Regular Expressions

In [1]:
```
import re
text = 'The quick brown fox jumped over the lazy dog'
```

In [2]:
```
match = re.search('quick', text)
match.start()
```

Out[2]: 4

In [3]:
```
match.end()
```

Out[3]: 9

In [4]:
```
match = re.match('quick', text)
print match
```

```
None
```

In [5]:
```
match = re.match('.*quick', text)
print match
```

```
<_sre.SRE_Match object at 0x247b718>
```

In [6]:
```
match = re.search('brown ([a-z]+) jumped', text)
match
```

Out[6]: <_sre.SRE_Match at 0x2486030>

In [7]:
```
match.groups()
```

Out[7]: ('fox',)

In [8]:
```
match = re.search('brown ([a-z]+)', text)
match
```

Out[8]: <_sre.SRE_Match at 0x2486120>

In [9]:
```
match.groups()
```

Out[9]: ('fox',)

In [10]:
```
match = re.search('brown ([a-z]+?)', text)
match
```

Out[10]: <_sre.SRE_Match at 0x2486198>

In [11]:
```
match.groups()
```

Out[11]: ('f',)

In [12]:
```
s = 'foo bar baz baz'
re.search('foo (.*) baz', s).groups()
```

Out[12]:
```
('bar baz',)
```

In [13]:
```
s = 'foo bar baz baz'
re.search('foo (.*?) baz', s).groups()
```

Out[13]:
```
('bar',)
```

In [14]:
```
m = re.search('brown (?P<animal>[a-z]+) jumped', text)
m.groups()
```

Out[14]:
```
('fox',)
```

In [15]:
```
m.groupdict()
```

Out[15]:
```
{'animal': 'fox'}
```

In [16]:
```
re_time = re.compile('(\\d{2}):(\\d{2}):(\\d{2})')
```

In [17]:
```
re_time.match('01:23:45').groups()
```

Out[17]:
```
('01', '23', '45')
```

You can make regex strings more readable by using 'raw strings':

In [18]:
```
re_time = re.compile(r'(\d{2}):(\d{2}):(\d{2})')
```

In [19]:
```
re_time.match('01:23:45').groups()
```

Out[19]:
```
('01', '23', '45')
```

You can also use 'verbose mode':

In [20]:
```
re_time = re.compile(r'''
(\d{2}) # hour
:
(\d{2}) # minute
:
(\d{2}) # second
''', re.VERBOSE)
```

In [21]:
```
re_time.match('01:23:45').groups()
```

Out[21]:
```
('01', '23', '45')
```

In [22]:
```
text = '''The quick
brown fox
jumped over
the lazy dog'''
```

In [23]:
```
print re.search('quick brown', text)
```

```
None
```

In [24]:
```
print re.search('quick( |\n)brown', text)
```

```
<_sre.SRE_Match object at 0x2486468>
```

In [25]:
```
print re.search('quick.brown', text)
```

```
None
```

In [26]:
```
print re.search('quick.brown', text, re.DOTALL)
```

```
<_sre.SRE_Match object at 0x247b718>
```

In [27]:
```
for m in re.finditer('(\w+)', text):
    print m.group(1)
```

```
The
quick
brown
fox
jumped
over
the
lazy
dog
```

In [28]:
```
print re.sub('quick', 'slow', text)
```

```
The slow
brown fox
jumped over
the lazy dog
```

In [29]:
```
def sub_length(match):
    return str(len(match.group(1)))

print re.sub('(\w+)', sub_length, text)
```

```
3 5
5 3
6 4
3 4 3
```

In [30]:
```
print re.sub('(\w+)', sub_length, text, 4)
```

```
3 5
5 3
jumped over
the lazy dog
```

In [31]:
```
large_text = open('Regular Expressions.ipynb').read()
```

In [32]:
```
regular_split = large_text.split()
```

In [33]:
```
regular_split[:10]
```

Out[33]:
```
['{',
 '"metadata":',
 '{',
 '"name":',
 '"Regular',
 'Expressions"',
 '},',
 '"nbformat":',
 '2,',
 '"worksheets":']
```

In [34]:
```
better_split = re.split('\W*', large_text)
better_split[:10]
```

Out[34]:
```
['',
 'metadata',
 'name',
 'Regular',
 'Expressions',
 'nbformat',
 '2',
 'worksheets',
 'cells',
 'cell_type']
```

Exercises

- Write a function that finds all integers in a file using regular expressions
- Write a function that finds all capitalized words in a file
- Write a function that replaces all instances of `'<br>'` in a file with `'<br/>'`