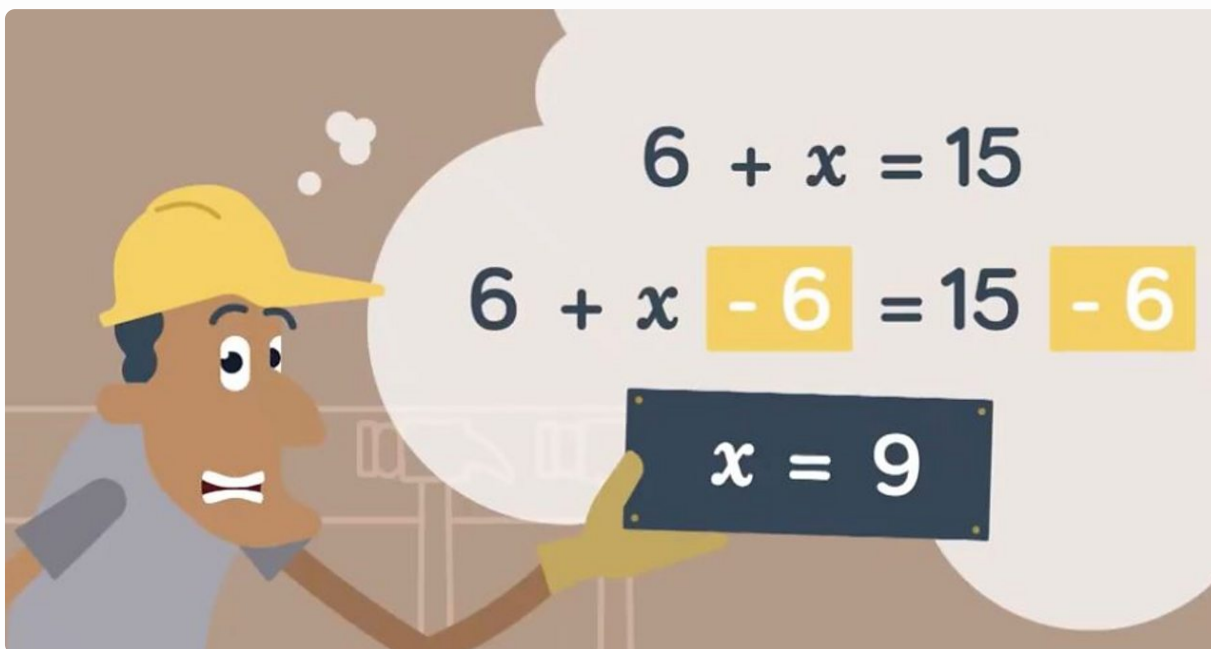


0_3_Variable

數學中的變數



數學中的變數

= 等號表示相等關係的符號，讀作「等於」[1]，是在西元1557年由Robert Recorde發明的。
在數學等式中，等號被放置在具有相同值的兩個（或更多個）表達式之間。(維基百科)

