

Python Lookahead

Python look ahead explained.

WE'LL COVER THE FOLLOWING ^

- Positive Lookahead
- Example
- Neagative Lookahead
- Example

Positive Lookahead

Python positive lookahead matches at a position where the pattern inside the lookahead can be matched. Matches only the position. It does not consume any characters or expand the match.

Example

Consider the following string:

```
begin:learner1:scientific:learner2:scientific:learner3:end
```

Positive lookahead assertion can help us to find all words followed by the word `scientific`.

```
import re

string = "begin:learner1:scientific:learner2:scientific:learner3:end"
print re.findall(r"(\w+)(?=scientific)", string)
```



Note the output `learner1` and `learner2`, but not `learner3`, which is followed by the word `:end`.

Neegative Lookahead

Similar to positive lookahead, except that negative lookahead only succeeds if the regex inside the lookahead fails to match.

Example

Let's now proceed to an example, where we find the word (**learner3**) followed by **end** .

```
import re

string = "begin:learner1:scientific:learner2:scientific:learner3:end"
print re.findall(r"(learner\d+)(?!:scientific)", string)
```



This matched all the words, not followed by the word **scientific** !