



# SKILLPILLS

## Skill Pill: Introduction to Programming

### Lecture 4: File I/O

Ankur Dhar

*ankur.dhar@oist.jp*

March 27th, 2019



- 1 Introduction to Files
- 2 Opening and Reading Files
- 3 Writing Files and Formatting

Files are sequences of data stored on a disk. These are commonly stored in the form of text, binary, or hexadecimal characters.

Text files can be opened with text editors to view their contents:

- Windows: Notepad++
- Mac: Atom
- Linux: VIM (or Emacs)

# Why are they important?

Files are crucial to storing data needed as input, or storing output.

Historically files were crucial for working over large data sets, when computer memory was limited and could not load everything at once.

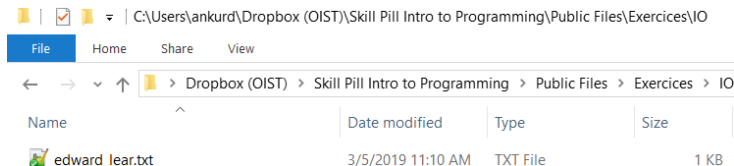


**Figure:** Jeanette A. Scissum working with sunspot data at NASA

There are two ways to open a file, depending on where the file is:

## Open relative to working directory

```
file = open('edward_lear.txt')
```



## Open an absolute path

```
file = open('C:\Users\ankurd\Dropbox (OIST)\Misc\Programming Skill Pill\edward_lear.txt')
```

## Read a file all at once

```
file = open('edward_lear.txt')  
s = file.read()  
print(s)  
file.close()
```

Edward Lear

The Owl and the Pussy-cat went to sea  
In a beautiful pea-green boat  
They took some honey, and plenty of money  
Wrapped up in a five-pound note  
The Owl looked up to the stars above  
And sang to a small guitar  
"O lovely Pussy! O Pussy, my love  
What a beautiful Pussy you are  
You are  
You are!  
What a beautiful Pussy you are!"

Reading advances forward through the file like a roll of tape, so using `read(n)` will read the next *n* characters.

## Reading *n* characters

```
file = open('edward_lear.txt')
print(file.read(14))
print(file.read(10))
file.seek(5)
print(file.read(10))
file.close()
```

Edward Lear

Th

e Owl and

d Lear

The

However you can use *readline()* to read until the end of line, denoted by the newline character `\n`.

## Read a single line

```
file = open('edward_lear.txt')  
s = file.readline()  
print(s)
```

Edward Lear



Use for loop to iterate through the file

```
file = open('edward_lear.txt')  
for s in file:  
    print('line :' + s)
```

line :Edward Lear

line :The Owl and the Pussy-cat went to sea

line :In a beautiful pea-green boat

line :They took some honey, and plenty of money

line :Wrapped up in a five-pound note

## Read a whole file into a list

```
file = open('edward_lear.txt')
all_lines = file.readlines()
print(all_lines)
```

## Read a whole file into a list and strip \n

```
file = open('edward_lear.txt')
all_lines = []
for s in file:
    all_lines.append(s.rstrip())
```

## Exercise 1

Open the file 'edward\_lear.txt' and print the first 4 lines.

## Exercise 2

Open the file 'fasta.txt' and print the first 50 nucleotides.

```
GGAGAGTTTGATCCTGGCTCAGGACGAACGCTGGCGGCGTGCCTAACACA
```

## Creating a new file to write text

```
file = open("example.txt", "w")  
file.write("Beans beans\nThe musical fruit\n")  
file.close()
```

## Writing more characters afterwards

```
file = open("example.txt", "a")  
file.write("The more you eat\n")  
file.write("The more you toot\n")  
file.close()
```

Reading a file, transforming it, and write it out

```
with open("edward_lear.txt") as f1:
    with open("edward_lear_2.txt", "w") as f2:
        for line in f1:
            line2 = line.rstrip() + ",\n"
            f2.write(line2)
```

Edward Lear,  
The Owl and the Pussy-cat went to sea,  
In a beautiful pea-green boat,

## Exercise 1

Write a program which creates a file 'squares.txt' consisting of the first 5 square numbers.

## Exercise 2

Write a program which reads 'edward\_lear.txt' and writes out 'edward\_lear\_numbered.txt'.

```
1 Edward Lear
2 The Owl and the Pussy-cat went to sea
3 In a beautiful pea-green boat
4 They took some honey, and plenty of money
5 Wrapped up in a five-pound note
...
```

## Tabular data can be stored in delimited text

Each line represents a row, while columns are marked by delimiting characters, most common of which are commas (CSV), and tabs. Using the `split()` function you can parse each line into a list of data.

	A	B	C	D	E	
1	city	city_ascii	lat	lng	pop	country
2	Qal eh-ye Now	Qal eh-ye	34.98300013	63.13329964	2997	Afghanistan
3	Chaghcharan	Chaghcharan	34.5167011	65.25000063	15000	Afghanistan
4	Lashkar Gah	Lashkar Gah	31.58299802	64.35999955	201546	Afghanistan
5	Zaranj	Zaranj	31.11200108	61.88699752	49851	Afghanistan
6	Tarin Kowt	Tarin Kowt	32.63329815	65.86669865	10000	Afghanistan
7	Zareh Sharan	Zareh Sharan	32.85000016	68.41670453	13737	Afghanistan
8	Asadabad	Asadabad	34.86600004	71.15000459	48400	Afghanistan
9	Talogan	Talogan	36.72999904	69.54000364	64256	Afghanistan
10	Mahmud-E Eraqi	Mahmud-E Eraqi	35.01669608	69.33330065	7407	Afghanistan
11	Mehtar Lam	Mehtar Lam	34.65000001	70.16670052	17345	Afghanistan
12	Baraki Barak	Baraki Barak	33.9667021	68.96670354	22305	Afghanistan
13	Aybak	Aybak	36.26100015	68.04000051	24000	Afghanistan
14	Mayda Shahr	Mayda Shahr	34.45000209	68.79999663	35008	Afghanistan
15	Karokh	Karokh	34.48676963	62.59177608	14388.5	Afghanistan

Figure: Example of CSV file

Print all players older than 30

```
with open('rugby_world_cup_20151.csv') as file:
    file.readline()
    for line in file:
        data = line.strip().split(',')
        age = int(data[4])
        name = data[1]
        if age > 30:
            print('{} ({} )'.format(name, age))
```



## Exercise 1

Write a program which creates a file 'powers.csv' where each row is 1 to 5 and each successive column is that number raised to the second, third, and fourth powers

```
1,1,1,1
2,4,8,16
3,9,27,81
...
```

## Exercise 2

Read the file `simplemaps-worldcities-basic.csv`, and print the names and populations of all cities which have a population larger than 10,000,000.