data-cleaning-2

September 2, 2024

[ ]:
```
import pandas as pd
data={
    "name":["alice","bob","Charlie"],
    "age":[24,25,26],
    "salary":[10000,None,3000],
    "gender":["F","M","F"],
    "height":[1.8,1.7,None]
}
df=pd.DataFrame(data)
df.dropna(how="all",inplace=True)
df
```

[ ]:

	name	age	salary	gender	height
0	alice	24	10000.0	F	1.8
1	bob	25	NaN	M	1.7
2	Charlie	26	3000.0	F	NaN

[ ]:
```
import pandas as pd
data={
    "name":["alice","bob","Charlie"],
    "age":[24,25,26],
    "salary":[10000,None,3000],
    "gender":["F","M","F"],
    "height":[1.8,1.7,None]
}
df=pd.DataFrame(data)
df.dropna(subset="salary",inplace=True)
df
```

[ ]:

	name	age	salary	gender	height
0	alice	24	10000.0	F	1.8
2	Charlie	26	3000.0	F	NaN

[ ]:
```
import pandas as pd
data={
    "name":["alice","bob","Charlie"],
    "age":[24,25,26],
```

```
    "salary":[10000,None,3000],
    "gender":["F","M","F"],
    "height":[1.8,1.7,None]
}
df=pd.DataFrame(data)
df.dropna(how="any",inplace=True)
df
```

[ ]:
```
    name  age   salary gender  height
0  alice   24  10000.0      F     1.8
```

[ ]:
```
import pandas as pd
import numpy as np
data={
    "name":["alice","bob","Charlie","dave","eve","bob","Charlie"],
    "age":[24,np.nan,35,41,np.nan,np.nan,85],
    "salary":[10000,np.nan,2000,np.nan,3000,np.nan,4000]
}
df=pd.DataFrame(data)
~df.duplicated()
#
```

[ ]:
```
0     True
1     True
2     True
3     True
4     True
5    False
6     True
dtype: bool
```

[ ]:
```
import pandas as pd
import numpy as np
data={
    "name":["alice","bob","Charlie","dave","eve","bob","Charlie"],
    "age":[24,np.nan,35,41,np.nan,np.nan,85],
    "salary":[10000,np.nan,2000,np.nan,3000,np.nan,4000]
}
df=pd.DataFrame(data)
df_filled=df.fillna(10,inplace=True)
df
```

[ ]:
```
      name   age   salary
0    alice  24.0  10000.0
1      bob  10.0     10.0
2  Charlie  35.0   2000.0
3     dave  41.0     10.0
```

```
4      eve  10.0   3000.0
5      bob  10.0     10.0
6  Charlie  85.0   4000.0
```

[ ]:
```
import pandas as pd
import numpy as np
data={
    "name":["alice","bob","Charlie","dave","eve","bob","Charlie"],
    "age":[24,np.nan,35,41,np.nan,np.nan,85],
    "salary":[10000,np.nan,2000,np.nan,3000,np.nan,4000]
}
df=pd.DataFrame(data)
df_filled=df.fillna(10)
df_filled
```

[ ]:
```
      name   age   salary
0    alice  24.0  10000.0
1      bob  10.0     10.0
2  Charlie  35.0   2000.0
3     dave  41.0     10.0
4      eve  10.0   3000.0
5      bob  10.0     10.0
6  Charlie  85.0   4000.0
```

[ ]:
```
import pandas as pd
import numpy as np
data={
    "name":["alice","bob","Charlie","dave","eve","bob","Charlie"],
    "age":[24,np.nan,35,41,np.nan,np.nan,85],
    "salary":[10000,np.nan,2000,np.nan,3000,np.nan,4000]
}
df=pd.DataFrame(data)
df_filled=df.fillna(method="ffill")
df_filled
```

```
<ipython-input-25-7cc257af65c2>:9: FutureWarning: DataFrame.fillna with 'method'
is deprecated and will raise in a future version. Use obj.ffill() or obj.bfill()
instead.
  df_filled=df.fillna(method="ffill")
```

[ ]:
```
      name   age   salary
0    alice  24.0  10000.0
1      bob  24.0  10000.0
2  Charlie  35.0   2000.0
3     dave  41.0   2000.0
4      eve  41.0   3000.0
5      bob  41.0   3000.0
```

```
6  Charlie  85.0   4000.0
```

[ ]:
```
import pandas as pd
import numpy as np
data={
    "name":["alice","bob","Charlie","dave","eve","bob","Charlie"],
    "age":[24,np.nan,35,41,np.nan,np.nan,85],
    "salary":[10000,np.nan,2000,np.nan,3000,np.nan,4000]
}
df=pd.DataFrame(data)
df_filled=df.fillna(method="bfill")
df_filled
```

```
<ipython-input-26-b0061b5aa1c5>:9: FutureWarning: DataFrame.fillna with 'method'
is deprecated and will raise in a future version. Use obj.ffill() or obj.bfill()
instead.
  df_filled=df.fillna(method="bfill")
```

[ ]:

	name	age	salary
0	alice	24.0	10000.0
1	bob	35.0	2000.0
2	Charlie	35.0	2000.0
3	dave	41.0	3000.0
4	eve	85.0	3000.0
5	bob	85.0	4000.0
6	Charlie	85.0	4000.0

[ ]:
```
import pandas as pd
import numpy as np
data={
    "name":["alice","bob","Charlie","dave","eve","bob","Charlie"],
    "age":[24,np.nan,35,41,np.nan,np.nan,85],
    "salary":[10000,np.nan,2000,np.nan,3000,np.nan,4000]
}
df=pd.DataFrame(data)
df_filled=df.fillna(df["salary"].mean())
df_filled
```

[ ]:

	name	age	salary
0	alice	24.0	10000.0
1	bob	4750.0	4750.0
2	Charlie	35.0	2000.0
3	dave	41.0	4750.0
4	eve	4750.0	3000.0
5	bob	4750.0	4750.0
6	Charlie	85.0	4000.0

[ ]:
```
import pandas as pd
df=pd.read_csv("/content/SAMPLEIDS.csv")
df.fillna(0)
```

[ ]:

