REPLACING THE MISSING VALUE

```
import numpy as n
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import sklearn
from sklearn.preprocessing import LabelEncoder
from sklearn.preprocessing import MinMaxScaler

import joblib
import pickle

%matplotlib inline
```

Loading the Dataset

In [2]:

```
data=pd.read_csv("/content/chronickidneydisease.csv")
```

In [3]:

```
data.head()
```

Out[3]:

	id	age	bp	sg	al	su	rbc	pc	pcc	ba	...	pcv	wc	rc	htn	dm	cad	appet	pe	ane	classification
0	0	48.0	80.0	1.020	1.0	0.0	NaN	normal	notpresent	notpresent	...	44	7800	5.2	yes	yes	no	good	no	no	ckd
1	1	7.0	50.0	1.020	4.0	0.0	NaN	normal	notpresent	notpresent	...	38	6000	NaN	no	no	no	good	no	no	ckd
2	2	62.0	80.0	1.010	2.0	3.0	normal	normal	notpresent	notpresent	...	31	7500	NaN	no	yes	no	poor	no	yes	ckd
3	3	48.0	70.0	1.005	4.0	0.0	normal	abnormal	present	notpresent	...	32	6700	3.9	yes	no	no	poor	yes	yes	ckd

	id	age	bp	sg	al	su	rbc	pc	pcc	ba	...	pcv	wc	rc	htn	dm	cad	appet	pe	ane	classification
4	4	51.0	80.0	1.0100	2.0	0.0	normal	normal	notpresent	notpresent	...	35	7300	4.6	no	no	no	good	no	no	ckd

5 rows × 26 columns

In [4]:

```
data.tail()
```

Out[4]:

	id	age	bp	sg	al	su	rbc	pc	pcc	ba	...	pcv	wc	rc	htn	dm	cad	appet	pe	ane	classification
395	395	55.0	80.0	1.020	0.0	0.0	normal	normal	notpresent	notpresent	...	47	6700	4.9	no	no	no	good	no	no	notckd
396	396	42.0	70.0	1.025	0.0	0.0	normal	normal	notpresent	notpresent	...	54	7800	6.2	no	no	no	good	no	no	notckd
397	397	12.0	80.0	1.020	0.0	0.0	normal	normal	notpresent	notpresent	...	49	6600	5.4	no	no	no	good	no	no	notckd
398	398	17.0	60.0	1.025	0.0	0.0	normal	normal	notpresent	notpresent	...	51	7200	5.9	no	no	no	good	no	no	notckd
399	399	58.0	80.0	1.025	0.0	0.0	normal	normal	notpresent	notpresent	...	53	6800	6.1	no	no	no	good	no	no	notckd

5 rows × 26 columns

In [5]:

```
data.head(10)
```

Out[5]:

	id	age	bp	sg	al	su	rbc	pc	pcc	ba	...	pcv	wc	rc	htn	dm	cad	appet	pe	ane	classification
0	0	48.0	80.0	1.020	1.0	0.0	NaN	normal	notpresent	notpresent	...	44	7800	5.2	yes	yes	no	good	no	no	ckd
1	1	7.0	50.0	1.020	4.0	0.0	NaN	normal	notpresent	notpresent	...	38	6000	NaN	no	no	no	good	no	no	ckd
2	2	62.0	80.0	1.010	2.0	3.0	normal	normal	notpresent	notpresent	...	31	7500	NaN	no	yes	no	poor	no	yes	ckd
3	3	48.0	70.0	1.005	4.0	0.0	normal	abnormal	present	notpresent	...	32	6700	3.9	yes	no	no	poor	yes	yes	ckd
4	4	51.0	80.0	1.010	2.0	0.0	normal	normal	notpresent	notpresent	...	35	7300	4.6	no	no	no	good	no	no	ckd
5	5	60.0	90.0	1.015	3.0	0.0	NaN	NaN	notpresent	notpresent	...	39	7800	4.4	yes	yes	no	good	yes	no	ckd
6	6	68.0	70.0	1.010	0.0	0.0	NaN	normal	notpresent	notpresent	...	36	NaN	NaN	no	no	no	good	no	no	ckd
7	7	24.0	NaN	1.015	2.0	4.0	normal	abnormal	notpresent	notpresent	...	44	6900	5	no	yes	no	good	yes	no	ckd
8	8	52.0	100.0	1.015	3.0	0.0	normal	abnormal	present	notpresent	...	33	9600	4.0	yes	yes	no	good	no	yes	ckd
9	9	53.0	90.0	1.020	2.0	0.0	abnormal	abnormal	present	notpresent	...	29	12100	3.7	yes	yes	no	poor	no	yes	ckd

