

Prošireni regularni izrazi

Veličina zapisa regularnog izraza brzo raste kako raste složenost regularnog jezika. Radi skraćivanja zapisa i povećanja čitljivosti, uvode se tzv. **prošireni regularni izrazi** koji predstavljaju kraće zapise nekih osnovnih regularnih izraza.

Prošireni regularni izraz	Regularni izraz
$[a_1a_2\dots a_n]$	$a_1 a_2 \dots a_n$
$[\^a_1a_2\dots a_n]$	Svi karakteri iz azbuke bez $\{a_1, a_2, \dots, a_n\}$
$[a_1-a_2]$	Disjunkcija karaktera od a_1 do a_2 iz ASCII tabele
<code>\w</code>	$[A-Za-z0-9_]$
<code>\W</code>	$[\^A-Za-z0-9_]$
<code>\d</code>	$[0-9]$
<code>\D</code>	$[\^0-9]$
<code>\s</code>	$[\ \backslash t \backslash r \backslash n \backslash f]$ tj. sve moguće bjeline
<code>\S</code>	$[\^ \backslash t \backslash r \backslash n \backslash f]$
<code>.</code>	$[\^ \backslash n]$ tj. sve osim novog reda
a^+	aa^* tj. jedno ili više pojavljivanja slova a
$a^?$	ϵa tj. nula ili jedno pojavljivanje slova a
$a\{n\}$	$aa\dots a$ (a se ponavlja n puta)
$a\{n, m\}$	$aa\dots a$ ($n \leq$ broj pojavljivanja slova $a \leq m$)
$a\{n, \}$	$aa\dots a$ ($n \leq$ broj pojavljivanja slova a)

Karakteri koji imaju specijalno značenje u okviru karakterske klase su: -
(ukoliko se nađe između 2 simbola), \,] i ^.

Karakteri koji imaju specijalno značenje van karakterske klase su: *, +, ?, |, . ,
\, (,), [,...

Za **oduzimanje** specijalnog značenja koristimo karakter \ (**backslash**).

Gramzivost ili pohlepnost - poklapanje izraza se vrši u poslednjem trenutku kada je to moguće. Gramzivi operatori: *, +, ?, { }. Oduzimanje gramzivosti se može postići korišćenjem operatora ? nakon gramzivog operatora.

Nestrpljivost ili lenjo izračunavanje - ako možemo u disjunkciji prepoznati prvi disjunkt u tekstu, drugi se uopšte ne razmatra.

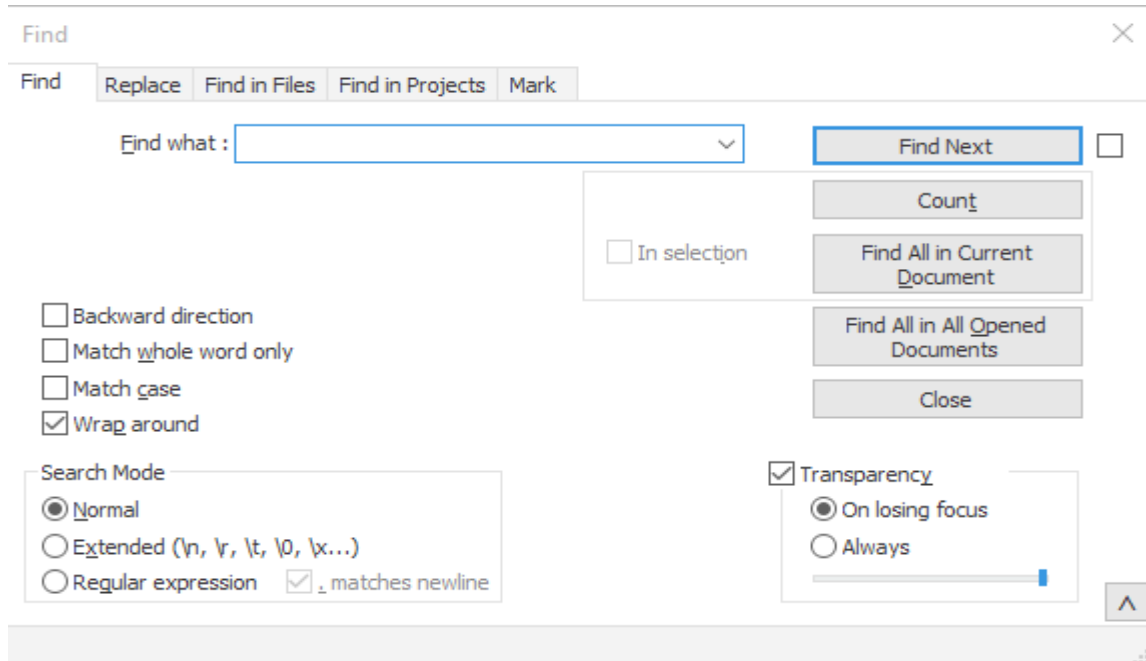
Sidra ne prepoznaju nijedan karakter, već se koriste prilikom definisanja pozicije na kojoj treba prepoznati tekst. Primeri:

