

## BÀI THỰC HÀNH SỐ 3: BIỂU THỨC CHÍNH QUY

**Mục tiêu:** làm quen với module “re”, là thư viện chuẩn trong Python, giải quyết các vấn đề liên quan đến biểu thức chính quy (btcq)

### I. Kiến thức:

Module “re” được sử dụng để tìm kiếm và thao tác với chuỗi, tương tự các module khác, cần import trước khi sử dụng module này.

```
>>> import re
>>>
```

### Các phương thức thông dụng trong thư viện “re”

TT	Phương thức	Mô tả
1	match	trả về true nếu một kết quả khớp được tìm thấy trong chuỗi
2	search	trả về đối tượng khớp nếu có một kết quả khớp được tìm thấy trong chuỗi
3	findall	trả về một danh sách chứa tất cả các kết quả khớp của một mẫu trong chuỗi
4	split	trả về một danh sách trong đó chuỗi đã được phân chia theo mỗi kết quả khớp
5	sub	tìm tất cả các chuỗi khớp và thay thế

### Các ký hiệu đại diện

Metacharacter	Mô tả	Ví dụ
[]	một tập các ký tự	"[a-z]"
\	ký tự đặc biệt	"\r"
.	bất kỳ ký tự nào xuất hiện ở một số nơi cụ thể	"Ja.v."
^	mẫu có mặt ở đầu chuỗi	"^Java"
\$	mẫu có mặt ở cuối chuỗi.	"viettuts"

*	không hoặc nhiều lần xuất hiện của một mẫu trong chuỗi.	"hello*"
+	một hoặc nhiều lần xuất hiện của một mẫu trong chuỗi.	"hello+"
{ }	số lần xuất hiện đã chỉ định của một mẫu trong chuỗi.	"java{2}"
	cái này hoặc cái kia (điều kiện or).	"python2 python3"
()	Nhóm các thành phần.	

**Ký tự đặc biệt**

Ký tự đặt biệt là các chuỗi có chứa \ theo sau là một trong các ký tự.

<b>Ký tự</b>	<b>Mô tả</b>
\A	ký tự được chỉ định có mặt ở đầu chuỗi.
\b	các ký tự được chỉ định có mặt ở đầu hoặc cuối chuỗi.
\B	các ký tự được chỉ định có mặt ở đầu chuỗi nhưng không ở cuối chuỗi.
\d	chuỗi chứa các chữ số [0-9].
\D	chuỗi không chứa các chữ số [0-9].
\s	chuỗi chứa bất kỳ ký tự khoảng trắng nào.
\S	chuỗi không chứa bất kỳ ký tự khoảng trắng nào.
\w	chuỗi chứa bất kỳ ký tự từ nào.
\W	chuỗi không chứa bất kỳ từ nào.
\Z	các ký tự được chỉ định ở cuối chuỗi.

**Set:** một set là một nhóm các ký tự được đưa ra bên trong một cặp dấu ngoặc vuông. Nó đại diện cho ý nghĩa đặc biệt.

STT	Set	Mô tả
1	[arn]	Trả về một kết quả khớp nếu chuỗi chứa bất kỳ ký tự nào được chỉ định trong tập hợp.
2	[a-n]	Trả về một kết quả khớp nếu chuỗi chứa bất kỳ ký tự nào từ a đến n.
3	[^arn]	Trả về một kết quả khớp nếu chuỗi chứa các ký tự ngoại trừ a, r và n.
4	[0123]	Trả về một kết quả khớp nếu chuỗi chứa bất kỳ chữ số nào được chỉ định.
5	[0-9]	Trả về một kết quả khớp nếu chuỗi chứa bất kỳ chữ số nào trong khoảng từ 0 đến 9.
6	[0-5][0-9]	Trả về một kết quả khớp nếu chuỗi chứa bất kỳ chữ số nào trong khoảng từ 00 đến 59.
10	[a-zA-Z]	Trả về một kết quả khớp nếu chuỗi chứa bất kỳ bảng chữ cái nào (chữ thường hoặc chữ hoa).

### 1. Sử dụng `findall()` để tìm các từ thỏa biểu thức chính quy:

Giả sử cần tìm tất cả các từ bắt đầu bằng ‘wo’ trong một đoạn text ngắn, phương thức `re.findall()` sẽ giúp thực hiện điều này, phương thức này có 2 đối số: biểu thức chính quy và chuỗi gốc

```
>>> wood = 'How much wood would a woodchuck chuck if a woodchuck could
chuck wood?'
>>> re.findall(r'wo\w+', wood)          # r'...' for raw string
['wood', 'would', 'woodchuck', 'woodchuck', 'wood']
>>>
```

Kết quả trả về là một danh sách gồm các từ thỏa biểu thức chính quy đã cho. Lưu ý: **`r'wo\w+'`** là biểu thức chính quy mô tả các từ bắt đầu bằng “wo” trong đó tiếp đầu ngữ “r” báo hiệu đây là biểu thức chính quy, “\w” là **một ký tự bất kỳ thuộc [a-zA-Z0-9\_]**, có thể là chữ, hoặc số, hoặc dấu gạch dưới.