	SNO	REGNO	NAME	DOB	GENDER	ADDRESS	M1	M2	M3	\
0	1	1220121	ARUN	2000-02-10	MALE	THANDALAM	82.0	81.0	90.0	
1	2	1220122	BABU	1999-01-25	MALE	KANCHIPURAM	56.0	61.0	80.0	
2	3	1220123	CHARAN	2000.09.21	MALE	THANDALAM	0.0	59.0	60.0	
3	4	1220124	DEVA	2000-11-09	MALE	POONAMALEE	74.0	79.0	80.0	
4	5	1220125	ESTER	2000-11-21	FEMALE	CHITHUR	92.0	95.0	96.0	
5	6	1220126	FARHANA	1999-03-05	FEMALE	THANDALAM	91.0	88.0	90.0	
6	7	1220127	GANI	2000-10-02	MALE	KANCHIPURAM	49.0	51.0	70.0	
7	7	1220127	GANI	2000-10-02	MALE	KANCHIPURAM	49.0	51.0	70.0	
8	8	1220128	HEMA	1999-01-25	FEMALE	POONAMALEE	95.0	96.0	90.0	
9	9	1220129	INDRA	2000.09.21	FEMALE	KANCHIPURAM	64.0	0.0	0.0	
10	10	1220130	JAHITH	2000-11-09	MALE	THANDALAM	34.0	45.0	50.0	
11	11	1220131	KANI	2000-11-21	FEMALE	CHITHUR	96.0	95.0	96.0	
12	12	1220132	LATHESSH	1999-03-05	MALE	THANDALAM	0.0	68.0	70.0	
13	13	1220133	MANI	2000-10-02	MALE	KANCHIPURAM	71.0	76.0	0.0	
14	14	1220134	NANI	20001109	MALE	POONAMALEE	79.0	77.0	80.0	
15	15	1220135	0	19990125	0	0	0.0	0.0	0.0	
16	16	1220136	PRATHAP	20000921	MALE	KANCHIPURAM	86.0	84.0	90.0	
17	17	1220137	RAGHU	20001109	MALE	POONAMALEE	67.0	64.0	70.0	
18	18	1220138	RATHI	20001121	FEMALE	KANCHIPURAM	81.0	86.0	90.0	
19	19	1220139	SARVESH	19990305	MALE	THANDALAM	84.0	87.0	0.0	
20	20	1220140	SANTHOSH	20001002	MALE	KANCHIPURAM	76.0	69.0	80.0	

	M4	TOTAL	AVG
0	0.0	0.0	0.000000
1	56.0	253.0	84.333333
2	70.0	0.0	0.000000
3	74.0	307.0	102.333333
4	92.0	375.0	125.000000
5	91.0	360.0	120.000000
6	49.0	219.0	73.000000
7	49.0	219.0	73.000000
8	95.0	376.0	125.333333
9	64.0	0.0	0.000000
10	34.0	163.0	54.333333
11	96.0	383.0	127.666667
12	70.0	208.0	69.333333
13	71.0	0.0	0.000000
14	79.0	315.0	105.000000
15	0.0	0.0	0.000000
16	86.0	346.0	115.333333
17	0.0	201.0	67.000000
18	81.0	338.0	112.666667

```
19  84.0    0.0    0.000000
20  76.0  301.0  100.333333
```

[ ]:

[ ]:
```
df.head(10)
```

[ ]:
```
   SNO    REGNO     NAME         DOB  GENDER      ADDRESS    M1    M2    M3  \
0    1  1220121     ARUN  2000-02-10    MALE    THANDALAM  82.0  81.0  90.0
1    2  1220122     BABU  1999-01-25    MALE  KANCHIPURAM  56.0  61.0  80.0
2    3  1220123   CHARAN  2000.09.21    MALE    THANDALAM   NaN  59.0  60.0
3    4  1220124     DEVA  2000-11-09    MALE   POONAMALEE  74.0  79.0  80.0
4    5  1220125    ESTER  2000-11-21  FEMALE      CHITHUR  92.0  95.0  96.0
5    6  1220126  FARHANA  1999-03-05  FEMALE    THANDALAM  91.0  88.0  90.0
6    7  1220127     GANI  2000-10-02    MALE  KANCHIPURAM  49.0  51.0  70.0
7    7  1220127     GANI  2000-10-02    MALE  KANCHIPURAM  49.0  51.0  70.0
8    8  1220128     HEMA  1999-01-25  FEMALE   POONAMALEE  95.0  96.0  90.0
9    9  1220129    INDRA  2000.09.21  FEMALE  KANCHIPURAM  64.0   NaN   NaN

     M4  TOTAL         AVG
0   NaN    NaN         NaN
1  56.0  253.0   84.333333
2  70.0    NaN    0.000000
3  74.0  307.0  102.333333
4  92.0  375.0  125.000000
5  91.0  360.0  120.000000
6  49.0  219.0   73.000000
7  49.0  219.0   73.000000
8  95.0  376.0  125.333333
9  64.0    NaN    0.000000
```

[ ]:
```
df.tail(10)
```

[ ]:
```
    SNO    REGNO      NAME         DOB  GENDER      ADDRESS    M1    M2    M3  \
11   11  1220131      KANI  2000-11-21  FEMALE      CHITHUR  96.0  95.0  96.0
12   12  1220132  LATHESSH  1999-03-05    MALE    THANDALAM   NaN  68.0  70.0
13   13  1220133      MANI  2000-10-02    MALE  KANCHIPURAM  71.0  76.0   NaN
14   14  1220134      NANI    20001109    MALE   POONAMALEE  79.0  77.0  80.0
15   15  1220135       NaN    19990125     NaN          NaN   NaN   NaN   NaN
16   16  1220136   PRATHAP    20000921    MALE  KANCHIPURAM  86.0  84.0  90.0
17   17  1220137     RAGHU    20001109    MALE   POONAMALEE  67.0  64.0  70.0
18   18  1220138     RATHI    20001121  FEMALE  KANCHIPURAM  81.0  86.0  90.0
19   19  1220139   SARVESH    19990305    MALE    THANDALAM  84.0  87.0   NaN
20   20  1220140  SANTHOSH    20001002    MALE  KANCHIPURAM  76.0  69.0  80.0

      M4  TOTAL         AVG
11  96.0  383.0  127.666667
```

```
12  70.0  208.0   69.333333
13  71.0    NaN    0.000000
14  79.0  315.0  105.000000
15   NaN    0.0    0.000000
16  86.0  346.0  115.333333
17   NaN  201.0   67.000000
18  81.0  338.0  112.666667
19  84.0    NaN    0.000000
20  76.0  301.0  100.333333
```

[ ]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 21 entries, 0 to 20
Data columns (total 12 columns):
 #   Column   Non-Null Count  Dtype
---  ------   --------------  -----
 0   SNO      21 non-null     int64
 1   REGNO    21 non-null     int64
 2   NAME     20 non-null     object
 3   DOB      21 non-null     object
 4   GENDER   20 non-null     object
 5   ADDRESS  20 non-null     object
 6   M1       18 non-null     float64
 7   M2       19 non-null     float64
 8   M3       17 non-null     float64
 9   M4       18 non-null     float64
 10  TOTAL    16 non-null     float64
 11  AVG      20 non-null     float64
dtypes: float64(6), int64(2), object(4)
memory usage: 2.1+ KB
```

[ ]:
```
df.describe()
```

[ ]:
```
              SNO         REGNO         M1         M2         M3         M4  \
count   21.000000  2.100000e+01  18.000000  19.000000  17.000000  18.000000
mean    10.333333  1.220130e+06  73.666667  74.315789  79.529412  73.166667
std      5.816643  5.816643e+00  17.580069  15.836149  13.010177  17.426315
min      1.000000  1.220121e+06  34.000000  45.000000  50.000000  34.000000
25%      6.000000  1.220126e+06  64.750000  62.500000  70.000000  65.500000
50%     10.000000  1.220130e+06  77.500000  77.000000  80.000000  75.000000
75%     15.000000  1.220135e+06  85.500000  86.500000  90.000000  85.500000
max     20.000000  1.220140e+06  96.000000  96.000000  96.000000  96.000000

