Python File Handling Methods - Complete Guide with Simple Examples

## Python File Handling Methods - Complete Guide with Simple Examples

# \*\*1. read()\*\*

Reads the entire file content as a single string.

## Examples:

file = open("sample.txt", "r")  
print(file.read())  
file.close()

file = open("sample.txt", "r")  
text = file.read()  
print(text)  
file.close()

with open("sample.txt", "r") as file:  
 print(file.read())

# \*\*2. readline()\*\*

Reads one line at a time.

## Examples:

file = open("sample.txt", "r")  
print(file.readline())  
file.close()

file = open("sample.txt", "r")  
line = file.readline()  
print(line)  
file.close()

with open("sample.txt", "r") as file:  
 print(file.readline())

# \*\*3. readlines()\*\*

Reads all lines into a list.

## Examples:

file = open("sample.txt", "r")  
lines = file.readlines()  
print(lines)  
file.close()

with open("sample.txt", "r") as file:  
 lines = file.readlines()  
 print(lines)

with open("sample.txt", "r") as file:  
 print(len(file.readlines()))

# \*\*4. write()\*\*

Writes data to the file. Overwrites if the file exists.

## Examples:

file = open("output.txt", "w")  
file.write("Hello!")  
file.close()

file = open("output.txt", "w")  
file.write("Line one\n")  
file.close()

with open("output.txt", "w") as file:  
 file.write("Simple text")

# \*\*5. writelines()\*\*

Writes a list of strings to the file.

## Examples:

file = open("items.txt", "w")  
file.writelines(["Apple\n", "Banana\n"])  
file.close()

with open("list.txt", "w") as file:  
 file.writelines(["One\n", "Two\n"])

lines = ["Dog\n", "Cat\n"]  
file = open("pets.txt", "w")  
file.writelines(lines)  
file.close()

# \*\*6. close()\*\*

Closes the file to free up system resources.

## Examples:

file = open("sample.txt", "r")  
file.close()

file = open("sample.txt", "w")  
file.write("Text")  
file.close()

file = open("sample.txt", "r")  
print("Opened file")  
file.close()

# \*\*7. seek(position)\*\*

Moves the file cursor to the specified byte.

## Examples:

file = open("sample.txt", "r")  
file.seek(0)  
print(file.read())  
file.close()

file = open("sample.txt", "r")  
file.seek(5)  
print(file.read())  
file.close()

file = open("sample.txt", "r")  
file.seek(2)  
print(file.read())  
file.close()

# \*\*8. tell()\*\*

Returns the current cursor position.

## Examples:

file = open("sample.txt", "r")  
file.read(3)  
print(file.tell())  
file.close()

with open("sample.txt", "r") as file:  
 file.read(5)  
 print(file.tell())

file = open("sample.txt", "r")  
print(file.tell())  
file.close()

# \*\*9. truncate(size)\*\*

Cuts the file to a specific size in bytes.

## Examples:

file = open("sample.txt", "w")  
file.write("Hello, Python!")  
file.truncate(5)  
file.close()

with open("sample.txt", "w") as file:  
 file.write("Example text")  
 file.truncate(7)

file = open("sample.txt", "w")  
file.write("Testing truncate")  
file.truncate(4)  
file.close()

# \*\*10. flush()\*\*

Immediately writes content to disk without waiting.

## Examples:

file = open("log.txt", "w")  
file.write("Start\n")  
file.flush()  
file.close()

file = open("demo.txt", "w")  
file.write("Flushing data")  
file.flush()  
file.close()

with open("flush\_test.txt", "w") as file:  
 file.write("Flush this")  
 file.flush()

# \*\*11. with open(...) as\*\*

Automatically handles opening and closing the file safely.

## Examples:

with open("sample.txt", "r") as file:  
 print(file.read())

with open("output.txt", "w") as file:  
 file.write("Hello")

with open("notes.txt", "a") as file:  
 file.write("Note added\n")