

File Handle as a Sequence

A file handle open for read can be treated as a sequence of strings where each line in the file is a string in the sequence.

We can use a for statement to iterate through a sequence.

- ```
xfile = open('mbox.txt')
for cheese in xfile:
 print(cheese)
```

Remember - a sequence is an ordered set.

## Counting lines in a File

Open a file read-only

- `fhand = open('mbox.txt')`

Use a for loop to read each line.

- `fhand = open('mbox.txt')`  
`count = 0`  
`for line in fhand:`  
`count = count + 1`

Count the lines and print out the number of lines.

- `fhand = open('mbox.txt')`  
`count = 0`  
`for line in fhand:`  
`count = count + 1`  
`print('Line Count:', count)`

## Reading the \*Whole\* File

We can read the whole file (newlines and all) into a single string

- `fhand = open('mbox-short.txt')`  
`inp = fhand.read()`
  - `print(len(inp))`  
94626
    - `print(inp[:20])`  
From stephen marquar

## Searching through a File

We can put an if statement in our for statement to only print lines that meet some criteria.

- ```
fhand = open('mbox-short.txt')
for line in fhand:
    if line.startswith('From:'):
        print(line)
```

OOPS!

What are all these blank lines doing here?

- ```
From: stephen.marquard@utc.ac.za\n
\n
From: louis@media.berkeley.edu\n
\n
From: zqian@umich.edu\n
\n
From: rjlowe@iupui.edu\n
\n
```

Each line from the file actually has a newline at the end.

The print statement adds a newline to each line.

## Searching through the File (fixed)

We can strip the whitespace from the right-hand side of the string using `rstrip()` from the string library.

- ```
fhand = open('mbox-short.txt')
for line in fhand:
    line = line.rstrip()
    if line.startswith('From:'):
        print(line)
```

The newline is considered “white space” and is stripped.

- ```
From: stephen.marquard@utc.ac.za
From: louis@media.berkeley.edu
From: zqian@umich.edu
From: rjlowe@iupui.edu
```

## Skipping with continue

We can conveniently skip a line using the continue statement.

- ```
fhand = open('mbox-short.txt')
for line in fhand:
    line = line.rstrip()
    if not line.startswith('From:'):
        continue
    print(line)
```

Using in to select lines

We can look for a string anywhere in a line as our selection criteria

- ```
fhand = open('mbox-short.txt')
for line in fhand:
 line = line.rstrip()
 if not line.startswith('From:'):
 continue
 print(line)
```

### **Prompt for File Name**

```
fname = input('Enter file name: ")
fhand = open(fname)
count = 0
For line in fhand:
 If line.startswith('Subject:'):
 count = count + 1
print('There were', count, 'subject lines in', fname)
```



## Bad File Names

```
fname = input('Enter file name: ")
try:
 fhand = open(fname)
except:
 print('File name cannot be opened:', fname)
 quit()

count = 0
for line in fhand:
 If line.startswith('Subject:'):
 count = count + 1
print('There were', count, 'subject lines in', fname)
```

Good:

Enter file name: mbox.txt

There were 1797 subject lines in mbox.txt

Bad:

Enter file name: na na boo boo

File name cannot be opened: na na boo boo

The quit() function is a way to kil the program silently without a Traceback.