            TOTAL         AVG
count   16.000000   20.000000
mean   272.750000   72.733333
```

```
std    102.048681   48.017127
min      0.000000    0.000000
25%    216.250000   40.750000
50%    304.000000   78.666667
75%    349.500000  113.333333
max    383.000000  127.666667
```

[ ]:
```
df.shape
```

[ ]:
```
(21, 12)
```

[ ]:
```
df.isnull().sum()
```

[ ]:
```
SNO        0
REGNO      0
NAME       1
DOB        0
GENDER     1
ADDRESS    1
M1         3
M2         2
M3         4
M4         3
TOTAL      5
AVG        1
dtype: int64
```

[ ]:
```
import pandas as pd
df=pd.read_csv("/content/SAMPLEIDS.csv")
df.nunique()
```

[ ]:
```
SNO        20
REGNO      20
NAME       19
DOB        13
GENDER      2
ADDRESS     4
M1         17
M2         17
M3          6
M4         16
TOTAL      15
AVG        15
dtype: int64
```

[ ]:
```
df.shape
```

[ ]: (21, 12)

[ ]:
```
df['GENDER'].value_counts()
```

[ ]:
```
GENDER
MALE      14
FEMALE     6
Name: count, dtype: int64
```

[ ]:
```
df.dropna(how="any").shape
```

[ ]: (13, 12)

[ ]:
```
x=df.dropna(how="any")
x
```

[ ]:

	SNO	REGNO	NAME	DOB	GENDER	ADDRESS	M1	M2	M3	\
1	2	1220122	BABU	1999-01-25	MALE	KANCHIPURAM	56.0	61.0	80.0	
3	4	1220124	DEVA	2000-11-09	MALE	POONAMALEE	74.0	79.0	80.0	
4	5	1220125	ESTER	2000-11-21	FEMALE	CHITHUR	92.0	95.0	96.0	
5	6	1220126	FARHANA	1999-03-05	FEMALE	THANDALAM	91.0	88.0	90.0	
6	7	1220127	GANI	2000-10-02	MALE	KANCHIPURAM	49.0	51.0	70.0	
7	7	1220127	GANI	2000-10-02	MALE	KANCHIPURAM	49.0	51.0	70.0	
8	8	1220128	HEMA	1999-01-25	FEMALE	POONAMALEE	95.0	96.0	90.0	
10	10	1220130	JAHITH	2000-11-09	MALE	THANDALAM	34.0	45.0	50.0	
11	11	1220131	KANI	2000-11-21	FEMALE	CHITHUR	96.0	95.0	96.0	
14	14	1220134	NANI	20001109	MALE	POONAMALEE	79.0	77.0	80.0	
16	16	1220136	PRATHAP	20000921	MALE	KANCHIPURAM	86.0	84.0	90.0	
18	18	1220138	RATHI	20001121	FEMALE	KANCHIPURAM	81.0	86.0	90.0	
20	20	1220140	SANTHOSH	20001002	MALE	KANCHIPURAM	76.0	69.0	80.0	

	M4	TOTAL	AVG
1	56.0	253.0	84.333333
3	74.0	307.0	102.333333
4	92.0	375.0	125.000000
5	91.0	360.0	120.000000
6	49.0	219.0	73.000000
7	49.0	219.0	73.000000
8	95.0	376.0	125.333333
10	34.0	163.0	54.333333
11	96.0	383.0	127.666667
14	79.0	315.0	105.000000
16	86.0	346.0	115.333333
18	81.0	338.0	112.666667
20	76.0	301.0	100.333333

[ ]:
```
x2=df.dropna(how="all").shape
x2
```

[ ]: (21, 12)

[ ]:
```
tot=df.dropna(subset=["TOTAL"],how="any").shape
tot
```

[ ]: (16, 12)

[ ]:
```
tot=df.dropna(subset=["TOTAL"],how="any")
tot
```

[ ]:

	SNO	REGNO	NAME	DOB	GENDER	ADDRESS	M1	M2	M3	\
1	2	1220122	BABU	1999-01-25	MALE	KANCHIPURAM	56.0	61.0	80.0	
3	4	1220124	DEVA	2000-11-09	MALE	POONAMALEE	74.0	79.0	80.0	
4	5	1220125	ESTER	2000-11-21	FEMALE	CHITHUR	92.0	95.0	96.0	
5	6	1220126	FARHANA	1999-03-05	FEMALE	THANDALAM	91.0	88.0	90.0	
6	7	1220127	GANI	2000-10-02	MALE	KANCHIPURAM	49.0	51.0	70.0	
7	7	1220127	GANI	2000-10-02	MALE	KANCHIPURAM	49.0	51.0	70.0	
8	8	1220128	HEMA	1999-01-25	FEMALE	POONAMALEE	95.0	96.0	90.0	
10	10	1220130	JAHITH	2000-11-09	MALE	THANDALAM	34.0	45.0	50.0	
11	11	1220131	KANI	2000-11-21	FEMALE	CHITHUR	96.0	95.0	96.0	
12	12	1220132	LATHESSH	1999-03-05	MALE	THANDALAM	NaN	68.0	70.0	
14	14	1220134	NANI	20001109	MALE	POONAMALEE	79.0	77.0	80.0	
15	15	1220135	NaN	19990125	NaN	NaN	NaN	NaN	NaN	
16	16	1220136	PRATHAP	20000921	MALE	KANCHIPURAM	86.0	84.0	90.0	
17	17	1220137	RAGHU	20001109	MALE	POONAMALEE	67.0	64.0	70.0	
18	18	1220138	RATHI	20001121	FEMALE	KANCHIPURAM	81.0	86.0	90.0	
20	20	1220140	SANTHOSH	20001002	MALE	KANCHIPURAM	76.0	69.0	80.0	

	M4	TOTAL	AVG
1	56.0	253.0	84.333333
3	74.0	307.0	102.333333
4	92.0	375.0	125.000000
5	91.0	360.0	120.000000
6	49.0	219.0	73.000000
7	49.0	219.0	73.000000
8	95.0	376.0	125.333333
10	34.0	163.0	54.333333
11	96.0	383.0	127.666667
12	70.0	208.0	69.333333
14	79.0	315.0	105.000000
15	NaN	0.0	0.000000
16	86.0	346.0	115.333333
17	NaN	201.0	67.000000
18	81.0	338.0	112.666667

