

SAU BÀI NÀY EM SẼ:

- Biết và thực hiện được một số lệnh vào, ra đơn giản.
- Thực hiện được một số chuyển đổi dữ liệu giữa các kiểu dữ liệu cơ bản.



Để tương tác với người sử dụng trong khi thực hiện chương trình, các ngôn ngữ lập trình có các câu lệnh để đưa dữ liệu ra màn hình hay nhập dữ liệu vào từ bàn phím. Em đã biết Python có lệnh `print()` dùng để đưa dữ liệu ra màn hình. Để nhập dữ liệu từ bàn phím khi thực hiện chương trình, Python sử dụng câu lệnh `input()`.

Em dự đoán lệnh nhập dữ liệu `input()` có cú pháp và chức năng như thế nào?

Hãy nhập tên đầy đủ và tên viết tắt của bạn vào các ô sau:

Tên đầy đủ:

Tên viết tắt:

## 1. CÁC LỆNH VÀO RA ĐƠN GIẢN

Trong Bài 1 chúng ta đã biết quá trình xử lý thông tin trong máy tính thực chất cũng là xử lý dữ liệu. Do vậy, trong các ngôn ngữ lập trình bậc cao cần có các câu lệnh tương ứng để nhập, xuất dữ liệu. Trong Python, lệnh `print()` có chức năng đưa dữ liệu ra, còn lệnh `input()` đưa dữ liệu vào.

### Hoạt động 1 Tìm hiểu chức năng của lệnh `input()`

Quan sát lệnh sau và trả lời các câu hỏi: Lệnh `input()` cho phép nhập dữ liệu từ đâu? Giá trị được nhập sẽ là số hay chuỗi?

```
>>> input("Nhập một số: ")
Nhập một số: 12
'12'
```



Lệnh `input()` có chức năng nhập dữ liệu từ thiết bị vào chuẩn, thường là bàn phím. Nội dung nhập có thể nhập số, biểu thức hay chuỗi và cho kết quả là một chuỗi ký tự. Cú pháp của lệnh `input()` như sau:

```
<biến> = input(<Dòng thông báo>)
```

Cần nhập một chuỗi ký tự thì có thể dùng lệnh `input()` tương tự như sau:

```
name = input("Nhập họ tên em: ")
print("Xin chào ", name)
```

Lệnh `print()` có chức năng đưa dữ liệu ra thiết bị ra chuẩn, thường là màn hình. Thông tin cần đưa ra có thể bao gồm một hay nhiều dữ liệu với kiểu khác nhau, cho phép cả biểu thức tính toán.

Các lệnh vào ra đơn giản của Python bao gồm lệnh `input()` và lệnh `print()`.

## 2. CHUYỂN ĐỔI KIỂU DỮ LIỆU CƠ BẢN CỦA PYTHON

### Hoạt động 2 Nhận biết kiểu dữ liệu của biến

Chúng ta đã biết một số kiểu dữ liệu cơ bản như số nguyên, số thực và chuỗi ký tự. Trong Python có cách nào để nhận biết được kiểu dữ liệu của biến không?



Quan sát các lệnh sau để biết kiểu dữ liệu của mỗi biến.

```
>>> n, x, s = 10, 1.8, "One"  ← Gán n = 10, x = 1.8, s = "One" đồng thời.
>>> n
10
>>> type(n)
<class 'int'>  ← Biến n thuộc kiểu int – số nguyên.
>>> type(x)
<class 'float'>  ← Biến x thuộc kiểu float – số thực.
>>> type(s)
<class 'str'>  ← Biến s thuộc kiểu str – chuỗi ký tự.
```

Kiểu dữ liệu logic cũng là kiểu dữ liệu cơ bản và dữ liệu kiểu này chỉ có hai giá trị là True (đúng) và False (sai). Ví dụ dữ liệu kiểu logic là kết quả phép so sánh:

```
>>> 3 > 2
True
>>> 5 < 0
False
>>> b = 10 > 3
>>> type(b)
<class 'bool'>  ← Tên kiểu dữ liệu logic là bool.
```

- Một số kiểu dữ liệu cơ bản của Python bao gồm: `int` (số nguyên), `float` (số thực), `str` (chuỗi ký tự), `bool` (logic).
- Lệnh `type()` dùng để nhận biết kiểu dữ liệu của biến trong Python.



Xác định kiểu và giá trị của các biểu thức sau:

- a) `"15 + 20 - 7"`      b) `32 > 45`      c) `13 != 8 + 5`      d) `1 == 2`

### Hoạt động 3 Tìm hiểu cách chuyển đổi giữa các kiểu dữ liệu

- Có chuyển đổi dữ liệu kiểu này sang kiểu khác được không?
- Giả sử có biến `s` với giá trị `"123"`. Nếu muốn biến `s` có giá trị là số nguyên 123 chứ không phải là chuỗi `"123"` thì em phải làm gì?



