Регулярные выражения в Java

Регулярные выражения (Regular expressions) – это способ описания набора строк, на основе общих характеристик каждой строки в этом наборе.

Регулярные выражения могут быть использованы для поиска, редактирования и манипулирования текстом и данными. Для этого нужно изучить специфический синтаксис, который не имеет отношения к синтаксису Java (так же, как и к другим языкам).

Регулярные выражения могут быть различны по сложности, однажды усвоив как они устроены можно создавать любые выражения.

В Java регулярные выражения реализованы в пакете java.util.regex.

Существует множество реализаций технологии регулярных выражений, такие как grep, Perl, Tcl, Python, PHP, awk.

Синтаксис java.util.regex API наиболее похож на Perl.

java.util.regex API в основном представлен такими классами:

* Объект **Pattern** – это скомпилированное представление регулярного выражения. Pattern не предоставляет публичных конструкторов, для создания вызывается статический метод, который возвращает объект. Этот метод принимает регулярное выражение как аргумент;
* Объект **Matcher** – это движок, который интерпретирует Pattern и выполняет операции согласно входной строке. Так же, как и Pattern он не имеет публичных конструкторов;
* Объект **PatternSyntaxException** это неконтролируемое исключение, которое говорил об ошибке в регулярном выражении;

Квантом входных и выходных данных для регулярных выражений является **строка**. Важно понимать, что процессор выражений принимает строки и возвращает строки, а значит нужно учитывать разделены ли входные данные на строки.

**Преимущества и недостатки регулярных выражений**

|  |  |
| --- | --- |
| **Преимущества** | **Недостатки** |
| * Большая гибкость и огромные возможности по манипулированию строками; * Очень короткий код выполняющий огромную работу; | * Работает медленнее по сравнению с доступом к строке или массиву; * Существование многих диалектов, относительная сложность изучения; * В случае ошибки выбрасывается неконтролируемое исключение в рантайме, если шаблон неправильный. Нет возможности проверить на этапе компиляции; |

Можно посоветовать не использовать регулярные выражения для простого поиска и замены в строке. Но если нужен анализ по многим критериям, то регулярные выражения незаменимы, поскольку поддерживаются большинством языков программирования на уровне библиотек.

1. <https://docs.oracle.com/javase/tutorial/essential/regex/>