Nếu không tìm thấy chuỗi thỏa yêu cầu, kết quả trả về là một danh sách rỗng. Ví dụ sau trả về các chuỗi có chứa ít nhất 1 ký tự ‘o’ hoặc ‘e’.

```
>>> re.findall(r'o+', wood)
['o', 'oo', 'o', 'oo', 'oo', 'o', 'oo']
>>> re.findall(r'e+', wood)
[]
```

Trường hợp tìm chuỗi không phân biệt chữ hoa và chữ thường: bổ sung thêm đối số thứ 3 **re.IGNORECASE** trong **findall()**

```
>>> foo = 'This and that and those'
>>> re.findall(r'th\w+', foo)
['that', 'those']
>>> re.findall(r'th\w+', foo, re.IGNORECASE)
['This', 'that', 'those']
>>>
```

S

## 2. *Thay thế tất cả các chuỗi thỏa biểu thức chính quy sử dụng phương thức **re.sub()***

Giả sử cần tìm tất cả các nguyên âm trong chuỗi **wood** và thay bằng dấu ‘-’

```
>>> wood
'How much wood would a woodchuck chuck if a woodchuck could chuck wood?'
>>> re.sub(r'[aeiou]+', '-', wood)      # 3 args: regex, replacer string, target string
'H-w m-ch w-d w-ld - w-dch-ck ch-ck -f - w-dch-ck c-ld ch-ck w-d?'
>>>
```

Ngoài ra, có thể xóa bỏ một phần chuỗi thỏa biểu thức chính quy bằng chuỗi rỗng ‘’

## 3. *Phương thức **re.compile()***

Trường hợp cần xem xét btcq ở các chuỗi khác nhau chúng ta nên tạo btcq như một object trong Python, dùng phương thức **re.compile()**

```
>>> myre = re.compile(r'\w+ou\w+')      # compiling myre as a reg ex
>>> myre.findall(wood)                  # calling .findall() directly on myre
['would', 'could']
>>> myre.findall('Colorless green ideas sleep furiously')
['furiously']
>>> myre.findall('The thirty-three thieves thought that they thrilled
the throne throughout Thursday.')
['thought', 'throughout']
```

## 4. *Kiểm tra một chuỗi có chứa btcq hay không?*

Phương thức **re.search()** tìm xem một chuỗi có chứa btcq hay không, sẽ thoát khi tìm thấy

chuỗi đầu tiên.

```
>>> re.search(r'e+', 'Colorless green ideas sleep furiously')
<_sre.SRE_Match object at 0x02D9CB48>
>>> re.search(r'e+', wood)
>>>
```

Tham khảo: <https://docs.python.org/3/library/re.html>

## II. Bài tập:

### A. *Đọc file txt, và xây dựng btcq để tìm và in:*

- 1) Các dòng bắt đầu bằng 't' hoặc 'h', và có chứa 're' (sử dụng phương thức re.match())
- 2) Các dòng có chiều dài tối thiểu 20 ký tự
- 3) Các dòng kết thúc bởi cặp dấu '?!'
- 4) Các dòng chứa những ký tự: a, r, s, m, l (không cần liên tục)
- 5) Nội dung file ko chứa các dấu ',' và '.'
- 6) Các dòng có chứa chữ "mouse"
- 7) Các từ có số chữ 'a' bất kỳ và theo sau bởi 'b'
- 8) Domain địa chỉ mail (ví dụ: abc@gmail.com, in gmail.com )
- 9) Nội dung giữa cặp tag <head> </head> (gợi ý: sử dụng open và read để mở và đọc file)

### B. *Viết chương trình kiểm tra một câu có đúng cú pháp hay không?*

- 1) Câu được bắt đầu bằng ký tự in hoa, theo sau bởi ký tự thường
- 2) Kết thúc bằng dấu chấm hoặc sau một ký tự in hoa
- 3) Các từ cách nhau bằng một khoảng trắng, không chấp nhận nhiều hơn 1 khoảng trắng liên tiếp
- 4) Không tồn tại hai ký tự liên tiếp viết hoa

### C. *Viết chương trình in ra các từ thỏa một trong các trường hợp sau:*

- 1) Các từ có chứa các ký tự thường 'a-z' và số từ '0-9'
- 2) Các từ có chứa ký tự 'a' theo sau bởi b (b xuất hiện ít nhất 0 lần)
- 3) Các từ bắt đầu bằng 'a', theo sau là ký tự bất kỳ và kết thúc bằng 'b'
- 4) Các từ chỉ chứa ký tự thường 'a-z' và '\_'
- 5) Các từ có chiều dài là 5
- 6) Các từ có chứa ký tự 'h'
- 7) Các từ bắt đầu là số từ '0-9'
- 8) Các từ có chứa dấu '\_' và thay bằng khoảng trắng
- 9) Có chứa định dạng mm-dd-yy, và chuyển thành định dạng dd-mm-yy

### D. *Sử dụng biểu thức chính quy để giải quyết bài tập ở bài thực hành số 1*