# Reading and writing files - Automate the boring stuff Chapter 8

## Some basic information

Reading and writing files in Windows/Mac/Linux.

We are going to use our OS module for this chapter. First we are going to import our os.module

*import* os

If you want to make a program that will work and all 3 of these operating systems we should use the os.path.join().

So if we want to navigate to c:/users/ghira/downloads we can use.

desired\_path = os.path.join('c:\\users\\ghira\\downloads\\')

our result is:



## Absolute and Relative paths

An absolute path always begins with the root folder example:

C:\Users\ghira\Desktop\Programming\Python Projects\automate\_the\_boring\_stuff\

And a relative path is relative to the programs current working directory, so if we are already in the automate\_the\_boring\_stuff\_folder we could just type :

print(os.listdir(os.curdir))

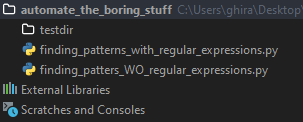
and this will print all the files in our current directory

## Creating New Folders with os.makedirs()

Creating a folder with the os package is really easy we just have to type

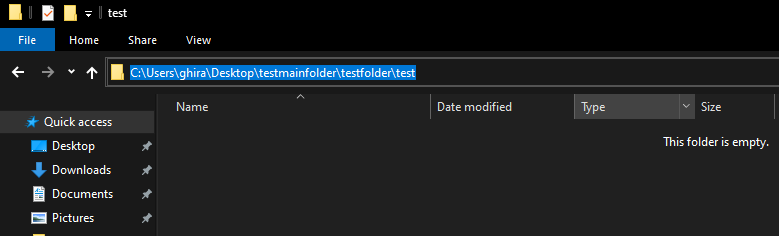
os.makedirs('testdir')

This will make a folder called testdir in our current working directory which is the automate\_the\_boring\_stuff folder.



Or if the folders don’t exist it will create them for us:

os.makedirs('c:\\users\\ghira\\desktop\\testmainfolder\\testfolder\\test\\')



## Os.path Module

The os.path module contains many helpful functions related to filenames and filepaths like os.path.join() which we used to build paths in a way that will work on all operating systems.

### Handling absolute and relative paths

Calling os.path.abspath(path) will return a string of our absolute path

path = 'testdir' 🡪 located in our current working folder

os.path.abspath(path) will result in

'C:\\Users\\ghira\\Desktop\\Programming\\Python Projects\\automate\_the\_boring\_stuff\\testdir'

path = 'testdir'  
os.path.isabs(path)

This will return a Boolean if our path is relative or not

If you want to find out what is the path before the last backslash we can do it in the following way:

path = 'C:\\Windows\\System32\\calc.exe'  
print(os.path.dirname(path))

This will print C:\Windows\System32

calc\_file\_path = 'C:\\Windows\\System32\\calc.exe'  
print(os.path.split(calc\_file\_path))

## Usefull OS Commands

path = 'C:\\Windows\\System32\\calc.exe'

Os.getcwd() 🡪 gets current working directory

Os.chdir(path) 🡪changes the directory to the one we want

Os.listdir(path) 🡪 lists all the directories in that path

Os.makedirs(path) 🡪 makes a directory or more directories

Os.path.abspath(path) 🡪 writes out the abs path for the specified path

Returns:

'C:\\Windows\\System32\\calc.exe'

Os.path.isabs(path) 🡪 tells us if path is an absolute path 🡪 returns True

Os.path.basename(path) 🡪 returns ‘calc.exe’

Os.path.dirname(path) 🡪 returns 'C:\\Windows\\System32'

Os.path.getsize(path) 🡪 tells us the size of the file 🡪 returns 27648 bytes

>>> os.path.exists('C:\\Windows')

True

>>> os.path.exists('C:\\some\_made\_up\_folder')

False

>>> os.path.isdir('C:\\Windows\\System32')

True

>>> os.path.isfile('C:\\Windows\\System32')

False

>>> os.path.isdir('C:\\Windows\\System32\\calc.exe')

False

>>> os.path.isfile('C:\\Windows\\System32\\calc.exe'**)**