TASK-6- [PYTHON - MEDICORE LVL]

NAME-Buchupalli Keerthi lakshmi

Roll.No-CH.EN.U4CSE20112

Discord server-keerthi#3423

Question -1

Write a python program that reads the contents from the given file 'onelinefile.txt'. The file contains a single line which is of the format (int)(string)(float)(string) repeatedly. For e.g.

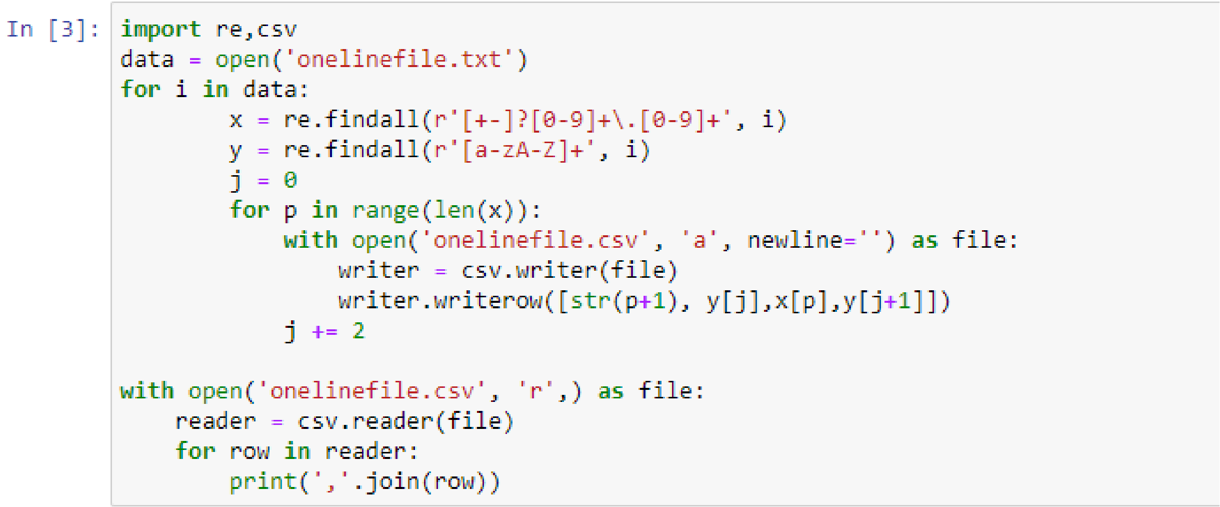
1Aaa3.5Maths2Bbb4.2Physics3Ccc7.62Chemistry

Your main task is to split the contents of the given file based on their format and write it into a .csv file say 'Filename2.csv'. For e.g. the above txt file should be converted into a csv file such that the contents look like this:

1,Aaa,3.5,Maths

2,Bbb,4.2,Physics

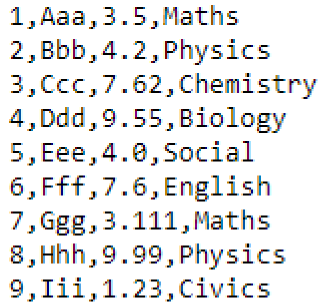
3,Ccc,7.62,Chemistry



```
In [3]: import re,csv
        data = open('onelinefile.txt')
        for i in data:
                x = re.findall(r'[+-]?[0-9]+\.[0-9]+', i)
                y = re.findall(r'[a-zA-Z]+', i)
                j = 0
                for p in range(len(x)):
                    with open('onelinefile.csv', 'a', newline='') as file:
                        writer = csv.writer(file)
                        writer.writerow([str(p+1), y[j],x[p],y[j+1]])
                    j += 2

        with open('onelinefile.csv', 'r',) as file:
            reader = csv.reader(file)
            for row in reader:
                print(','.join(row))
```

Output



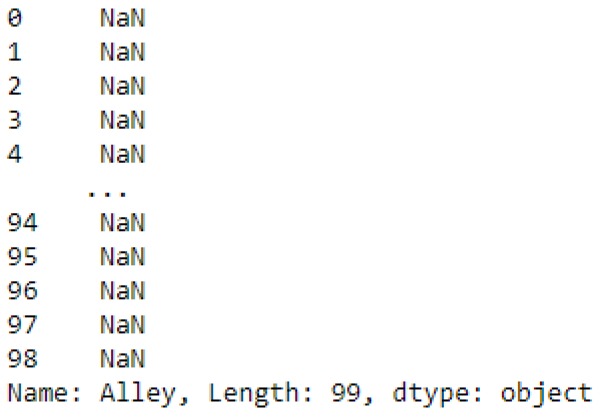
```
1,Aaa,3.5,Maths
2,Bbb,4.2,Physics
3,Ccc,7.62,Chemistry
4,Ddd,9.55,Biology
5,Eee,4.0,Social
6,Fff,7.6,English
7,Ggg,3.111,Maths
8,Hhh,9.99,Physics
9,Iii,1.23,Civics
```

