



# STRING





Xâu kí tự là chuỗi các kí tự được đặt giữa dấu nháy đơn hoặc nháy kép. Xâu trong Python được hỗ trợ một tập các công cụ xử lý cực kì mạnh mẽ.



## 1. Tạo string:



Để tạo string ta đặt nội dung của chuỗi giữa dấu nháy đơn hoặc dấu nháy kép. Chuỗi có thể có hoặc không có ký tự.

### EXAMPLE

```
s = '28tech' #single quotes
t = "python" #double quotes
#Multiline string
u = """
cuoc song
tuoi dep
"""

print(s, t)
print(u)
```

### OUTPUT

```
28tech python
cuoc song
tuoi dep
```

## 2. str() constructor:



Bạn có thể chuyển đổi hầu hết các đối tượng trong Python sang chuỗi ký tự bằng hàm khởi tạo str().


### EXAMPLE

```
s = str(100)
t = str(100.20)
u = str(10 + 28j)
print(type(s), type(t), type(u))
```

### OUTPUT

```
<class 'str'> <class 'str'> <class 'str'>
```

### 3. Truy cập kí tự thông qua chỉ số và duyệt xâu:

 Để truy cập vào các kí tự trong string, ta dùng chỉ số bắt đầu từ 0, ngoài ra Python còn hỗ trợ chỉ số âm (negative index).

Index	0	1	2	3	4	5
S	2	8	T	E	C	H
Neg index	-6	-5	-4	-3	-2	-1

#### EXAMPLE

```
s = "28TECH"
print(s[1])
print(s[-2])
```

#### OUTPUT

8  
C

### 3. Truy cập kí tự thông qua chỉ số và duyệt xâu:

Bạn có thể sử dụng string slicing tương tự như list

EXAMPLE

```
s = "28TECH"
print(s[1:4])
print(s[2:])
print(s[::-1])
```

OUTPUT

```
8TE
TECH
HCET82
```

Duyệt xâu

EXAMPLE

```
s = "28tech dsa"
for x in s :
    print(x, end = ' ')
print()
for i in range(len(s)):
    print(s[i], end = ' ')
```

OUTPUT

```
2 8 t e c h   d s a
2 8 t e c h   d s a
```

## 4. String is immutable:



Bạn có thể truy cập các kí tự trong str thông qua chỉ số nhưng bạn không thể thay đổi được các kí tự đó bằng cách gán nó thành các kí tự hoặc xâu khác.

### EXAMPLE

```
s = "28TECH"  
s[0] = "@"  
print(s)
```

### OUTPUT

```
TypeError: 'str' object does  
not support item assignment
```

## 5. Nối xâu:



Trong Python để nối 2 xâu bạn có thể sử dụng toán tử +, để nhân bản xâu bạn có thể dùng toán tử \*.

### EXAMPLE

```
s = "28tech"  
t = "python"  
s = s + " " + t  
print(s)  
a = "ab"  
b = a * 3  
print(b)
```

### OUTPUT

```
28tech python  
ababab
```

## 6. Hàm replace:



Hàm **replace()** được sử dụng để thay thế 1 xâu cũ bằng một xâu mới.



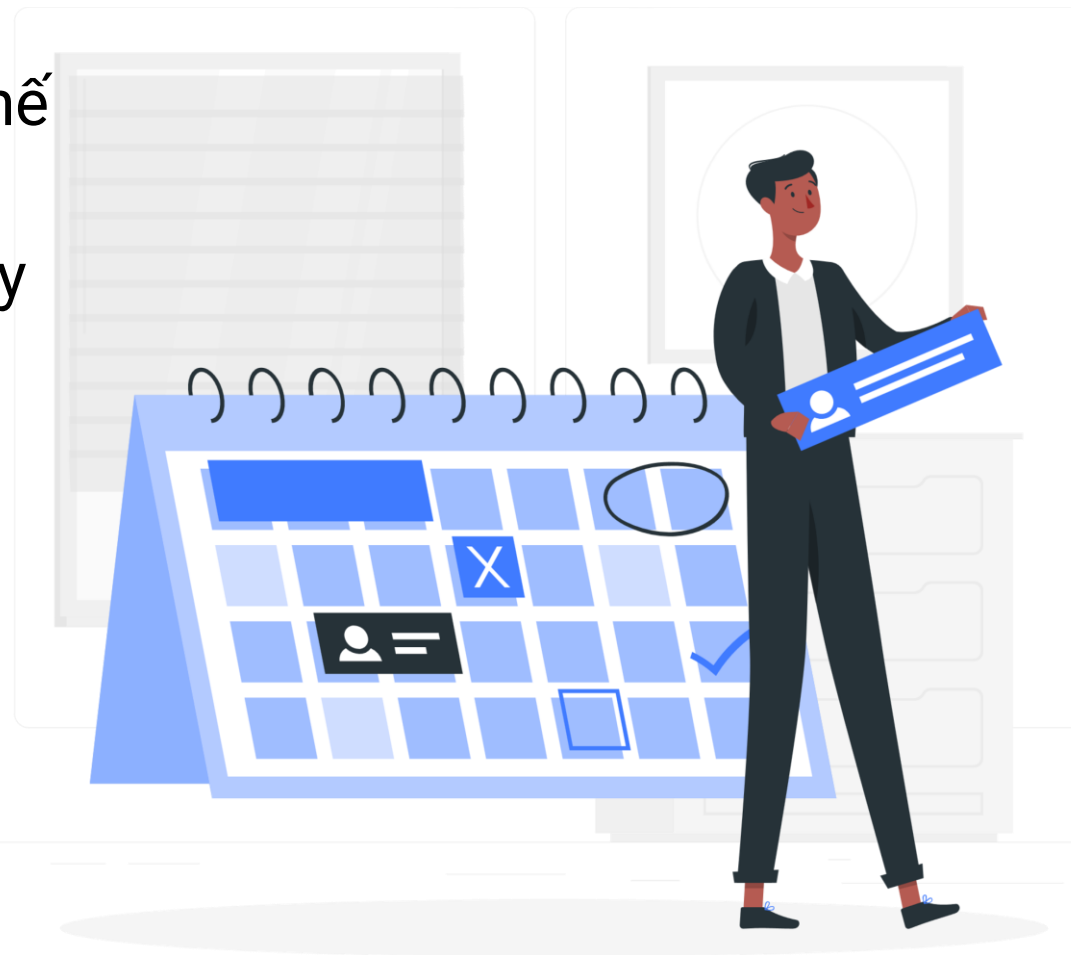
**Chú ý:** Các hàm xử lý của str không thay đổi chính nó mà trả về xâu tương ứng.

