Некоторые наборы символов, например, [0-9] или [^0-9] и другие довольно часто используются на практике, поэтому им были назначены специальные краткие формы:

|  |  |
| --- | --- |
| Символ | Значение |
| . | Соответствует любому символу, кроме символа переноса строки (‘\n’). Но, если установлен флаг re.DOTALL, то точка соответствует вообще любому символу в тексте. Однако, если она записана внутри символьного класса [.], то воспринимается как символ точки. |
| \d | Соответствует любой цифре, если используется кодировка Юникода. Если же установлен флаг re.ASCII, то диапазону цифр [0-9]. |
| \D | Соответствует любому не цифровому символу для Юникода или символьному классу [^0-9] при установленном флаге re.ASCII |
| \s | Для Юникода – любой пробельный символ. Для re.ASCII – символьному классу [ \t\n\r\f\v] |
| \S | Для Юникода – любой не пробельный символ. Для re.ASCII – символьному классу [^ \t\n\r\f\v] |
| \w | Для Юникода – любой символ слова. При флаге re.ASCII – набору символов [a-zA-Z0-9\_] |
| \W | Для Юникода – любой не символ слова. При флаге re.ASCII – набору символов [^a-zA-Z0-9\_] |

Квантификаторы можно записывать и в кратких формах, например:

* {m} – повторение выражения ровно m раз (эквивалент {m,m});
* {m,} – повторения от m и более раз;
* {, n} – повторения не более n раз.

Разумеется, для последних двух форм также можно использовать минорный режим:

{m,}?    {,n}?

Для квантификаторов {0,} и {1,} существуют специальные символы:

* ? – от нуля до одного (аналог {0,1});
* \* – от нуля и до «бесконечности» (в действительности, большого числа – от 32767), соответствует квантификатору {0,};
* + – от единицы и до «бесконечности» (также большого числа – от 32767), соответствует квантификатору {1,}.

Все эти сокращения также можно использовать в минорном режиме:

??    \*?    +?

Обратите внимание, проверки не являются частью совпадения строки по шаблону, они лишь проверяют определенные условия, поэтому сам по себе символ \b в строке text не ищется, а определяется граница слова в шаблоне, где он записан.

В общем случае, для регулярных выражений доступны следующие проверки:

|  |  |
| --- | --- |
| Символ | Описание |
| ^ | Начало текста (с флагом re.MULTILINE – начало строки) |
| $ | Конец текста (с флагом re.MULTILINE – позиция перед символом переноса строки \n) |
| \A | Начало текста |
| \b | Граница слова (внутри символьных классов [] соответствует символу BACKSPACE) |
| \B | Граница не слова (зависим от флага re.ASCII) |
| \Z | Конец текста |
| (?=exp) | Проверка на совпадение с выражением exp продолжения строки. При этом позиция поиска не смещается на выражение exp (опережающая проверка). |
| (?!exp) | Проверка на несовпадение с выражением exp продолжения строки. (Также опережающая проверка). |
| (?<=exp) | Проверка на совпадение с выражением exp хвоста уже обработанной (проверенной) строки. Она также называется *позитивной ретроспективной проверкой*. |
| (?<!exp) | Проверка на несовпадение с выражением exp хвоста уже обработанной (проверенной) строки. Еще она называется *негативной ретроспективной проверкой*. |

Набор флагов, которые можно назначать к регулярным выражениям:

|  |  |
| --- | --- |
| Флаг | Описание |
| re.A или re.ASCII | При этом флаге проверки \b, \B, \s, \S, \w и \W действуют так, как если бы они применялись к тексту, содержащему только символы ASCII (по умолчанию используется Юникод re.U / re.UNICODE и лучше оставаться в этом режиме) |
| re.I или re.IGNORECASE | Проверка без учета регистра символов |
| re.M или  re.MULTILINE | Влияет на проверки ^ и $. Начало ^ считается началом строки (сразу после символа \n или начало текста). Конец $ считается в позиции перед \n (или конец строки) |
| re.S или  re.DOTALL | При установке этого флага символ . также включает символ перевода строки \n. |
| re.X или  re.VERBOSE | Позволяет включать в регулярные выражения пробелы и комментарии |
| re.DEBUG | Включает режим отладки при компиляции регулярного выражения |

Флаги можно указывать и непосредственно внутри выражения, используя синтаксис:

(?flags),

где flags – один или несколько флагов. Причем, их имена, следующие:

* a – то же самое, что и re.ASCII;
* i – соответствует re.IGNORECASE;
* m – для re.MULTILINE;
* s – для re.DOTALL;
* x – для re.VERBOSE.