Question -2

Data formatting

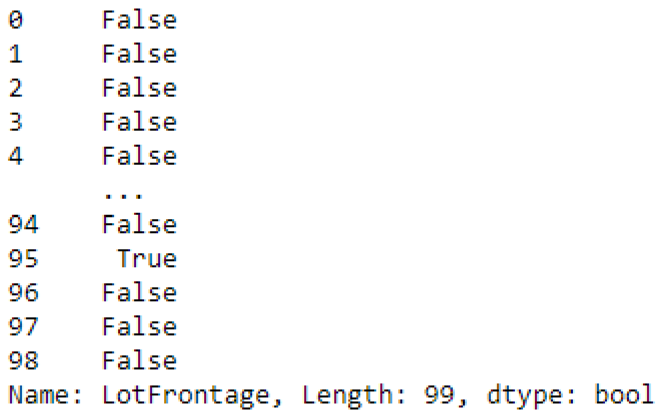
Python libraries represent missing numbers as nan which is short for “not a number”. Most libraries (including scikit-learn) will give you an error if you try to build a model using data with missing values. One of the common solution to get around this issue is to impute or fill in the missing value with a number or value of same format. From the given dataset, find the missing values(Nan/NA/-/Nil) and change those values into an appropriate number.

```
import pandas as pd
import numpy as np
df = pd.read_csv("https://raw.githubusercontent.com/cognizance-amrita/AI-
Tasks/main/Task-1/Q2-Dataset.csv")
df.head()
missing_value_formats = ["n.a.","?","NA","n/a", "na", "--"]
df = pd.read_csv("https://raw.githubusercontent.com/cognizance-amrita/AI-
Tasks/main/Task-1/Q2-Dataset.csv", na_values = missing_value_formats)
print(df['Alley'].head(100))
```



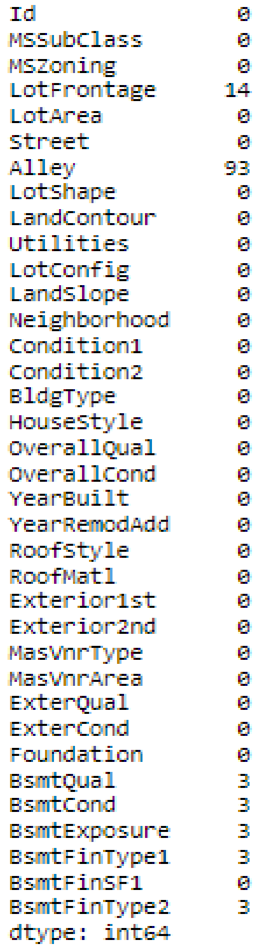
```
0     NaN
1     NaN
2     NaN
3     NaN
4     NaN
     ...
94    NaN
95    NaN
96    NaN
97    NaN
98    NaN
Name: Alley, Length: 99, dtype: object
```

```
print(df['LotFrontage'].isnull())
```



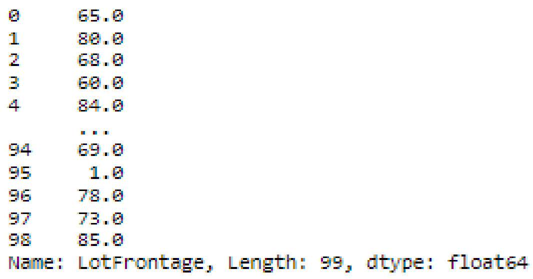
```
0     False
1     False
2     False
3     False
4     False
      ...
94    False
95     True
96    False
97    False
98    False
Name: LotFrontage, Length: 99, dtype: bool
```

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



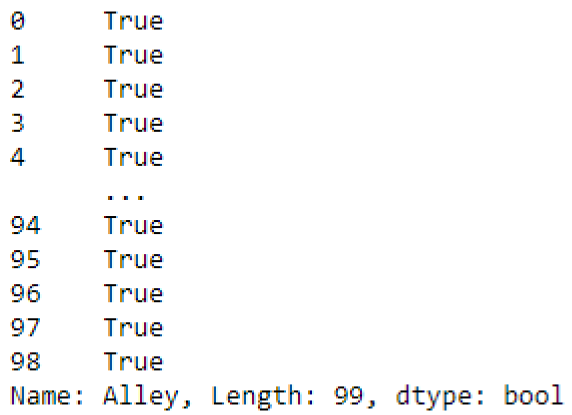
```
Id               0
MSSubClass       0
MSZoning         0
LotFrontage     14
LotArea          0
Street           0
Alley           93
LotShape         0
LandContour      0
Utilities        0
LotConfig        0
LandSlope        0
Neighborhood     0
Condition1       0
Condition2       0
BldgType         0
HouseStyle       0
OverallQual      0
OverallCond      0
YearBuilt        0
YearRemodAdd     0
RoofStyle        0
RoofMatl         0
Exterior1st      0
Exterior2nd      0
MasVnrType       0
MasVnrArea       0
ExterQual        0
ExterCond        0
Foundation       0
BsmtQual         3
BsmtCond         3
BsmtExposure     3
BsmtFinType1     3
BsmtFinSF1       0
BsmtFinType2     3
dtype: int64
```

```
df['LotFrontage'].fillna(1, inplace=True)
print(df['LotFrontage'])
```



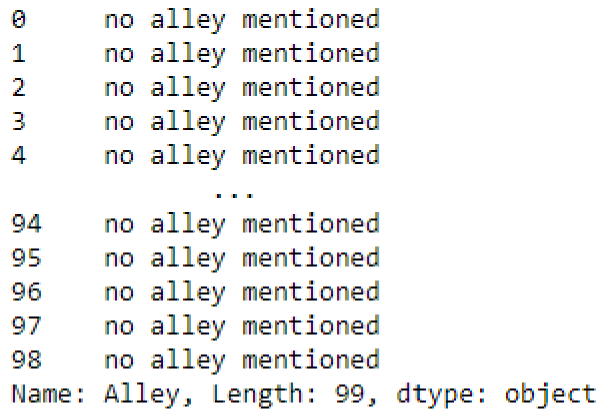
```
0     65.0
1     80.0
2     68.0
3     60.0
4     84.0
      ...
94    69.0
95     1.0
96    78.0
97    73.0
98    85.0
Name: LotFrontage, Length: 99, dtype: float64
```

```
print(df['Alley'].isnull())
```



