String Pattern Matching

Regular Expression

- It is a special text string used for a search pattern. It is useful for extracting information from text like code, files, log, spreadsheets or even documents.
- Python has a built-in package called re Module, which helps to work with Regular Expressions.
- It is required to import the re Module before using it. e.g. `import re`

Regular Expression functions

- `match()`: checks for a match only at the beginning of the string.
- `search()`: checks for a match anywhere in the string.
- `findall()`: checks for all the match in the string and returns the list.
- `split()`: gives a list where the string has been split at each match.
- `sub()`: replaces one or many matches with a string.

Metacharacters

Metacharacters are characters with specific meaning.

<table>
<thead>
<tr>
<th>Character</th>
<th>Description</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>[ ]</td>
<td>A set of characters</td>
<td>&quot;[a-m]&quot;</td>
</tr>
<tr>
<td>\</td>
<td>Signals a special sequence (can also be used to escape special characters)</td>
<td>&quot;\d&quot;</td>
</tr>
<tr>
<td>.</td>
<td>Any character (except newline character)</td>
<td>&quot;he..o&quot;</td>
</tr>
<tr>
<td>^</td>
<td>Starts with</td>
<td>&quot;^hello&quot;</td>
</tr>
<tr>
<td>$</td>
<td>Ends with</td>
<td>&quot;world$&quot;</td>
</tr>
<tr>
<td>*</td>
<td>Zero or more occurrences</td>
<td>&quot;aix*&quot;</td>
</tr>
<tr>
<td>+</td>
<td>One or more occurrences</td>
<td>&quot;aix+&quot;</td>
</tr>
<tr>
<td>{ }</td>
<td>Exactly the specified number of occurrences</td>
<td>&quot;al{2}&quot;</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Either or</td>
</tr>
<tr>
<td>( )</td>
<td>Capture and group</td>
<td></td>
</tr>
</tbody>
</table>
Example

```python
import re

txt = "The dog chase the cat"
x = re.search("^The.*cat$", txt)
if (x):
    print("String found!")
else:
    print("String not found")
```

**Special Sequences**

A special sequence is a \ followed by one of the characters.

<table>
<thead>
<tr>
<th>Character</th>
<th>Description</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>\A</td>
<td>Check if the specified characters are at the beginning of the string</td>
<td>&quot;\AThe&quot;</td>
</tr>
<tr>
<td>\b</td>
<td>Check the specified characters are at the beginning or at the end of a word (the &quot;r&quot; in the beginning is making sure that the string is being treated as a &quot;raw string&quot;)</td>
<td>r&quot;\bain&quot; r&quot;ain\b&quot;</td>
</tr>
<tr>
<td>\B</td>
<td>Check the specified characters are present, but NOT at the beginning (or at the end) of a word (the &quot;r&quot; in the beginning is making sure that the string is being treated as a &quot;raw string&quot;)</td>
<td>r&quot;\Bain&quot; r&quot;ain\B&quot;</td>
</tr>
<tr>
<td>\d</td>
<td>Checks the string contains digits (numbers from 0-9)</td>
<td>&quot;\d&quot;</td>
</tr>
<tr>
<td>\D</td>
<td>Checks the string DOES NOT contain digits</td>
<td>&quot;\D&quot;</td>
</tr>
<tr>
<td>\s</td>
<td>Checks the string contains a white space character</td>
<td>&quot;\s&quot;</td>
</tr>
<tr>
<td>\S</td>
<td>Checks the string DOES NOT contain a white space character</td>
<td>&quot;\S&quot;</td>
</tr>
<tr>
<td>\w</td>
<td>Checks the string contains any word characters (characters from a to z, digits from 0-9, and the underscore _ character)</td>
<td>&quot;\w&quot;</td>
</tr>
<tr>
<td>\W</td>
<td>Checks the string DOES NOT contain any word characters</td>
<td>&quot;\W&quot;</td>
</tr>
<tr>
<td>\Z</td>
<td>Checks the specified characters are at the end of the string</td>
<td>&quot;Spain\Z&quot;</td>
</tr>
</tbody>
</table>

Example

```python
import re

txt = "The rain in Train"
x = re.search("ai", txt)
print(x)    #this will print an object
```

Output

<re.Match object; span=(5, 7), match='ai'>
Example
import re

#searches all the words starting with r
txt = "rain in train"
x = re.findall("r\w+", txt)
print(x)

#searches the occurrence of words starting with r, in the beginning of string.
txt = "rain in train"
x = re.match("r\w+", txt)
print(x.group())

#searches the occurrence of words starting with r, anywhere of string.
txt = "pain in train"
x = re.search("r\w+", txt)
print(x)

#split the string, wherever the occurrence of word found.
txt = "The rain in Spain"
x = re.split("ai", txt)
print(x)

Output
['rain', 'rain']
<re.Match object; span=(0, 4), match='rain'>
None
<re.Match object; span=(9, 13), match='rain'>
['The r', 'n in Sp', 'n']

Example
import re

#splits the string wherever the whitespace found

txt = "The rain in train"
x = re.split("s", txt, 1)
print(x)
x = re.split("s", txt, 2)
print(x)
x = re.split("s", txt)
print(x)

Output
['The', 'rain in train']
['The', 'rain', 'in train']
['The', 'rain', 'in', 'train']