```
20  76.0  301.0  100.333333
```

[ ]:
```
tot=df.dropna(subset=['M1','M2','M3','M4'],how="any")
tot
```

[ ]:
```
     SNO    REGNO      NAME         DOB  GENDER      ADDRESS    M1    M2    M3  \
1      2  1220122      BABU  1999-01-25    MALE  KANCHIPURAM  56.0  61.0  80.0
3      4  1220124      DEVA  2000-11-09    MALE   POONAMALEE  74.0  79.0  80.0
4      5  1220125     ESTER  2000-11-21  FEMALE      CHITHUR  92.0  95.0  96.0
5      6  1220126   FARHANA  1999-03-05  FEMALE    THANDALAM  91.0  88.0  90.0
6      7  1220127      GANI  2000-10-02    MALE  KANCHIPURAM  49.0  51.0  70.0
7      7  1220127      GANI  2000-10-02    MALE  KANCHIPURAM  49.0  51.0  70.0
8      8  1220128      HEMA  1999-01-25  FEMALE   POONAMALEE  95.0  96.0  90.0
10    10  1220130    JAHITH  2000-11-09    MALE    THANDALAM  34.0  45.0  50.0
11    11  1220131      KANI  2000-11-21  FEMALE      CHITHUR  96.0  95.0  96.0
14    14  1220134      NANI    20001109    MALE   POONAMALEE  79.0  77.0  80.0
16    16  1220136   PRATHAP    20000921    MALE  KANCHIPURAM  86.0  84.0  90.0
18    18  1220138     RATHI    20001121  FEMALE  KANCHIPURAM  81.0  86.0  90.0
20    20  1220140  SANTHOSH    20001002    MALE  KANCHIPURAM  76.0  69.0  80.0

      M4  TOTAL         AVG
1   56.0  253.0   84.333333
3   74.0  307.0  102.333333
4   92.0  375.0  125.000000
5   91.0  360.0  120.000000
6   49.0  219.0   73.000000
7   49.0  219.0   73.000000
8   95.0  376.0  125.333333
10  34.0  163.0   54.333333
11  96.0  383.0  127.666667
14  79.0  315.0  105.000000
16  86.0  346.0  115.333333
18  81.0  338.0  112.666667
20  76.0  301.0  100.333333
```

[ ]:
```
tot=df.dropna(subset=['M1','M2','M3','M4'],how="any").shape
tot
```

[ ]:
```
(13, 12)
```

[ ]:
```
s=df.fillna(method="ffill")
s
```

```
<ipython-input-33-b8fa547bb146>:1: FutureWarning: DataFrame.fillna with 'method'
is deprecated and will raise in a future version. Use obj.ffill() or obj.bfill()
instead.
  s=df.fillna(method="ffill")
```

[ ]:

	SNO	REGNO	NAME	DOB	GENDER	ADDRESS	M1	M2	M3	\
0	1	1220121	ARUN	2000-02-10	MALE	THANDALAM	82.0	81.0	90.0	
1	2	1220122	BABU	1999-01-25	MALE	KANCHIPURAM	56.0	61.0	80.0	
2	3	1220123	CHARAN	2000.09.21	MALE	THANDALAM	56.0	59.0	60.0	
3	4	1220124	DEVA	2000-11-09	MALE	POONAMALEE	74.0	79.0	80.0	
4	5	1220125	ESTER	2000-11-21	FEMALE	CHITHUR	92.0	95.0	96.0	
5	6	1220126	FARHANA	1999-03-05	FEMALE	THANDALAM	91.0	88.0	90.0	
6	7	1220127	GANI	2000-10-02	MALE	KANCHIPURAM	49.0	51.0	70.0	
7	7	1220127	GANI	2000-10-02	MALE	KANCHIPURAM	49.0	51.0	70.0	
8	8	1220128	HEMA	1999-01-25	FEMALE	POONAMALEE	95.0	96.0	90.0	
9	9	1220129	INDRA	2000.09.21	FEMALE	KANCHIPURAM	64.0	96.0	90.0	
10	10	1220130	JAHITH	2000-11-09	MALE	THANDALAM	34.0	45.0	50.0	
11	11	1220131	KANI	2000-11-21	FEMALE	CHITHUR	96.0	95.0	96.0	
12	12	1220132	LATHESSH	1999-03-05	MALE	THANDALAM	96.0	68.0	70.0	
13	13	1220133	MANI	2000-10-02	MALE	KANCHIPURAM	71.0	76.0	70.0	
14	14	1220134	NANI	20001109	MALE	POONAMALEE	79.0	77.0	80.0	
15	15	1220135	NANI	19990125	MALE	POONAMALEE	79.0	77.0	80.0	
16	16	1220136	PRATHAP	20000921	MALE	KANCHIPURAM	86.0	84.0	90.0	
17	17	1220137	RAGHU	20001109	MALE	POONAMALEE	67.0	64.0	70.0	
18	18	1220138	RATHI	20001121	FEMALE	KANCHIPURAM	81.0	86.0	90.0	
19	19	1220139	SARVESH	19990305	MALE	THANDALAM	84.0	87.0	90.0	
20	20	1220140	SANTHOSH	20001002	MALE	KANCHIPURAM	76.0	69.0	80.0	

	M4	TOTAL	AVG
0	NaN	NaN	NaN
1	56.0	253.0	84.333333
2	70.0	253.0	0.000000
3	74.0	307.0	102.333333
4	92.0	375.0	125.000000
5	91.0	360.0	120.000000
6	49.0	219.0	73.000000
7	49.0	219.0	73.000000
8	95.0	376.0	125.333333
9	64.0	376.0	0.000000
10	34.0	163.0	54.333333
11	96.0	383.0	127.666667
12	70.0	208.0	69.333333
13	71.0	208.0	0.000000
14	79.0	315.0	105.000000
15	79.0	0.0	0.000000
16	86.0	346.0	115.333333
17	86.0	201.0	67.000000
18	81.0	338.0	112.666667
19	84.0	338.0	0.000000
20	76.0	301.0	100.333333

[ ]:
```
df.isna().sum()
```

```
[ ]: SNO        0
     REGNO      0
     NAME       1
     DOB        0
     GENDER     1
     ADDRESS    1
     M1         3
     M2         2
     M3         4
     M4         3
     TOTAL      5
     AVG        1
     dtype: int64
```

```
[ ]: df['M1']
```

```
[ ]: 0     82.0
     1     56.0
     2      NaN
     3     74.0
     4     92.0
     5     91.0
     6     49.0
     7     49.0
     8     95.0
     9     64.0
     10    34.0
     11    96.0
     12     NaN
     13    71.0
     14    79.0
     15     NaN
     16    86.0
     17    67.0
     18    81.0
     19    84.0
     20    76.0
     Name: M1, dtype: float64
```

```
[ ]: df.isnull()
```

```
[ ]:       SNO  REGNO   NAME    DOB  GENDER  ADDRESS     M1     M2     M3     M4  \
     0   False  False  False  False   False    False  False  False  False   True
     1   False  False  False  False   False    False  False  False  False  False
     2   False  False  False  False   False    False   True  False  False  False
     3   False  False  False  False   False    False  False  False  False  False
     4   False  False  False  False   False    False  False  False  False  False
```