10 rows × 26 columns

Drop id Column

In [6]:

```
data.drop(["id"],axis=1,inplace=True)
data.columns
```

Out[6]:

```
Index(['age', 'bp', 'sg', 'al', 'su', 'rbc', 'pc', 'pcc', 'ba', 'bgr', 'bu',
       'sc', 'sod', 'pot', 'hemo', 'pcv', 'wc', 'rc', 'htn', 'dm', 'cad',
       'appet', 'pe', 'ane', 'classification'],
      dtype='object')
```

Renaming the columns

In [7]:

```
data.columns=['age','blood_pressure','specific_gravity','albumin','sugar','re
d_blood_cells','pus_cell','pus_cell_clumps','bacteria','blood glucose
random','blood_urea','serum_creatinine','sodium','potassium','hemoglobin','pa
cked_cell_volume','white_blood_cell_count','red_blood_cell_count','hypertensi
on','diabetesmellitus','coronary_artery_disease','appetite','pedal_edema','an
emia','class']
data.columns
```

Out[7]:

```
Index(['age', 'blood_pressure', 'specific_gravity', 'albumin', 'sugar',
       'red_blood_cells', 'pus_cell', 'pus_cell_clumps', 'bacteria',
       'blood glucose random', 'blood_urea', 'serum_creatinine', 'sodium',
       'potassium', 'hemoglobin', 'packed_cell_volume',
       'white_blood_cell_count', 'red_blood_cell_count', 'hypertension',
       'diabetesmellitus', 'coronary_artery_disease', 'appetite',
       'pedal_edema', 'anemia', 'class'],
      dtype='object')
```

In [8]:

```
data.info()
```

```
RangeIndex: 400 entries, 0 to 399
Data columns (total 25 columns):
 #   Column                   Non-Null Count  Dtype
---  ------                   --------------  -----
 0   age                      391 non-null    float64
 1   blood_pressure           388 non-null    float64
 2   specific_gravity         353 non-null    float64
 3   albumin                  354 non-null    float64
 4   sugar                    351 non-null    float64
 5   red_blood_cells          248 non-null    object
 6   pus_cell                 335 non-null    object
 7   pus_cell_clumps          396 non-null    object
 8   bacteria                 396 non-null    object
 9   blood glucose random     356 non-null    float64
 10  blood_urea               381 non-null    float64
 11  serum_creatinine         383 non-null    float64
 12  sodium                   313 non-null    float64
 13  potassium                312 non-null    float64
 14  hemoglobin               348 non-null    float64
 15  packed_cell_volume       330 non-null    object
 16  white_blood_cell_count   295 non-null    object
 17  red_blood_cell_count     270 non-null    object
```

```
 18  hypertension              398 non-null    object
 19  diabetesmellitus          398 non-null    object
 20  coronary_artery_disease   398 non-null    object
 21  appetite                  399 non-null    object
 22  pedal_edema               399 non-null    object
 23  anemia                    399 non-null    object
 24  class                     400 non-null    object
dtypes: float64(11), object(14)
memory usage: 78.2+ KB
```

Target Column

In [9]:

```
data['class'].unique()
```

Out[9]:

```
array(['ckd', 'ckd\t', 'notckd'], dtype=object)
```

In [10]:

```
data['class']=data['class'].replace("ckd\t","ckd")
```

In [11]:

```
data['class'].unique()
```

Out[11]:

```
array(['ckd', 'notckd'], dtype=object)
```

In [12]:

```
catcols=set(data.dtypes[data.dtypes=='O'].index.values)
print(catcols)
```

```
{'packed_cell_volume', 'red_blood_cells', 'white_blood_cell_count', 'red_bloo
d_cell_count', 'diabetesmellitus', 'anemia', 'bacteria', 'coronary_artery_dis
ease', 'pedal_edema', 'appetite', 'pus_cell', 'pus_cell_clumps', 'class', 'hy
pertension'}
```

