

File I/O



TH.S TRẦN ĐỨC LỢI
PYTHONVIETNAM.INFO

Ôn tập bài cũ



- Ôn lại nội dung đã học về **Function, module & package**

Mục đích bài học



- Tìm hiểu về file I/O trong python

I/O



- `Raw_input()`
- `Input()`
- `Print()`

Biểu diễn dữ liệu



Type	Meaning
'b'	Binary format. Outputs the number in base 2.
'c'	Character. Converts the integer to the corresponding unicode character before printing.
'd'	Decimal Integer. Outputs the number in base 10.
'o'	Octal format. Outputs the number in base 8.
'x'	Hex format. Outputs the number in base 16, using lower- case letters for the digits above 9.
'X'	Hex format. Outputs the number in base 16, using upper- case letters for the digits above 9.
'n'	Number. This is the same as 'd', except that it uses the current locale setting to insert the appropriate number separator characters.
None	The same as 'd'.

Mở một file



- Tác dụng của hàm `open()`
- Giá trị trả về
- Filehandle = ***open***(filename, mode, buffer)
- Ví dụ:
 - `Open('file.loi', 'r')`

File I/O: Mode



- Mặc định là read.
- r: reading only, pointer at beginning. Error if not exists
- r+: both read & write
- w: writting only. Overwrite if existed.
- w+: both writting & reading. Ponter at the beginning.
- a: appending. pointer at the end if file exists. create new if not exists
- a+: appending & reading. pointer at the end.
- b: in binary mode

File I/O: Buffer



- nếu = 0, không sử dụng buffer
- nếu = 1: line buffering
- nếu > 1: buffer with buffer size
- nếu < 0: buffer size = system default

File I/O



- Trong trường hợp không có file?
- Try ... except khi mở file
- File handle with for loop
 - For l in handle:
- Bài tập:
 - Mở một file do người dùng nhập vào
 - Đếm số dòng trong một file đó?

File I/O: read & readline



- `Handle.read()`
- `Handle.read(xbytes)`
- `Handle.readline()`
- `Handle.readline(xbytes)`
- `Handle.readlines(xbytes)`
- EOF: trả về null



- **Bài tập:**
 - In ra nội dung của 1 file sample với từng dòng
 - Sử dụng các hàm xử lý chuỗi để đảm bảo in ra đúng

File I/O: pointer



- `Handle.write('x')`
- `Handle.tell()`
- `Handle.seek(offset, from)`
- **From:**
 - 0: đầu file
 - 1: vị trí hiện thời
 - 2: cuối file
- **Offset: bytes**

OS module: file & folder



- `Os.mkdir()`
- `Os.chdir()`
- `Os.getcwd()`
- `Os.rmdir()`
- `Os.remove()`
- `Os.path.isfile()`
- `Os.path.exists()`
- `Os.access(filename, os.R_OK)`

Zip files

`z =
zipfile.ZipFile(filename)`

`z.namelist()`

`zipfile
module`

`z.write()`

`z.extractall(pwd='123456')`

CSV file



open

CSV

write

read

CSV file



- Import csv
- `writer = csv.writer(fh, delimiter=',', quotechar='|', quoting=csv.QUOTE_MINIMAL)`
 - Delimiter: A one-character string used to separate fields. It defaults to ','.
 - Quotechar: A one-character string used to quote fields containing special characters, such as the *delimiter* or *quotechar*, or which contain new-line characters. It defaults to '"'.
 - Quoting: Controls when quotes should be generated by the writer and recognised by the reader. Default = QUOTE_MINIMAL
 - Lineterminator: The string used to terminate lines produced by the [writer](#). It defaults to '\r\n'.

CSV file



- Import csv
- `reader = csv.reader(fh, delimiter=',', quotechar='|')`

JSON



- `import json`
- `jsonout = json.dumps(input, sort_keys=True)`
- `orig = json.loads(jsonout)`

Tổng kết bài học



- File open
- File read
- File write
- Zip file
- CSV file
- JSON
- Folder/file in OS module