

# PYTHON

## CHEATSHEET CHƯƠNG 6

### CẤU TRÚC CƠ BẢN

#### Chuỗi là một chuỗi

**(Sequence):** Một chuỗi ký tự được sắp xếp theo thứ tự

**Lấy độ dài:** Dùng hàm `len()`

để lấy số lượng ký tự trong chuỗi

#### Duyệt qua chuỗi

**(Traversal):** Có thể dùng vòng lặp `for` để xử lý từng ký tự trong chuỗi

### PHÂN TÍCH PHƯƠNG PHÁP

#### Chức năng: Phương thức `s.split()`

được dùng để chia một chuỗi thành một **List** (danh sách) các chuỗi con. Đây là một công cụ cực kỳ hữu ích khi làm việc với dữ liệu có cấu trúc, ví dụ như một dòng trong tệp văn bản.

**Hoạt động mặc định:** Nếu không cung cấp đối số nào cho `split()`,

, nó sẽ tự động tìm kiếm và chia chuỗi dựa trên **bất kỳ chuỗi khoảng trắng nào** (spaces, tabs, newlines).

Kết quả là một danh sách chứa các "từ" (word) trong chuỗi. **Sử dụng với ký tự phân cách (Delimiter):**

Bạn có thể truyền một chuỗi làm đối số để chỉ định ký tự phân cách

### THAO TÁC VỚI CHUỖI

**Slices (Lát cắt):** Lấy một phần của chuỗi bằng cách sử dụng ký hiệu dấu ngoặc vuông

`[n:m]`

Kết quả bao gồm ký tự từ vị trí `n` đến **trước** vị trí `m`

`s[:n]`

là từ đầu đến trước `n`

`s[m:]`

là từ

`m`

đến cuối

#### Chuỗi là bất biến (Immutable):

Bạn **không thể thay đổi** một ký tự trong chuỗi sau khi nó đã được tạo. Để thay đổi chuỗi, bạn phải tạo một chuỗi mới

#### Toán tử

`in`

Là một toán tử **Boolean** kiểm tra xem một chuỗi con có xuất hiện trong một chuỗi khác hay không

### TOÁN TỬ FORMAT

#### Toán tử %

Cho phép bạn xây dựng một chuỗi, sau đó chèn các giá trị vào các vị trí đã đánh dấu

### GỠ LỖI

Các vấn đề thường gặp có thể liên quan đến

**không nhớ**

rằng chuỗi là **bất biến**

, hoặc các vấn đề với các lát cắt và chỉ mục (index)

### PHƯƠNG THỨC STRING

**Tính bất biến:** Chuỗi không thể thay đổi;

các phương thức **trả về chuỗi mới** hoặc

giá trị **Chuyển đổi:**

`s.upper()`

: Chữ hoa

`s.lower()`

: Chữ thường. **Tìm kiếm/Thay thế:**

`s.find(sub)`

: Index của

`sub`

(hoặc -1)

`s.replace(old, new)`

Thay thế **Xóa khoảng trắng:**

`s.strip()`

: Xóa khoảng trắng ở hai đầu.

`s.lstrip()`

: Xóa bên trái

`s.rstrip()`

: Xóa bên phải **Kiểm tra Boolean:**

`s.startswith(prefix)`

`s.endswith(suffix)`

**Toán tử**

`in`

Kiểm tra sự tồn tại của chuỗi con