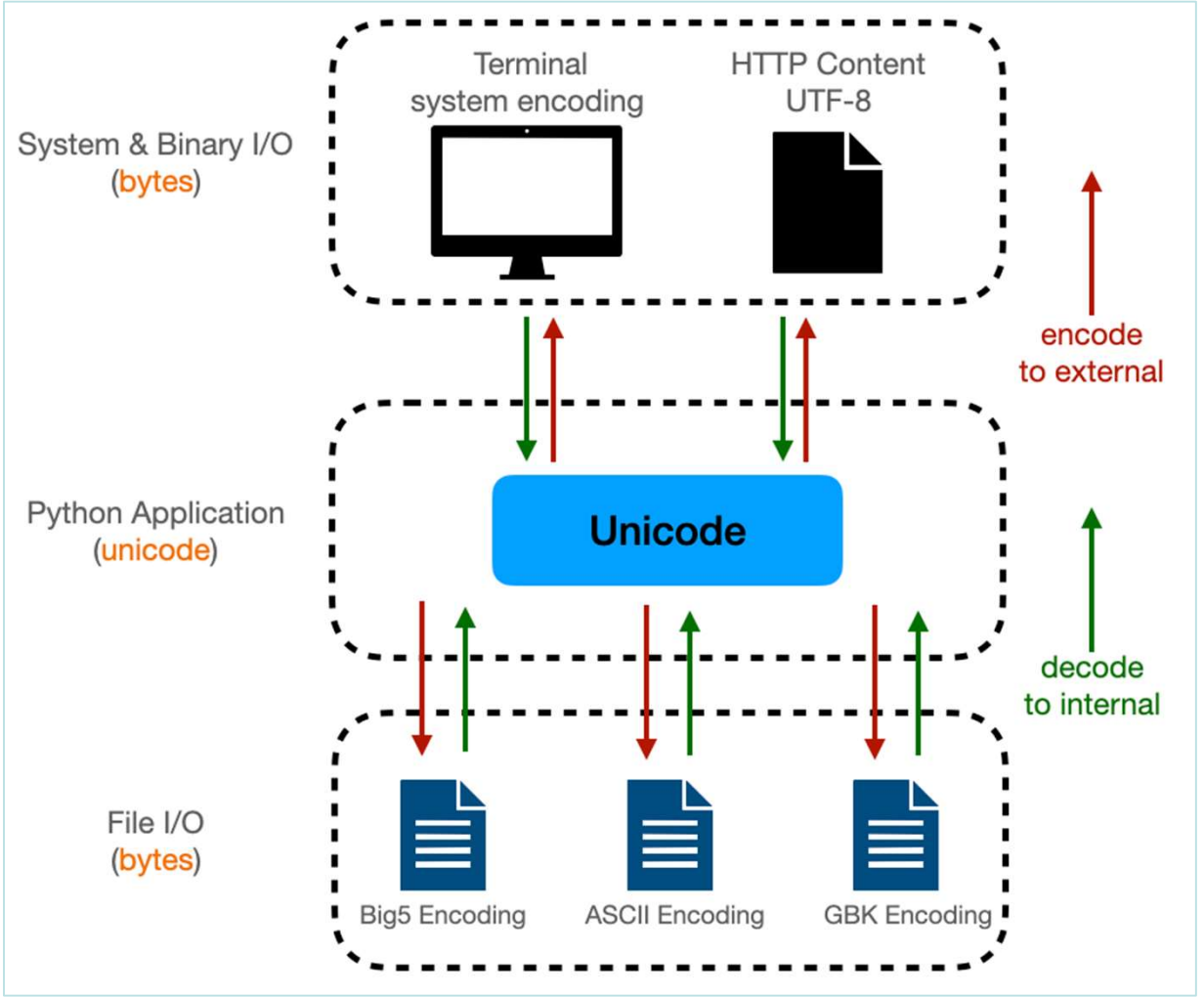
Python
字串 string 、 byte 轉換

臺北科技大學資訊工程系

string 與 byte 轉換

- 在 Python 程式裡
  - 所有「內部」的操作，都使用 Unicode 來處理，
  - 在與「外界」溝通時，才需要轉換成 bytes。



https://medium.com/gofreight_hq/python-diving-unicode-%E6%B7%B1%E5%85%A5%E6%B7%BA%E5%87%BA-8cdbee3fe81c

string 與 byte 轉換

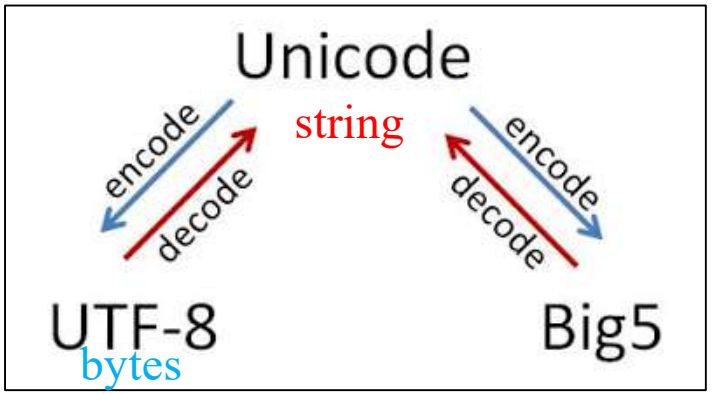
- 「外界」包含下列
  - File I/O (read, write)
  - System I/O (stdin, stdout, stderr)
  - Binary I/O (http content)
- 在 Python3 裡的 function 參數
  - 都傳 bytes
  - 若傳 Unicode 字串就會 Exception
- Use .encode() to convert human text to bytes.
- Use .decode() to convert bytes to human text.

https://medium.com/gofreight_hq/python-diving-unicode-%E6%B7%B1%E5%85%A5%E6%B7%BA%E5%87%BA-8cdbee3fe81c

string 與 byte 轉換

- Python
  - 文字/字串使用 Unicode，由 str 型別表示
  - 二進制資料，以 bytes 型別表示

- str 跟 bytes 可互相轉換
  - bytes = str.encode()
    - 預設編碼 utf-8
  - str = b bytes.decode()



str (Unicode)	⇔	bytes (utf-8)
str.encode() (utf-8)	⇔	bytes.decode() (Unicode)

string 與 byte 轉換

- 字符集
  - ASCII
  - Unicode (\uXXXX)
  - Big5　　「大五碼」收錄13060字
  - GB 2312　《信息交換用漢字編碼字符集·基本集》收錄6763字
  - GBK　　《漢字內碼擴展規範》收錄21886字
  - GB 18030　《信息技術中文編碼字符集》、完全支援Unicode
  - 不同字符集間，可互相轉換。

- 字符編碼 (0, 1 訊號 對應 字符 之間的映射規則) UTF-8
  - 是Unicode的一種電腦儲存實現方式，即字元編碼格式。
  - UTF-8 一般用於網路傳輸。

[東] 全字庫 CNS11643

0	0	0000
1	1	0001
2	2	0010
3	3	0011
4	4	0100
5	5	0101
6	6	0110
7	7	0111
8	8	1000
9	9	1001
10	A	1010
11	B	1011
12	C	1100
13	D	1101
14	E	1110
15	F	1111

- human text
- Unicode

- U+6771

0110 0111

0111 0001

- bytes

- UTF-8 encoder/decoder
  - https://mothereff.in/utf-8

\xE6	\x9D	\xB1
1110 0110	1001 1101	1011 0001

字串編碼

- 字串前加 u，如u"中文"，
  - 可建立 unicode 物件實例，
  - str 型別

```
s = u'\u4eba\u751f\u82e6\u77ed\uff0cpy\u662f\u5cb8'
print(type(s))  # <class 'str'>
print(s)       #人生苦短，py是岸

s_utf8 = s.encode(encoding='utf-8')
print(s_utf8)
#b'\xe4\xba\xba\xe7\x94\x9f\xe8\x8b\xa6\xe7\x9f\xad\xef\xbc\x8cpy\xe6\x98\xaf\xe5\xb2\xb8'
print(type(s_utf8))       #<class 'bytes'>
```

字串編碼

- str 型別
- bytes 型別

```
print(type("中文"))
print(type("中文".encode("utf-8")))
print(type(u"中文"))
print(len("中文"))
```

```
<class 'str'>
<class 'bytes'>
<class 'str'>
2
```

字串編碼

- bytes 型別

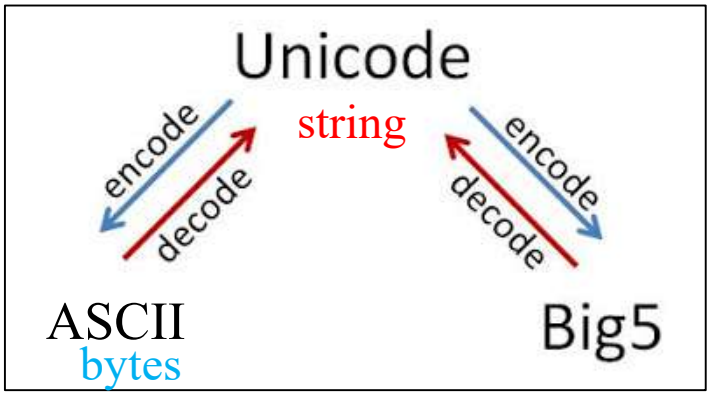
```
a = bytes([1,2,3,4,5,6,7,8,9])
b = bytes('python', 'ascii')
print(type(a))    # <class 'bytes'>
print(type(b))    # <class 'bytes'>
print(a)          # b'\x01\x02\x03\x04\x05\x06\x07\x08\t'
print(b)          # b'python'
```