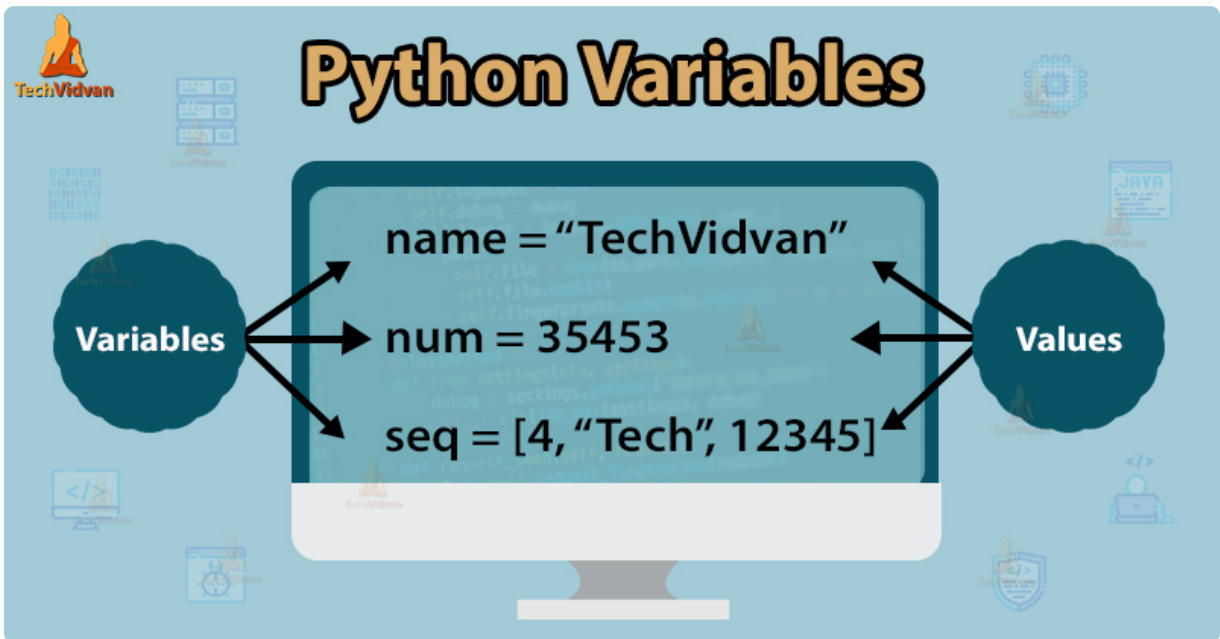
$$x = 1$$

$$y = 3 \Rightarrow y + 1 = 4$$

$$(x+1)^2 = x^2 + 2x + 1$$

$$x = x + 1 \quad \text{✗}$$

程式中的變數 Variable



程式中的變數

觀察變數

程式中的變數

```
x = 2 # 是將 2 放到 x 中; x 放入 2
```

= 是指定，assign

Write code in Python 3.6 (drag lower right corner to resize code editor)

→ 1 x = 2

Global frame
x 2

→ line that just executed
→ next line to execute

<< First < Prev Next > Last >>

Done running (1 steps)

Variable Example

```
1 x = 123      # 把 123 數值存在 x 變數中
2 print(x)     # 印出 x 變數的數值
3 print(type(x)) # 印出 x 變數的型態
```

