

Kotta praful ranganath

Ch.en.u4cce21027

DISCORD HANDLE: Ranganath kotta#0230

TASK – 3 [PYTHON - MEDICORE LVL]

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
```

OUTPUT

```
In [5]: import re, csv

file = open('onelinefile.txt', "w")
file.write("1Aaa3.5Maths2Bbb4.2Physics3Ccc7.62Chemistry4Ddd9.55Biology5Eee4.0Social6Fff7.6English7Ggg3.111Maths8Hhh9.99Physics9Iii1.23Civics")
file = open("onelinefile.txt")

for i in file:
    n = re.findall(r'[+-]?[0-9]+\.[0-9]+', i)
    a = re.findall(r'[a-zA-Z]+', i)
    j = 0
    for p in range(len(n)):
        with open('Filename2.csv', 'a', newline='') as file:
            writer = csv.writer(file)
            writer.writerow([str(p+1), a[j], n[p], a[j+1]])
            j += 2

with open('Filename2.csv', 'r',) as file:
    reader = csv.reader(file)
    for row in reader:
        print(','.join(row))
```



```
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

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 solutions 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. **OUTPUT**

```
In [4]: import pandas as pd
import numpy as np
```

```
In [2]: df = pd.read_csv("https://raw.githubusercontent.com/cognizance-amrita/AI-Tasks/main/Task-1/Q2-Dataset.csv")
```

```
In [3]: df.head()
```

```
Out[3]:
```

	Id	MSSubClass	MSZoning	LotFrontage	LotArea	Street	Alley	LotShape	LandContour	Utilities	...	MasVnrArea	ExterQual	ExterCond	Foundation	Bsmnt
0	1	60	RL	65.0	8450	Pave	NaN	Reg	Lvl	AllPub	...	196	Gd	TA	PConc	
1	2	20	RL	80.0	9600	Pave	NaN	Reg	Lvl	AllPub	...	0	TA	TA	CBlock	
2	3	60	RL	68.0	11250	Pave	NaN	IR1	Lvl	AllPub	...	162	Gd	TA	PConc	
3	4	70	RL	60.0	9550	Pave	NaN	IR1	Lvl	AllPub	...	0	TA	TA	BrkTil	
4	5	60	RL	84.0	14260	Pave	NaN	IR1	Lvl	AllPub	...	350	Gd	TA	PConc	

5 rows × 36 columns

```
In [5]: 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(10))
```

```
0    NaN
1    NaN
2    NaN
3    NaN
4    NaN
5    NaN
6    NaN
7    NaN
8    NaN
9    NaN
```

Name: Alley, dtype: object

```
In [7]: 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

```
In [7]: 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

```
In [11]: 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
```

```
In [12]: df['Alley'].fillna('no alley name mentioned', inplace=True)
print(df['Alley'])
```

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

```
In [20]: 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
```

```
In [32]: df[df['BsmtQual'].isnull()]
```

```
Out[32]:
```

id	LandContour	Utilities	...	MasVnrArea	ExterQual	ExterCond	Foundation	BsmtQual	BsmtCond	BsmtExposure	BsmtFinType1	BsmtFinSF1	BsmtFinType2
leg	Lvl	AllPub	...	0	TA	TA	Slab	NaN	NaN	NaN	NaN	0	NaN
leg	Lvl	AllPub	...	0	TA	TA	PConc	NaN	NaN	NaN	NaN	0	NaN
leg	Lvl	AllPub	...	0	TA	TA	Slab	NaN	NaN	NaN	NaN	0	NaN

```
In [35]: df['BsmtQual'].fillna('no value given here', inplace=True)
```

```
In [36]: df[df['BsmtQual'].isnull()]
```

```
Out[36]:
```

id	MSSubClass	MSZoning	LotFrontage	LotArea	Street	Alley	LotShape	LandContour	Utilities	...	MasVnrArea	ExterQual	ExterCond	Foundation	BsmtC
0 rows × 16 columns															

```
In [49]: 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
```

```
In [50]: df[df['BsmtFinType2'].isnull()]
```

Out[50]:

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

3 rows × 36 columns

```
In [51]: df['BsmtFinType2'].fillna('values not found', inplace=True)
```

```
In [52]: df[df['BsmtFinType2'].isnull()]
```

Out[52]:

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

0 rows × 36 columns

```
In [53]: 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."

OUTPUT

```
In [8]: count = 0;
word = "";
maxCount = 0;
words = []

file = open("about.txt", "w")
file.write("Python has tools for almost every aspect of scientific computing. The Bank of America uses Python to crunch its finan
file.close()

file = open("about.txt", "r")

for line in file:
    string = line.lower().replace(' ', '').replace('.', '').split(" ");

    for s in string:
        if len(s) == 6:
            words.append(s);

for i in range(0, len(words)):
    count = 1;

    for j in range(i+1, len(words)):
        if(words[i] == words[j]):
            count = count + 1;

    if(count > maxCount):
        maxCount = count;
        word = words[i];

print("Most repeated word: " + word);
file.close();
```

Most repeated word: python