

Author: Lê Trung Thực

Email: thuclt12a@gmail.com / Skype: vpar.thuclt

PHƯƠNG THỨC XỬ LÝ CHUỖI

STT	Phương thức và Miêu tả
1	Phương thức capitalize() Viết hoa chữ cái đầu tiên của chuỗi
2	Phương thức center(width, fillchar) Trả về một chuỗi mới, trong đó chuỗi ban đầu đã được cho vào trung tâm và hai bên đó là các fillchar sao cho tổng số ký tự của chuỗi mới là width
3	Phương thức count(str, beg= 0,end=len(string)) Đếm xem chuỗi str này xuất hiện bao nhiêu lần trong chuỗi string hoặc chuỗi con của string nếu bạn cung cấp chỉ mục ban đầu start và chỉ mục kết thúc end
4	Phương thức decode(encoding='UTF-8',errors='strict') Giải mã chuỗi bởi sử dụng encoding đã cho
5	Phương thức encode(encoding='UTF-8',errors='strict') Trả về phiên bản chuỗi đã được mã hóa của chuỗi ban đầu. Nếu có lỗi xảy ra, thì chương trình sẽ tạo một ValueError trừ khi các lỗi này được cung cấp với ignore hoặc replace
6	Phương thức endswith(suffix, beg=0, end=len(string)) Xác định xem nếu chuỗi string hoặc chuỗi con đã cho của string (nếu bạn cung cấp chỉ mục bắt đầu beg và chỉ mục kết thúc end) kết thúc với hậu tố suffix thì trả về true, nếu không thì phương thức này trả về false
7	Phương thức expandtabs(tabsize=8) Mở rộng các tab trong chuỗi tới số khoảng trống đã cho; mặc định là 8 space cho mỗi tab nếu bạn không cung cấp tabsize

Author: Lê Trung Thực

Email: thuclt12a@gmail.com / **Skype:** vpar.thuclt

8	Phương thức find(str, beg=0 end=len(string)) Xác định xem chuỗi str có xuất hiện trong chuỗi string hoặc chuỗi con đã cho của string (nếu bạn cung cấp chỉ mục bắt đầu beg và chỉ mục kết thúc end), nếu xuất hiện thì trả về chỉ mục của str, còn không thì trả về -1
9	Phương thức index(str, beg=0, end=len(string)) Tương tự như find(), nhưng tạo ra một ngoại lệ nếu str là không được tìm thấy
10	Phương thức isalnum() Trả về true nếu chuỗi có ít nhất một ký tự và tất cả ký tự là chữ số. Nếu không hàm sẽ trả về false
11	Phương thức isalpha() Trả về true nếu chuỗi có ít nhất 1 ký tự và tất cả ký tự là chữ cái. Nếu không phương thức sẽ trả về false
12	Phương thức isdigit() Trả về true nếu chuỗi chỉ chứa các chữ số, nếu không là false
13	Phương thức islower() Trả về true nếu tất cả ký tự trong chuỗi là ở dạng chữ thường, nếu không là false
14	Phương thức isnumeric() Trả về true nếu một chuỗi dạng Unicode chỉ chứa các ký tự số, nếu không là false
15	Phương thức isspace() Trả về true nếu chuỗi chỉ chứa các ký tự khoảng trắng whitespace, nếu không là false
16	Phương thức istitle() Trả về true nếu chuỗi là ở dạng titlecase, nếu không là false

Author: Lê Trung Thực

Email: thuclt12a@gmail.com / **Skype:** vpar.thuclt

17	Phương thức isupper() Trả về true nếu tất cả ký tự trong chuỗi là chữ hoa
18	Phương thức join(seq) Nối chuỗi các biểu diễn chuỗi của các phần tử trong dãy seq thành một chuỗi
19	Phương thức len(string) Trả về độ dài của chuỗi
20	Phương thức ljust(width[, fillchar]) Trả về một chuỗi mới, trong đó có chuỗi ban đầu được căn chỉnh vào bên trái và bên phải là các fillchar sao cho tổng số ký tự là width
21	Phương thức lower() Chuyển đổi tất cả chữ hoa trong chuỗi sang kiểu chữ thường
22	Phương thức lstrip() Xóa tất cả các khoảng trống trắng ban đầu (leading) trong chuỗi
23	Phương thức maketrans() Trả về một bảng thông dịch được sử dụng trong hàm translate
24	Phương thức max(str) Trả về ký tự chữ cái lớn nhất từ chuỗi str đã cho
25	Phương thức min(str) Trả về ký tự chữ cái nhỏ nhất từ chuỗi str đã cho

Author: Lê Trung Thực

Email: thuclt12a@gmail.com / **Skype:** vpar.thuclt

26	Phương thức <code>replace(old, new [, max])</code> Thay thế tất cả sự xuất hiện của old trong chuỗi với new với số lần xuất hiện max (nếu cung cấp)
27	Phương thức <code>rfind(str, beg=0,end=len(string))</code> Tương tự hàm <code>find()</code> , nhưng trả về chỉ mục cuối cùng
28	Phương thức <code>rindex(str, beg=0, end=len(string))</code> Giống <code>index()</code> , nhưng trả về chỉ mục cuối cùng nếu tìm thấy
29	Phương thức <code>rjust(width,[, fillchar])</code> Trả về một chuỗi mới, trong đó có chuỗi ban đầu được căn chỉnh vào bên phải và bên trái là các fillchar sao cho tổng số ký tự là width
30	Phương thức <code>rstrip()</code> Xóa bỏ tất cả các khoảng trống trắng ở cuối (trailing) của chuỗi
31	Phương thức <code>split(str="", num=string.count(str))</code> Chia chuỗi theo delimiter đã cho (là space nếu không được cung cấp) và trả về danh sách các chuỗi con; nếu bạn cung cấp num thì chia chuỗi thành num chuỗi con
32	Phương thức <code>splitlines(num=string.count('\n'))</code> Trả về một List gồm tất cả các dòng trong chuỗi, và tùy ý xác định các ngắt dòng (nếu num được cung cấp và là true).
33	Phương thức <code>startswith(str, beg=0,end=len(string))</code> Xác định xem chuỗi hoặc chuỗi con (nếu bạn cung cấp chỉ mục bắt đầu beg và chỉ mục kết thúc end) có bắt đầu với chuỗi con str không, nếu có trả về true, nếu không là false
34	Phương thức <code>strip([chars])</code>

Author: Lê Trung Thực

Email: thuclt12a@gmail.com / **Skype:** vpar.thuclt

	Thực hiện cả hai phương thức lstrip() và rstrip() trên chuỗi
35	Phương thức swapcase() Đảo ngược kiểu của tất cả ký tự trong chuỗi
36	Phương thức title() Trả về một bản sao của chuỗi trong đó tất cả ký tự đầu tiên của tất cả các từ là ở kiểu chữ hoa.
37	Phương thức translate(table, deletechars="") Trả về một bản sao đã được thông dịch của chuỗi
38	Phương thức upper() Chuyển đổi các chữ thường trong chuỗi thành chữ hoa
39	Phương thức zfill (width) Trả về một chuỗi mới, trong đó bao gồm chuỗi ban đầu và được đệm thêm với các số 0 vào bên trái sao cho tổng ký tự là width
40	Phương thức isdecimal() Kiểm tra xem chuỗi đó có phải là decimal hay không