Szukanie i zamiana

Poziom: podstawy **Dotyczy: Vim 9.x** **Czas: ~20 min**

Szukanie fragmentu tekstu i podmiana go w całym pliku to jedna z najczęstszych czynności administratora w plikach konfiguracyjnych. Vim robi to bez wychodzenia z edytora i bez otwierania osobnego narzędzia.

Skróty użyte na tej stronie (1)

Skrót	Co znaczy
IP	Internet Protocol: adres komputera w sieci; w elektryce: stopień ochrony obudowy (Ingress Protection)

Fiszka

TYLKO ODCZYT

Szukanie w przód: /wzorzec

```
/localhost
```

`/` wchodzi w tryb szukania w dół pliku, wpisujemy szukany tekst i naciskamy Enter. Kursor skacze na pierwsze dopasowanie od bieżącej pozycji w dół, z zawijaniem na początek pliku, gdy dojdzie do końca.

TYLKO ODCZYT

Szukanie w tył: ?wzorzec

```
?localhost
```

`?` robi to samo, tylko szuka w górę pliku od bieżącej pozycji. Przydatne, gdy wiemy, że szukany fragment jest wcześniej w pliku, niż stoi kursor.

◫◫ TYLKO ODCZYT

Kolejne dopasowanie: n N

```
n
N
```

`n` powtarza ostatnie wyszukiwanie w tym samym kierunku, w jakim je rozpoczęliśmy, `N` w kierunku przeciwnym. Działa niezależnie od tego, czy szukaliśmy przez `/`, czy przez `?`.

◫◫ TYLKO ODCZYT

Szukanie słowa pod kursorem: *

```
*
#
```

`*` szuka w dół kolejnego wystąpienia słowa, na którym akurat stoi kursor, bez wpisywania go ręcznie. `#` szuka tego samego słowa w górę. Szybszy sposób na sprawdzenie, gdzie jeszcze w pliku pojawia się nazwa zmiennej czy usługi.

◫◫ TYLKO ODCZYT

Zamiana w jednej linii: :s

```
:s/stare/nowe/
```

`:s` (substitute) zamienia w bieżącej linii pierwsze dopasowanie `stare` na `nowe`. Bez `g` na końcu zamienia tylko jedno wystąpienie w linii, nawet jeśli jest ich więcej.

◫◫ TYLKO ODCZYT

Zamiana w całym pliku: :%s/stare/nowe/g

```
:%s/localhost/docker01/g
```

`%` na początku oznacza cały plik, `g` (global) na końcu oznacza wszystkie wystąpienia w każdej linii, nie tylko pierwsze. Bez `g` zamieni tylko jedno dopasowanie w każdej linii, co jest częstym powodem wrażenia, że „zamiana zadziałała tylko połowicznie".

TYLKO ODCZYT

Zamiana z potwierdzeniem: /gc

```
:%s/localhost/docker01/gc
```

Dokłada potwierdzenie przy każdym dopasowaniu: `y` zamienia, `n` pomija, `a` zamienia wszystkie pozostałe bez dalszych pytań, `q` przerywa. Bezpieczniejsze, gdy wzorzec może trafić w coś niezamierzonego.

Przykład:

```
replace with docker01 (y/n/a/q/l/^E/^Y)?
```

TYLKO ODCZYT

Zamiana w zakresie linii: :N,Ms/a/b/g

```
:10,40s/localhost/docker01/g
```

Ogranicza zamianę do linii od 10 do 40, przydatne w dużych plikach, gdzie ten sam wzorzec powtarza się w niepowiązanych sekcjach i nie chcemy dotknąć wszystkich naraz.

Z życia wzięte

Z ŻYCIA WZIĘTE

Ola migruje konfigurację testową z serwera `pg01` na nowy adres i musi podmienić wszystkie wystąpienia starego hostname w pliku `docker-compose.yaml`. Zamiast przeklikiwać każdą linię ręcznie, wpisuje `:%s/pg01-stary/pg01/gc` i przechodzi przez kilkanaście dopasowań, potwierdzając każde osobno, bo dwa z nich okazują się komentarzem, który ma zostać bez zmian.

Jak to działa

> **Pojęcie: wyrażenie regularne.** Wzorzec, który opisuje nie jeden konkretny tekst, tylko całą rodzinę pasujących do niego ciągów znaków, na przykład dowolny numer portu albo dowolny adres IP. Vim rozumie własną odmianę wyrażeń regularnych, zbliżoną do wariantu rozszerzonego. Pełne omówienie trzech odmian wyrażeń regularnych i ich różnic jest na stronie Wyrażenia regularne: BRE, ERE, PCRE, tu wystarczy znajomość zwykłego tekstu bez znaków specjalnych, jak w przykładach powyżej.

Częste błędy

Objaw	Przyczyna	Co robimy
Zamiana działa tylko w części linii, w których wystąpień jest kilka	brak flagi `g` na końcu, `:s` bez niej zamienia tylko pierwsze dopasowanie w linii	dopisujemy `g`: `:%s/stare/nowe/g`
Zamiana dotyka linii, których nie chcieliśmy ruszać	użyliśmy `%` (cały plik) zamiast zakresu	ograniczamy do konkretnych linii: `:10,40s/stare/nowe/g`
`/wzorzec` nie znajduje nic, mimo że tekst na pewno jest w pliku	literówka we wzorcu albo szukanie w złym kierunku (`/` zamiast `?`)	sprawdzamy pisownię, próbujemy `?` zamiast `/`

Ćwiczenia

1. Chcemy zamienić każde wystąpienie `10.20.10.42` na `10.20.10.43` w całym pliku, bez pytania o potwierdzenie. Jakiej komendy użyjemy?

Odpowiedź

`:%s/10.20.10.42/10.20.10.43/g`. `%` obejmuje cały plik, `g` zamienia wszystkie wystąpienia w każdej linii, nie tylko pierwsze.

1. Dlaczego warto czasem dopisać `c` do komendy zamiany, skoro wydłuża to pracę o potwierdzanie każdego dopasowania?

Odpowiedź

Bo wzorzec bywa szerszy, niż nam się wydaje, i trafia też w miejsca, których nie chcieliśmy zmieniać, na przykład komentarze albo fragmenty innych nazw zawierające ten sam ciąg znaków. Potwierdzenie każdego dopasowania pozwala pominąć te przypadkowe trafienia,

zamiast poprawiać je ręcznie po fakcie.

Powiązane: Edycja: kasowanie, kopiowanie, wklejanie, cofanie · Wyrażenia regularne: BRE, ERE, PCRE · Makra, rejestry, bloki i inne sztuczki

Źródła: `man 1 vim` · `:help pattern.txt`, `:help change.txt` w samym Vimie

Revision #30
Created 2026-09-27 11:19:47 UTC by Tomasz Konieczny
Updated 2026-10-26 14:07:26 UTC by Tomasz Konieczny