In [14]:

```
from collections import Counter as c
```

In [15]:

```
for i in catcols:
    print("Columns :",i)
    print(c(data[i]))
    print('*'*120+'\n')
```

```
Columns : packed_cell_volume
Counter({nan: 70, '52': 21, '41': 21, '44': 19, '48': 19, '40': 16, '43': 14,
'45': 13, '42': 13, '32': 12, '36': 12, '33': 12, '28': 12, '50': 12, '37': 1
1, '34': 11, '35': 9, '29': 9, '30': 9, '46': 9, '31': 8, '39': 7, '24': 7, '
26': 6, '38': 5, '47': 4, '49': 4, '53': 4, '51': 4, '54': 4, '27': 3, '22':
3, '25': 3, '23': 2, '19': 2, '16': 1, '\t?': 1, '14': 1, '18': 1, '17': 1, '
15': 1, '21': 1, '20': 1, '\t43': 1, '9': 1})
************************************************************************
****************************************

Columns : red_blood_cells
Counter({'normal': 201, nan: 152, 'abnormal': 47})
************************************************************************
****************************************
```

```
Columns : white_blood_cell_count
Counter({nan: 105, '9800': 11, '6700': 10, '9600': 9, '9200': 9, '7200': 9, '
6900': 8, '11000': 8, '5800': 8, '7800': 7, '9100': 7, '9400': 7, '7000': 7,
'4300': 6, '6300': 6, '10700': 6, '10500': 6, '7500': 5, '8300': 5, '7900': 5
, '8600': 5, '5600': 5, '10200': 5, '5000': 5, '8100': 5, '9500': 5, '6000':
4, '6200': 4, '10300': 4, '7700': 4, '5500': 4, '10400': 4, '6800': 4, '6500'
: 4, '4700': 4, '7300': 3, '4500': 3, '8400': 3, '6400': 3, '4200': 3, '7400'
: 3, '8000': 3, '5400': 3, '3800': 2, '11400': 2, '5300': 2, '8500': 2, '1460
0': 2, '7100': 2, '13200': 2, '9000': 2, '8200': 2, '15200': 2, '12400': 2, '
12800': 2, '8800': 2, '5700': 2, '9300': 2, '6600': 2, '12100': 1, '12200': 1
, '18900': 1, '21600': 1, '11300': 1, '\t6200': 1, '11800': 1, '12500': 1, '1
1900': 1, '12700': 1, '13600': 1, '14900': 1, '16300': 1, '\t8400': 1, '10900
': 1, '2200': 1, '11200': 1, '19100': 1, '\t?': 1, '12300': 1, '16700': 1, '2
600': 1, '26400': 1, '4900': 1, '12000': 1, '15700': 1, '4100': 1, '11500': 1
, '10800': 1, '9900': 1, '5200': 1, '5900': 1, '9700': 1, '5100': 1})
**************************************************************************
****************************************

Columns : red_blood_cell_count
Counter({nan: 130, '5.2': 18, '4.5': 16, '4.9': 14, '4.7': 11, '3.9': 10, '4.
8': 10, '4.6': 9, '3.4': 9, '3.7': 8, '5.0': 8, '6.1': 8, '5.5': 8, '5.9': 8,
'3.8': 7, '5.4': 7, '5.8': 7, '5.3': 7, '4.3': 6, '4.2': 6, '5.6': 6, '4.4':
5, '3.2': 5, '4.1': 5, '6.2': 5, '5.1': 5, '6.4': 5, '5.7': 5, '6.5': 5, '3.6
': 4, '6.0': 4, '6.3': 4, '4.0': 3, '4': 3, '3.5': 3, '3.3': 3, '5': 2, '2.6'
: 2, '2.8': 2, '2.5': 2, '3.1': 2, '2.1': 2, '2.9': 2, '2.7': 2, '3.0': 2, '2
.3': 1, '8.0': 1, '3': 1, '2.4': 1, '\t?': 1})
**************************************************************************
****************************************

Columns : diabetesmellitus
Counter({'no': 258, 'yes': 134, '\tno': 3, '\tyes': 2, nan: 2, ' yes': 1})
**************************************************************************
****************************************

Columns : anemia
Counter({'no': 339, 'yes': 60, nan: 1})
**************************************************************************
****************************************

Columns : bacteria
Counter({'notpresent': 374, 'present': 22, nan: 4})
**************************************************************************
****************************************

Columns : coronary_artery_disease
Counter({'no': 362, 'yes': 34, '\tno': 2, nan: 2})
**************************************************************************
****************************************

Columns : pedal_edema
Counter({'no': 323, 'yes': 76, nan: 1})
**************************************************************************
****************************************
```

```
Columns : appetite
Counter({'good': 317, 'poor': 82, nan: 1})
********************************************************************************
****************************************

Columns : pus_cell
Counter({'normal': 259, 'abnormal': 76, nan: 65})
********************************************************************************
****************************************

Columns : pus_cell_clumps
Counter({'notpresent': 354, 'present': 42, nan: 4})
********************************************************************************
****************************************

Columns : class
Counter({'ckd': 250, 'notckd': 150})
********************************************************************************
****************************************

Columns : hypertension
Counter({'no': 251, 'yes': 147, nan: 2})
********************************************************************************
****************************************
```

Removing the column which are not categorized

In [16]:

```
catcols.remove('red_blood_cell_count')
catcols.remove('packed_cell_volume')
catcols.remove('white_blood_cell_count')
print(catcols)
```

```
{'red_blood_cells', 'diabetesmellitus', 'anemia', 'bacteria', 'coronary_arter
y_disease', 'pedal_edema', 'appetite', 'pus_cell', 'pus_cell_clumps', 'class'
, 'hypertension'}
```