Lệnh `int()` có chức năng chuyển đổi số thực hoặc xâu chứa số nguyên thành số nguyên. Quan sát các lệnh sau:

```
>>> int(12.6)
12
>>> int("123")
123
>>> int("10.35")
Traceback (most recent call last):
  File "<pyshell#21>", line 1, in <module>
    int("10.35")
ValueError: invalid literal for int() with base 10: '10.35'
```

Lệnh `int()` không chuyển đổi được xâu chứa số thực.

Lệnh `float()` dùng để chuyển đổi số nguyên và xâu kí tự thành số thực.

```
>>> float(8)
8.0
>>> float("10.23")
10.23
```

Lệnh `str()` dùng để chuyển đổi các kiểu dữ liệu khác thành xâu kí tự.

```
>>> str(12+34)
'46'
>>> str(12.567)
'12.567'
>>> str(2>3)
'False'
```

Chú ý: Các lệnh `int()`, `float()` chỉ có thể chuyển đổi các xâu ghi giá trị số trực tiếp, không chuyển đổi xâu có công thức, ví dụ:

```
>>> int("12+45")
Traceback (most recent call last):
  File "<pyshell#27>", line 1, in <module>
    int("12+45")
ValueError: invalid literal for int() with base 10: '12+45'
```

- Các lệnh `int()`, `float()`, `str()` có chức năng chuyển đổi dữ liệu từ các kiểu khác tương ứng về kiểu số nguyên, số thực và xâu kí tự.
- Các lệnh `int()`, `float()` không thực hiện xâu là biểu thức toán.



1. Mỗi lệnh sau sẽ trả lại các giá trị nào?

- a) `str(150)`                      b) `int("1110")`                      c) `float("15.0")`

2. Lệnh nào sau đây sẽ báo lỗi?

- A. `int("12.0")`                      B. `float(13+1)`                      C. `str(17.001)`

#### Hoạt động 4 Nhập dữ liệu kiểu số nguyên hoặc số thực từ bàn phím

Dữ liệu nhập từ bàn phím bằng lệnh `input()` luôn là xâu kí tự nên muốn nhập dữ liệu đầu vào là số nguyên hay số thực thì phải làm thế nào?



Nếu cần nhập số nguyên thì sau khi nhập giá trị số cần dùng lệnh `int()` để chuyển đổi sang kiểu số nguyên như sau:

```
>>> n = int(input("Nhập số tự nhiên: "))
Nhập số tự nhiên: 13
```

Nếu cần nhập số thực thì sau khi nhập giá trị số cần dùng lệnh `float()` để chuyển đổi sang kiểu số thực như sau:

```
>>> x = float(input("Nhập số thực x: "))
```



Dùng lệnh `x = input("Nhập số x: ")` để nhập số cho biến `x` là đúng hay sai?



## THỰC HÀNH. Nhập dữ liệu từ bàn phím bằng lệnh `input()`.

**Nhiệm vụ 1.** Viết chương trình nhập lần lượt ba số tự nhiên `m`, `n`, `p`, sau đó in ra tổng của ba số này.

**Hướng dẫn.** Cần thực hiện lần lượt ba lệnh nhập các số `m`, `n`, `p`. Chú ý cách nhập số nguyên cần dùng lệnh `int()` để chuyển đổi dữ liệu nhập từ bàn phím. Chương trình có thể viết như sau:

```
m = int(input("Nhập số nguyên m: "))
n = int(input("Nhập số nguyên n: "))
p = int(input("Nhập số nguyên p: "))
print("Tổng ba số đã nhập là", m+n+p)
```

**Nhiệm vụ 2.** Viết chương trình nhập họ tên, sau đó nhập tuổi của học sinh. Chương trình đưa ra thông báo, ví dụ: Bạn Nguyễn Hoà Bình 15 tuổi.

**Hướng dẫn.** Cần thực hiện hai lệnh nhập dữ liệu, một lệnh nhập tên học sinh, lệnh thứ hai nhập tuổi, sau đó thông báo ra màn hình. Chú ý khi nhập tuổi cần chuyển đổi dữ liệu.

```
ten = input("Nhập họ tên học sinh: ")
tuoi = int(input("Nhập tuổi: "))
print("Bạn", ten, tuoi, "tuổi")
```



## LUYỆN TẬP

1. Những lệnh nào trong các lệnh sau sẽ báo lỗi?

a) `int("12+45")`      b) `float(123.56)`      c) `float("123,5.5")`

2. Vì sao khi nhập một số thực cần viết lệnh `float(input())`?



## VẬN DỤNG

1. Viết chương trình nhập giá trị `ss` là số giây từ bàn phím. Thông báo ra màn hình thời gian `ss` giây này sau khi đổi thành thời gian tính bằng ngày, giờ, phút, giây.

2. Viết chương trình nhập ba số thực dương `a`, `b`, `c` và tính chu vi, diện tích của tam giác có độ dài các cạnh là `a`, `b`, `c` ( $a, b, c > 0$  và thoả mãn bất đẳng thức tam giác).

Gợi ý: Công thức Heron tính diện tích tam giác:  $S = \sqrt{p(p-a)(p-b)(p-c)}$  với  $p$  là nửa chu vi tam giác.