lcp-2

Mani Sai Srinivas Kandukuri

Class id-20

1. First input was taken and then another list was taken counting the length of each element in the list.

   Now zip operator was used to combine 2 lists and then they are sorted acc. To requirements

   Input:

```
: def tuple_sort():
      inp1=["PHP","Exercises","Backend"]
      '''counting the length of each input'''
      len_inp1=[len(inp1[0]),len(inp1[1]),len(inp1[2])]
      '''converting into tuple'''
      new_inp1=list(zip(len_inp1,inp1))
      '''sorting using first key'''
      new_inp1.sort(key=lambda new_inp1:new_inp1[0])
      return new_inp1[2]
  print(tuple_sort())
```

   Output:

```
(9, 'Exercises')
```

2. First file was opened using python open function.
   Now readlines() was used to read and return a list containing the lines
   So we separated the list using strip() and found the length of each one

   Input:

practise_code.py | input_file

```
Saria
Deep learning
Python
```

pr2.py | input_file

```
filename="input_file"
with open(filename,"r") as f:
    for line in f.readlines():
        words = len(line.strip())
        line=line.strip()
        print(line,words)
```

Output:

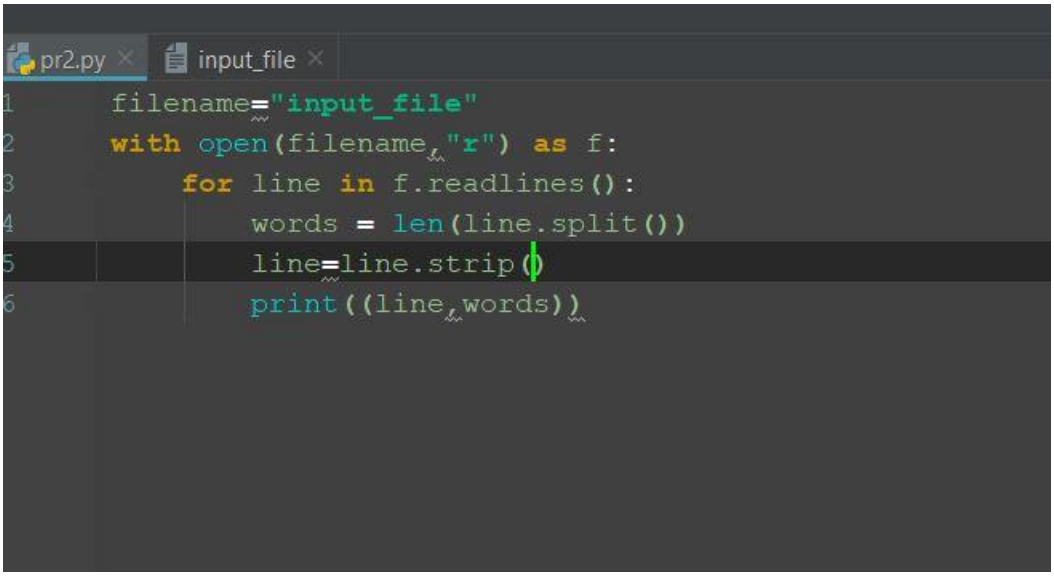
pr2

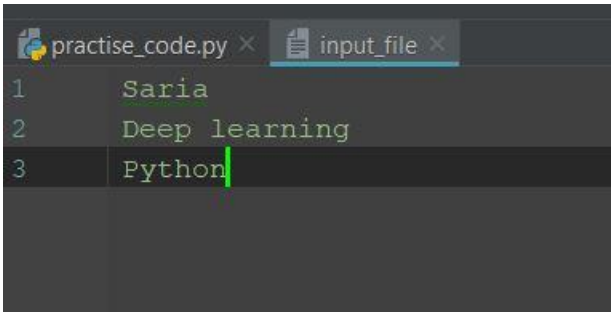
```
C:\Users\srini\AppData\Local\Programs\Python\Pyth
Saria 5
Deep learning 13
Python 6

Process finished with exit code 0
```

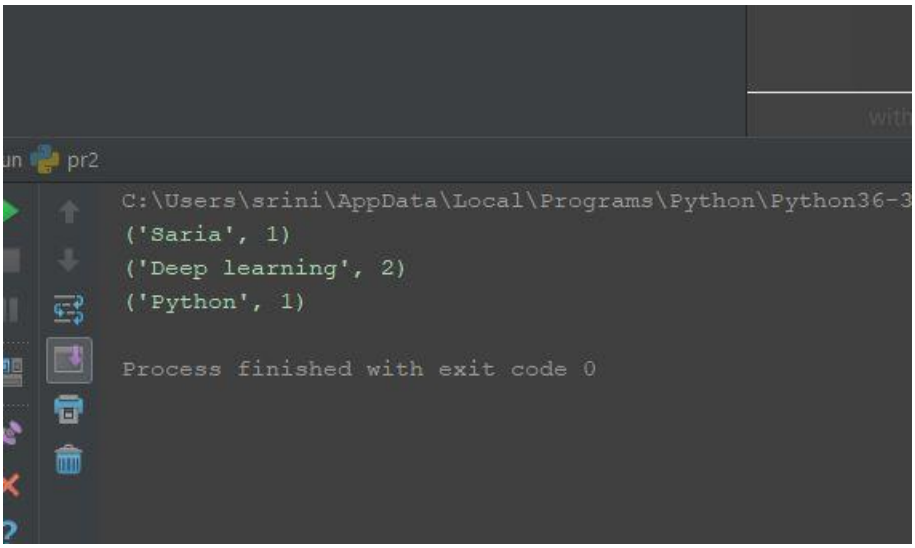
3. Everything is same as the previous one except we use split() instead of strip()
   First file was opened using python open function.
   Now readlines() was used to read and return a list containing the lines
   So we separated the list using split() and found the length of each one

   Input:





Output:



4. First

   Input:

   The output here is displayed according to the user input.

   Print statement was used here.

   Also here the user input should be separated by '*' Like 2*4

```
def get_dimensions():
    request = input("Please choose the dimensions of the game board. ")
    while True:
        try:
            request = request.strip().split("*")
            if len(request) != 2:
                int("error")
            dimensions = (int(request[0]), int(request[1]))
            if dimensions[0] >= 0 and dimensions[1] >= 0:
                return dimensions
            else:
                int("error")
        except:
            request = input("Please choose two natural numbers seperated by an '*'. For example '3*3'. ")

def draw (rows, columns):
    top = columns *     " ----"
    strich = columns *  "|    " + "|"
    for i in range(0, rows):
        print(top)
        print(strich)
    print(top)

dim = get_dimensions()
x = dim[0]
y = dim[1]
draw(x, y)
```

Output:

```
Please choose the dimensions of the game board. 2*2
 ---- ----
|    |    |
 ---- ----
|    |    |
 ---- ----
```