

`match = re.match(pattern, string, flags=0)`
searches only the beginning of the string
does not work with multiline strings

`match = re.search(pattern, string, flags=0)`
searches anywhere within string
flags specify special options (i.e. ignore case etc)
returns the first pattern instance in the string
works with multiline strings

`match.group()` : pattern instance
`match.group(0)` : pattern instance
`match.group(i)` : i = 1 to number of groups in the pattern: returns the i.th group instance
`match.span(i)` : i = 1 to number of groups in the pattern: returns the i.th group instance's span

`match.start()` : pattern instance start index
`match.end()` : pattern instance end index

`re.findall(pattern, string, flags=0)`
pulls out all instances of the pattern

`re.split(pattern, string, maxsplit=0, flags=0)`
splits the string with a given pattern with pattern instance not included

P.S `re.split` & `re.findall` can be used for the same use case
where they have different patterns

`re.sub(pattern, replacement_string, string)` (sub: substitute)

For each match instance of the pattern, `re.sub()` replaces the instance with `replacement_string`.

`re.sub()` can operate on a multiline string
`re.sub()` can receive lambda expressions in place of `replacement_string`