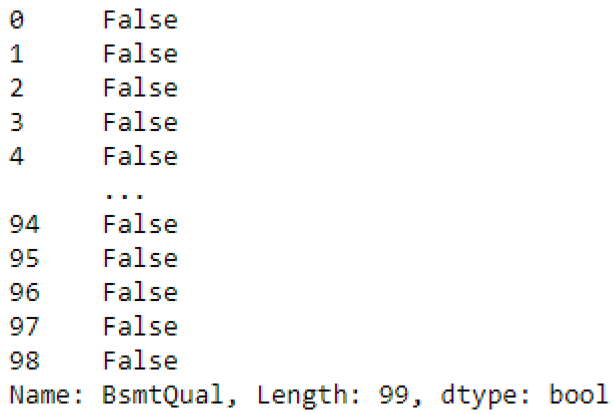
```
0     True
1     True
2     True
3     True
4     True
      ...
94    True
95    True
96    True
97    True
98    True
Name: Alley, Length: 99, dtype: bool
```

```
df['Alley'].fillna('no alley mentioned', inplace=True)
print(df['Alley'])
```



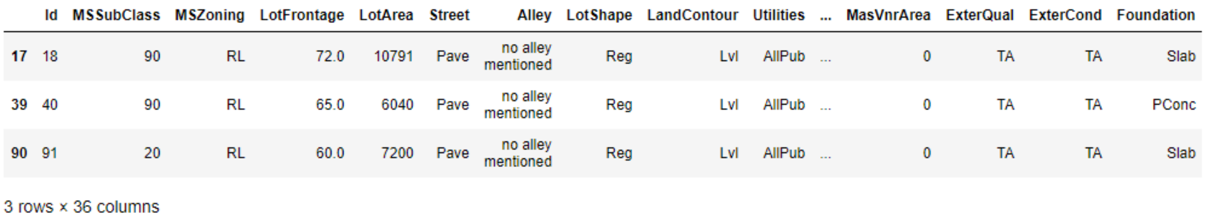
```
0     no alley mentioned
1     no alley mentioned
2     no alley mentioned
3     no alley mentioned
4     no alley mentioned
             ...
94    no alley mentioned
95    no alley mentioned
96    no alley mentioned
97    no alley mentioned
98    no alley mentioned
Name: Alley, Length: 99, dtype: object
```

```
print(df['BsmtQual'].isnull())
```



```
0     False
1     False
2     False
3     False
4     False
      ...
94    False
95    False
96    False
97    False
98    False
Name: BsmtQual, Length: 99, dtype: bool
```

```
df[df['BsmtQual'].isnull()]
```

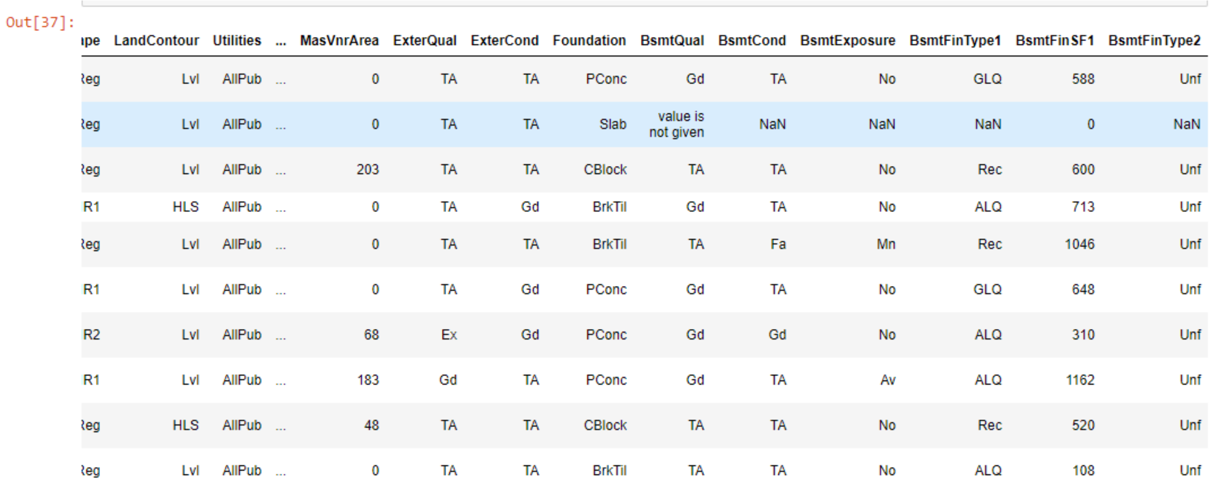


	Id	MSSubClass	MSZoning	LotFrontage	LotArea	Street	Alley	LotShape	LandContour	Utilities	...	MasVnrArea	ExterQual	ExterCond	Foundation
17	18	90	RL	72.0	10791	Pave	no alley mentioned	Reg	Lvl	AllPub	...	0	TA	TA	Slab
39	40	90	RL	65.0	6040	Pave	no alley mentioned	Reg	Lvl	AllPub	...	0	TA	TA	PConc
90	91	20	RL	60.0	7200	Pave	no alley mentioned	Reg	Lvl	AllPub	...	0	TA	TA	Slab

3 rows × 36 columns

df['BsmtQual'].fillna('value is not given here', inplace=True

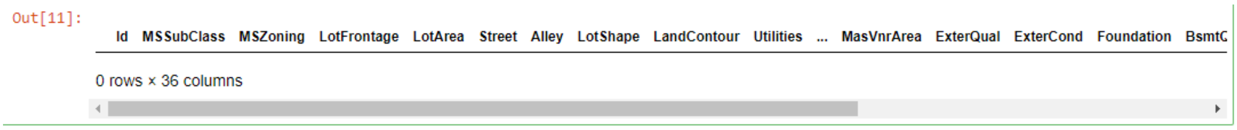
df.tail(10)



