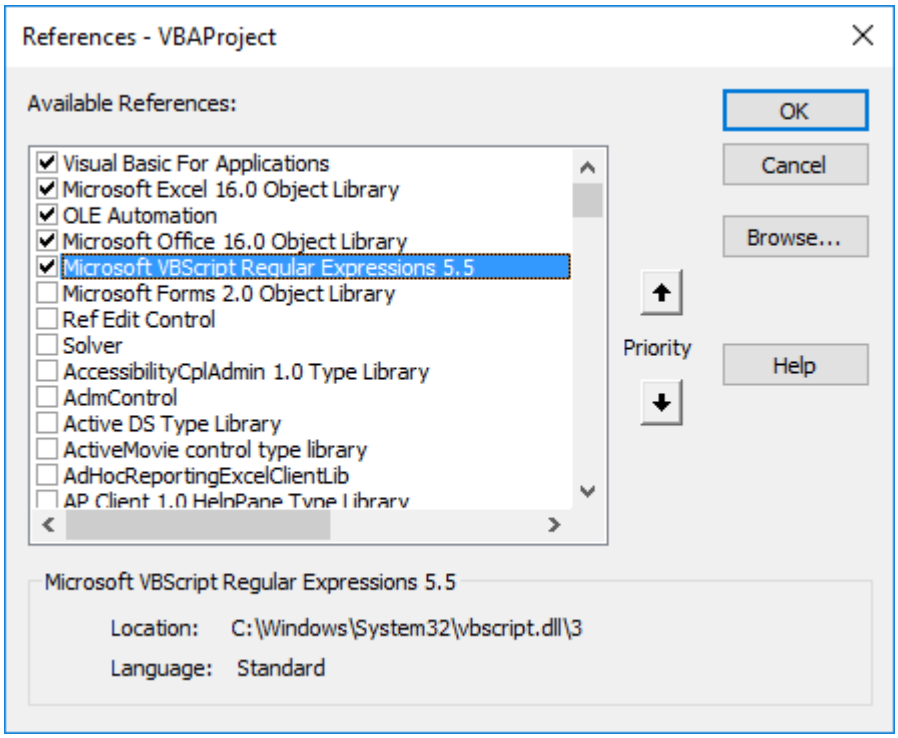
Regulární výrazy v excelu

Regulární výraz (zkracováno na regexp, regex či jen RE podle anglického regular expression) je řetězec popisující celou množinu řetězců, konkrétně regulární jazyk. Tolik Wikipedie. Nejčastější využití je vyhledávání textu nebo manipulace s textem. Pojďme se podívat na využití v Excelu.

Abychom mohli regulární výrazy v Excelu používat, nejdříve je nutné připojit knihovnu. To provedete následujícím způsobem.

- Otevřete VBA editor (Alt+F11) nebo na kartě Vývojář klikněte na ikonu Visual Basic.
- Ve VBA editoru přejděte v menu do Tools - Preferences.
- Vyhledejte a zaškrtněte **Microsoft VBScript Regular Expressions 5.5**
- Potvrďte klávesou OK.



Teď k vlastnímu příkladu. Použiji vzorová data z účetnictví ve třech sloupcích. V prvním sloupci jsou čísla účtů. V druhém jsou počáteční stavy a ve třetím jsou koncové stavy. Pro vytvoření kontrolních sestav (cashflow) bývají v účetních programech použity následující definice P21 + P22 + P25 - P252 - K321. Písmeno P zde představuje data ve sloupci počáteční stavy a písmeno K ve sloupci koncové stavy. Takže definice říká, sečti všechna data v počátečních stavech všech účtů začínajících 21 a k tomu přičti data počátečních stavů účtů začínajících 22 atd. Takových definic použiji v příkladu několik a vždy se liší. Definice samozřejmě lze editovat a tím měnit výsledný výpočet. Úkolem bude pomocí VBA a v něm použitého regulárního výrazu výpočet zprovoznit.