```
5   False  False  False  False   False    False  False  False  False  False
6   False  False  False  False   False    False  False  False  False  False
7   False  False  False  False   False    False  False  False  False  False
8   False  False  False  False   False    False  False  False  False  False
9   False  False  False  False   False    False  False   True   True  False
10  False  False  False  False   False    False  False  False  False  False
11  False  False  False  False   False    False  False  False  False  False
12  False  False  False  False   False    False   True  False  False  False
13  False  False  False  False   False    False  False  False   True  False
14  False  False  False  False   False    False  False  False  False  False
15  False  False   True  False    True     True   True   True   True   True
16  False  False  False  False   False    False  False  False  False  False
17  False  False  False  False   False    False  False  False  False   True
18  False  False  False  False   False    False  False  False  False  False
19  False  False  False  False   False    False  False  False   True  False
20  False  False  False  False   False    False  False  False  False  False

    TOTAL    AVG
0    True   True
1   False  False
2    True  False
3   False  False
4   False  False
5   False  False
6   False  False
7   False  False
8   False  False
9    True  False
10  False  False
11  False  False
12  False  False
13   True  False
14  False  False
15  False  False
16  False  False
17  False  False
18  False  False
19   True  False
20  False  False
```

[ ]:
```
df.notnull()
```

[ ]:
```
     SNO  REGNO   NAME   DOB  GENDER  ADDRESS     M1     M2     M3     M4  \
0   True   True   True  True    True     True   True   True   True  False
1   True   True   True  True    True     True   True   True   True   True
2   True   True   True  True    True     True  False   True   True   True
3   True   True   True  True    True     True   True   True   True   True
```

```
4   True   True   True  True    True     True   True   True   True   True
5   True   True   True  True    True     True   True   True   True   True
6   True   True   True  True    True     True   True   True   True   True
7   True   True   True  True    True     True   True   True   True   True
8   True   True   True  True    True     True   True   True   True   True
9   True   True   True  True    True     True   True  False  False   True
10  True   True   True  True    True     True   True   True   True   True
11  True   True   True  True    True     True   True   True   True   True
12  True   True   True  True    True     True  False   True   True   True
13  True   True   True  True    True     True   True   True  False   True
14  True   True   True  True    True     True   True   True   True   True
15  True   True  False  True   False    False  False  False  False  False
16  True   True   True  True    True     True   True   True   True   True
17  True   True   True  True    True     True   True   True   True  False
18  True   True   True  True    True     True   True   True   True   True
19  True   True   True  True    True     True   True   True  False   True
20  True   True   True  True    True     True   True   True   True   True

    TOTAL    AVG
0   False  False
1    True   True
2   False   True
3    True   True
4    True   True
5    True   True
6    True   True
7    True   True
8    True   True
9   False   True
10   True   True
11   True   True
12   True   True
13  False   True
14   True   True
15   True   True
16   True   True
17   True   True
18   True   True
19  False   True
20   True   True
```

[ ]:
```
df.dropna(axis=0)
```

[ ]:
```
    SNO    REGNO      NAME         DOB  GENDER      ADDRESS    M1    M2    M3  \
1     2  1220122      BABU  1999-01-25    MALE  KANCHIPURAM  56.0  61.0  80.0
3     4  1220124      DEVA  2000-11-09    MALE   POONAMALEE  74.0  79.0  80.0
4     5  1220125     ESTER  2000-11-21  FEMALE      CHITHUR  92.0  95.0  96.0
```

5	6	1220126	FARHANA	1999-03-05	FEMALE	THANDALAM	91.0	88.0	90.0
6	7	1220127	GANI	2000-10-02	MALE	KANCHIPURAM	49.0	51.0	70.0
7	7	1220127	GANI	2000-10-02	MALE	KANCHIPURAM	49.0	51.0	70.0
8	8	1220128	HEMA	1999-01-25	FEMALE	POONAMALEE	95.0	96.0	90.0
10	10	1220130	JAHITH	2000-11-09	MALE	THANDALAM	34.0	45.0	50.0
11	11	1220131	KANI	2000-11-21	FEMALE	CHITHUR	96.0	95.0	96.0
14	14	1220134	NANI	20001109	MALE	POONAMALEE	79.0	77.0	80.0
16	16	1220136	PRATHAP	20000921	MALE	KANCHIPURAM	86.0	84.0	90.0
18	18	1220138	RATHI	20001121	FEMALE	KANCHIPURAM	81.0	86.0	90.0
20	20	1220140	SANTHOSH	20001002	MALE	KANCHIPURAM	76.0	69.0	80.0

	M4	TOTAL	AVG
1	56.0	253.0	84.333333
3	74.0	307.0	102.333333
4	92.0	375.0	125.000000
5	91.0	360.0	120.000000
6	49.0	219.0	73.000000
7	49.0	219.0	73.000000
8	95.0	376.0	125.333333
10	34.0	163.0	54.333333
11	96.0	383.0	127.666667
14	79.0	315.0	105.000000
16	86.0	346.0	115.333333
18	81.0	338.0	112.666667
20	76.0	301.0	100.333333

[ ]:
```
df.dropna(axis=1)
```

[ ]:

	SNO	REGNO	DOB
0	1	1220121	2000-02-10
1	2	1220122	1999-01-25
2	3	1220123	2000.09.21
3	4	1220124	2000-11-09
4	5	1220125	2000-11-21
5	6	1220126	1999-03-05
6	7	1220127	2000-10-02
7	7	1220127	2000-10-02
8	8	1220128	1999-01-25
9	9	1220129	2000.09.21
10	10	1220130	2000-11-09
11	11	1220131	2000-11-21
12	12	1220132	1999-03-05
13	13	1220133	2000-10-02
14	14	1220134	20001109
15	15	1220135	19990125
16	16	1220136	20000921
17	17	1220137	20001109