Out[37]:

pe	LandContour	Utilities	...	MasVnrArea	ExterQual	ExterCond	Foundation	BsmtQual	BsmtCond	BsmtExposure	BsmtFinType1	BsmtFinSF1	BsmtFinType2
Reg	Lvl	AllPub	...	0	TA	TA	PConc	Gd	TA	No	GLQ	588	Unf
Reg	Lvl	AllPub	...	0	TA	TA	Slab	value is not given	NaN	NaN	NaN	0	NaN
Reg	Lvl	AllPub	...	203	TA	TA	CBlock	TA	TA	No	Rec	600	Unf
R1	HLS	AllPub	...	0	TA	Gd	BrkTil	Gd	TA	No	ALQ	713	Unf
Reg	Lvl	AllPub	...	0	TA	TA	BrkTil	TA	Fa	Mn	Rec	1046	Unf
R1	Lvl	AllPub	...	0	TA	Gd	PConc	Gd	TA	No	GLQ	648	Unf
R2	Lvl	AllPub	...	68	Ex	Gd	PConc	Gd	Gd	No	ALQ	310	Unf
R1	Lvl	AllPub	...	183	Gd	TA	PConc	Gd	TA	Av	ALQ	1162	Unf
Reg	HLS	AllPub	...	48	TA	TA	CBlock	TA	TA	No	Rec	520	Unf
Reg	Lvl	AllPub	...	0	TA	TA	BrkTil	TA	TA	No	ALQ	108	Unf

df[df['BsmtQual'].isnull()]

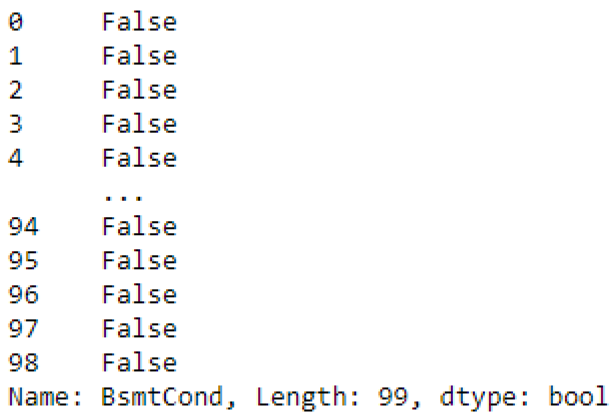


Out[11]:

Id	MSSubClass	MSZoning	LotFrontage	LotArea	Street	Alley	LotShape	LandContour	Utilities	...	MasVnrArea	ExterQual	ExterCond	Foundation	BsmtC

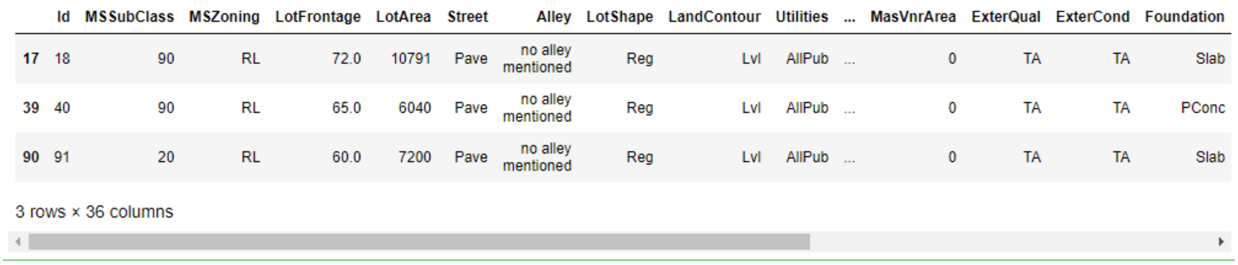
0 rows × 36 columns

print(df['BsmtCond'].isnull())



```
0     False
1     False
2     False
3     False
4     False
      ...
94    False
95    False
96    False
97    False
98    False
Name: BsmtCond, Length: 99, dtype: bool
```

```
df[df['BsmtCond'].isnull()]
```

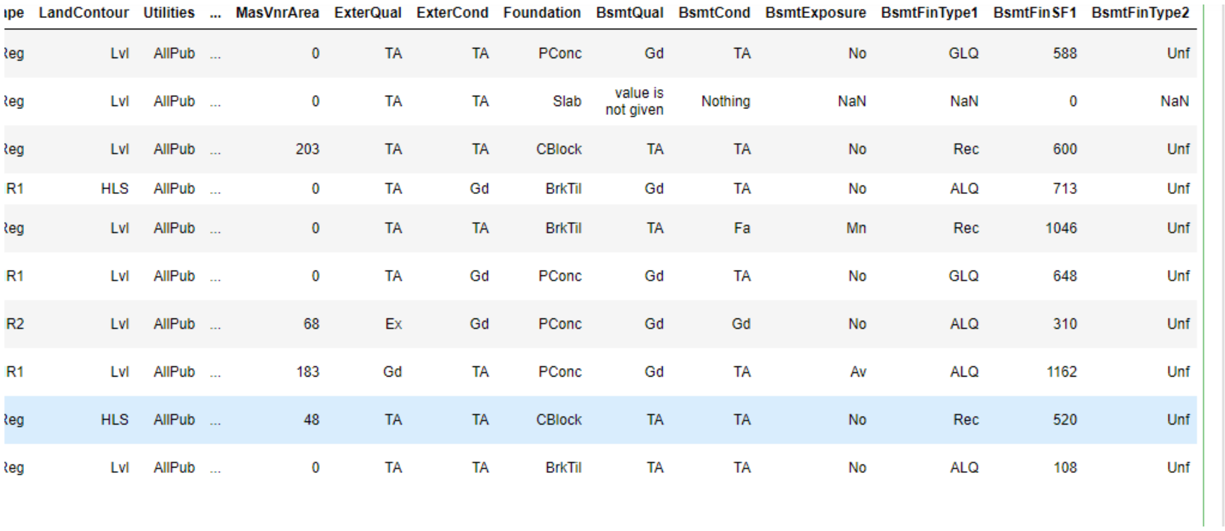


	Id	MSSubClass	MSZoning	LotFrontage	LotArea	Street	Alley	LotShape	LandContour	Utilities	...	MasVnrArea	ExterQual	ExterCond	Foundation
17	18	90	RL	72.0	10791	Pave	no alley mentioned	Reg	Lvl	AllPub	...	0	TA	TA	Slab
39	40	90	RL	65.0	6040	Pave	no alley mentioned	Reg	Lvl	AllPub	...	0	TA	TA	PConc
90	91	20	RL	60.0	7200	Pave	no alley mentioned	Reg	Lvl	AllPub	...	0	TA	TA	Slab

