1. ***capitalize()*** : trả về một bản sao của chuỗi với chữ cái đầu tiên được viết hoa.
2. ***casefold()*** : chuyển đổi tất cả các ký tự của chuỗi đó thành chữ thường và trả về một chuỗi mới.
3. ***center()*** : trả về một chuỗi mới, trong đó chuỗi ban đầu đã được cho vào trung tâm và hai đầu của chuối mới là các ký tự fillchar sao cho tổng số ký tự của chuỗi mới là width. Ký tự cần điền Fillchar mặc định là một khoảng trống.
4. ***count()*** : trả về số lần xuất hiện của chuỗi con trong khoảng [start, end]. 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.
5. ***encode()*** **:** 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. ***endswith()*** : 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. ***expandtabs()*** : 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. Trả về một bản sao của chuỗi với tab đã được mở rộng (chẳng hạn bởi cac khoảng trống).
8. ***find()*** : 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. ***format()*** : chuyển chuỗi thành các thành phần trong hàm để in ra màn hình bằng {}

***10) index()*** : tạo ra một ngoại lệ nếu str là không được tìm thấy. 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, nếu không tìm thấy thì trả về một.

***11) isalnum()*** :  trả về true nếu chuỗi có ít nhất một ký tự và tất cả ký tự là chữ hoặc số. Nếu không hàm sẽ trả về false.

***12)*** ***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.

***13) isascii()*** : thức trả về True nếu tất cả các ký tự đều là ký tự ascii (az).

***14) isdecimal()*** : trả về true nếu một chuỗi dạng Unicode chỉ chứa các ký tự thập phân, nếu không là false.

***15) isdigit() :*** trả về true nếu chuỗi chỉ chứa các chữ số, nếu không là false.

***16) isdentifier() :*** Nó được sử dụng để kiểm tra xem một chuỗi có phải là một định danh hợp lệ hay không. Một định danh là một từ, tên biến hoặc tên hàm mà Python sử dụng để xác định một thực thể trong chương trình.

***17) islower()*** : trả về true nếu tất cả ký tự trong chuỗi là ở dạng chữ thường, nếu không là false.

***18) isnumeric()*** : trả về true nếu một chuỗi dạng Unicode chỉ chứa các ký tự số, nếu không là false.

***19) isoprintable()*** : là một hàm xác định xem tất cả các ký tự trong một chuỗi có thể được in ra màn hình hay không. Nó trả về giá trị True nếu tất cả các ký tự trong chuỗi là các ký tự in được, và False nếu có ít nhất một ký tự không thể in ra màn hình.

***20 isspace()*** : trả về true nếu chuỗi chỉ chứa các ký tự khoảng trắng whitespace, nếu không là false.

***21) istitle()*** : trả về true nếu chuỗi là ở dạng titlecase - ký tự đầu tiên của mỗi chữ trong chuỗi là chữ hoa, nếu không là false.

***22) isupper()*** : trả về true nếu tất cả ký tự trong chuỗi là chữ hoa.

***23) join()*** : nối chuỗi các biểu diễn chuỗi của các phần tử trong dãy seq thành một chuỗi.

***24) len()*** : kích thước chuỗi

***25) ljust()*** : 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.

***26)lower()*** : chuyển đối tất cả chữ hoa trong chuỗi sang kiểu chữ thường.

***27) lstrip()*** : trả về một bản sao của chuỗi ban đầu sau khi đã cắt tất cả các ký tự chars đã cung cấp từ phần đầu của chuỗi (các ký tự mặc định là khoảng trắng).