```
18   18  1220138    20001121
19   19  1220139    19990305
20   20  1220140    20001002
```

[ ]:
```
df.duplicated()
```

[ ]:
```
0     False
1     False
2     False
3     False
4     False
5     False
6     False
7      True
8     False
9     False
10    False
11    False
12    False
13    False
14    False
15    False
16    False
17    False
18    False
19    False
20    False
dtype: bool
```

[ ]:
```
m=df.drop_duplicates(inplace=False)
m
```

[ ]:

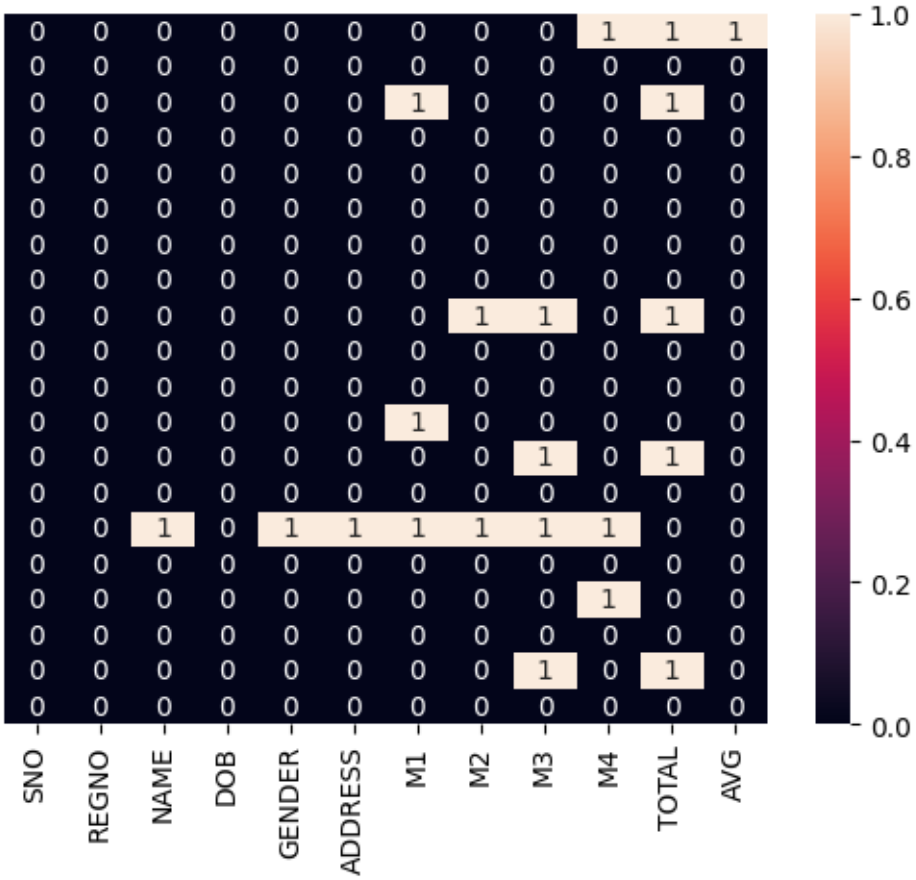
	SNO	REGNO	NAME	DOB	GENDER	ADDRESS	M1	M2	M3	\
0	1	1220121	ARUN	2000-02-10	MALE	THANDALAM	82.0	81.0	90.0	
1	2	1220122	BABU	1999-01-25	MALE	KANCHIPURAM	56.0	61.0	80.0	
2	3	1220123	CHARAN	2000.09.21	MALE	THANDALAM	NaN	59.0	60.0	
3	4	1220124	DEVA	2000-11-09	MALE	POONAMALEE	74.0	79.0	80.0	
4	5	1220125	ESTER	2000-11-21	FEMALE	CHITHUR	92.0	95.0	96.0	
5	6	1220126	FARHANA	1999-03-05	FEMALE	THANDALAM	91.0	88.0	90.0	
6	7	1220127	GANI	2000-10-02	MALE	KANCHIPURAM	49.0	51.0	70.0	
8	8	1220128	HEMA	1999-01-25	FEMALE	POONAMALEE	95.0	96.0	90.0	
9	9	1220129	INDRA	2000.09.21	FEMALE	KANCHIPURAM	64.0	NaN	NaN	
10	10	1220130	JAHITH	2000-11-09	MALE	THANDALAM	34.0	45.0	50.0	
11	11	1220131	KANI	2000-11-21	FEMALE	CHITHUR	96.0	95.0	96.0	
12	12	1220132	LATHESSH	1999-03-05	MALE	THANDALAM	NaN	68.0	70.0	
13	13	1220133	MANI	2000-10-02	MALE	KANCHIPURAM	71.0	76.0	NaN	
14	14	1220134	NANI	20001109	MALE	POONAMALEE	79.0	77.0	80.0	

