percentage of blood leaving	Percentage
high_blood_pressure	If a patient has hypertension	Boolean
platelets	Platelets in the blood	kiloplatelets/mL
serum_creatinine	Level of creatinine in the blood	mg/dL
serum_sodium	Level of sodium in the blood	mEq/L
sex	Woman or man	Binary
smoking	If the patient smokes	Boolean
patient_dead	If the patient died during the follow-up period	Boolean

In [44]:
```
#to change the details in explanation column
column_deatils_df['explanation']['anaemia', 'diabetes', 'ejection_fraction', 'high_blood_pressure
```

In [45]:
```
column_deatils_df['explanation']
```

Out[45]:
```
feature
age                                                   Age of the patient
anaemia                                 True, if the patient has Anaemia
diabetes                               True, if the patient has Diabetes
ejection_fraction       % of blood leaving the heart at each contraction
high_blood_pressure         True, if the patient has High blood pressure
platelets                               Amount of platelets in the blood
serum_creatinine                        Level of creatinine in the blood
serum_sodium                                Level of sodium in the blood
sex                                                       Male or Female
smoking                                      True, if the patient smokes
patient_dead           True, if the patient died during the follow-up...
Name: explanation, dtype: object
```

In [47]:
```
#to change the details in measurement unit column
column_deatils_df.measurement_unit['sex', 'platelets','serum_creatinine','serum_sodium'] = ['Boole
                                                                                            'kilo-
                                                                                            'mg/dL
                                                                                            'mEq/L
                                                                                            ]
```

In [48]:
```
#let's add another column to mention normal values of the attributes
column_deatils_df["normal_value"] = ['None',
                                    'None',
                                    'None',
                                    '55% - 70%',
                                    'None',
                                    '150 - 400 kilo-platelets / mcL',
                                    '0.6 - 1.2 mg/dL',
                                    '135 - 145 mEq /L',
                                    'None', 'None', 'None'
                                   ]
```

In [49]:
```
column_deatils_df
```

Out[49]:

feature	explanation	measurement_unit	normal_value
age	Age of the patient	Years	None
anaemia	True, if the patient has Anaemia	Boolean	None
diabetes	True, if the patient has Diabetes	Boolean	None
ejection_fraction	% of blood leaving the heart at each contraction	Percentage	55% - 70%
high_blood_pressure	True, if the patient has High blood pressure	Boolean	None
platelets	Amount of platelets in the blood	kilo-platelets / mcL (microliter)	150 - 400 kilo-platelets / mcL
serum_creatinine	Level of creatinine in the blood	mg/dL (milligrams per deciliter)	0.6 - 1.2 mg/dL
serum_sodium	Level of sodium in the blood	mEq/L (milliequivalents per litre)	135 - 145 mEq /L
sex	Male or Female	Boolean	None
smoking	True, if the patient smokes	Boolean	None
patient_dead	True, if the patient died during the follow-up...	Boolean	None

In [ ]:
```
# Question 1:

# How many number of patient are there in our observation? out of them how many male and female p
```

In [54]:
```
heart_failure_df.shape[0]
```

Out[54]:
```
299
```

In [59]:
```
heart_failure_df.sex.value_counts()
```

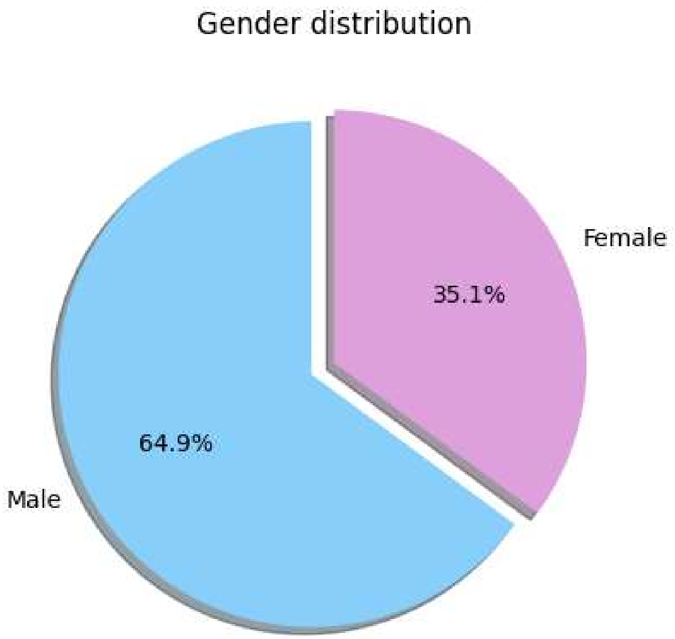
Out[59]:
```
Male      194
Female    105
Name: sex, dtype: int64
```