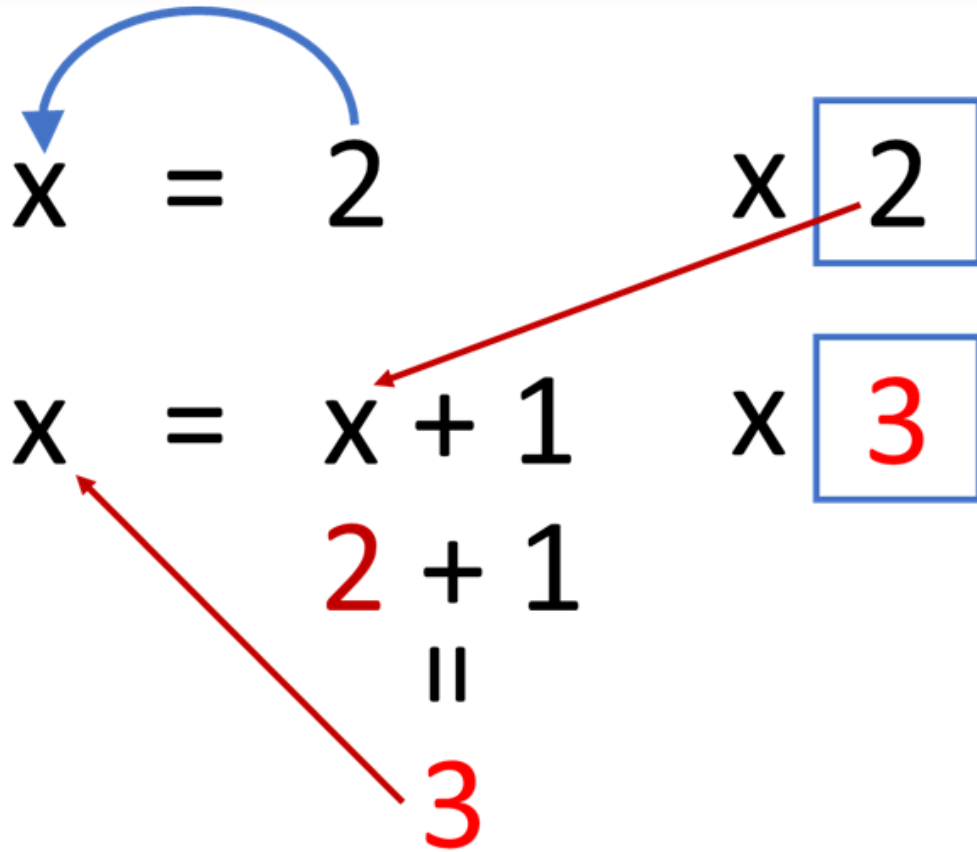
```
123
<class 'int'>
```

程式中的變數

x = --> x 在 = 左邊 代表容器 (variable)
= x --> x 在 = 右邊 代表內容物 (value)
x 單獨出現 代表內容物 (value)

```
x = 2
x = x + 1 ✓
```

```
x = 2
x = x + 1
==> x = 2 + 1
==> x = 3
```



程式中的變數

```
銀行帳號 = --> 代表銀行帳號
= 銀行帳號 --> 代表銀行帳號餘額
```

```
1 | 銀行帳號 = 200 # 銀行帳號 = (放入) 200
2 | print("銀行帳號 = 200          後銀行帳號有", 銀行帳號)
3 | # 有 300 要轉入銀行帳號
4 | 銀行帳號 = 銀行帳號 + 300 # 銀行帳號 = (放入) 銀行帳號餘額 + 200
5 | print("銀行帳號 = 銀行帳號 + 300 後銀行帳號有", 銀行帳號)
```

```
銀行帳號 = 200          後銀行帳號有 200
銀行帳號 = 銀行帳號 + 300 後銀行帳號有 500
```

變數使用前要先定義且指定一個值

```
1 | print("var ", var) # 未定義也未指定一個值
```

```
-----
NameError                                Traceback (most recent call last)
<ipython-input-13-60ab7fdb6ffd> in <module>
----> 1 print("var ", var)

NameError: name 'var' is not defined
```

```
1 | var # 定義 var 但未指定一個值
2 | print("var ", var)
```

```
-----
NameError                                Traceback (most recent call last)
<ipython-input-15-00639764908d> in <module>
----> 1 var
      2 print("var ", var)

NameError: name 'var' is not defined
```

變數使用前要先定義且指定一個值

```
1 | var = 100 # 定義 var 且指定一個值 100
2 | print("var ", var)
```

```
var 100
```

Variable can change type after they have been set

```
1 | x = 4          # x is of type int
2 | print("x = 4", x)
3 | x = "Sally"    # x is now of type str
4 | print('x = "Sally"', x)
```

```
x = 4 4
x = "Sally" Sally
```

