

- Инвертированный символьный класс представляет собой сокращённое обозначение обычного символьного класса, означающее все символы, кроме перечисленных. Следовательно, выражение `['^x']` означает не «всё что угодно кроме x», а любой символ «отличный от x». Различие тонкое, но важное. Например, в первой интерпретации совпадение будет найдено в пустой строке, чего быть не должно.
- Полезный ключ `-i` игнорирует различия в регистре букв при сравнении (стр.36).

Даже те метасимволы, которые мы рассмотрели, способны принести несомненную пользу, но настоящие возможности регулярных выражений начинаются с применения квантификаторов.

## Необязательные элементы

Допустим, вам понадобилось найти слово `color` или `colour`. Эти два слова почти одинаковы, только в одном есть буква `u`, а в другом её нет. Регулярное выражение `['colou?r']` позволяет найти любой из этих вариантов. Метасимвол `?` (вопросительный знак) означает «необязательный символ». Он ставится после символа, который может находиться в данной позиции текста, но наличие которого не требуется для успешного совпадения.

В отличие от других упоминавшихся метасимволов, вопросительный знак относится только к символу, расположенному непосредственно перед ним. Выражение `['colou?r']` интерпретируется как «`c`, затем `o`, затем `l`, затем `u`, затем `?`, затем `r`».

Сопоставление `['?']` с текстом всегда оказывается успешным: иногда ему соответствует символ `u` в тексте, иногда не соответствует. Собственно, главная особенность необязательного элемента `?` состоит в том, что совпадение для него находится всегда. Это вовсе не означает, что совпадение будет найдено для любого регулярного выражения, содержащего метасимвол `?`. Например, при поиске в строке `semicolon` будут найдены совпадения для `['colo']` и `['u?']` (соответственно `colo` и ничего). Тем не менее завершающее `['']` не находится, из-за чего `['colou?r']` в конечном счёте и не совпадает в строке `semicolon`.

Рассмотрим другой пример. Представьте, что вам требуется найти дату — четвёртое июля, в которой месяц обозначается `July` или `Jul`, а число — `fourth`, `4th` или просто цифра `4`. Конечно, можно просто воспользоваться выражением `['(July|Jul) • (fourth|4th|4)']`, но давайте рассмотрим другие варианты выражения той же идеи.

Во-первых, `['(July|Jul)']` сокращается до `['(July?)']`. Вы видите, что это одно и то же? При удалении `['']` круглые скобки перестают быть нужными. Вреда от скобок не будет, но без них выражение `['July? • (fourth|4th|4)']` становится чуть более понятным. Получается `['July? • (fourth|4th|4)']`.