In [60]:
```
print(f'total number of patient in our observation is {heart_failure_df.shape[0]}')
#heart_failure_df.sex.value_counts()
print(f'number of Male patient in our observation is {heart_failure_df.sex.value_counts()[0]}')
print(f'number of FeMale patient in our observation is {heart_failure_df.sex.value_counts()[1]}')
```

```
total number of patient in our observation is 299
number of Male patient in our observation is 194
number of FeMale patient in our observation is 105
```

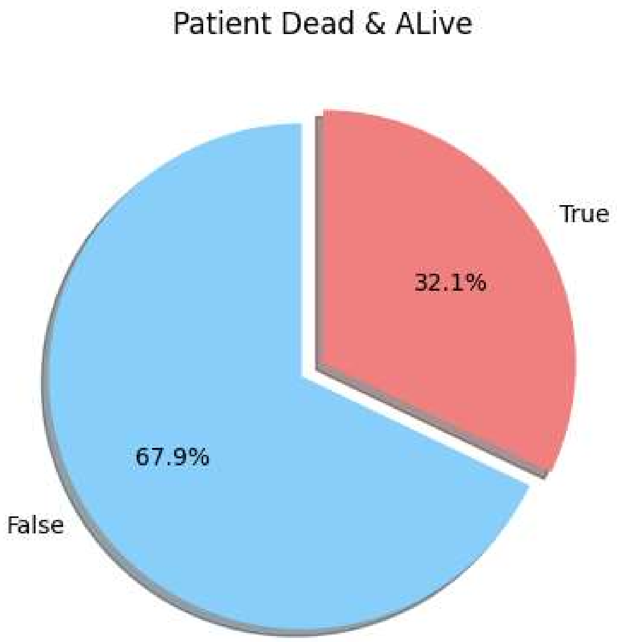
In [61]:

```
gender_counts = heart_failure_df.sex.value_counts()

plt.pie(gender_counts,labels=gender_counts.index,autopct='%.1f%%',
        explode=[0.1,0],startangle=90, colors=['lightskyblue','plum'],shadow = True)
plt.title("Gender distribution")
plt.show()
```