Numerical Columns

In [18]:

```
contcols=set(data.dtypes[data.dtypes!='O'].index.values)
print(contcols)
```

```
{'blood glucose random', 'potassium', 'sugar', 'serum_creatinine', 'blood_ure
a', 'hemoglobin', 'sodium', 'specific_gravity', 'blood_pressure', 'albumin',
'age'}
```

In [19]:

```
for i in contcols:
    print("Continous Columns :",i)
    print(c(data[i]))
    print('*'*120+'\n')
```

```
Continous Columns : blood glucose random
Counter({99.0: 10, 100.0: 9, 93.0: 9, 107.0: 8, 117.0: 6, 140.0: 6, 92.0: 6,
109.0: 6, 131.0: 6, 130.0: 6, 70.0: 5, 114.0: 5, 95.0: 5, 123.0: 5, 124.0: 5,
```

102.0: 5, 132.0: 5, 104.0: 5, 125.0: 5, 122.0: 5, 121.0: 4, 106.0: 4, 76.0: 4
, 91.0: 4, 129.0: 4, 133.0: 4, 94.0: 4, 88.0: 4, 118.0: 4, 139.0: 4, 111.0: 4
, 113.0: 4, 120.0: 4, 119.0: 4, 74.0: 3, 108.0: 3, 171.0: 3, 137.0: 3, 79.0:
3, 150.0: 3, 112.0: 3, 127.0: 3, 219.0: 3, 172.0: 3, 89.0: 3, 128.0: 3, 214.0
: 3, 105.0: 3, 78.0: 3, 103.0: 3, 82.0: 3, 97.0: 3, 81.0: 3, 138.0: 2, 490.0:
2, 208.0: 2, 98.0: 2, 204.0: 2, 207.0: 2, 144.0: 2, 253.0: 2, 141.0: 2, 86.0:
2, 360.0: 2, 163.0: 2, 158.0: 2, 165.0: 2, 169.0: 2, 210.0: 2, 101.0: 2, 153.
0: 2, 213.0: 2, 424.0: 2, 303.0: 2, 192.0: 2, 80.0: 2, 110.0: 2, 96.0: 2, 85.
0: 2, 83.0: 2, 75.0: 2, nan: 1, 423.0: 1, 410.0: 1, 380.0: 1, 157.0: 1, 263.0
: 1, 173.0: 1, nan: 1, nan: 1, nan: 1, 156.0: 1, 264.0: 1, nan: 1, 159.0: 1,
270.0: 1, nan: 1, nan: 1, nan: 1, 162.0: 1, nan: 1, 246.0: 1, nan: 1, nan: 1,
nan: 1, nan: 1, 182.0: 1, 146.0: 1, nan: 1, 425.0: 1, 250.0: 1, nan: 1, nan:
1, nan: 1, 415.0: 1, 251.0: 1, 280.0: 1, 295.0: 1, 298.0: 1, 226.0: 1, 143.0:
1, 115.0: 1, 297.0: 1, 233.0: 1, 294.0: 1, nan: 1, nan: 1, nan: 1, nan: 1, 32
3.0: 1, nan: 1, nan: 1, 90.0: 1, 308.0: 1, 224.0: 1, nan: 1, 268.0: 1, nan: 1
, 256.0: 1, nan: 1, 84.0: 1, nan: 1, 288.0: 1, 273.0: 1, 242.0: 1, 148.0: 1,
nan: 1, 160.0: 1, nan: 1, 307.0: 1, 220.0: 1, 447.0: 1, 309.0: 1, 22.0: 1, 26
1.0: 1, 215.0: 1, 234.0: 1, 352.0: 1, nan: 1, nan: 1, 239.0: 1, nan: 1, nan:
1, 184.0: 1, nan: 1, 252.0: 1, 230.0: 1, 341.0: 1, nan: 1, 255.0: 1, nan: 1,
238.0: 1, 248.0: 1, 241.0: 1, 269.0: 1, nan: 1, nan: 1, 201.0: 1, 203.0: 1, 4
63.0: 1, 176.0: 1, nan: 1, nan: 1, 116.0: 1, nan: 1, nan: 1, nan: 1, 134.0: 1
, 87.0: 1, nan: 1})
****
****