Č. účtu	Počátek	Konec
211100	12 400	14 070
221100	3 707 173	3 469 527
223100	15 288	15 288
225100	36 224	24 366
252100	56 120	21 250
255100	12 330	18 990
372100	36 840	36 840
374100	1 650	1 680
461100	100 000	100 000
469100	2 425 290	2 425 290
479102	1 691 450	1 691 450
479500	336 000	336 000
541100	98 000	112 580
543200	12 580	12 580
546500	120 350	98 260
549100	4 054	4 054
551150	13 630	53 630
641100	80 950	10 950
661101	713 833	789 833

Generuj sestavu

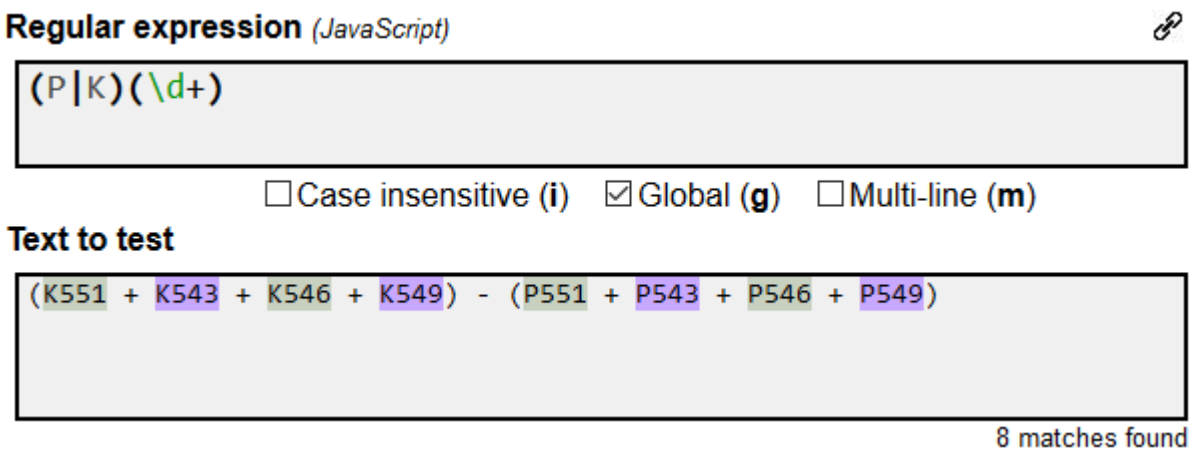
Řádek	Text	Skutečnost	Definice výpočtu
1.	Stav peněžních prostředků na začátku účetního období	3 771 085	P21 + P22 + P25 + P26 + P29 - P252 - P255 - P257
2.	Odpisy dlouhodobého majetku a pohledávek	17 910	(K551 + K543 + K546 + K549) - (P551 + P543 + P546 + P549)
3.	Zisk/ztráta z běžné činnosti před zdaněním	8 580	((K541 - K641 + K561 - K661) - (P541 - P641 + P561 - P661))
4.	Změna stavu dlouhodobých závazků	6 690	(K255 + K372 + K374 + K46 + K47) - (P255 + P372 + P374 + P46 + P47)
5.	Konečný stav peněžní prostředků pro kontrolu	3 523 251	K21 + K22 + K25 + K26 + K29 - K252 - K255 - K257

Na Internetu je mnoho informací o regulárních výrazech. Uvádím odkaz na dokumentaci od Microsoftu Regular

Expression Language - Quick Reference, kde lze následně stáhnout v PDF. Zajímavý článek je na stránce stackoverflow.com. Pomocí regulárního výrazu ve VBA budu potřebovat najít všechny texty, kde je uveden znak představující sloupec s výpočtem (P nebo K) a za ním následující sekvenci čísel, které představují první čísla účtu. Následně pak v makru pomocí textové funkce dále zpracuji. Regulární výraz, který texty vyhledá, bude vypadat takto: `(P|K)(\d+)`. Jak to funguje:

- `(P|K)` - najde subřetězec odpovídající P nebo K.
- `(\d+)` - najde všechny subřetězce odpovídající číslici 1 a víckrát,