3 rows × 36 columns

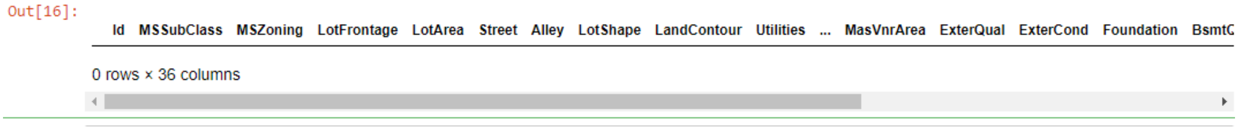
```
df['BsmtCond'].fillna('Nothing', inplace=True)

df.tail(10)
```



ipe	LandContour	Utilities	...	MasVnrArea	ExterQual	ExterCond	Foundation	BsmtQual	BsmtCond	BsmtExposure	BsmtFinType1	BsmtFinSF1	BsmtFinType2
Reg	Lvl	AllPub	...	0	TA	TA	PConc	Gd	TA	No	GLQ	588	Unf
Reg	Lvl	AllPub	...	0	TA	TA	Slab	value is not given	Nothing	NaN	NaN	0	NaN
Reg	Lvl	AllPub	...	203	TA	TA	CBlock	TA	TA	No	Rec	600	Unf
R1	HLS	AllPub	...	0	TA	Gd	BrkTil	Gd	TA	No	ALQ	713	Unf
Reg	Lvl	AllPub	...	0	TA	TA	BrkTil	TA	Fa	Mn	Rec	1046	Unf
R1	Lvl	AllPub	...	0	TA	Gd	PConc	Gd	TA	No	GLQ	648	Unf
R2	Lvl	AllPub	...	68	Ex	Gd	PConc	Gd	Gd	No	ALQ	310	Unf
R1	Lvl	AllPub	...	183	Gd	TA	PConc	Gd	TA	Av	ALQ	1162	Unf
Reg	HLS	AllPub	...	48	TA	TA	CBlock	TA	TA	No	Rec	520	Unf
Reg	Lvl	AllPub	...	0	TA	TA	BrkTil	TA	TA	No	ALQ	108	Unf

```
df[df['BsmtCond'].isnull()]
```

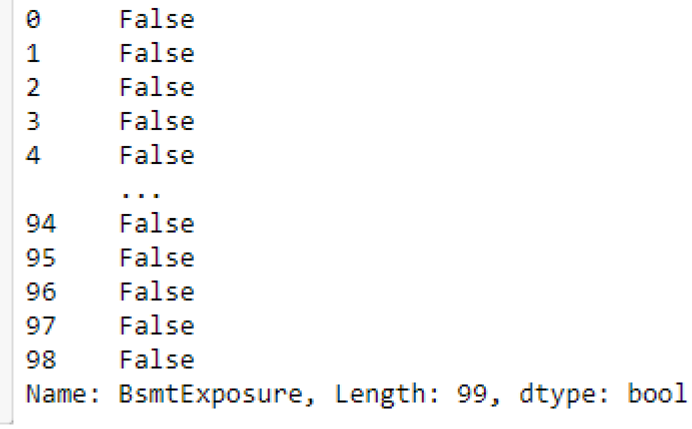


Out[16]:

	Id	MSSubClass	MSZoning	LotFrontage	LotArea	Street	Alley	LotShape	LandContour	Utilities	...	MasVnrArea	ExterQual	ExterCond	Foundation	BsmtC

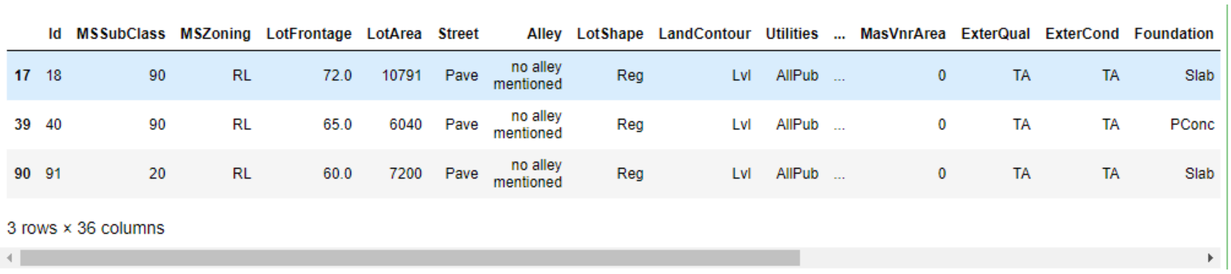
0 rows × 36 columns

```
print(df['BsmtExposure'].isnull())
```



```
0      False
1      False
2      False
3      False
4      False
       ...
94     False
95     False
96     False
97     False
98     False
Name: BsmtExposure, Length: 99, dtype: bool
```

```
df[df['BsmtExposure'].isnull()]
```

