

# Working with files in python 13



# File handling

File handling is an important part of any web application.

Python has several functions for creating, reading, updating, and deleting files.

# open() function

The key function for working with files in Python is the `open()` function.

The `open()` function takes two parameters; *filename*, and *mode*.

There are four different methods (modes) for opening a file:

`"r"` - Read - Default value. Opens a file for reading, error if the file does not exist

`"a"` - Append - Opens a file for appending, creates the file if it does not exist

`"w"` - Write - Opens a file for writing, creates the file if it does not exist

`"x"` - Create - Creates the specified file, returns an error if the file exists

`"t"` - Text - Default value. Text mode

`"b"` - Binary - Binary mode (e.g. images)

## syntax

```
f = open("text.txt")
```

the same as

```
f = open("text.txt", "rt")
```

Make sure the file exists, or else you will get an error.

## read() files

To open the file, use the built-in `open()` function.

The `open()` function returns a file object, which has a `read()` method for reading the content of the file:

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

```
print(f.read())
```

```
>>Hello! Welcome to text.txt
```

```
This file is for testing purposes.
```

```
Good Luck!
```

## Read only parts of a text

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

```
print(f.read(5))
```

```
>>Hello
```

## readline()

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

```
print(f.readline())
```

```
print(f.readline())
```

```
>>Hello! Welcome to text.txt
```

```
This file is for testing purposes.
```

# Close files

it is a good practice to always close the file when you are done with it.

Close the file when you are finish with .You should always close your files, in some cases, due to buffering, changes made to a file may not show until you close the file.

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

```
print(f.readline())
```

```
f.close()
```



# Write a file in python

To write to an existing file, you must add a parameter to the `open()` function:

"a" - Append - will append to the end of the file

"w" - Write - will overwrite any existing content

## create a new file

To create a new file in Python, use the `open()` method, with one of the following parameters:

"x" - Create - will create a file, returns an error if the file exist

"a" - Append - will create a file if the specified file does not exist

"w" - Write - will create a file if the specified file does not exist

```
f = open("text.txt", "a")
```

```
f.write("Now the file has more content!")
```

```
f.close()
```

#open and read the file after the appending:

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

```
print(f.read())
```

```
>>Hello! Welcome to demofile.txt
```

```
This file is for testing purposes.
```

```
Good Luck!Now the file has more content!
```

## with statement in Python

```
with open('text.txt', 'a') as file:  
    file.write('hello world !')
```

```
file.closed
```

```
>>True
```

"r" Opens a file for reading only.

"r+" Opens a file for both reading and writing.

"rb" Opens a file for reading only in binary format.

"rb+" Opens a file for both reading and writing in binary format.

"w" Opens a file for writing only.

"a" Open for writing. The file is created if it does not exist.

"a+" Open for reading and writing. The file is created if it does not exist.

# delete a file

To delete a file, you must import the OS module, and run its `os.remove()` function:

```
import os
```

```
os.remove("demofile.txt")
```

# check if file exists

```
import os
```

```
if os.path.exists("demofile.txt"):
```

```
    os.remove("demofile.txt")
```

```
else:
```

```
    print("The file does not exist")
```

# delete folder

use rmdir() method

```
import os
```

```
os.rmdir("myfolder")
```

You can only remove *empty* folders.



# exercises

1. Write a function in python to read the content from a text file "poem.txt" line by line and display the same on screen.
2. Write a function in Python to count and display the total number of words in a text file.
3. Write a python program to add text to a file and display the text in python.txt.
4. Write a function `display_words()` in python to read lines from a text file "story.txt", and display those words, which are less than 4 characters.
5. Write a python program to read a file, a.txt line by line.