

# Python for Informatics

1

## LESSON 9

# Computer Housekeeping and Maintenance

2

- In addition to the cool informatics applications that we have explored, Python can also be used to perform computer housekeeping and maintenance operations.
- For example, the automated processing of a large set of selected files can be a valuable capability.

# Computer Housekeeping and Maintenance

3

- The ***os*** library module offers features that help us deal with our local file system.
- More specifically, ***os.getcwd()*** returns the name of the current directory, and ***os.walk()*** enables us to traverse all of the directories and files within a given directory tree.

# Computer Housekeeping and Maintenance

4

OS Function	Function Effect
<b><i>walk()</i></b>	Walks the elements of a directory tree.
<b><i>getcwd()</i></b>	Gets the current directory.
<b><i>listdir(path)</i></b>	Lists the directory of the specified path.
<b><i>path.abspath(s)</i></b>	Returns the absolute file path of the specified filename.
<b><i>path.exists(s)</i></b>	Returns true if the specified file or directory name exists.
<b><i>path.isdir(s)</i></b>	Returns true if the specified name is a directory.
<b><i>path.isfile(s)</i></b>	Returns true if the specified name is a file.

<https://docs.python.org/2/library/os.html#files-and-directories>

# Computer Housekeeping and Maintenance

5

**"""**

***This program counts the number of .txt files within the directory tree, starting at the current directory.***

**"""**

***import os***

***count = 0***

***for dirname, dirs, files in os.walk('.'):***

***for filename in files:***

***if filename.endswith('.txt') :***

***count = count + 1***

***print 'Files:', count***

# Computer Housekeeping and Maintenance

6

**"""**

***This program determines the size of all .txt files within the directory tree, starting at the current directory.***

***Note that we use “join” to perform concatenation because it accounts for the path delimiter differences across different operating systems (“/” vs “\”).***

**"""**

```
import os  
from os.path import join  
for (dirname, dirs, files) in os.walk('.')  
    for filename in files:  
        if filename.endswith('.txt') :  
            thefile = os.path.join(dirname, filename)  
            print os.path.getsize(thefile), thefile
```

# Computer Housekeeping and Maintenance

7

```
"""
```

```
This program looks all files except those of specific  
sizes 2578 and 2565 (these are vendor headers that  
we don't care about) or files that don't have more  
than one line.
```

```
"""
```

```
import os
```

```
from os.path import join
```

```
for (dirname, dirs, files) in os.walk('.'):
    for filename in files:
```

```
        if filename.endswith('.txt'):
```

```
            thefile = os.path.join(dirname, filename)
```

```
            size = os.path.getsize(thefile)
```

```
            if size == 2578 or size == 2565:
```

```
                continue
```

```
            fhand = open(thefile, 'r')
```

```
            lines = list()
```

```
            for line in fhand:
```

```
                lines.append(line)
```

```
            fhand.close()
```

```
            if len(lines) > 1:
```

```
                print len(lines), thefile
```

```
                print lines[:4]
```

# Computer Housekeeping and Maintenance

```
"""
```

```
This program reverses the logic of the  
previous one—we will soon want to  
delete all files we are not interested in.  
(we also deal with 3 line footer files that  
we are not interested in).
```

```
"""
```

```
import os  
from os.path import join  
for (dirname, dirs, files) in os.walk('.'):   
    for filename in files:  
        if filename.endswith('.txt') :  
            thefile = os.path.join(dirname,filename)  
            size = os.path.getsize(thefile)  
            if size == 2578 or size == 2565:  
                print 'T-Mobile:',thefile  
                continue  
            fhand = open(thefile,'r')  
            lines = list()  
            for line in fhand:  
                lines.append(line)  
            fhand.close()  
            if len(lines) == 3 and lines[2].startswith('Sent from my iPhone') :  
                print 'iPhone:', thefile  
                continue
```



# Computer Housekeeping and Maintenance

*"""*

*This program actually deletes the files that  
we've determined we want to delete.*

*"""*

```
import os
from os.path import join
for (dirname, dirs, files) in os.walk('.'):
    for filename in files:
        if filename.endswith('.txt') :
            thefile = os.path.join(dirname,filename)
            size = os.path.getsize(thefile)
            if size == 2578 or size == 2565:
                print 'T-Mobile:',thefile
                os.remove(thefile)
                continue
            fhand = open(thefile,'r')
            lines = list()
            for line in fhand:
                lines.append(line)
            fhand.close()
            if len(lines) == 3 and lines[2].startswith('Sent from my iPhone') :
                print 'iPhone:', thefile
                os.remove(thefile)
                continue
```

# Computer Housekeeping and Maintenance

10

*"""*

*This program how to process  
command line arguments.*

*"""*

*import sys*

*print 'Count:', len(sys.argv)*

*print 'Type:', type(sys.argv)*

*for arg in sys.argv:*

*print 'Argument:', arg*

# Computer Housekeeping and Maintenance

11

**"""**

***This program takes a file name as  
a command line argument, opens  
the specified file, and reads it.***

**"""**

***import sys***

***name = sys.argv[1]***

***handle = open(name, 'r')***

***text = handle.read()***

***print name, 'is', len(text), 'bytes'***

# Computer Housekeeping and Maintenance

12

**"""**

***This program defines a command str,  
opens that str as a pipe (which executes  
the command), and reads all of the lines  
returned by the command. The pipe is then  
closed, and the status is printed.***

**"""**

***import os***

***cmd = 'ls -l'***

***file\_pointer = os.popen(cmd)***

***all\_lines = file\_pointer.read()***

***print all\_lines***

***status = file\_pointer.close()***

***print status***