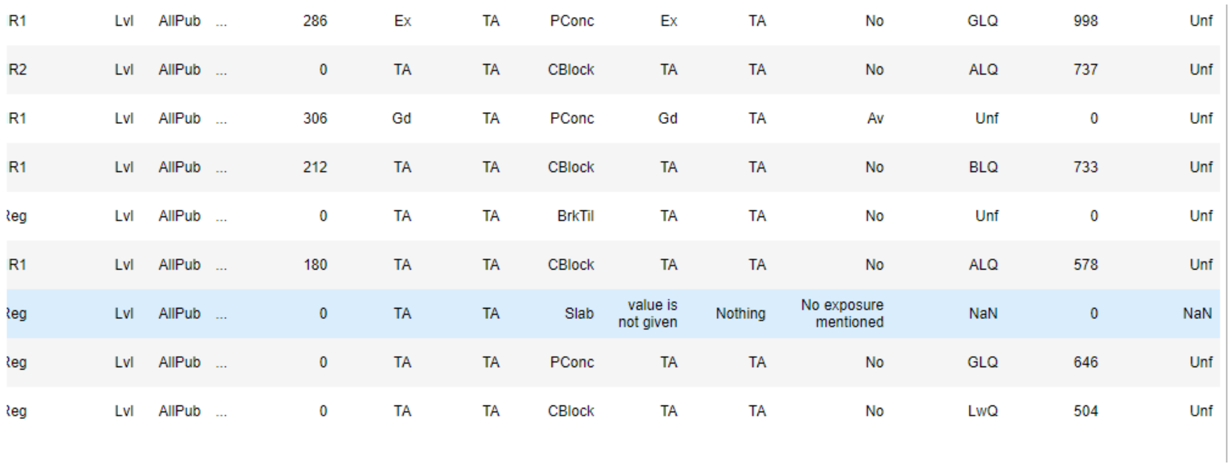


	Id	MSSubClass	MSZoning	LotFrontage	LotArea	Street	Alley	LotShape	LandContour	Utilities	...	MasVnrArea	ExterQual	ExterCond	Foundation
17	18	90	RL	72.0	10791	Pave	no alley mentioned	Reg	Lvl	AllPub	...	0	TA	TA	Slab
39	40	90	RL	65.0	6040	Pave	no alley mentioned	Reg	Lvl	AllPub	...	0	TA	TA	PConc
90	91	20	RL	60.0	7200	Pave	no alley mentioned	Reg	Lvl	AllPub	...	0	TA	TA	Slab

3 rows × 36 columns

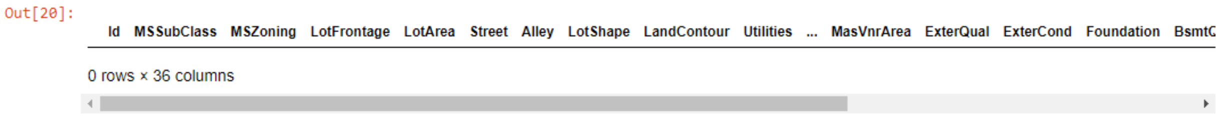
```
df['BsmtExposure'].fillna('No exposure mentioned', inplace=True)
```

```
df.head(20)
```



R1	Lvl	AllPub	...	286	Ex	TA	PConc	Ex	TA	No	GLQ	998	Unf
R2	Lvl	AllPub	...	0	TA	TA	CBlock	TA	TA	No	ALQ	737	Unf
R1	Lvl	AllPub	...	306	Gd	TA	PConc	Gd	TA	Av	Unf	0	Unf
R1	Lvl	AllPub	...	212	TA	TA	CBlock	TA	TA	No	BLQ	733	Unf
Reg	Lvl	AllPub	...	0	TA	TA	BrkTil	TA	TA	No	Unf	0	Unf
R1	Lvl	AllPub	...	180	TA	TA	CBlock	TA	TA	No	ALQ	578	Unf
Reg	Lvl	AllPub	...	0	TA	TA	Slab	value is not given	Nothing	No exposure mentioned	NaN	0	NaN
Reg	Lvl	AllPub	...	0	TA	TA	PConc	TA	TA	No	GLQ	646	Unf
Reg	Lvl	AllPub	...	0	TA	TA	CBlock	TA	TA	No	LwQ	504	Unf

```
df[df['BsmtExposure'].isnull()]
```

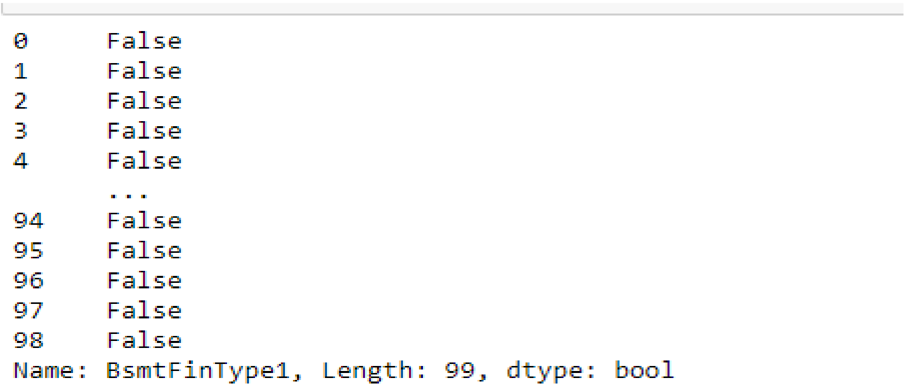


Out[20]:

Id	MSSubClass	MSZoning	LotFrontage	LotArea	Street	Alley	LotShape	LandContour	Utilities	...	MasVnrArea	ExterQual	ExterCond	Foundation	BsmtC

