

NỀN TẢNG CÔNG NGHỆ THÔNG TIN 2

CÁC LOẠI DỮ LIỆU VÀ TOÁN TỬ CƠ BẢN

L^AT_EX bởi Nguyễn “Sorai” Lê Đăng Khoa

I. Các kiểu dữ liệu trong Python

Text Type: `str`

Numeric Types: `int`, `float`, `complex`

Sequence Types: `list`, `tuple`, `range`

Mapping Type: `dict`

Set Types: `set`, `frozenset`

Boolean Type: `bool`

Binary Types: `bytes`, `bytearray`, `memoryview`

None Type: `NoneType`

Hình 1. Danh sách các loại dữ liệu trong Python

(Nguồn: https://www.w3schools.com/python/python_datatypes.asp)

Cách tạo biến trong Python

```
a = 1 #int
b = 1.5 #float
c = 'string' #str with ' '
d = "another string" #str with " "
e = True #bool
f = [1, 2, 3] #list
g = (1, 2, 3) #tuple
h = {1, 2, 3} #set
i = {1: 'one', 2: 'two', 3: 'three'} #dict
j = input() # input function
k = input("Enter a number: ") # input function with prompt
l = int(input("Enter a number: ")) # input function with prompt and convert to int
```

Lưu ý về việc đặt tên cho biến:

1. Tên không được cách ra, phải ghi chung thành 1 chữ (có thể dùng dấu underscore _ để thay thế như: python_code = 1 ✓; python code = 2 ✗)
2. Tên không được bắt đầu bằng chữ số (text = "text" ✓; 1number = 1 ✗)
3. Ngôn ngữ Python có các từ khóa (**Keywords**) với những chức năng đặc biệt, không thể dùng để đặt tên biến. Để kiểm tra những **Keywords** ấy, ta có thể dùng lệnh trong Python như sau (Chương trình sẽ output các từ khóa cần chú ý ra từng dòng):

```
import keyword
for i in keyword.kwlist:
    print(i)
```

II. Các toán tử trong Python

A. Toán tử số học (Arithmetic Operators)

Toán tử	Tên/Định nghĩa	Ví dụ
+	Cộng	x + y
-	Trừ	x - y
*	Nhân	x * y
/	Chia	x / y
%	Modulo	x % y
**	Lũy thừa	x ** y
//	Chia lấy phần nguyên	x // y

B. Toán tử gán (Assignment Operators)

Toán tử	Ví dụ	Ý nghĩa
=	<code>x = 7</code>	<code>x = y</code>
<code>+=</code>	<code>x += 7</code>	<code>x = x + 7</code>
<code>-=</code>	<code>x -= 7</code>	<code>x = x - 7</code>
<code>*=</code>	<code>x *= 7</code>	<code>x = x * 7</code>
<code>/=</code>	<code>x /= 7</code>	<code>x = x / 7</code>
<code>%=</code>	<code>x %= 7</code>	<code>x = x % 7</code>
<code>**=</code>	<code>x **= 7</code>	<code>x = x ** 7</code>
<code>//=</code>	<code>x //= 7</code>	<code>x = x // 7</code>
<code>&=</code>	<code>x &= 7</code>	<code>x = x & 7</code>
<code> =</code>	<code>x = 7</code>	<code>x = x 7</code>
<code>^=</code>	<code>x ^= 7</code>	<code>x = x ^ 7</code>
<code>»=</code>	<code>x »= 7</code>	<code>x = x » 7</code>
<code>«=</code>	<code>x «= 7</code>	<code>x = x « 7</code>
<code>:=</code>	<code>print(x := 7)</code>	<code>x = 7; print(x)</code>

C. Toán tử so sánh (Comparison Operators)

Toán tử	Tên/Định nghĩa	Ví dụ
<code>==</code>	Bằng nhau	<code>x == y</code>
<code>!=</code>	Khác nhau	<code>x != y</code>
<code>></code>	Lớn hơn	<code>x > y</code>
<code><</code>	Bé hơn	<code>x < y</code>
<code>>=</code>	x Lớn hơn hoặc bằng y	<code>x >= y</code>
<code><=</code>	x Bé hơn hoặc bằng y	<code>x <= y</code>