^ - treba izvršiti poklapanje na početku teksta (ako je uključen *flag multiline* treba izvršiti poklapanje na početku linije)

\$ - treba izvršiti poklapanje na kraju teksta (ako je uključen *flag multiline* treba izvršiti poklapanje na kraju linije)

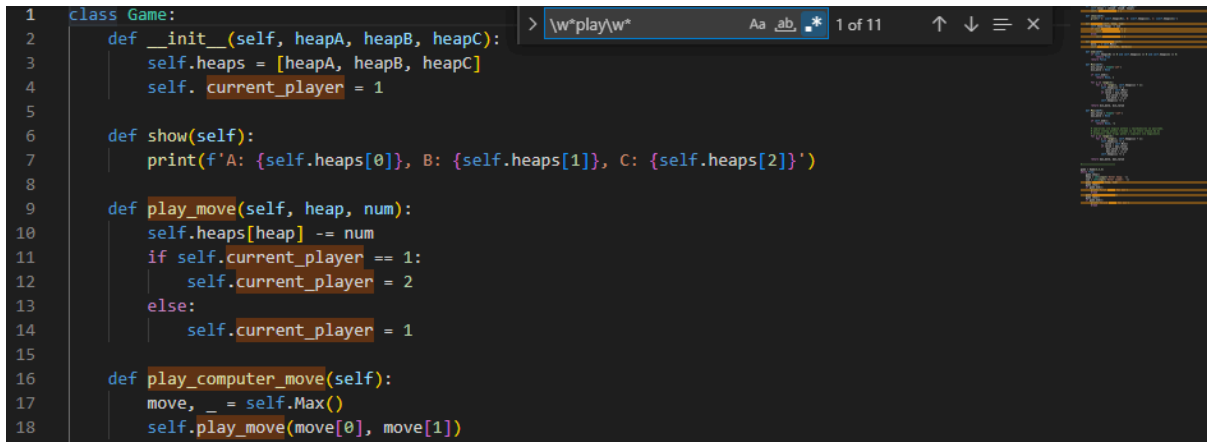
\b - koristi se kao granica za reči (riječ posmatramo kao \w+ pa je granica bilo koji karakter iz klase \W)

Upotreba regularnih izraza u Notepad++



Komanda **ctrl+F** omogućava pretraživanje dokumenta. Ukoliko želimo da vršimo pretragu koristeći regularne izraze prvi korak je podešavanje *Search Mode* na *Regular expression*. Nakon toga u okviru praznog polja *Find what* unosimo željeni regularni izraz. Opcija *Find Next* nam omogućava da nađemo sledeće poklapanje, opcija *Count* da prebrojimo koliko poklapanja ima u trenutnom dokumentu, opcija *Find All in Current Document* pronalazi sva poklapanja boji ih i ispisuje u kojoj su liniji pronađeni...

Upotreba regularnih izraza u Visual Studio Code



```
1 class Game:
2     def __init__(self, heapA, heapB, heapC):
3         self.heaps = [heapA, heapB, heapC]
4         self.current_player = 1
5
6     def show(self):
7         print(f'A: {self.heaps[0]}, B: {self.heaps[1]}, C: {self.heaps[2]}')
8
9     def play_move(self, heap, num):
10        self.heaps[heap] -= num
11        if self.current_player == 1:
12            self.current_player = 2
13        else:
14            self.current_player = 1
15
16    def play_computer_move(self):
17        move, _ = self.Max()
18        self.play_move(move[0], move[1])
```

The search bar at the top right contains the text: `\w*play\w*`. The search results on the right show matches for `play_move` and `play_computer_move` in the code.

Komanda `ctrl+F` omogućava pretraživanje dokumenta. U okviru prozora za unos odgovarajućeg izraza treba uključiti opciju `.*` koja omogućava korišćenje regularnih izraza.