0 rows × 36 columns

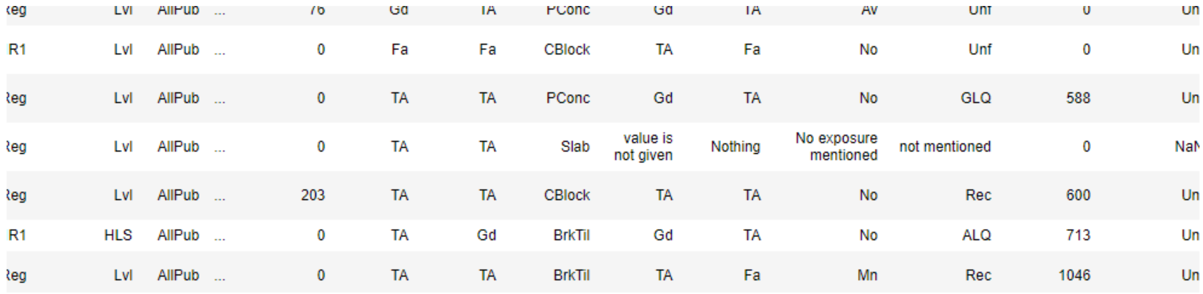
```
print(df['BsmtFinType1'].isnull())
```



```
0      False
1      False
2      False
3      False
4      False
       ...
94     False
95     False
96     False
97     False
98     False
Name: BsmtFinType1, Length: 99, dtype: bool
```

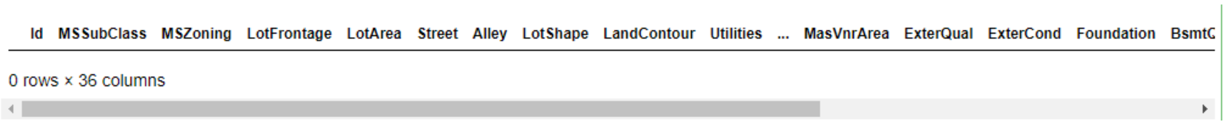
df['BsmtFinType1'].fillna(' not mentioned', inplace=True)

df.tail(20)



Reg	Lvl	AllPub	...	76	Gd	TA	PConc	Gd	TA	Av	Unf	0	Un
R1	Lvl	AllPub	...	0	Fa	Fa	CBlock	TA	Fa	No	Unf	0	Un
Reg	Lvl	AllPub	...	0	TA	TA	PConc	Gd	TA	No	GLQ	588	Un
Reg	Lvl	AllPub	...	0	TA	TA	Slab	value is not given	Nothing	No exposure mentioned	not mentioned	0	NaN
Reg	Lvl	AllPub	...	203	TA	TA	CBlock	TA	TA	No	Rec	600	Un
R1	HLS	AllPub	...	0	TA	Gd	BrkTil	Gd	TA	No	ALQ	713	Un
Reg	Lvl	AllPub	...	0	TA	TA	BrkTil	TA	Fa	Mn	Rec	1046	Un

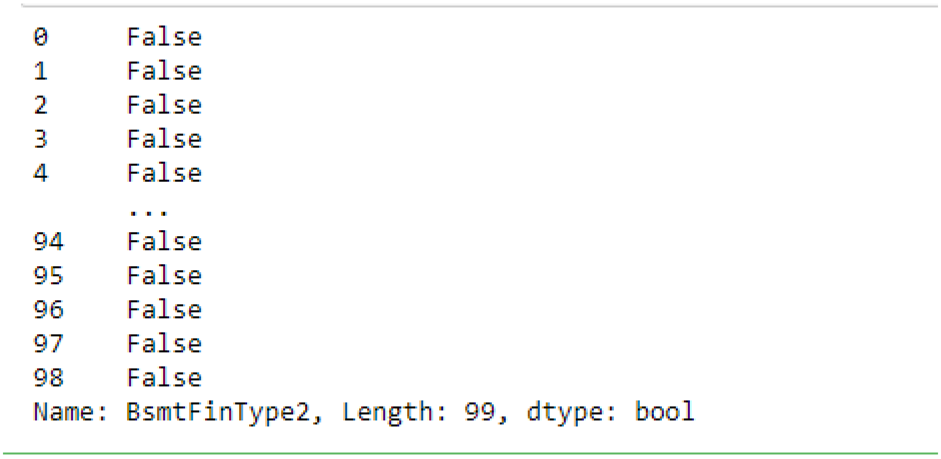
df[df['BsmtFinType1'].isnull()]



Id	MSSubClass	MSZoning	LotFrontage	LotArea	Street	Alley	LotShape	LandContour	Utilities	...	MasVnrArea	ExterQual	ExterCond	Foundation	BsmtC

0 rows × 36 columns

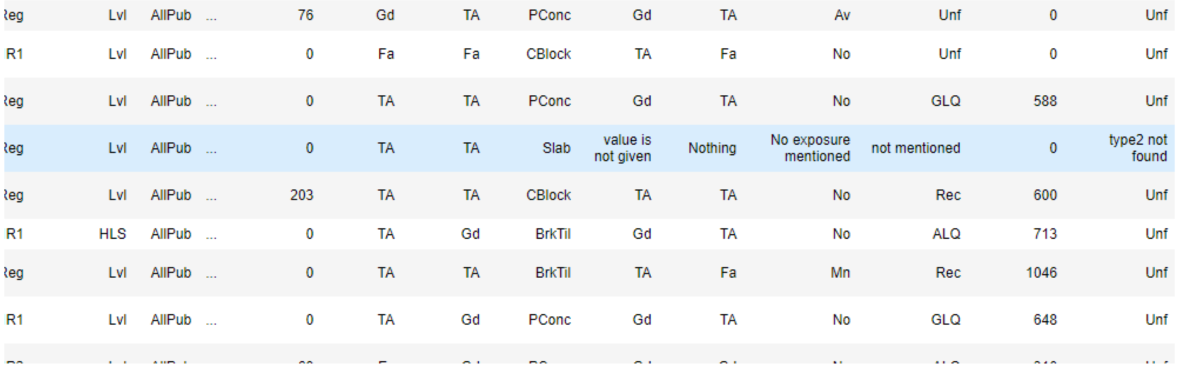
print(df['BsmtFinType2'].isnull())



```
0     False
1     False
2     False
3     False
4     False
      ...
94    False
95    False
96    False
97    False
98    False
Name: BsmtFinType2, Length: 99, dtype: bool
```