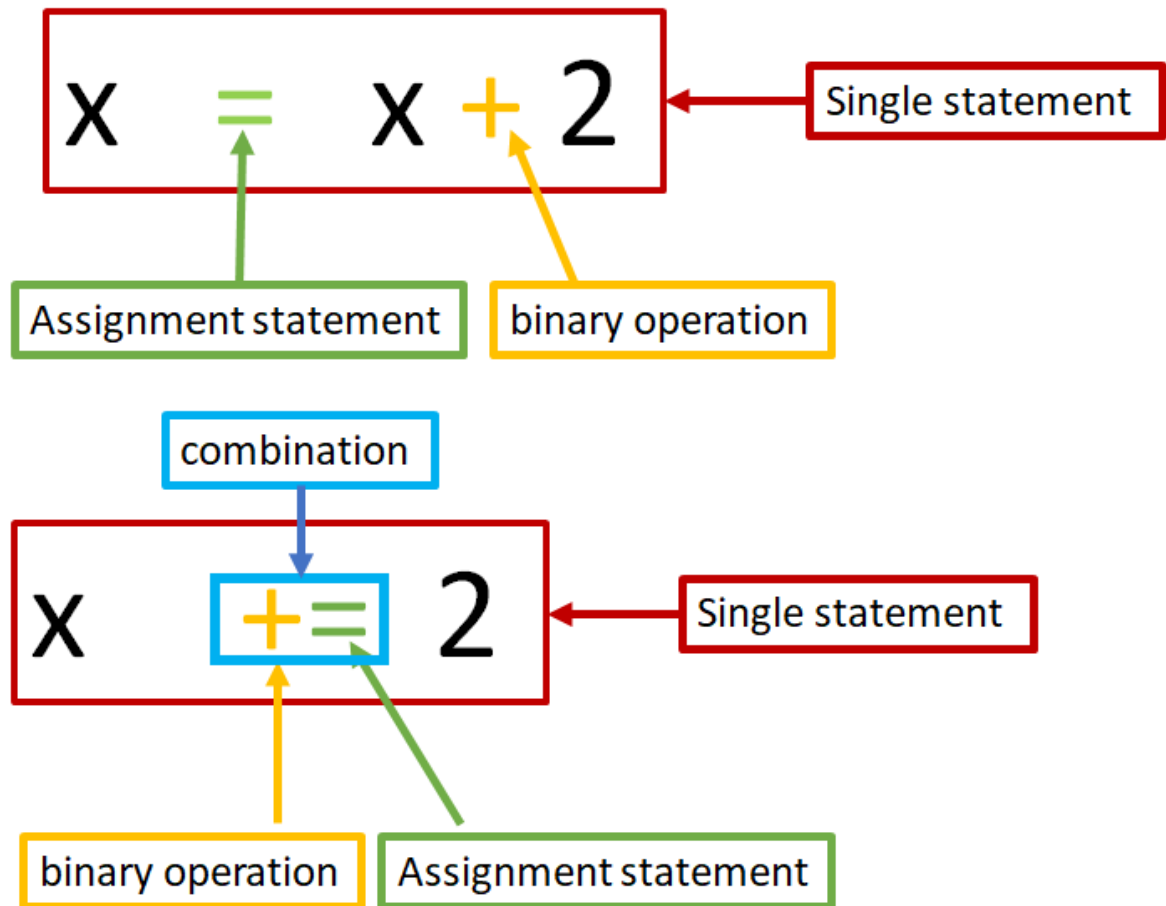
Augmented Assignment

augment : to increase the size or value of something by adding something to it

```
1 | x = 3
2 | x = x + 3
3 | print(" x = 3 ==> x = x + 3 ==> x is", x)
4 | x = 3
5 | x += 3
6 | print(" x = 3 ==> x += 3      ==> x is", x)
```

```
x = 3 ==> x = x + 3 ==> x is 6
x = 3 ==> x += 3      ==> x is 6
```

Augmented assignment is the combination, in a single statement, of a binary operation and an assignment statement



Augmented Assignment

`+=, -=, *=, /=, //=, %=, **=`

What is a after running following program? 🤔

```
a = 10
a += 10
a -= 5
a *= 3
a //= 2
a %= 5
a **= 2
```

Variable Name 變數命名

會區分英文大小寫

可以使用中文，但不建議

變數名稱只能包括以下字元：

1. 小寫字母 (a~z)
2. 大寫字母 (A~Z)
3. 數字 (0~9)
4. 底線 (_)

變數名稱第一個字 **不能** 是數字

變數名稱第一個字是數字 ❌

```
1 | 2ndhu = "NDHU"
```

```
File "<ipython-input-7-9e41c27b6cd6>", line 1
    2ndhu = "NDHU"
      ^
SyntaxError: invalid syntax
```

區分英文大小寫

```
1 | ndhu = "ndhu"
2 | NDHU = "NDHU"
3 | print("ndhu is", ndhu)
4 | print("NDHU is", NDHU)
```

```
ndhu is ndhu
NDHU is NDHU
```