In [1]:
```
import numpy as np
import pandas as pd
```

In [2]:
```
dataset=pd.read_csv("PreProcessed_kidney_disease.csv")
dataset
```

Out[2]:

	id	age	F	sg	al	su	rbc	pc	pcc	ba	...	pcv	wc	rc	htn	dm	cad	appet	pe	ane	classifica
0	0	48.0	80.0	1.0200	1.0	0.0	normal	normal	notpresent	notpresent	...	44.0	7800.0	5.2000	yes	yes	no	good	no	no	
1	1	7.5	55.0	1.0200	4.0	0.0	normal	normal	notpresent	notpresent	...	38.0	6000.0	4.7000	no	no	no	good	no	no	
2	2	62.0	80.0	1.0100	2.0	0.0	normal	normal	notpresent	notpresent	...	31.0	7500.0	4.7000	no	yes	no	poor	no	yes	
3	3	48.0	70.0	1.0075	4.0	0.0	normal	abnormal	present	notpresent	...	32.0	6700.0	3.9000	yes	no	no	poor	yes	yes	
4	4	51.0	80.0	1.0100	2.0	0.0	normal	normal	notpresent	notpresent	...	35.0	7300.0	4.6000	no	no	no	good	no	no	
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	
379	395	55.0	80.0	1.0200	0.0	0.0	normal	normal	notpresent	notpresent	...	47.0	6700.0	4.9000	no	no	no	good	no	no	nc
380	396	42.0	70.0	1.0250	0.0	0.0	normal	normal	notpresent	notpresent	...	54.0	7800.0	5.8125	no	no	no	good	no	no	nc
381	397	12.0	80.0	1.0200	0.0	0.0	normal	normal	notpresent	notpresent	...	49.0	6600.0	5.4000	no	no	no	good	no	no	nc
382	398	17.0	60.0	1.0250	0.0	0.0	normal	normal	notpresent	notpresent	...	51.0	7200.0	5.8125	no	no	no	good	no	no	nc
383	399	58.0	80.0	1.0250	0.0	0.0	normal	normal	notpresent	notpresent	...	53.0	6800.0	5.8125	no	no	no	good	no	no	nc

384 rows × 26 columns

In [3]:
```
dataset.isnull().sum().T
```

Out[3]:
```
id                0
age               0
F                 0
sg                0
al                0
su                0
rbc               0
pc                0
pcc               0
ba                0
bgr               0
bu                0
sc                0
sod               0
pot               0
hemo              0
pcv               0
wc                0
rc                0
htn               0
dm                0
cad               0
appet             0
pe                0
ane               0
classification    0
dtype: int64
```

In [4]:
```
def quanQual(dataset):
    quan=[]
    qual=[]
    for columnName in dataset.columns:
        #print(columnName)
        if(dataset[columnName].dtype=='O'):
            #print("qual")
            qual.append(columnName)
        else:
            #print("quan")
            quan.append(columnName)
    return quan,qual
```

In [5]:
```
quan,qual=quanQual(dataset)
```

In [6]:
```
descriptive=pd.DataFrame(index=["Mean","Median","Mode","Q1:25%","Q2:50%",
                                "Q3:75%","99%","Q4:100%","IQR","1.5rule","Lesser","Greater","Min","Max"])
for columnName in quan:
    descriptive[columnName]["Mean"]=dataset[columnName].mean()
    descriptive[columnName]["Median"]=dataset[columnName].median()
    descriptive[columnName]["Mode"]=dataset[columnName].mode()[0]
    descriptive[columnName]["Q1:25%"]=dataset.describe()[columnName]["25%"]
    descriptive[columnName]["Q2:50%"]=dataset.describe()[columnName]["50%"]
    descriptive[columnName]["Q3:75%"]=dataset.describe()[columnName]["75%"]
    descriptive[columnName]["99%"]=np.percentile(dataset[columnName],99)
    descriptive[columnName]["Q4:100%"]=dataset.describe()[columnName]["max"]
    descriptive[columnName]["IQR"]=descriptive[columnName]["Q3:75%"]-descriptive[columnName]["Q1:25%"]
    descriptive[columnName]["1.5rule"]=1.5*descriptive[columnName]["IQR"]
    descriptive[columnName]["Lesser"]=descriptive[columnName]["Q1:25%"]-descriptive[columnName]["1.5rul
    descriptive[columnName]["Greater"]=descriptive[columnName]["Q3:75%"]+descriptive[columnName]["1.5ru
    descriptive[columnName]["Min"]=dataset[columnName].min()
    descriptive[columnName]["Max"]=dataset[columnName].max()
    descriptive
```

In [7]:
```
lesser=[]
greater=[]

for columnName in quan:
    if(descriptive[columnName]["Lesser"]>descriptive[columnName]["Min"]):
        lesser.append(columnName)
    if(descriptive[columnName]["Greater"]<descriptive[columnName]["Q4:100%"]):
        greater.append(columnName)
```

In [8]:
```
lesser
```

Out[8]: ['rc']

In [9]:
```
greater
```

Out[9]: []