

Python Regex Sets & Ranges

If this Python Tutorial saves you
hours of work, please **whitelist it in**
your ad blocker 🙏 and

Donate Now

(<https://www.pythontutorial.net/donation/>)

to help us ❤️ pay for the web
hosting fee and CDN to keep the

website running.

Summary: in this tutorial, you'll learn how to use the sets and ranges to create patterns that match a set of characters.

Several characters or character sets inside square brackets `[]` mean matching for any character or character set among them.

Sets

For example, `[abc]` means any of three characters. `'a'`, `'b'`, or `'c'`. The `[abc]` is called a set. And you can use the set with regular characters to construct a search pattern.

For example, the following program uses the pattern `licen[cs]e` that matches both `license` and `licence` :

```
import re

s = 'A licence or license'

pattern = 'licen[cs]e'
matches = re.finditer(pattern, s)
```

```
for match in matches:
    print(match.group())
```

Output:

```
licence
license
```

The pattern `licen[cs]e` searches for:

- `licen`
- then one of the letters `[cs]`
- then `e` .

Therefore, it matches `license` and `licence` .

Ranges

When a set consists of many characters in e.g., from `a` to `z` or `1` to `9` , it'll tedious to list them in a set. Instead, you can use character ranges in square brackets. For example, `[a-z]` is a character in the range from `a` to `z` and `[0-9]` is a digit from `0` to `9` .

Also, you can use multiple ranges within the same square brackets. For example, `[a-z0-9]` has two ranges that match for a character that is either from `a` to `z` or a digit from `0` to `9` .

Similarly, you can use one or more character sets inside the square brackets like `[\d\s]` means a digit or a space character.

Likewise, you can mix the character with character sets. For example, `[\d_]` matches for a digit or an underscore.

Excluding sets & ranges

To negate a set or a range, you use the caret character (`^`) at the beginning of the set and range. For example, the range `[^0-9]` matches any character except a digit. It is the same as the character set `\D` .

Notice that regex also uses the caret (`^`) as an anchor that matches at the beginning of a string. However, if you use the caret (`^`) inside the square brackets, the regex will treat it as a negation operator, not an anchor.

The following example uses the caret (`^`) to negate the set `[aeoiu]` to match the consonants in the string `'Python'` :

```
import re

s = 'Python'

pattern = '[^aeoiu]'
matches = re.finditer(pattern, s)

for match in matches:
    print(match.group())
```

Output:

```
P
y
t
h
n
```

Summary

- A set or a range matches any single character or character set specified in square brackets [...].
- Use the caret (`^`) operator to negate a set or a range like `[^...]` .