15	15	1220135	NaN	19990125	NaN	NaN	NaN	NaN	NaN
16	16	1220136	PRATHAP	20000921	MALE	KANCHIPURAM	86.0	84.0	90.0
17	17	1220137	RAGHU	20001109	MALE	POONAMALEE	67.0	64.0	70.0
18	18	1220138	RATHI	20001121	FEMALE	KANCHIPURAM	81.0	86.0	90.0
19	19	1220139	SARVESH	19990305	MALE	THANDALAM	84.0	87.0	NaN
20	20	1220140	SANTHOSH	20001002	MALE	KANCHIPURAM	76.0	69.0	80.0

	M4	TOTAL	AVG
0	NaN	NaN	NaN
1	56.0	253.0	84.333333
2	70.0	NaN	0.000000
3	74.0	307.0	102.333333
4	92.0	375.0	125.000000
5	91.0	360.0	120.000000
6	49.0	219.0	73.000000
8	95.0	376.0	125.333333
9	64.0	NaN	0.000000
10	34.0	163.0	54.333333
11	96.0	383.0	127.666667
12	70.0	208.0	69.333333
13	71.0	NaN	0.000000
14	79.0	315.0	105.000000
15	NaN	0.0	0.000000
16	86.0	346.0	115.333333
17	NaN	201.0	67.000000
18	81.0	338.0	112.666667
19	84.0	NaN	0.000000
20	76.0	301.0	100.333333

[ ]:
```
import seaborn as sns
sns.heatmap(df.isnull(),yticklabels=False,annot=True)
```

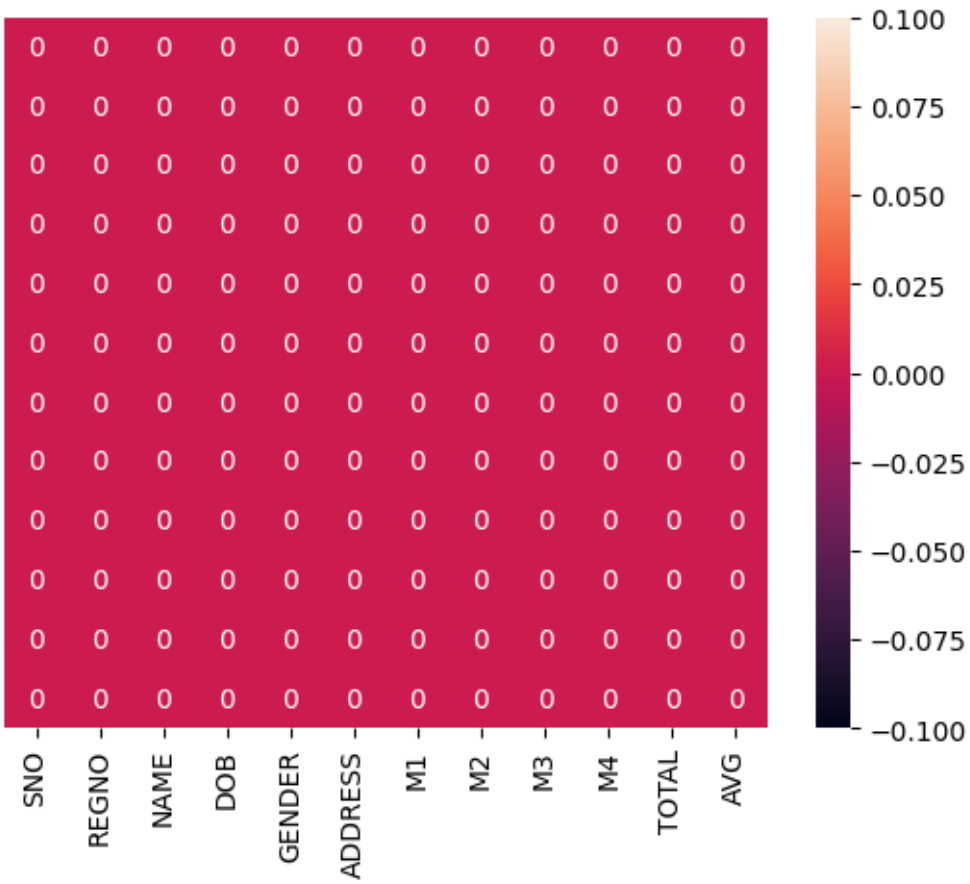
[ ]: <Axes: >



[ ]:
```
df.dropna(inplace=True)
```

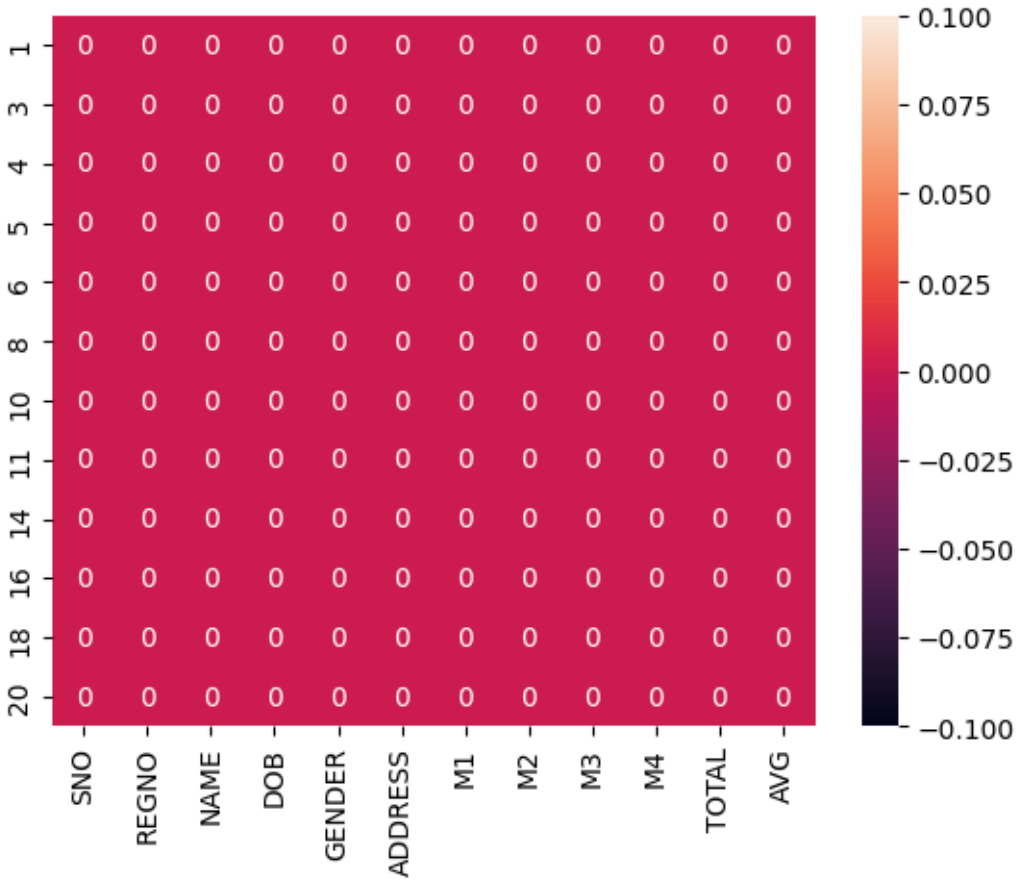
[ ]:
```
import seaborn as sns
sns.heatmap(df.isnull(),yticklabels=False,annot=True)
```

[ ]: <Axes: >



[ ]:
```
import seaborn as sns
sns.heatmap(df.isnull(),yticklabels=True,annot=True)
```

[ ]: <Axes: >



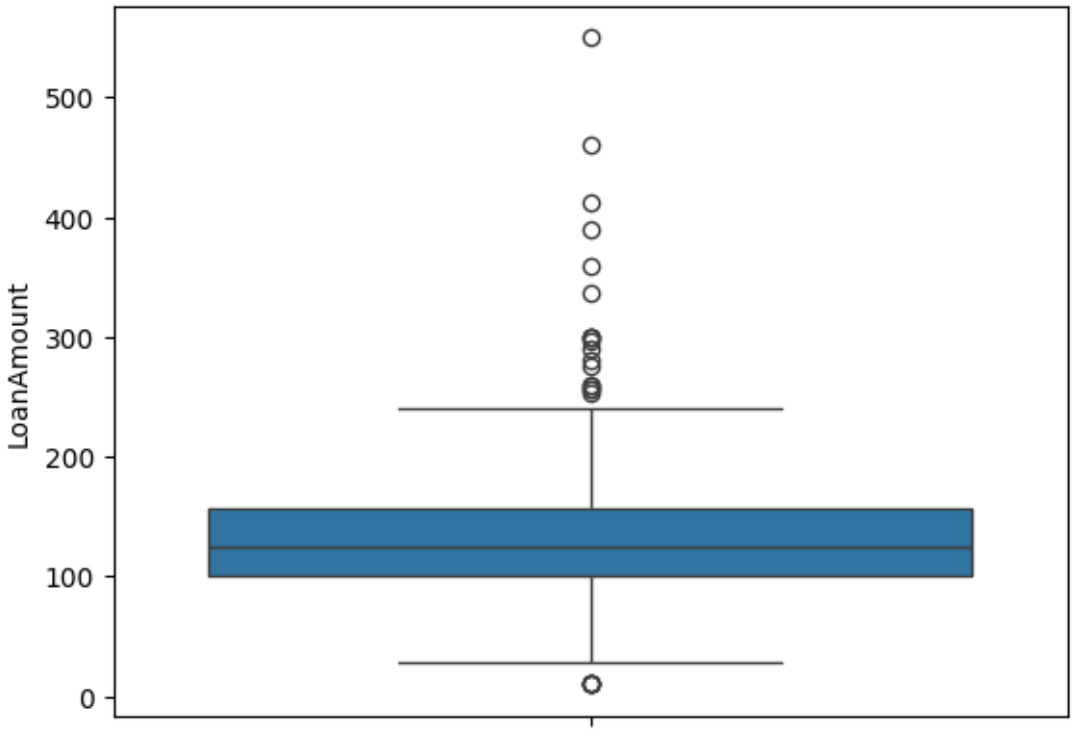
[ ]:
```
df.dtypes
```

[ ]:
```
SNO          int64