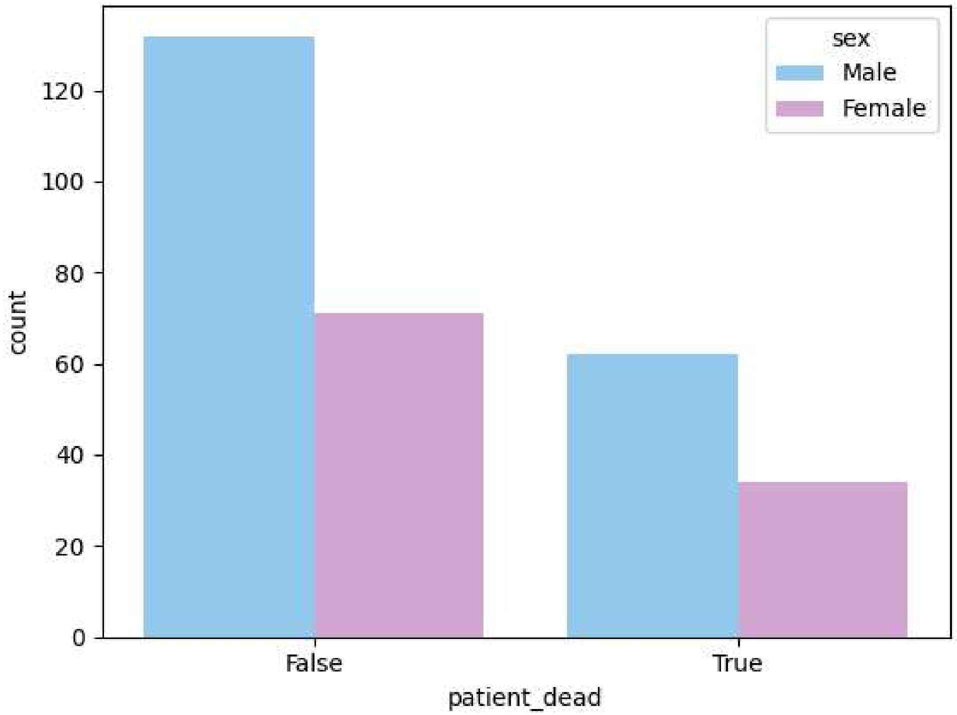


In [64]:

```
dead_counts = heart_failure_df.patient_dead.value_counts()

plt.pie(dead_counts,labels=dead_counts.index,autopct='%.1f%%',
        explode=[0.1,0],startangle=90, colors=['lightskyblue','lightcoral'],shadow = True)
plt.title("Patient Dead & ALive")
plt.show()
```



```
In [65]: # death and gender
         sns.countplot(x=heart_failure_df.patient_dead, hue=heart_failure_df.sex,palette=['lightskyblue','
         plt.show()
```



```
In [ ]: # Question 2

        #What is the normal level of Ejection Function? how many had abnormal ejection function?

        # hint : 55% - 70% - Normal value range
```

```
In [66]: print(" the normal level of  ejection_fraction is  {}".format(column_deatils_df.normal_value['eje

          the normal level of  ejection_fraction is  55% - 70%
```

```
In [67]: normal_ejection_fraction = heart_failure_df[(heart_failure_df.ejection_fraction >= 55) & (heart_f
         normal_ejection_fraction

Out[67]: 38
```

```
In [69]: abnormal_ejection_fraction = len(heart_failure_df) - normal_ejection_fraction
         abnormal_ejection_fraction

Out[69]: 261
```

```
In [ ]: ## Q3 what is the normal level of Platelets counts? How many patient had abnormal Platelets count
```

```
In [70]: column_deatils_df.normal_value['platelets']

Out[70]: '150 - 400 kilo-platelets / mcL'
```

In [71]:

```
# 1.5 lac - 4lac
normal_platelets = heart_failure_df[(heart_failure_df.platelets >= 150) & (heart_failure_df.plate
print("normal {}".format(normal_platelets))

abnormal = len(heart_failure_df) - normal_platelets
print("abnormal {}".format(abnormal))
```

```
normal 252
abnormal 47
```

In [ ]:

```
# Q4 what is the normal level of serum_creatinine counts? How many patient had abnormal serum_crea
# Q5 what is the normal level of serum_sodium counts? How many patient had abnormal serum_sodium c
```

In [ ]:

```
# Q6 How many patients had smoking habit? out of them how many male and female patients are there
```

In [74]:

```
total_number_smoking_habit = len(heart_failure_df[heart_failure_df.smoking == True])
total_number_smoking_habit
```

Out[74]:

```
96
```

In [82]:

```
smokers = heart_failure_df[heart_failure_df.smoking== True]
smokers =smokers.groupby('sex').count()
# smokers.smoking
print("Female smokers is {}".format(smokers.smoking[0]))
print("male smokers is {}".format(smokers.smoking[1]))
```

```
Female smokers is 4
male smokers is 92
```

In [78]:

```
# Q7 How many Patients had anemia or diabetics or high blood pressure?
total_number_anaemia = len(heart_failure_df[heart_failure_df.anaemia == True])
total_number_diabetes = len(heart_failure_df[heart_failure_df.diabetes == True])
total_number_high_blood_pressure = len(heart_failure_df[heart_failure_df.high_blood_pressure == Tr
print(total_number_anaemia,total_number_diabetes,total_number_high_blood_pressure)
```

```
129 125 105
```

In [ ]: