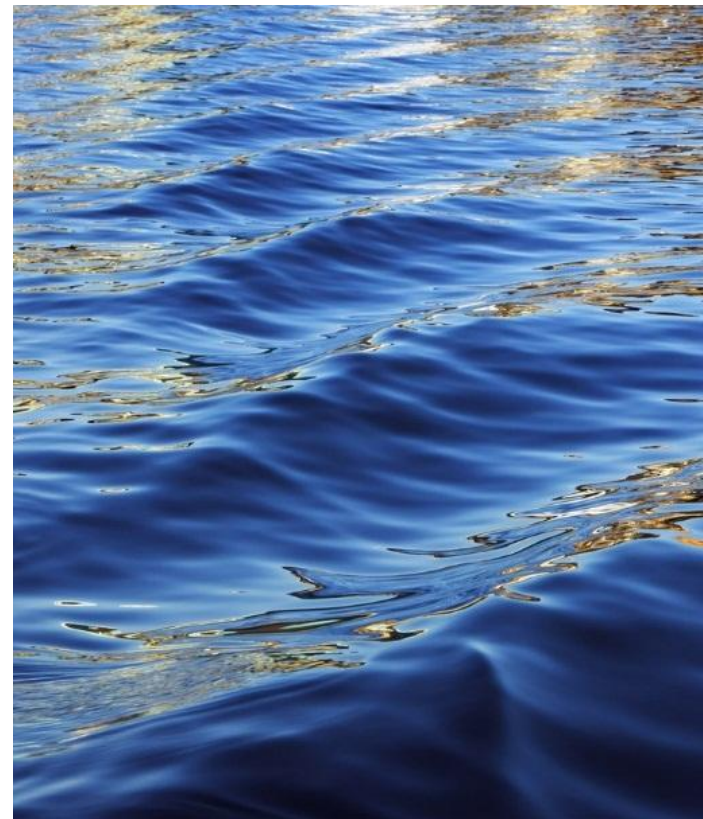





Python

Ricky





4. 資料型態

1. 種類與Type()函數
 2. 數值資料型態
 3. 2進位整數與函數bin()
 4. 8進位整數與函數oct()
 5. 16進位整數與函數hex()
 6. 數值運算常用的函數
 7. 布林值資料型態
 8. 字串資料型態
- 

種類與Type()函數

- Python的基本資料型態有下列幾種：
- 數值資料型態(numeric type)
- 布林值(Boolean)資料型態
- 文字序列型態(text sequence type)
- 字元組(bytes，有的書稱字節)資料型態
- 序列型態(sequence type)
- 對映型態(mapping type)
- 集合型態(set type)

種類與Type()函數

- 如何確認變數種類
- 使用內建函數 Type()

```
x = 10
y = x / 3
print(x)
print(type(x))
print(y)
print(type(y))
```

數值資料型態

- 整數int

- Python 3已經將整數可以儲存空間大小的限制拿掉了
- 英文googol是指自然數 10^{100}

[illegible]

- 浮點數float

- 帶有小數點的數字我們稱之為浮點數
- $x = 10.31$

- Python在宣告變數時可以不用設定這個變數的資料型態

```
x = 10.0
print(x)
print(type(x))
```

數值資料型態

- 整數與浮點數的運算

- Python具有簡單自動轉換能力，在計算時會將整數轉換為浮點數再執行運算。

```
#不同資料型態的運算
```

```
x = 10
y = x + 5.5
print(x)
print(type(x))
print(y)
print(type(y))
```

```
#整數轉換成浮點數的應用
```

```
a = 10
print(a)
print(type(a))          # 加法前列出a資料型態
a = a + 5.5
print(a)
print(type(a))          # 加法後列出a資料型態
```

數值資料型態

- 強制資料型態的轉換

- `int()`：將資料型態強制轉換為整數
- `float()`：將資料型態強制轉換為浮點數

- 將浮點數強制轉換為整數的運算

```
x = 10.5
print(x)
print(type(x))           # 加法前列出x資料型態
y = int(x) + 5
print(y)
print(type(y))           # 加法後列出y資料型態
```

- 將整數強制轉換為浮點數的運算

```
x = 10
print(x)
print(type(x))           # 加法前列出x資料型態
y = float(x) + 10
print(y)
print(type(y))           # 加法後列出y資料型態
```

2進位整數與函數bin()

- 凡是0b開頭的數字，代表這是2進位的整數。
- bin()函數可以將一般整數數字轉換為2進位。
- 下面是10進位數值與2進位數值互轉的應用。

```
x = 0b1101      # 這是2進位整數
print(x)         # 列出10進位的結果
y = 13           # 這是10進為整數
print(bin(y))    # 列出轉換成2進位的結果
```


8進位整數與函數oct()

- Python中定義凡是0o開頭的數字，代表這是8進位的整數。
- oct()函數可以將一般數字轉換為8進位。
- 將10進位數值與8進位數值互轉的應用。

```
x = 0o57          # 這是8進位整數
print(x)          # 列出10進位的結果
y = 47            # 這是10進為整數
print(oct(y))     # 列出轉換成8進位的結果
```

16進位整數與函數hex()

- 凡是0x開頭的數字，代表這是16進位的整數。
- hex()函數可以將一般數字轉換為16進位。
- 10進位數值與8進位數值互轉的應用

```
x = 0x5D          # 這是16進位整數
print(x)          # 列出10進位的結果
y = 93            # 這是10進為整數
print(hex(y))     # 列出轉換成16進位的結果
```

數值運算常用的函數

- `abs()`
 - 計算絕對值
- `pow(x,y)`
 - 返回x的y次方
- `round()`
 - 這是採用演算法則的Bankers Rounding觀念，如果處理位數左邊是奇數則使用四捨五入，如果處理位數左邊是偶數則使用五捨六入，例如：`round(1.5)=2`，`round(2.5)=2`
 - 處理小數時，第2個參數代表取到小數第幾位，小數位數的下一個小數位數採用“5”以下捨去，“5”以上進位，例如：`round(2.15,1)=2.1`，`round(2.25,1)=2.2`，`round(2.151,1)=2.2`，`round(2.251,1)=2.3`

數值運算常用的函數

- abs()、pow()、round()、round(x,n)函數的應用

```
x = -10
print("以下輸出abs( )函數的應用")
print(x)          # 輸出x變數
print(abs(x))     # 輸出abs(x)
x = 5
y = 3
print("以下輸出pow( )函數的應用")
print(pow(x, y))  # 輸出pow(x,y)
x = 47.5
print("以下輸出round(x)函數的應用")
print(x)          # 輸出x變數
print(round(x))   # 輸出round(x)
x = 48.5
print(x)          # 輸出x變數
print(round(x))   # 輸出round(x)
x = 49.5
print(x)          # 輸出x變數
print(round(x))   # 輸出round(x)
print("以下輸出round(x,n)函數的應用")
x = 2.15
print(x)          # 輸出x變數
print(round(x,1)) # 輸出round(x,1)
x = 2.25
print(x)          # 輸出x變數
print(round(x,1)) # 輸出round(x,1)
x = 2.151
print(x)          # 輸出x變數
print(round(x,1)) # 輸出round(x,1)
x = 2.251
print(x)          # 輸出x變數
print(round(x,1)) # 輸出round(x,1)
```

布林值資料型態

- 布林值(Boolean)資料型態的值有兩種，True(真)或False(偽)
- 常用在流程控制
- 列出布林值True與布林值False的資料型態，並將布林值強制轉換為整數，同時列出轉換的結果。
- 將布林值與整數值相加的應用，並確認最後變數資料型態是甚麼

```
x = True
print(x)
print(type(x))          # 列出x資料型態
#轉換型態
print(int(x))
print(type(x))          # 列出x資料型態

y = False
print(y)
print(type(y))          # 列出y資料型態
#轉換型態
print(int(y))
print(type(y))          # 列出y資料型態
```

```
xt = True
x = 1 + xt
print(x)
print(type(x))          # 列出x資料型態

yt = False
y = 1 + yt
print(y)
print(type(y))          # 列出y資料型態
```

布林值資料型態

- 在程式設計中False值不一定是一定要經過條件判斷是False，才可以得到False，下列情況也會被視為False。
 - 布林值False
 - 整數0
 - 浮點數0.0
 - 空字串 ''
 - 空串列 []
 - 空元組 ()
 - 空字典 {}
 - 空集合 set()
 - None
 - 至於其它的皆會被視為True。

字串資料型態

- 字串(string)資料是指兩個單引號(')之間或是兩個雙引號(")之間任意個數字元符號的資料
- 英文字串的使用中常會發生某字中間有單引號，其實這是文字的一部份，如下所示

- This is James' s ball

```
>>> x = 'This is James's ball'
SyntaxError: invalid syntax
>>>
```

- 我們可以用雙引號解決

```
>>> x = "This is James's ball"
>>> print(x)
This is James's ball
>>>
```

