

<PKU 통계스터디 4번째 >

Read file & Histogram

2018/03/24

이준호

Read txt file with “readline()”

Format1 : on same location

Say there is a file name “n1.txt”

```
f = open('n1.txt', 'r')
Firstline = f.readline()
Secondline = f.readline()
print(Firstline)
print(Secondline)
f.close()
```

=====

“c1.py” on
“TESTs/project_180324/lecture_readfile”

Format1 : on same location (no change line)

```
f = open('n1.txt', 'r')
Firstline = f.readline()
Secondline = f.readline()
print(Firstline, end='')
print(Secondline)
f.close()
```

=====

“c2.py” on
“TESTs/project_180324/lecture_readfile”

Format2 : on different location

On
“TESTs/project_180324/lecture_readfile/diff_loca”.

```
f = open("../n1.txt",'r')
Firstline = f.readline()
Secondline = f.readline()
```

```
print(Firstline,end="")
print(Secondline,end="")
f.close()
```

=====

“diff_c1.py” on
“TESTs/project_180324/lecture_readfile/diff_loca”

More useful to read in file : for line in f

General

```
f = open("n1.txt","r")
```

```
for line in f:  
    print(line, end = "")
```

```
f.close()
```

=====

```
c3.py
```

From different directory

```
f = open("../n1.txt","r")
```

```
for line in f:  
    print(line, end = "")
```

```
f.close()
```

=====

```
diff_loca/diff_c3.py
```

Interfacing “list”

```
f = open("../n1.txt","r")
```

```
LL = []
```

```
for line in f:  
    print(line, end = "")  
    LL.append(line)
```

```
f.close()
```

=====

```
diff_loca/diff_c5.py
```

The “split()” function

General format

```
f = open("n1.txt",'r')

for line in f:
    num1, num2 = line.split()
    print(num1, end=' ')
    print(num2)

print(type(num1))

f.close()
=====
c4.py
```

List infercafed

```
f = open("n1.txt",'r')
LL1 = []
LL2 = []
for line in f:
    num1, num2 = line.split()
    print(num1, end=' ')
    print(num2)
    LL1.append(num1)
    LL2.append(num2)
print(LL1);print(LL2)
f.close()
=====
c5.py
```

“root0_project.txt”

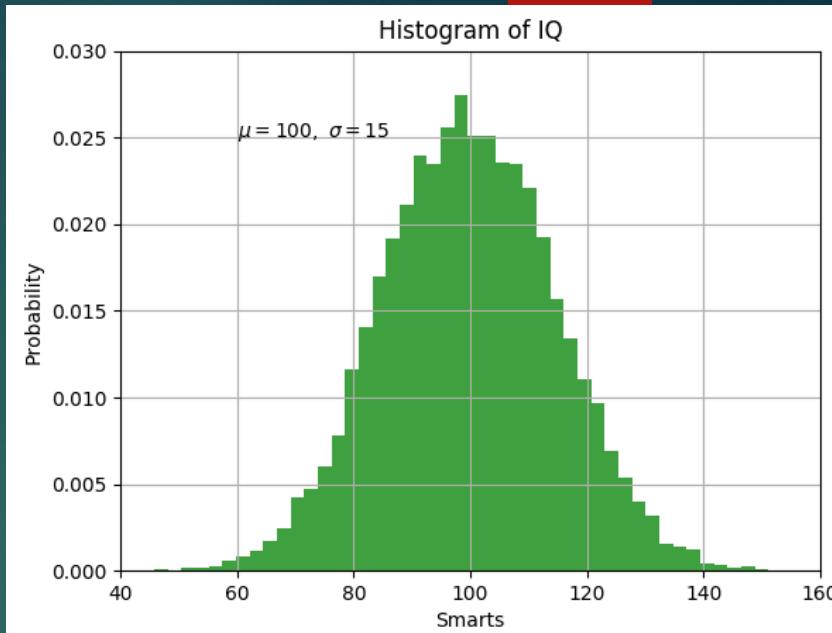
```
f = open("root0_project.txt",'r')
LL = []
for line in f:
    _, _, _, num = line.split()
    LL.append(num)
print(LL)
f.close()
=====
c6.py
```

Plotting in shape of histogram using input data file

- ▶ ## please refer
“TESTs/project_180324/lecture_readfile/histogram/c1.py”

Project 1. (totally 3 plots)

- ▶ There is a file named “root1_project.txt” on
TESTs/project_180324/lecture_readfile/root0_project.txt,
- ▶ Draw histogram, with Xaxis label, Yaxis label. (pro1-1.pdf)
- ▶ **Calculate total entry number**
- ▶ **Normalize the histogram and plot.** (pro1-2.pdf)
- ▶ **Calculate mean and standard deviation, and variation of this specific histogram**
- ▶ **Print the Mean, standard deviation and total Entry number on the plot.**
(pro1-3.pdf)



Project 2. (totally 4 plots)

- ▶ There is a file named “root2_project.txt” on
TESTs/project_180324/lecture_readfile/root1_project.txt
- ▶ Draw histogram, with Xaxis label, Yaxis label. (pro2-1.pdf)
- ▶ **Calculate total entry number**
- ▶ **Normalize the histogram and Plot (pro2-2.pdf)**
- ▶ **Calculate mean and standard deviation, and variation of this specific histogram**
- ▶ **Print the Mean, standard deviation and total Entry number on the plot. (pro2-3.pdf)**

- ▶ Put the Normalized “root1_project.txt” and “root2_project.txt” on a same canvas (pro2-4.pdf)

Additional project :: printing prime numbers

- ▶ Make a function which prints prime-number between 0 to given number :

ex)

```
def primenumber(NUM):
    ...
    ...
    return List_of_prime_number
```

The function should return list of prime numbers, between 0 to NUM.