File Handling

30 January 2025 00:02

Python provides built-in functions to handle files.

Method	Description
open()	Opens a file and returns a file object. It takes parameters like file name and mode (r, w, a, etc.).
read(size)	Reads and returns a string of the specified size from the file. If no size is given, it reads the entire file.
readline()	Reads a single line from the file. The cursor moves to the next line after reading.
readlines()	Reads all lines from the file and returns them as a list. Each item in the list represents a line.
write()	Writes a string to the file. It does not add a newline unless specified.
writelines(list)	Writes a list of strings to the file. It does not add a newline unless specified.
close()	Closes the file, freeing up resources. It's important to call this method after file operations are completed.
seek(cursor_position)	Moves the file cursor to a specific position, denoted by the cursor_position (an integer, where 0 is the beginning of the file).
tell()	Returns the current position of the cursor in the file. It returns an integer indicating the byte offset of the cursor.

"python is programming language"

→ f.read(6)
f.read(6)
" is pr"
ogramml

Python also offers different modes.

Mode	Description
r	Read mode. Opens the file for reading (default mode). The file must exist; otherwise, an error is raised.
w	Write mode. Opens the file for writing. If the file exists, it is overwritten(deletes the existing content while opening the file itself); if it doesn't exist, a new file is created.
a	Append mode. Opens the file for appending. If the file exists, new data is added to the end; if it doesn't exist, a new file is created.
x	Exclusive creation/write mode. Create new file and opens the file for writing . If the file exists, an error is raised.
r+	Read/Write mode. Opens the file for both reading and writing. The file must exist; otherwise, an error is raised. It will not delete the existing content while file is opened, but can overwrite.
w+	Write/Read mode. Opens the file for both writing and reading. If the file exists, it is overwritten(deletes the existing content while opening the file itself); if it doesn't exist, a new file is created.
a+	Append/Read mode. Opens the file for both appending and reading. If the file exists, new data is added to the end; if it doesn't exist, a new file is created.

With keyword:
==================

→ when you open wing with Keyword, Python automatically close the file not like normal file opening where you suppose to close the file Explicitly.

Handling other file formats

File Format	Library	Methods	Example

.csv	csv	csv.reader(), csv.writer()	import csv
			csv.reader(file)

.csv (alternative)	pandas	pd.read_csv(), df.to_csv()	import pandas as pd pd.read_csv('file.csv')

.json	json	json.load(), json.dump()	import json
			json.load(file)

.xlsx	pandas	pd.read_excel(), df.to_excel()	import pandas as pd pd.read_excel('file.xlsx')
.xlsx (alternative)	openpyxl	load_workbook(), save()	import openpyxl openpyxl.load_workbook('file

			.xlsx')

.xml	xml.etree.ElementTree	ElementTree.parse(), ElementTree.write()	import xml.etree.ElementTree as ET
			ET.parse(file)

.pkl	pickle	pickle.load(), pickle.dump()	import pickle
			pickle.load(file)

.jpg, .png	Pillow (PIL)	Image.open(), Image.save()	from PIL import Image
			Image.open('image.jpg')

.jpg, .png	opencv	cv2.imread(), cv2.imwrite()	import cv2
			cv2.imread('image.jpg')

.pdf	PyPDF2	PdfReader(), page.extract_text()	import PyPDF2
			PyPDF2.PdfReader(file)