Continous Columns : potassium
Counter({5.0: 30, 3.5: 30, 4.9: 27, 4.7: 17, 4.8: 16, 4.0: 14, 4.2: 14, 4.1:
14, 3.8: 14, 3.9: 14, 4.4: 14, 4.5: 13, 3.7: 12, 4.3: 12, 3.6: 8, 4.6: 7, 3.4
: 5, 5.2: 5, 5.7: 4, 5.3: 4, 3.2: 3, 5.5: 3, 2.9: 3, 5.4: 3, 6.3: 3, 3.3: 3,
2.5: 2, 5.8: 2, 5.9: 2, 5.6: 2, 3.0: 2, 6.5: 2, nan: 1, nan: 1, nan: 1, nan:
1, nan: 1, nan: 1, nan: 1, 6.4: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, na
n: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1,
nan: 1, 6.6: 1, 39.0: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan:
1, 7.6: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, na
n: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, 47.0: 1
, nan: 1, 5.1: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan
: 1, nan: 1, nan: 1, 2.8: 1, nan: 1, nan: 1, nan: 1, nan: 1, 2.7: 1, nan: 1,
nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan:
1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, na
n: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1})
****
****

Continous Columns : sugar
Counter({0.0: 290, 2.0: 18, 3.0: 14, 4.0: 13, 1.0: 13, 5.0: 3, nan: 1, nan: 1
, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan
: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1,
nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan:
1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, na
n: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1})
****
****

Continous Columns : serum_creatinine
Counter({1.2: 40, 1.1: 24, 1.0: 23, 0.5: 23, 0.7: 22, 0.9: 22, 0.6: 18, 0.8: 17, 2.2: 10, 1.5: 9, 1.7: 9, 1.3: 8, 1.6: 8, 1.8: 7, 1.4: 7, 2.5: 7, 2.8: 7, 1.9: 6, 2.7: 5, 2.1: 5, 2.0: 5, 3.2: 5, 3.3: 5, 3.9: 4, 7.3: 4, 4.0: 3, 2.4: 3, 3.4: 3, 2.9: 3, 5.3: 3, 2.3: 3, 7.2: 2, 4.6: 2, 4.1: 2, 5.2: 2, 6.3: 2, 3.0: 2, 6.1: 2, 6.7: 2, 5.6: 2, 6.5: 2, 4.4: 2, 6.0: 2, 3.8: 1, 24.0: 1, 9.6: 1, 76.0: 1, 7.7: 1, nan: 1, 10.8: 1, 5.9: 1, 3.25: 1, nan: 1, 9.7: 1, 6.4: 1, 32.0: 1, nan: 1, nan: 1, 8.5: 1, 15.0: 1, 3.6: 1, 10.2: 1, 11.5: 1, nan: 1, 12.2: 1, 9.2: 1, 13.8: 1, 16.9: 1, 7.1: 1, 18.0: 1, 13.0: 1, 48.1: 1, 14.2: 1, 16.4: 1, nan: 1, nan: 1, 2.6: 1, 7.5: 1, 4.3: 1, 18.1: 1, 11.8: 1, 9.3: 1, 6.8: 1, 13.5: 1, nan: 1, 12.8: 1, 11.9: 1, nan: 1, nan: 1, nan: 1, 12.0: 1, nan: 1, 13.4: 1, 15.2: 1, 13.3: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, 0.4: 1})
****
****

Continous Columns : blood_urea
Counter({46.0: 15, 25.0: 13, 19.0: 11, 40.0: 10, 18.0: 9, 50.0: 9, 15.0: 9, 48.0: 9, 26.0: 8, 27.0: 8, 32.0: 8, 49.0: 8, 36.0: 7, 28.0: 7, 20.0: 7, 17.0: 7, 38.0: 7, 16.0: 7, 30.0: 7, 44.0: 7, 31.0: 6, 45.0: 6, 39.0: 6, 29.0: 6, 24.0: 6, 37.0: 6, 22.0: 6, 23.0: 6, 53.0: 5, 55.0: 5, 33.0: 5, 66.0: 5, 35.0: 5, 42.0: 5, 47.0: 4, 51.0: 4, 34.0: 4, 68.0: 4, 41.0: 4, 60.0: 3, 107.0: 3, 80.0: 3, 96.0: 3, 52.0: 3, 106.0: 3, 125.0: 3, 56.0: 2, 54.0: 2, 72.0: 2, 86.0: 2, 90.0: 2, 87.0: 2, 155.0: 2, 153.0: 2, 77.0: 2, 89.0: 2, 111.0: 2, 73.0: 2, 98.0: 2, 82.0: 2, 132.0: 2, 58.0: 2, 10.0: 2, 162.0: 1, 148.0: 1, 180.0: 1, 163.0: 1, nan: 1, 75.0: 1, 65.0: 1, 103.0: 1, 70.0: 1, 202.0: 1, 114.0: 1, nan: 1, nan: 1, 164.0: 1, 142.0: 1, 391.0: 1, nan: 1, nan: 1, 92.0: 1, 139.0: 1, 85.0: 1, 186.0: 1, 217.0: 1, 88.0: 1, 118.0: 1, 50.1: 1, 71.0: 1, nan: 1, 21.0: 1, 219.0: 1, 166.0: 1, 208.0: 1, 176.0: 1, nan: 1, 145.0: 1, 165.0: 1, 322.0: 1, 235.0: 1, 76.0: 1, nan: 1, nan: 1, 113.0: 1, 1.5: 1, 146.0: 1, 133.0: 1, 137.0: 1, 67.0: 1, 115.0: 1, 223.0: 1, 98.6: 1, 158.0: 1, 94.0: 1, 74.0: 1, nan: 1, 150.0: 1, nan: 1, 61.0: 1, 57.0: 1, nan: 1, 95.0: 1, 191.0: 1, nan: 1, 93.0: 1, 241.0: 1, 64.0: 1, 79.0: 1, 215.0: 1, 309.0: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1})
****
****

