



sys module

stdin, stdout, stderr

stdin, stdout, stderr

- ▶ Standard streams are interconnected input and output communication channels between computer program and its environment when it begins execution.
- ▶ The three input/output connections are
- ▶ **standard input(stdin)**-input stream where data is sent to or taken or read by a program
- ▶ **standard output(stdout)**-output stream to write or print the output from the code.
- ▶ **standard error(stderr)**-it is the default file descriptor where a process can write error messages.
- ▶ *By default both stdout and stderr are written on the screen but print command goes to standard out and error messages go to standard error*

sys.stdin-replacement of input ()
sys.stdout and sys.stderr-replacement of print()
example:

- ▶ Import sys
- ▶ print("enter messge")
- ▶ mess=sys.stdin.readline()
- ▶ print("ur message")
- ▶ sys.stdout.write(mess)
- ▶ sys.stderr.write("error")

To read from file

- ▶ `a=open("abc.txt")`
- ▶ `for i in a:`
- ▶ `sys.stdout.write(i)`

To write or add info into a file

- ▶ `sys.stdout=open("abc.txt","a")`
- ▶ `print("hello")`
- ▶ `sys.stdout.close()`

Write a program that reads a file and makes two kinds of output: the no of words in the file on stdout and no of lines in the file on stderr.

- ▶ `a=open("abc.txt")`
- ▶ `c=0`
- ▶ `d=0`
- ▶ `for i in a:`
 - ▶ `print(i)`
 - ▶ `c=c+(len(i.split()))`
 - ▶ `d=d+1`
- ▶ `sys.stdout.write(str(c))`
- ▶ `sys.stderr.write(str(d))`

Flush()

- ▶ Flush()-It is used to clear the data inside the memory so that it can be used to write new data to it.
- ▶ Python automatically flushes the files when closing them. But we can flush a file before closing it by using flush() method.
- ▶ Flush() does not require any parameters.
- ▶ Flush() does not return anything
- ▶ Syntax:
 - ▶ fileobjectname.flush()

Example:

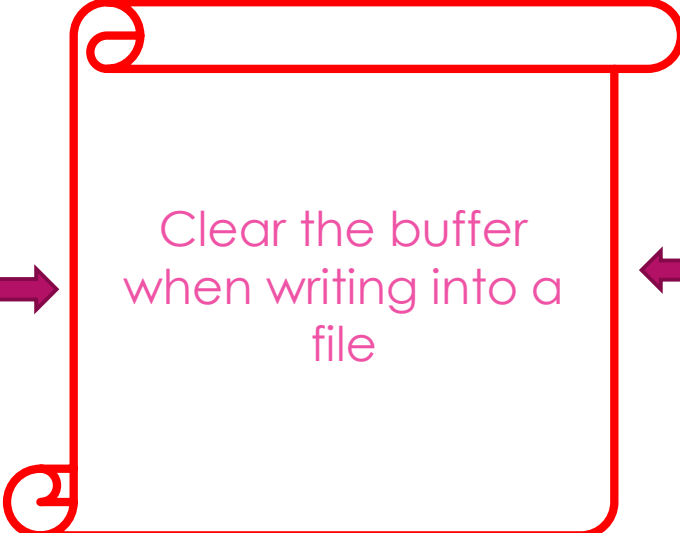
- ▶ `f=open("abc.txt")`
- ▶ `print("before flush()")`
- ▶ `print(f.read())`
- ▶ `f.flush()`
- ▶ `print("after flush")`
- ▶ `print(f.read())`
- ▶ `f.close()`



it clears the input
buffer of the file so
fileobject reads
nothing.
Hence nothing
displayed after flush()
method

Example:

- `import sys`
- `import time`
- `for i in range(10):`
- `print(i)`
- `sys.stdout.flush()`
- `time.sleep(1)`



Clear the buffer
when writing into a
file

- `f=open("abc.txt","a")`
- `f.write("new info")`
- `f.flush()`
- `f.write("another line")`