# Regular Expressions

# Metacharacters

Metacharacters are characters with a special meaning:

| Character | Description | Example |
|---|---|---|
| [] | A set of characters | "[a-m]" |
| \ | Signals a special sequence (can also be used to escape special characters) | "\d" |
| . | Any character (except newline character) | "he..o" |
| ^ | Starts with | "^hello" |
| $ | Ends with | "planet$" |
| * | Zero or more occurrences | "he.*o" |
| + | One or more occurrences | "he.+o" |
| ? | Zero or one occurrences | "he.?o" |
| {} | Exactly the specified number of occurrences | "he.{2}o" |
| \| | Either or | "falls\|stays" |
| () | Capture and group | |

# Special Sequences

A special sequence is a `\` followed by one of the characters in the list below, and has a special meaning:

| Character | Description | Example |
|---|---|---|
| \A | Returns a match if the specified characters are at the beginning of the string | "\AThe" |
| \b | Returns a match where the specified characters are at the beginning or at the end of a word<br>(the "r" in the beginning is making sure that the string is being treated as a "raw string") | r"\bain"<br>r"ain\b" |
| \B | Returns a match where the specified characters are present, but NOT at the beginning (or at the end) of a word<br>(the "r" in the beginning is making sure that the string is being treated as a "raw string") | r"\Bain"<br>r"ain\B" |
| \d | Returns a match where the string contains digits (numbers from 0-9) | "\d" |
| \D | Returns a match where the string DOES NOT contain digits | "\D" |
| \s | Returns a match where the string contains a white space character | "\s" |
| \S | Returns a match where the string DOES NOT contain a white space character | "\S" |
| \w | Returns a match where the string contains any word characters (characters from a to Z, digits from 0-9, and the underscore _ character) | "\w" |
| \W | Returns a match where the string DOES NOT contain any word characters | "\W" |
| \Z | Returns a match if the specified characters are at the end of the string | "Spain\Z" |

# Sets

A set is a set of characters inside a pair of square brackets `[]` with a special meaning:

| Set | Description |
| --- | --- |
| [arn] | Returns a match where one of the specified characters ( `a` , `r` , or `n` ) is present |
| [a-n] | Returns a match for any lower case character, alphabetically between `a` and `n` |
| [^arn] | Returns a match for any character EXCEPT `a` , `r` , and `n` |
| [0123] | Returns a match where any of the specified digits ( `0` , `1` , `2` , or `3` ) are present |
| [0-9] | Returns a match for any digit between `0` and `9` |
| [0-5][0-9] | Returns a match for any two-digit numbers from `00` and `59` |
| [a-zA-Z] | Returns a match for any character alphabetically between `a` and `z` , lower case OR upper case |
| [+] | In sets, `+` , `*` , `.` , `\|` , `()` , `$` , `{}` has no special meaning, so `[+]` means: return a match for any `+` character in the string |