Continous Columns : hemoglobin
Counter({15.0: 16, 10.9: 8, 9.8: 7, 11.1: 7, 13.0: 7, 13.6: 7, 11.3: 6, 10.3: 6, 12.0: 6, 13.9: 6, 15.4: 5, 11.2: 5, 10.8: 5, 9.7: 5, 12.6: 5, 7.9: 5, 10.0: 5, 14.0: 5, 14.3: 5, 14.8: 5, 12.2: 4, 12.4: 4, 12.5: 4, 15.2: 4, 9.1: 4, 11.9: 4, 13.5: 4, 16.1: 4, 14.1: 4, 13.2: 4, 13.8: 4, 13.7: 4, 13.4: 4, 17.0: 4, 15.5: 4, 15.8: 4, 9.6: 3, 11.6: 3, 9.5: 3, 9.4: 3, 12.7: 3, 9.9: 3, 10.1: 3, 8.6: 3, 11.0: 3, 15.6: 3, 8.1: 3, 8.3: 3, 10.4: 3, 11.8: 3, 11.4: 3, 11.5: 3, 15.9: 3, 14.5: 3, 16.2: 3, 14.4: 3, 14.2: 3, 16.3: 3, 16.5: 3, 15.7: 3, 16.4: 3, 14.9: 3, 15.3: 3, 17.8: 3, 12.1: 2, 9.3: 2, 10.2: 2, 10.5: 2, 6.0: 2, 11.7: 2, 8.0: 2, 12.3: 2, 8.7: 2, 13.1: 2, 8.8: 2, 13.3: 2, 14.6: 2, 16.9: 2, 16.0: 2, 14.7: 2, 16.6: 2, 16.7: 2, 16.8: 2, 15.1: 2, 17.1: 2, 17.2: 2, 17.4: 2, 5.6: 1, 7.6: 1, 7.7: 1, nan: 1, nan: 1, 12.9: 1, nan: 1, nan: 1, nan: 1, nan: 1, 6.6: 1, nan: 1, nan: 1, 7.5: 1, nan: 1, nan: 1, 4.8: 1, nan: 1, nan: 1, 7.1: 1, nan: 1, nan: 1, nan: 1, 9.2: 1, nan: 1, 6.2: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, 8.2: 1, nan: 1, nan: 1, 6.1: 1, nan: 1, nan: 1, nan: 1, nan: 1, 8.4: 1, nan: 1, 9.0: 1, nan: 1, nan: 1, 10.6: 1, nan: 1, nan:

1, nan: 1, 10.7: 1, nan: 1, 5.5: 1, nan: 1, 5.8: 1, 6.8: 1, 8.5: 1, 7.3: 1, n
an: 1, nan: 1, 12.8: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan:
1, 6.3: 1, nan: 1, 3.1: 1, nan: 1, 17.3: 1, nan: 1, nan: 1, nan: 1, nan: 1, 1
7.7: 1, 17.5: 1, nan: 1, 17.6: 1})
****
****

Continous Columns : sodium
Counter({135.0: 40, 140.0: 25, 141.0: 22, 139.0: 21, 142.0: 20, 138.0: 20, 13
7.0: 19, 136.0: 17, 150.0: 17, 147.0: 13, 145.0: 11, 132.0: 10, 146.0: 10, 13
1.0: 9, 144.0: 9, 133.0: 8, 130.0: 7, 134.0: 6, 143.0: 4, 127.0: 3, 124.0: 3,
114.0: 2, 125.0: 2, 128.0: 2, 122.0: 2, 113.0: 2, 120.0: 2, nan: 1, nan: 1, n
an: 1, 111.0: 1, nan: 1, 104.0: 1, nan: 1, nan: 1, nan: 1, nan: 1, 4.5: 1, na
n: 1, 129.0: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan:
1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, 163.0: 1, nan: 1, nan: 1, nan: 1,
nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan:
1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, na
n: 1, nan: 1, nan: 1, 126.0: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan:
1, nan: 1, nan: 1, nan: 1, nan: 1, 115.0: 1, nan: 1, nan: 1, nan: 1, nan: 1,
nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan:
1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, na
n: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1})
****
****