### EXAMPLE

```
s = "28tech python C++"  
t = s.replace("python", "java")  
print(s)  
print(t)
```

### OUTPUT

```
28tech python C++  
28tech java C++
```



## 7. Hàm `split()` và `join()`:



**`split()`:** Tách các từ trong xâu, mặc định nếu hàm này không có tham số sẽ tách theo dấu cách.

### EXAMPLE

```
s = "28tech    python  C++"  
a = s.split()  
print(a)  
t = "28tech,python,C++"  
b = t.split(",")  
print(b)
```

### OUTPUT

```
['28tech', 'python', 'C++']  
['28tech', 'python', 'C++']
```



**`join()`:** Kết hợp các từ rời rạc thành 1 xâu được phân cách bởi kí tự cho trước

### EXAMPLE

```
s = ["28tech", "python", "C++"]  
t = ','.join(s)  
print(t)  
u = '##'.join(s)  
print(u)
```

### OUTPUT

```
28tech,python,C++  
28tech##python##C++
```



## 8. Các hàm case conversion:

- **upper()**: Chuyển sang in hoa.
- **lower()**: Chuyển sang in thường.
- **capitalize()**: Viết hoa chữ cái đầu của xâu.
- **swapcase()**: Chuyển chữ hoa thành thường và ngược lại.
- **title()**: Viết hoa chữ cái đầu của từng từ, các kí tự còn lại viết thường

### EXAMPLE

```
s = "28TeCH jAVa PyThon"
s1 = s.upper()
print(s1)
s2 = s.lower()
print(s2)
s3 = s.swapcase()
print(s3)
s4 = s.title()
print(s4)
s5 = s.capitalize()
print(s5)
```

### OUTPUT

```
28TECH JAVA PYTHON
28tech java python
28tEch JavA pYtHON
28Tech Java Python
28tech java python
```

## 9. Kiểm tra xâu con:



Để kiểm tra xâu con có thể sử dụng **toán tử in** hoặc **hàm find**, hàm find sẽ trả về chỉ số nhỏ nhất của xâu con nếu nó tồn tại trong xâu lớn, trả về -1 nếu không tìm thấy xâu con.

### EXAMPLE

```
s = "28tech dsa 28tech"
if "28tech" in s:
    print('FOUND')
print(s.find("28tech"))
print(s.find("C++"))
```

### OUTPUT

```
FOUND
0
-1
```



## 10. String format :



Trong Python có 3 cách nhúng biến vào trong chuỗi ký tự.

**-% string formatting : %s : chuỗi ký tự, %d : số nguyên, %f : Số float**

**EXAMPLE**

```
s = '%s is %d years old' % ('CR7', 37)
t = 'Pi = %.2f' % (3.14159)
print(s)
print(t)
```

**OUTPUT**

```
CR7 is 37 years old
Pi = 3.14
```

## 10. String format:



Trong Python có 3 cách nhúng biến vào trong chuỗi kí tự.

### -str.format()

#### EXAMPLE

```
s = '{0} is {1} years old !'.format('CR7', 37)  
print(s)
```

#### OUTPUT

CR7 is 37 years old

### -f-String

#### EXAMPLE

```
name = "CR7"  
age = 37  
s = f'{name} is {age} years old !'  
print(s)
```

#### OUTPUT

CR7 is 37 years old