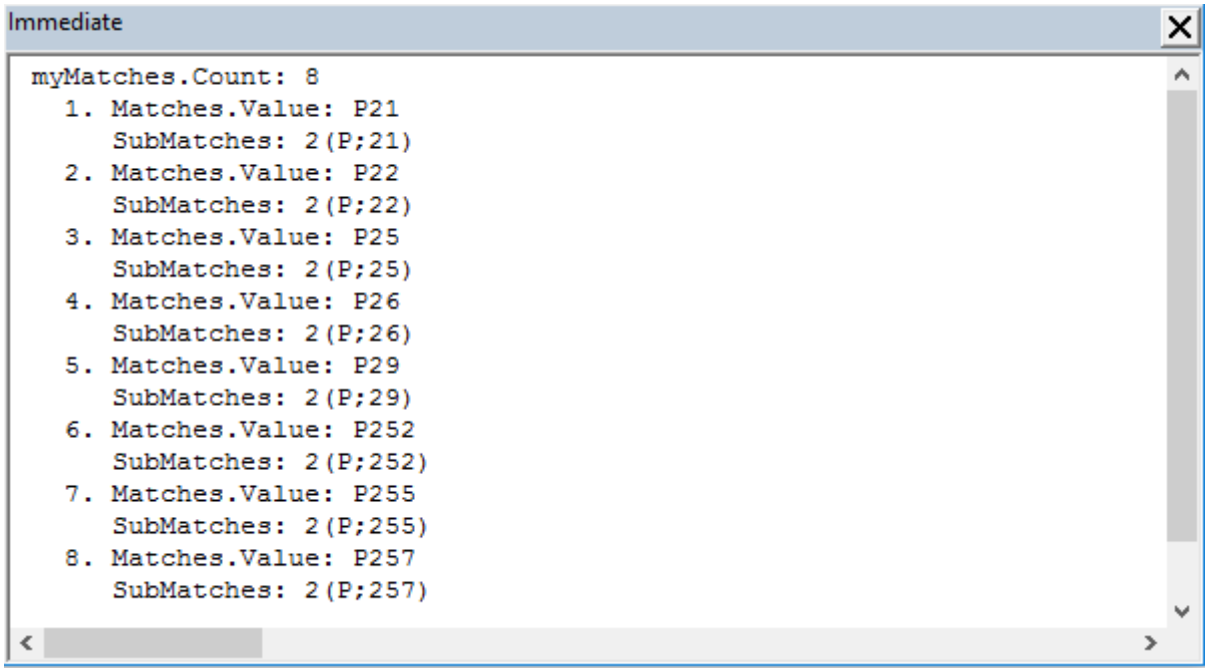
Kontrola regulárního výrazu na on-line testeru. Funguje jak má, najde všechny požadované řetězce.



Následně jsem připravil kód na test regulárního výrazu textu v buňce `I4`, kde je uvedeno `P21 + P22 + P25 + P26 + P29 - P252 - P255 - P257`.

```
Sub RegExpTest() Dim myMatch As Match Dim myMatches As MatchCollection Dim myRegExp As RegExp Dim myReplaceRegExp As RegExp Dim SubjectString As String Dim i As Integer Set myRegExp = New RegExp myRegExp.Global = True myRegExp.Pattern = "(P|K)(\d+)" Set myReplaceRegExp = New RegExp myReplaceRegExp.Global = True SubjectString = Range("I4") Set myMatches = myRegExp.Execute(SubjectString) Debug.Print "myMatches.Count: " & myMatches.Count i = 1 For Each myMatch In myMatches Debug.Print " " & i & ". Matches.Value: " & myMatch.Value Debug.Print " SubMatches: " & myMatch.SubMatches.Count & "(" & myMatch.SubMatches(0) & ";" & myMatch.SubMatches(1) & ")" i = i + 1 Next Set myRegExp = Nothing End Sub
```

Výpis testovací funkce. Všechny potřebné řetězce a subřetězce jsou nalezeny, takže tento základní kód lze aplikovat do výpočtu, který je zmíněn výše. Kompletní VBA řešení je v přiloženém souboru.



Uvedený příklad a další příklady si můžete stáhnout zde: Nejste přihlášen(a).

Autor: admin • **Vydáno:** 28.8.2016 19:45 • **Přečteno:** 23049x