Continous Columns : specific_gravity
Counter({1.02: 106, 1.01: 84, 1.025: 81, 1.015: 75, 1.005: 7, nan: 1, nan: 1,
nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan:
1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, na
n: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1,
nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan:
1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1})
****
****

Continous Columns : blood_pressure
Counter({80.0: 116, 70.0: 112, 60.0: 71, 90.0: 53, 100.0: 25, 50.0: 5, 110.0:
3, nan: 1, nan: 1, 140.0: 1, 180.0: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1
, nan: 1, nan: 1, 120.0: 1, nan: 1, nan: 1, nan: 1})
****
****

Continous Columns : albumin
Counter({0.0: 199, 1.0: 44, 2.0: 43, 3.0: 43, 4.0: 24, nan: 1, nan: 1, nan: 1
, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan
: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1,
5.0: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan:
1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1, na
n: 1, nan: 1, nan: 1, nan: 1, nan: 1, nan: 1})
****
****

Continous Columns : age

```
Counter({60.0: 19, 65.0: 17, 48.0: 12, 50.0: 12, 55.0: 12, 47.0: 11, 62.0: 10
, 45.0: 10, 54.0: 10, 59.0: 10, 56.0: 10, 61.0: 9, 70.0: 9, 46.0: 9, 34.0: 9,
68.0: 8, 73.0: 8, 64.0: 8, 71.0: 8, 57.0: 8, 63.0: 7, 72.0: 7, 67.0: 7, 30.0:
7, 42.0: 6, 69.0: 6, 35.0: 6, 44.0: 6, 43.0: 6, 33.0: 6, 51.0: 5, 52.0: 5, 53
.0: 5, 75.0: 5, 76.0: 5, 58.0: 5, 41.0: 5, 66.0: 5, 24.0: 4, 40.0: 4, 39.0: 4
, 80.0: 4, 23.0: 4, 74.0: 3, 38.0: 3, 17.0: 3, 8.0: 3, 32.0: 3, 37.0: 3, 25.0
: 3, 29.0: 3, 21.0: 2, 15.0: 2, 5.0: 2, 12.0: 2, 49.0: 2, 19.0: 2, 36.0: 2, 2
0.0: 2, 28.0: 2, 7.0: 1, nan: 1, 82.0: 1, 11.0: 1, 26.0: 1, nan: 1, nan: 1, n
an: 1, nan: 1, 81.0: 1, 14.0: 1, 27.0: 1, nan: 1, 83.0: 1, 4.0: 1, 3.0: 1, 6.
0: 1, nan: 1, 90.0: 1, 78.0: 1, nan: 1, 2.0: 1, nan: 1, 22.0: 1, 79.0: 1})
********************************************************************************
****************************************
```

Removing the Columns which are not Numerical

In [20]:

```
contcols.remove('specific_gravity')
contcols.remove('albumin')
contcols.remove('sugar')
print(contcols)
```

```
{'blood glucose random', 'potassium', 'serum_creatinine', 'blood_urea', 'hemo
globin', 'sodium', 'blood_pressure', 'age'}
```

Adding columns which we found Continous

In [21]:

```
contcols.add('red_blood_cell_count')
contcols.add('packed_cell_volume')
contcols.add('white_blood_cell_count')
print(contcols)
```

```
{'packed_cell_volume', 'blood glucose random', 'white_blood_cell_count', 'red
_blood_cell_count', 'potassium', 'serum_creatinine', 'blood_urea', 'hemoglobi
n', 'sodium', 'blood_pressure', 'age'}
```

Adding columns which we found Catergorical

In [22]:

```
catcols.add('specific_gravity')
catcols.add('albumin')
catcols.add('sugar')
print(catcols)
```

```
{'red_blood_cells', 'sugar', 'diabetesmellitus', 'anemia', 'bacteria', 'coron
ary_artery_disease', 'pedal_edema', 'appetite', 'specific_gravity', 'pus_cell
', 'albumin', 'pus_cell_clumps', 'class', 'hypertension'}
```

Rectifying the Categorical Columns Classes

In [23]:

```
data['coronary_artery_disease']=data.coronary_artery_disease.replace('\tno','
no')
c(data['coronary_artery_disease'])
```