字串資料型態-字串的連接

- 數學的運算子“ + ”，可以執行兩個字串相加，產生新的字串。

```
num1 = 222
num2 = 333
num3 = num1 + num2
print("以下是數值相加")
print(num3)
numstr1 = "222"
numstr2 = "333"
numstr3 = numstr1 + numstr2
print("以下是由數值組成的字串相加")
print(numstr3)
numstr4 = numstr1 + " " + numstr2
print("以下是由數值組成的字串相加，同時中間加上一空格")
print(numstr4)
str1 = "DeepStone "
str2 = "Deep Learning"
str3 = str1 + str2
print("以下是一般字串相加")
print(str3)
```


字串資料型態-處理多於一行的字串

- 程式設計時如果字串長度多於一行，可以使用三個單引號(或是3個雙引號)將字串包夾即可。
- 使用三個單引號處理多於一行的字串。

#不管追求什麼目標，都應堅持不懈。

```
str1 = '''Pursue your object, be it what it will,  
steadily and indefatigably.'''  
print(str1) # 字串內有沒分行符號
```

```
str1_1 = '''Pursue your object, be it what it will, \n  
steadily and indefatigably.'''  
print(str1_1) # 字串內有分行符號
```

```
str2 = "Pursue your object, be it what it will, " \n  
"steadily and indefatigably."  
print(str2) # 使用\\符號
```

```
str3 = ("Pursue your object, be it what it will,"  
"steadily and indefatigably.")  
print(str3) # 使用小括號
```

字串資料型態-逸出字元

要顯示一些特殊的字元時，需加入反斜線“ \ ”，加入反斜線的又叫作逸出字元，下面是其他的特殊功能

逸出字元	Hex 值	意義	逸出字元	Hex 值	意義
\'	27	單引號	\n	0A	換行
\"	22	雙引號	\o		8 進位表示
\\	5C	反斜線	\r	0D	游標移至最左位置
\a	07	響鈴	\x		16 進位表示
\b	08	BackSpace 鍵	\t	09	Tab 鍵效果
\f	0C	換頁	\v	0B	垂直定位

字串資料型態-逸出字元

- 逸出字元的應用，
- Str3 增加“\t”字元，所以“can’t”跳到下一個Tab鍵位置輸出。同時有“\n”字元，這是換行符號，所以“loving”跳到下一行輸出。

```
#以下輸出使用單引號設定的字串，需使用\  
str1 = 'I can\'t stop loving you.'  
print(str1)  
#以下輸出使用雙引號設定的字串，不需使用\  
str2 = "I can't stop loving you."  
print(str2)  
#以下輸出有\t和\n字元  
str3 = "I \tcan't stop \nloving you."  
print(str3)
```

字串資料型態-str()

- str()函數有好幾個用法：

1：可以設定空字串。

2：設定字串

3：可以強制將數值資料轉換為字串資料

```
x = str()
print("印出空字串")
print(x)

y = str("ABCD")
print("列印出字串")
print(y)

z = 12345
z = str(z)
print("印出轉換的字串")
print(z)
print("印出類型")
print(type(z))
```

字串資料型態-延伸應用

- 將字串轉換為整數: `int()` 函數可以將字串轉換為整數

```
x1 = "22"  
x2 = "33"  
x3 = x1 + x2  
print(x3)                # 列印字串相加  
x4 = int(x1) + int(x2)  
print(x4)                # 列印整數相加
```

- 字串與整數相乘產生字串複製效果: 字串與整數相乘

```
x1 = "A"  
x2 = x1 * 10  
print(x2)                # 列印字串乘以整數  
x3 = "ABC"  
x4 = x3 * 5  
print(x4)                # 列印字串乘以整數
```

字串資料型態-延伸應用

- 使用字串加法和換行字元\n

```
str1 = "資料型態"  
str2 = "字串類型應用"  
str3 = "斷行演練"  
str4 = str1 + "\n" + str2 + "\n" + str3  
print(str4)
```

- 字串前加上r，可以防止逸出字元(Escape Character)被轉譯

```
str1 = "Hello!\nPython"  
print("不含r字元的輸出")  
print(str1)  
str2 = r"Hello!\nPython"  
print("含r字元的輸出")  
print(str2)
```

字串資料型態-字串與字元

- 在Python沒有所謂的字元(character)資料，如果字串含一個字元，我們稱這是含一個字元的字串
- `chr(x)`：可以傳回函數x值的ASCII或Unicode字元。
- 從ASCII表可知，字元a的ASCII碼值是97，字元3的ASCII碼值是0x33可以使用下列方式印出此字元。

```
In [28]: chr(97)
Out[28]: 'a'
```

```
In [29]: chr(0x33)
Out[29]: '3'
```

HOMEWORK

- CH4 題目
- 1.從地球到月球約是384400公里，假設火箭的速度是每小時1225公里，設計一個程式計算需要多少天、多少小時才可抵達月球。這個程式省略分鐘數。