字串編碼

- 要表示 byte 字串，使用 b 前置符號

```
s = 'Cafe'
print(s)
print(type(s))
print(s.encode("ascii"))
print(type(s.encode("ascii")))
print(s.encode("ascii").decode('ascii'))
print(type(s.encode("ascii").decode('ascii')))

s = 'Café'
print(s)
print(type(s))

print(s.encode("utf-8"))
print(type(s.encode("utf-8")))

s.encode('utf-8').decode('utf-8')
print(type(s.encode('utf-8').decode('utf-8')))
print(s)
```

```
Cafe
<class 'str'>
b'Cafe'
<class 'bytes'>
Cafe
<class 'str'>


Café
<class 'str'>

b'Caf\xc3\xa9'
<class 'bytes'>


<class 'str'>
Café
```

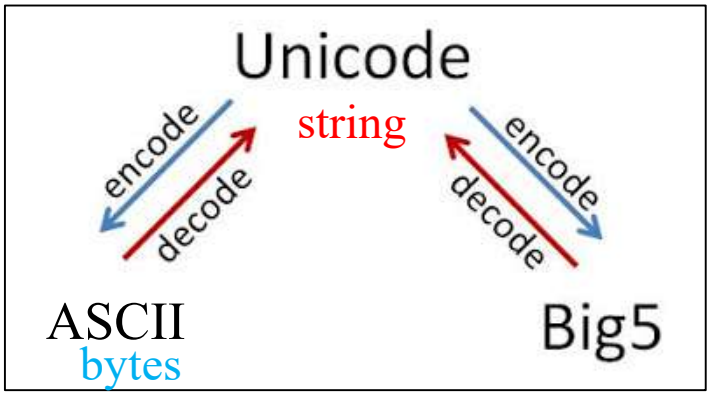


字串編碼

- 要表示 byte 字串，使用 b 前置符號

```
s = 'Café'
print([_c for _c in s])
print(len(s))
bs = bytes(s, encoding='utf-8')
print(bs)
print(type(bs))
```

```
['C', 'a', 'f', 'é']
4

b'Caf\xc3\xa9'
<class 'bytes'>
```



字串編碼

- 讀檔寫檔時，建立 I/O實例，透過參數 encoding 編碼。

```
with open("filename.txt",'w',encoding='utf-8') as outfile:
    outfile.write("anything you want to write")

with open("filename.txt",'r',encoding='utf-8') as infile:
    text = infile.read()

print(type(text))
```

```
<class 'str'>
```

string 與 byte 轉換

```
a = bytes([1,2,3,4,5,6,7,8,9])
b = bytes('python', 'ascii')  #指定ASCII編碼
print(type(a))    # <class 'bytes'>
print(type(b))    # <class 'bytes'>
print(a)          # b'\x01\x02\x03\x04\x05\x06\x07\x08\t'
print(b)          # b'python'
```

```
s = u'\u4eba\u751f\u82e6\u77ed\uff0cpy\u662f\u5cb8'
print(type(s))  # <class 'str'>
print(s)        #人生苦短，py是岸

s_utf8 = s.encode(encoding='utf-8')
print(s_utf8)
#b'\xe4\xba\xba\xe7\x94\x9f\xe8\x8b\xa6\xe7\x9f\xad\xef\xbc\x8cpy\xe6\x98\xaf\xe5\xb2\xb8'
print(type(s_utf8))       #<class 'bytes'>
```

string 與 byte 轉換

```
# bytes 轉字串 string 方式一
b = b'\xe9\x80\x86\xe7\x81\xab'
string = str(b,'utf-8')
print(string)

# bytes 轉字串 string 方式二
b = b'\xe9\x80\x86\xe7\x81\xab'
string = b.decode()   # 第一參數預設utf-8，第二參數預設 strict
print(string)

# bytes 轉字串 string 方式三
b = b'\xe9\x80\x86\xe7\x81haha\xab'
string = b.decode('utf-8', 'ignore')   # 忽略非法字符，用strict會拋出異常
print(string)

# bytes 轉字串 string 方式四
b = b'\xe9\x80\x86\xe7\x81haha\xab'
string = b.decode('utf-8', 'replace')   # 用？取代非法字符
print(string)
```

string 與 byte 轉換

```
# 字串 string 轉 bytes 方式一
str1 = '逆火'
b = bytes(str1, encoding='utf-8')
print(b)

# 字串 string 轉 bytes 方式二
b = str1.encode('utf-8')
print(b)
```

```
b'\xe9\x80\x86\xe7\x81\xab'
b'\xe9\x80\x86\xe7\x81\xab'
```

END