Out[23]:

```
Counter({'no': 364, 'yes': 34, nan: 2})
```

In [24]:

```
data['diabetesmellitus']=data.diabetesmellitus.replace(to_replace={'\tno':'no
','\tyes':'yes','yes':'yes'})
c(data['diabetesmellitus'])
```

Out[24]:

```
Counter({'yes': 136, 'no': 261, ' yes': 1, nan: 2})
```

Null Values

In [25]:

```
data.isnull().any()
```

Out[25]:

```
age                              True
blood_pressure                   True
specific_gravity                 True
albumin                          True
sugar                            True
red_blood_cells                  True
pus_cell                         True
pus_cell_clumps                  True
bacteria                         True
blood glucose random             True
blood_urea                       True
serum_creatinine                 True
sodium                           True
potassium                        True
hemoglobin                       True
packed_cell_volume               True
white_blood_cell_count           True
red_blood_cell_count             True
hypertension                     True
diabetesmellitus                 True
coronary_artery_disease          True
appetite                         True
pedal_edema                      True
anemia                           True
class                           False
dtype: bool
```

In [26]:

```
data.isnull().sum()#return the count
```

Out[26]:

```
age                              9
blood_pressure                  12
specific_gravity                47
albumin                         46
sugar                           49
red_blood_cells                152
pus_cell                        65
pus_cell_clumps                  4
bacteria                         4
blood glucose random            44
blood_urea                      19
serum_creatinine                17
```

```
sodium                      87
potassium                   88
hemoglobin                  52
packed_cell_volume          70
white_blood_cell_count     105
red_blood_cell_count       130
hypertension                 2
diabetesmellitus             2
coronary_artery_disease      2
appetite                     1
pedal_edema                  1
anemia                       1
class                        0
dtype: int64
```

In [27]:

```
data.packed_cell_volume=pd.to_numeric(data.packed_cell_volume,errors='coerce'
)
data.white_blood_cell_count=pd.to_numeric(data.white_blood_cell_count,errors=
'coerce')
data.red_blood_cell_count=pd.to_numeric(data.red_blood_cell_count,errors='coe
rce')
```

Replacing The Missing Values

Handling Continous/numerical columns Null values

In [28]:

```
#mean
data['blood glucose random'].fillna(data['blood glucose
random'].mean(),inplace=True)
data['blood_pressure'].fillna(data['blood_pressure'].mean(),inplace=True)
data['blood_urea'].fillna(data['blood_urea'].mean(),inplace=True)
data['hemoglobin'].fillna(data['hemoglobin'].mean(),inplace=True)
data['packed_cell_volume'].fillna(data['packed_cell_volume'].mean(),inplace=T
rue)
data['potassium'].fillna(data['potassium'].mean(),inplace=True)
data['red_blood_cell_count'].fillna(data['red_blood_cell_count'].mean(),inpla
ce=True)
data['serum_creatinine'].fillna(data['serum_creatinine'].mean(),inplace=True)
data['sodium'].fillna(data['sodium'].mean(),inplace=True)
data['white_blood_cell_count'].fillna(data['white_blood_cell_count'].mean(),i
nplace=True)
```

In [29]:

```
#mode
data['blood glucose random'].fillna(data['blood glucose
random'].mean(),inplace=True)
data['blood_pressure'].fillna(data['blood_pressure'].mean(),inplace=True)
data['blood_urea'].fillna(data['blood_urea'].mean(),inplace=True)
data['hemoglobin'].fillna(data['hemoglobin'].mean(),inplace=True)
data['packed_cell_volume'].fillna(data['packed_cell_volume'].mean(),inplace=T
rue)
```

```
data['potassium'].fillna(data['potassium'].mean(),inplace=True)
data['red_blood_cell_count'].fillna(data['red_blood_cell_count'].mean(),inpla
ce=True)
data['serum_creatinine'].fillna(data['serum_creatinine'].mean(),inplace=True)
data['sodium'].fillna(data['sodium'].mean(),inplace=True)
data['white_blood_cell_count'].fillna(data['white_blood_cell_count'].mean(),i
nplace=True)
```

In [34]:

```
data.isnull().sum()
```

Out[34]:

```
age                          9
blood_pressure               0
specific_gravity            47
albumin                     46
sugar                       49
red_blood_cells            152
pus_cell                    65
pus_cell_clumps              4
bacteria                     4
blood glucose random         0
blood_urea                   0
serum_creatinine             0
sodium                       0
potassium                    0
hemoglobin                   0
packed_cell_volume           0
white_blood_cell_count       0
red_blood_cell_count         0
hypertension                 2
diabetesmellitus             2
coronary_artery_disease      2
appetite                     1
pedal_edema                  1
anemia                       1
class                        0
dtype: int64
```