REGNO        int64
NAME        object
DOB         object
GENDER      object
ADDRESS     object
M1         float64
M2         float64
M3         float64
M4         float64
TOTAL      float64
AVG        float64
dtype: object
```

[ ]:
```
import pandas as pd
import numpy as np
import seaborn as sns
```

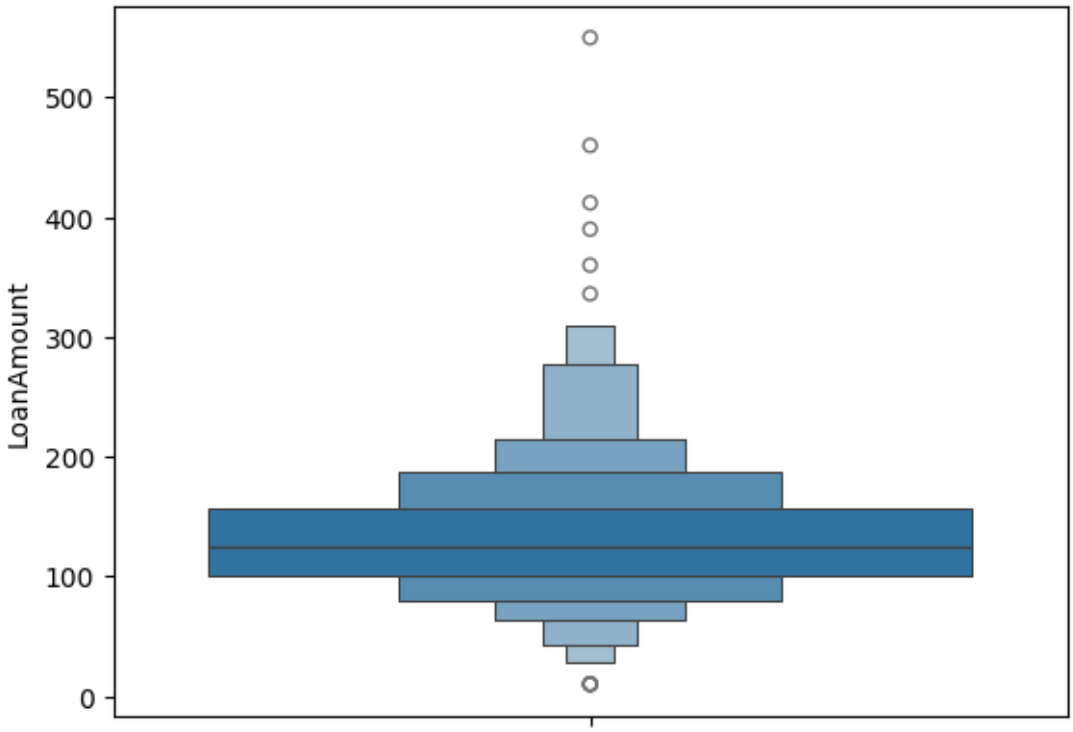
```
df=pd.read_csv("/content/Loan_data.csv")
df.fillna(10,inplace=True)
sns.boxplot(data=df['LoanAmount'])
```

[ ]: <Axes: ylabel='LoanAmount'>



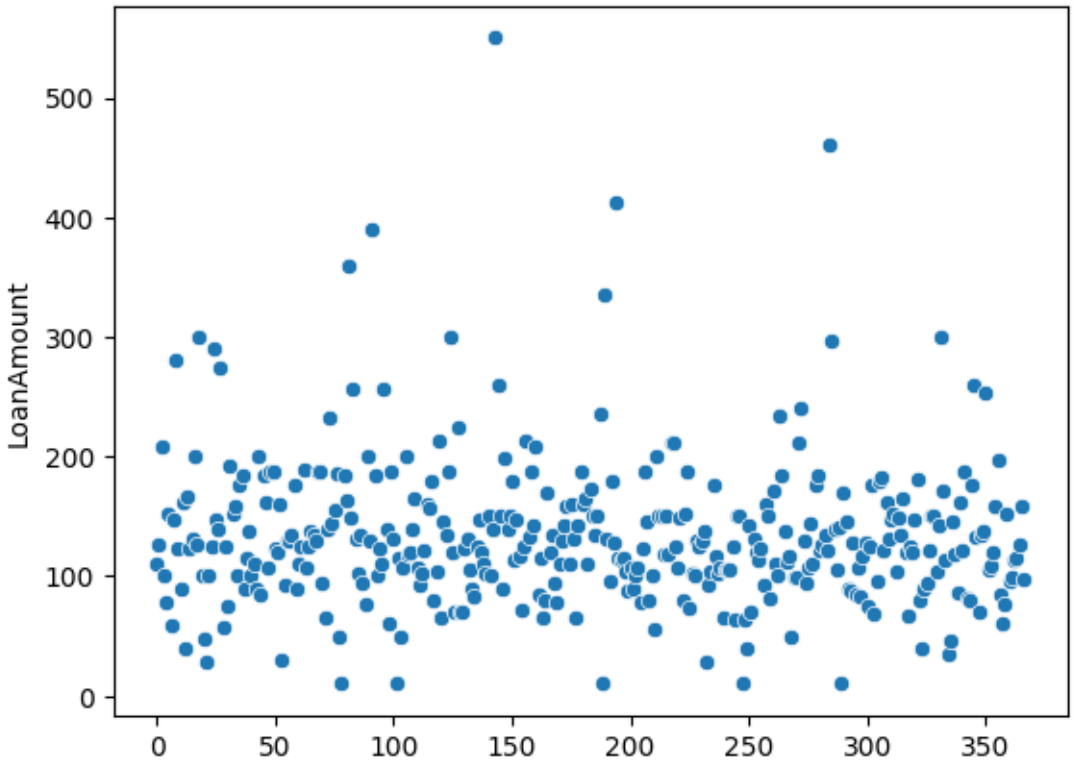
[ ]:
```
sns.boxenplot(data=df['LoanAmount'])
```

[ ]: <Axes: ylabel='LoanAmount'>



[ ]:
```
sns.scatterplot(data=df['LoanAmount'])
```

[ ]:
```
<Axes: ylabel='LoanAmount'>
```



[ ]:
```
q1=np.percentile(df['LoanAmount'],25)
q3=np.percentile(df['LoanAmount'],75)
iqr=q3-q1
lower_bound=q1-1.5*iqr
upper_bound=q3+1.5*iqr
print("LOWER BOUND",lower_bound)
print("UPPERBOUND",upper_bound)
af=df[((df['LoanAmount']>=lower_bound)&(df['LoanAmount']<=upper_bound))]
print("AFTER REMOVING OUTLIERS",af['LoanAmount'])
sns.boxplot(data=af['LoanAmount'])
```

```
LOWER BOUND 13.75
UPPERBOUND 243.75
AFTER REMOVING OUTLIERS 0      110.0
1      126.0
2      208.0
3      100.0
4       78.0
       …
362    113.0
363    115.0
364    126.0
```

```
365    158.0
366     98.0
Name: LoanAmount, Length: 344, dtype: float64
```

[ ]: <Axes: ylabel='LoanAmount'>