01-02-2026

Agenda :

- File Operations - II
- Exception Handling
- Exit commands
- ~~OOPs intro~~

File operation

f = open() → open the file → reading the file → mode

with open() as f :

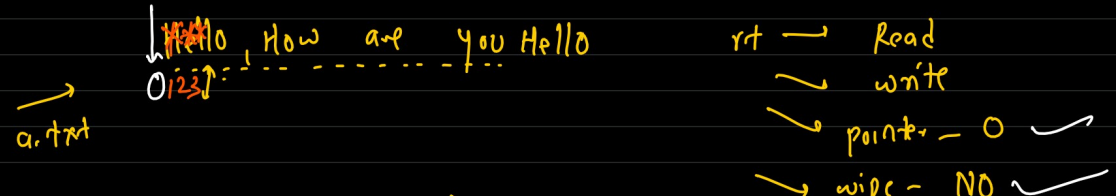
→ pointer

Hello world

f.read()

Mode Matrix/Table

Mode	Meaning	Pointer start	wipe file	usage	Create file ?
r	Read only	Beginning	NO	Standard Reading	Does not create file
w	write only	Beginning	Yes	Fresh start, deletes old data	Creates file if not present
a	Append	End	NO	Adding more content	Creates file if not present
r+	Read + write	Beginning	NO	Modifying existing content	Does not create file
w+	write + Read	Beginning	Yes	Overwrite	Creates file if not present
a+	Append + Read	End	NO		Creates file if not present



f = open(a.txt, 'rt'):

Scenario-1

f.read() → read all content

f.write() → Hello

Scenario-2

f.write() → ***
123

~~f.read()~~

→ what if i want to add new content in the middle of file?

→ pointers
→ some method which will help u move pointer
→ f.seek() # move the pointer/cursor

→ f.read() → seek
f.write() → fail

a, at → always append at the end of the file.
→ moving pointer using f.seek() will not work.

r, rt, wt, w → f.seek() will work.

Assignment:

Add new data in the middle of the file without overwriting the existing content.

→ rt, wt
→ f.write()
→ f.read() f.tell()
f.tell()
last_index last_index
→ middle_index → (last_index // 2) + 1
→ f.seek(middle_index)

→ from_centre_content = f.read()

10 → * * * → 13

→ f.write("***")

→ new_pointer_loc = f.tell()

→ f.write(from_centre_content)

if condition :

else :

try :

except :

→ error
exit()

→ error → print("___")

fin → part of name

Exception Type : Description

FileNotFoundError ←

TypeError ←

try :

except :

try : → file operations

except FileNotFound Error :

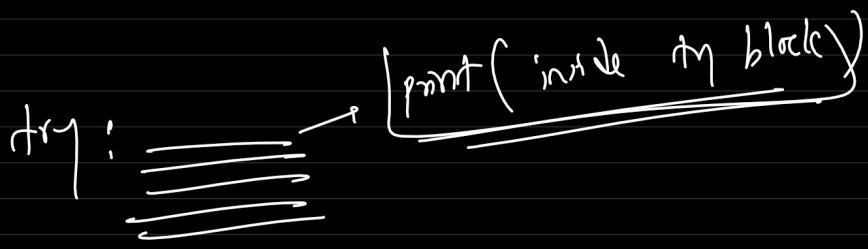
```
import os

try:
    file_path = "temp/temp_3.txt"
    with open(file_path, "r+") as f:
        contents = f.read()
        print(contents)
except FileNotFoundError:
    print("File not found,Please check correct file name or path")
    print("Current location is : ", os.getcwd())
    print("File name selected : ", "temp/temp_3.txt")
except TypeError:
    print("Issue with multiple type operation.")
print("Hello world")
```

try

except

1. Use it to handle errors gracefully.
2. Use it to write better debug comments.
3. Use it to execute the otherwise code. (recovery code)



except : print (inside except)

else : will run only when try is successfull , zero errors

try:

except:

else:

finally:

try:

except Exception as e:

else:

finally:

Run no matter what

mac OS → restart / shutdow → closing all application
windows → → force close at the end

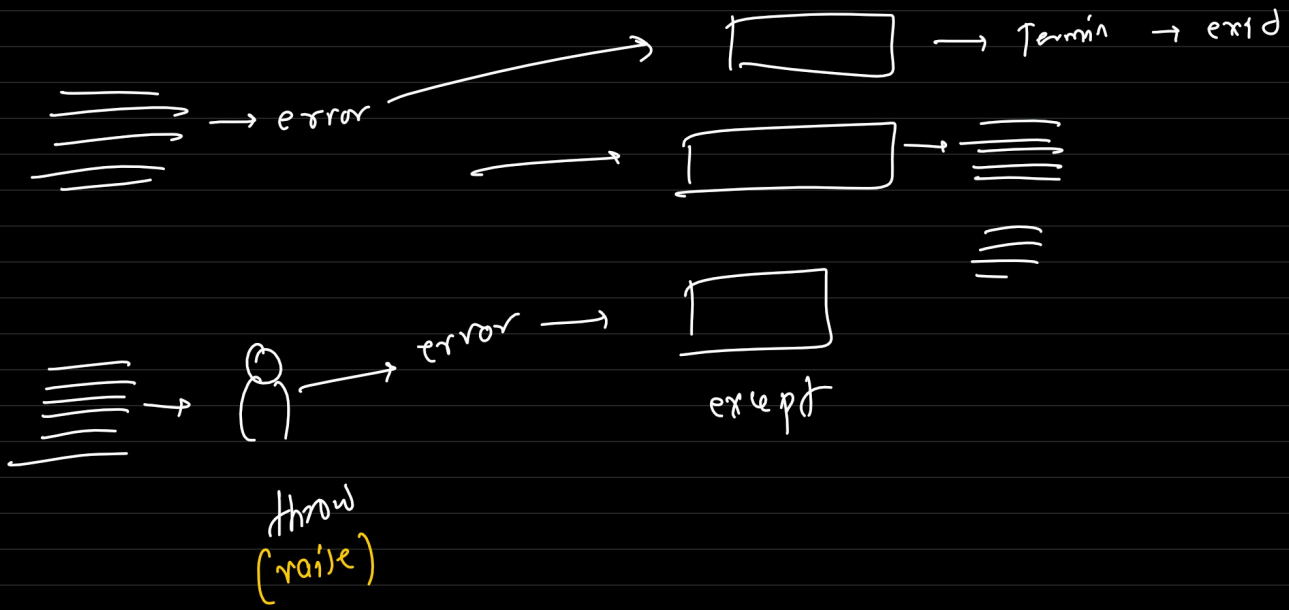
finally:

try:

safe close

finally:

force close



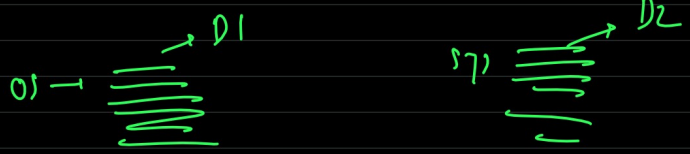
exit commands:

exit the application gracefully

(1) exit() → only works with terminal

(2) import os
os._exit()

(3) import sys
sys.exit()



break sys.exit()