Python List

2

Multi

Split into list

File processing – splitting out fields

Split into a list

Form:

(list) = line.split('delimiter')

Example:

Data: 20,40

(numb) = line.split(',')

numb[0] = 20 and numb[1] = 40

Create test file

Open notepad

Type in:

25,50,75

10,20,30

33,44,55

Save as testfile5.txt in c:\temp folder

Try
This
PT 1

```
def main():
    filename ='c:/temp/testfile5.txt'
    rcdcnt = 0
    datafile = open(filename,'r')
    line=datafile.readline()
    print('here are the sales amount: ')
```

```
while line != '':
    print(' line before split: ' + line)
    v1 =line.split(',')
    print(' A is: ' + v1[0])
    print(' B is: ' + v1[1])
    print(' C is: ' + v1[2])
    print(' B  v1[1] = ',v1[1], '\n')
    line=datafile.readline()
    rcdcnt = rcdcnt+1
```

Try This PT 2

Try This PT 3

Note: Indent – 4 spaces

```
    datafile.close()
    print('\n'+ 'Number of records read: ' + str(rcdcnt))
    print (' ---done')
main()
```

Whole program

```
def main():
    filename ='c:/temp/testfile5.txt'
    rcdcnt = 0
    datafile = open(filename,'r')
    line=datafile.readline()
    print('here are the sales amount: ')
    while line != '':
        print(' line before split: ' + line)
        v1 =line.split(',')
        numbel = len(v1)
        print('number of elements: ',numbel)
        print('v1: ',v1)
        print(' A is: ' + v1[0])
        print(' B is: ' + v1[1])
        print(' C is: ' + v1[2])
        print(' B  v1[1] = ',v1[1], '\n')
        line=datafile.readline()
        rcdcnt = rcdcnt+1
    datafile.close()
    print('\n'+ 'Number of records read: ' + str(rcdcnt))
    print (' ---done')
main()
```

Add validation for the split

Count the delimiters

Example:

```
comcount = line.count(‘,’)
If (comcount == 2):
```

Check list

Number of elements

Example:

```
lstcnt = len(numb)
print('number of fields: ',lstcnt)
```

Type in:

10,20,30
99,88,77
55,33,11
66,78
101,201,301

Create file

Save file as: testfile7.txt
In folder c:/temp

Try This
PT 1

```
def main():
    filename =‘c:\temp\n1.txt'
    rcdcnt = 0
    datafile = open(filename,'r')
    line=datafile.readline()
```

list11

Watch ident

4

```
while (line != ''):
    comcnt = line.count(',')
    print('Record: ', line)
    if (comcnt == 2):
        v1 =line.split(',')
        print(' A is: ' + v1[0])
        print(' B is: ' + v1[1])
        print(' C is: ' + v1[2])
    else:
        print('record missing fields\n')
    line=datafile.readline()
    rcdcnt = rcdcnt+1
```

Try This PT 2

Watch ident

Try This
PT 3

4

```
    datafile.close()
    print('\n'+ 'Number read: ‘ +   str(rcdcnt))
    print (' ---done')
main()
```

Whole program

```
def main():
    filename ='c:\temp\n1.txt'
    rcdcnt = 0
    datafile = open(filename,'r')
    line=datafile.readline()
    while (line != ''):
        comcnt = line.count(',')
        print('Record: ', line)
        if (comcnt == 2):
            v1 =line.split(',')
            print(' A is: ' + v1[0])
            print(' B is: ' + v1[1])
            print(' C is: ' + v1[2])
        else:
            print('record missing fields\n')
        line=datafile.readline()
        rcdcnt = rcdcnt+1
    datafile.close()
    print('\n'+ 'Number of records read: ' + str(rcdcnt))
    print (' ---done')
main()
```

Two-Dimensional list – Method 1

```
import random
rows=3
cols=4
def main():
    import random
    values = [[0,0,0,0],
              [0,0,0,0],
              [0,0,0,0]]
    for r in range(rows):
        for c in range(cols):
            values[r][c] =random.randint(1,100)
    print(values)
main()
```

Two-Dimensional list – Method 2

Install numpy

from

https://numpy.org/

Hint: pip install numpy

numpy

NumPy is the fundamental package for scientific computing with Python. It contains among other things:

- extension package to Python for multi-dimensional arrays
- closer to hardware (efficiency)
- designed for scientific computation (convenience)
- Also known as array oriented computing

https://www.geeksforgeeks.org/python-numpy/

https://scipy-lectures.org/intro/numpy/array_object.html

```
import numpy as np


# linear array
arr = np.array([1, 2, 3])
print("Linear Array: \n",arr)


# 2 x 2 array
arr = np.array([[1, 2, 3],
                [4, 5, 6]])
print("2 x 2 Array: \n", arr)
```

Numpy example try this part 1 of 2

Numpy example try this part 2 of 2

```
# 3 x 3 array
arr3 = np.array([[2,4,6],
                 [22,44,66],
                 [222,444,666]])
print('3 x 3 Array: \n', arr3)
```

Array characteristics

```
# basic array characteristics
import numpy as np


arr = np.array( [[ 1, 2, 3],
                 [ 4, 2, 5],
                 [6, 7, 8]] )
# ---  type of arr object
print("Array is of type: ", type(arr))
# --- array dimensions (axes)
print("No. of dimensions: ", arr.ndim)
```

Array characteristics

```
# ---   shape of array
print("Shape of array: ", arr.shape)
# --- size (total number of elements) of array
print("Size of array: ", arr.size)
# --- type of elements in array
print("Array stores elements of type: ", arr.dtype)
```

More

```
# ---- Creating a 3X4 array with all zeros
c = arr.zeros((3, 4))
print ("\nAn array initialized with all zeros:\n", c)

# ---- Creating array from list with type float
a = np.array([[1, 2, 4], [5, 8, 7]], dtype = 'float')
print ("Array created using passed list:\n", a
```

Dimensions example

```
import numpy as np


array1d = np.array([1, 2, 3, 4, 5, 6])
array2d = np.array([[1, 1, 1], [2, 2, 2]])
array3d = np.array([[[1, 1, 1], [2, 2, 2]], [[3, 3, 3], [4, 4, 4]]])
print(' 1 d array\n',array1d)
print('----------')
print(' 2 d array \n',array2d)
print('------')
print(' 3 d array \n',array3d)
```

Loading a 3 d array

```
array3d = np.array([[[1, 1, 1], [2, 2, 2]], [[3, 3, 3], [4, 4, 4]]])


for x in range(3):
    for y in range(3):
        for z in range(3):
            np.array3d(x)(y)(z)
```

TRY
THIS

```
import numpy as arr
def main():
    z = arr.zeros([3, 3, 3])
    print("\nMatrix z : \n", z)
    h = 1
    for a in range(3):
        for b in range(3):
            for c in range(3):
                z[a][b][c] =  h
                h = h + 1
     print(' ---------------- ‘)
     print("\nMatrix z : \n", z)
main()
```

done