D. Toán tử logic (Logical Operators)

Toán tử	Tên/Định nghĩa	Ví dụ
<code>and</code>	Và	<code>(x > 5 and x < 10)</code>
<code>or</code>	Hoặc	<code>(x < 0 or x > 69)</code>
<code>not</code>	Không phải	<code>not(x == 5 or x == 97)</code>

E. Toán tử định danh (Identity Operators)

Toán tử	Tên/Định nghĩa	Ví dụ
is	Là/Như	x is y
is not	Không như	x is not y

F. Toán tử thành viên (Membership Operators)

Toán tử	Tên/Định nghĩa	Ví dụ
in	Có trong	x in
not in	Không nằm trong	x not in y

G. Toán tử bit (Bitwise Operators)

Toán tử	Tên/Định nghĩa	Ví dụ
&	2 bit giống nhau $\rightarrow 1$	x & y
	Ít nhất 1 trong 2 bit là 1 $\rightarrow 1$	x y
^	2 bit khác nhau $\rightarrow 1$	x ^ y
~	Đảo ngược các bit	~x
«	Bỏ bit ngoài cùng bên trái và thêm 0 vào bên phải	x « 2 # thực hiện 2 lần
»	Bỏ bit ngoài cùng bên phải và thêm 0 vào bên trái	x » 3 # thực hiện 3 lần

H. Thứ tự ưu tiên của các toán tử (Operator Precedence)

Độ ưu tiên giảm từ trên xuống

Toán tử	Tên/Định nghĩa
()	Parentheses
**	Exponentiation
+x -x ~x	Unary plus, unary minus, and Bitwise NOT
* / // %	Multiplication, division, floor division, and modulus
+ -	Addition and subtraction
« »	Bitwise left and right shifts
&	Bitwise AND
^	Bitwise XOR
	Bitwise OR
== != > < >= <= is is not in not in	Comparisons, identity, and membership operators
not	Logical NOT
and	AND
or	OR

III. Bài tập làm quen

Bài 1. Nhập từ bàn phím 2 giá trị, tương ứng với 2 cạnh góc vuông của 1 tam giác vuông, với dữ liệu là kiểu **float**. Hãy in ra 3 dòng với 3 kết quả lần lượt là: *cạnh huyền*, *chu vi* và *diện tích* của tam giác.

Bài 2. Nhập từ bàn phím 2 giá trị cho 2 biến kiểu **string**, hãy kiểm tra xem có hay không việc 1 biến string là con của biến string còn lại. Nói cách khác, giả sử gọi 2 biến string a và b, kiểm tra xem a có trong b hay b có trong a hay không. Nếu thỏa ít nhất 1 trong 2 điều kiện thì in ra chữ "YES", không thì in ra chữ "NO".

IV. Tài liệu tham khảo

- [1] Das, U., Lawson, A., Mayfield, C., & Norouzi, N. (2024). *Introduction to Python Programming*. Open-Stax.
- [2] W3schools website (<https://www.w3schools.com/python/default.asp>)
- [3] Python developer roadmap (<https://roadmap.sh/python>)

V. Lời của Sorai the Author

- Với những kiến thức mình đã trình bày và 2 bài tập nho nhỏ để luyện tập, mình mong mọi người có thể nắm sơ qua về những kiểu dữ liệu thông thường và các toán tử biến đổi/so sánh.
- Nếu muốn tìm hiểu thêm về ngôn ngữ lập trình Python thì mọi người có thể lên phần 4 (Tài liệu tham khảo) và truy cập vào các tài nguyên mà mình đã liệt kê để học thêm.
- Nhớ đánh giá bài này 5* nha UwU.

———— HẾT ————