

Regular Expressions

Reminder: Commonly used special symbols in Python regular expressions

Symbol	Meaning
.	matches any character
+	1 or more
*	0 or more
()	capture group
\d	digit
\D	non-digit
\s	whitespace
\S	non-whitespace
\w	alphanumeric
\W	non-alphanumeric

Character sets:

[abcd] matches the letters a, b, c, d

Character sets:

[abcd] matches the letters a, b, c, d

String: gray

Regex: gr[abcd]y

Character sets:

`[abcd]` matches the letters a, b, c, d

String: `gray`

Regex: `gr[abcd]y`

Match: `gray`

Character sets:

[abcd] matches the letters a, b, c, d

String: grey

Regex: gr[abcd]y

Character sets:

[abcd] matches the letters a, b, c, d

String: grey

Regex: gr[abcd]y

Match: *Does not match!*

A character set matches exactly one letter

String: -ATGGTCTA-

Regex: -[ATGC]-

A character set matches exactly one letter

String: -ATGGTCTA-

Regex: -[ATGC]-

Match: *Does not match!*

A character set matches exactly one letter

String: -ATGGTCTA-

Regex: -[ATGC]+-

A character set matches exactly one letter

String: `-ATGGTCTA-`

Regex: `-[ATGC]+-`

Match: `-ATGGTCTA-`

Negative character sets:

`[^abcd]` matches anything but a, b, c, d

String: gray

Regex: `gr[^abcd]y`

Negative character sets:

`[^abcd]` matches anything but a, b, c, d

String: `gray`

Regex: `gr[^abcd]y`

Match: *Does not match!*

Negative character sets:

`[^abcd]` matches anything but a, b, c, d

String: `grey`

Regex: `gr[^abcd]y`

Negative character sets:

`[^abcd]` matches anything but a, b, c, d

String: `grey`

Regex: `gr[^abcd]y`

Match: `grey`