

Regular Expressions in Python

1. Importing the `re` Module:

```
import re
```

2. Basic Regex Functions:

- `re.match()`: Matches a pattern only at the start of the string.

Example:

```
result = re.match(r'Hello', 'Hello World')
```

- `re.search()`: Searches for the first occurrence of the pattern in the string.

Example:

```
result = re.search(r'World', 'Hello World')
```

- `re.findall()`: Finds all occurrences of a pattern in a string.

Example:

```
result = re.findall(r'\d+', 'My phone numbers are 12345 and 67890')
```

- `re.sub()`: Replaces all occurrences of a pattern in a string with a replacement.

Example:

```
result = re.sub(r'\d+', '###', 'My phone numbers are 12345 and 67890')
```

- `re.split()`: Splits the string by occurrences of the pattern.

Example:

```
result = re.split(r'\s+', 'Split by spaces in this sentence')
```

3. Special Characters:

- Dot ("."): Matches any character except newline.

Example: `re.findall(r'a.c', 'abc a1c a-c')`

- Caret ("^") and Dollar ("\$"): Matches the start or end of the string.

Example:

`result = re.search(r'^Hello', 'Hello World')`

`result = re.search(r'World$', 'Hello World')`

4. Quantifiers:

- Asterisk ("*"): Matches 0 or more occurrences.

Example: `re.findall(r'a*b', 'b ab aab aaab')`

- Plus ("+"): Matches 1 or more occurrences.

Example: `re.findall(r'a+b', 'b ab aab aaab')`

- Question Mark ("?"): Matches 0 or 1 occurrence.

Example: `re.findall(r'a?b', 'b ab aab aaab')`

5. Character Classes:

- "\d": Matches any digit.

Example: `re.findall(r'\d+', 'Phone numbers: 12345, 67890')`

- "\w": Matches any word character (alphanumeric + underscore).

Example: `re.findall(r'\w+', 'This is a test_123')`

6. Grouping and Backreferences:

- Grouping: (abc) groups patterns.

Example: `re.search(r'(Hello)', 'Hello World')`

- Backreferences: `"\1"` refers to the first captured group.

Example: `re.search(r'(\d+)-(\d+)', '123-456')`

7. Lookaheads and Lookbehinds:

- Positive Lookahead (`?=`): Ensures a pattern is followed by another pattern.

Example: `re.findall(r'\d(?:= dollars)', '5 dollars and 10 dollars')`

- Positive Lookbehind (`?<=`): Ensures a pattern is preceded by another pattern.

Example: `re.findall(r'(?<=\$)\d+', 'I paid $100 and $200')`

8. Flags:

- `re.IGNORECASE` (`re.I`): Ignores case sensitivity.

Example: `re.search(r'hello', 'Hello World', re.I)`

- `re.MULTILINE` (`re.M`): Treats each line independently for `^` and `$`.

Example: `re.findall(r'^\w+', text, re.M)`

- `re.DOTALL` (`re.S`): Makes the dot (`.`) match newline characters as well.