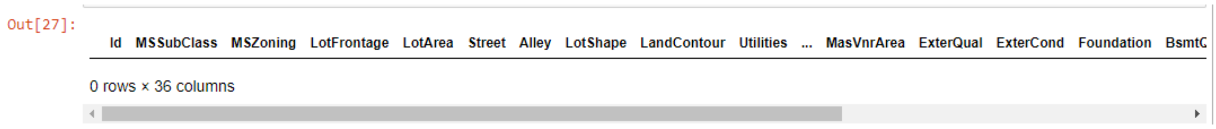
df['BsmtFinType2'].fillna(' type2 not mentioned', inplace=True)

df.tail(20)



Reg	Lvl	AllPub	...	76	Gd	TA	PConc	Gd	TA	Av	Unf	0	Unf
R1	Lvl	AllPub	...	0	Fa	Fa	CBlock	TA	Fa	No	Unf	0	Unf
Reg	Lvl	AllPub	...	0	TA	TA	PConc	Gd	TA	No	GLQ	588	Unf
Reg	Lvl	AllPub	...	0	TA	TA	Slab	value is not given	Nothing	No exposure mentioned	not mentioned	0	type2 not found
Reg	Lvl	AllPub	...	203	TA	TA	CBlock	TA	TA	No	Rec	600	Unf
R1	HLS	AllPub	...	0	TA	Gd	BrkTil	Gd	TA	No	ALQ	713	Unf
Reg	Lvl	AllPub	...	0	TA	TA	BrkTil	TA	Fa	Mn	Rec	1046	Unf
R1	Lvl	AllPub	...	0	TA	Gd	PConc	Gd	TA	No	GLQ	648	Unf

```
df[df['BsmtFinType2'].isnull()]
```

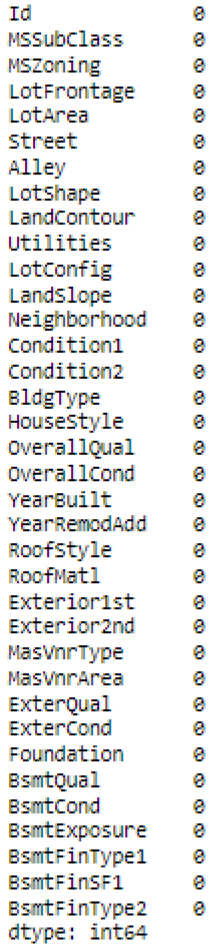


Out[27]:

	Id	MSSubClass	MSZoning	LotFrontage	LotArea	Street	Alley	LotShape	LandContour	Utilities	...	MasVnrArea	ExterQual	ExterCond	Foundation	BsmtC

0 rows × 36 columns

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



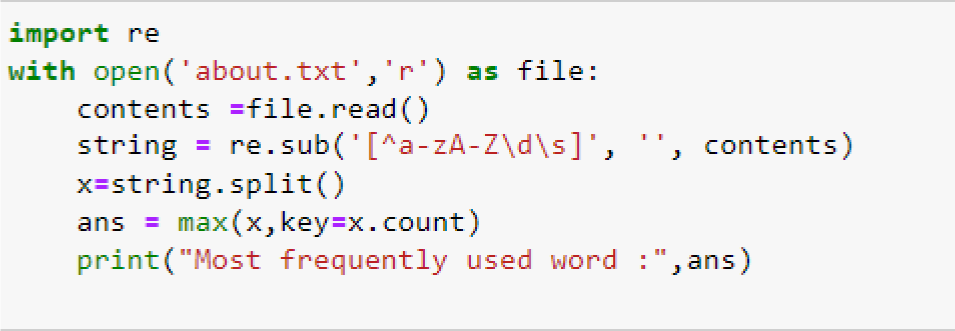
```
Id              0
MSSubClass      0
MSZoning        0
LotFrontage     0
LotArea         0
Street          0
Alley           0
LotShape        0
LandContour     0
Utilities       0
LotConfig       0
LandSlope       0
Neighborhood    0
Condition1      0
Condition2      0
BldgType        0
HouseStyle      0
OverallQual     0
OverallCond     0
YearBuilt       0
YearRemodAdd    0
RoofStyle       0
RoofMatl        0
Exterior1st     0
Exterior2nd     0
MasVnrType      0
MasVnrArea      0
ExterQual       0
ExterCond       0
Foundation      0
BsmtQual        0
BsmtCond        0
BsmtExposure    0
BsmtFinType1    0
BsmtFinSF1      0
BsmtFinType2    0
dtype: int64
```

Question -3

Read the file ‘about.txt’ and find the words with atleast 6 letters and the most frequently used word.

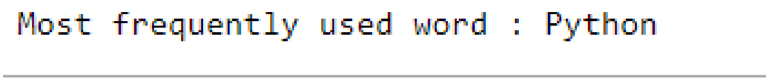
Contents of the file ‘about.txt’:

Python has tools for almost every aspect of scientific computing. The Bank of America uses Python to crunch its financial data and Facebook looks upon the Python library Pandas for its data analysis. While there are many libraries available to perform data analysis in Python, here are a few: NumPy, SciPy, Pandas and Matplotlib.



```python
import re
with open('about.txt','r') as file:
    contents =file.read()
    string = re.sub('[^a-zA-Z\d\s]', '', contents)
    x=string.split()
    ans = max(x,key=x.count)
    print("Most frequently used word :",ans)
```

Output



```
Most frequently used word : Python
```