

# Python Cheat Sheet: Biến, Biểu Thức & Câu Lệnh

## 1. Bản chất Hằng số & Biến

### Hằng số (Constants):

- Khái niệm logic: Giá trị không đổi.
- Python không ép buộc hằng số.
- Quy ước:** Viết hoa toàn bộ.
- VD: PI = 3.14159 [cite: 5, 6]

### Biến (Variables):

- Là tên **tham chiếu** tới đối tượng trong bộ nhớ (Names reference objects)[cite: 7, 8].
- Gán = Tạo liên kết, không phải bỏ giá trị vào ô nhớ[cite: 9].

### Aliasing (Lưu ý):

- Nhiều biến cùng trỏ tới 1 object.
- Cẩn thận với object *mutable* (có thể thay đổi)[cite: 12].

## 2. Quy tắc Đặt tên (PEP 8)

### Ký tự hợp lệ:

- Chữ cái, số, gạch dưới (\_).
- Không** bắt đầu bằng số[cite: 14].
- Phân biệt hoa/thường. Tránh từ khóa (if, for...)[cite: 15].

### Phong cách (Style Conventions): [cite: 16-19]

- Biến/Hàm:** snake\_case (vd: my\_variable)
- Lớp (Class):** PascalCase (vd: MyClass)
- Hằng:** ALL\_CAPS (vd: MAX\_SIZE)

## 3. Kiểu dữ liệu (Data Types)

### Kiểu cơ bản: [cite: 22]

- int, float, bool, str, NoneType

### Cấu trúc dữ liệu: [cite: 23]

- list, tuple, dict, set...

### Kiểm tra & Chuyển đổi: [cite: 24-26]

```
type(x)           # Xem kiểu
isinstance(x, int) # Kiểm tra True/False
int('3')          # -> 3
float('4.5')      # -> 4.5
str(10)           # -> '10'
```

Lưu ý: Ép kiểu sai có thể gây ValueError.

## 5. Nhập liệu (Input)

### Cơ chế:

- input() luôn trả về string (chuỗi)[cite: 34].
- Cần ép kiểu để tính toán.

### Xử lý lỗi (Validation): [cite: 35-36]

```
s = input('Nhập số: ')
try:
    n = int(s)
except ValueError:
    print('Không phải số nguyên!')
```

## 6. Thứ tự ưu tiên (Precedence)

Thứ tự thực hiện từ cao xuống thấp [cite: 38-39]:

- ( ) (Ngoặc)
- \*\* (Lũy thừa)
- +x, -x (Đầu dương/âm)
- \*, /, //, % (Nhân chia)
- +, - (Cộng trừ)
- <<, >, &, ■, | (Bitwise)
- ==, !=, <, >, <=, >= (So sánh)
- not, and, or (Logic)

Mẹo: Dùng ngoặc () để code rõ ràng hơn. [cite: 40]

## 4. Toán tử (Operators)

### Số học: [cite: 28]

- + - \* /
- // (Chia lấy nguyên), % (Chia lấy dư)
- \*\* (Lũy thừa)

### So sánh: ==, !=, <, >, <=, >= [cite: 29]

### Logic: [cite: 31]

- and, or, not (có tính chất short-circuiting).

### Membership: [cite: 32]

- in (dùng cho chuỗi, list, set).