Obsah

Předmluva

Popularita regulárních výrazů za poslední desetiletí značně narostla. Všechny programovací jazyky dnes obsahují knihovnu výkonných regulárních výrazů, anebo mají jejich podporu přímo zabudovanou. Mnozí vývojáři funkcí těchto regulárních výrazů využívají a uživatelům svých aplikací jejich prostřednictvím umožňují data vyhledávat a filtrovat. Regulární výrazy jsou všudypřítomné.

Na vlně nasazování regulárních výrazů se svezlo již mnoho knih. Většině z nich se daří vysvětlit syntaxi regulárních výrazů, jejich příklady a reference dobře. Nenajdete však knihu, která by na základě regulárních výrazů nabízela řešení široké škály praktických problémů z reálného života, s nimiž se setkáváme při práci s textem na počítači a v množství internetových aplikací. Jako autoři knihy (Steve a Jan) jsme se rozhodli, že zaplníme právě tuto mezeru.

Obzvláště vám chceme ukázat, jak můžete regulární výrazy užívat v situacích, ve kterých by vám lidé s omezenými zkušenostmi v práci s regulárními výrazy řekli, že je problém neřešitelný, anebo v takových situacích, v nichž by softwaroví puristi tvrdili, že se regulární výrazy na danou problematiku nehodí. Protože jsou dnes regulární výrazy všude okolo, mohou s nimi začít koncoví uživatelé přímo pracovat, aniž by museli do práce zapojovat tým programátorů. Dokonce i programátoři mohou nasazením regulárních výrazů ušetřit čas při získávání dat a změně úloh, jejichž ošetření by v procedurálním kódu trvalo hodiny či dny, nebo jejichž použití by vyžadovalo knihovnu třetí strany, kterou je třeba zkontrolovat a nechat schválit managementem.

Uvěznění mezi verzemi

Stejně jako je to se vším, co se stává v průmyslové oblasti výpočetních technologií populárním, i regulární výrazy přichází v mnoha různých implementacích, které se liší stupněm kompatibility. Vznikla tak řada různých *příchutí* regulárních výrazů. Ty s daným regulárním výrazem nezachází vždy stejně, případně s ním nezachází vůbec nijak.

Mnoho knih tyto různé příchuti zmiňuje a vyzdvihuje i některé z rozdílů, které se mezi nimi nachází. Obyčejně však některé z příchutí přeskakují – zvláště tehdy, když jim chybí některé konkrétní funkce – namísto toho, aby nabídly alternativní řešení či alespoň způsob, jak nedostatek obejít. Když pak máte pracovat s různými druhy regulárních výrazů ve více aplikacích či programovacích jazycích, budí to ve vás úzkost.

Ležérní výroky v literatuře ve stylu „dnes všichni pracují s perlovskými regulárními výrazy“ bohužel bagatelizují celou škálu problémů týkajících se nekompatibility. Dokonce i mezi „perlovskými“ balíky jsou význačné rozdíly, a Perl se přitom i nadále vyvíjí. Přespříliš zjednodušené výrazy mohou způsobit, že programátor stráví kupříkladu půl hodiny bezcílným laděním, namísto toho, aby zkontroloval podrobnosti týkající se své implementace regulárního výrazu. I když pak zjistí, že některá z funkcí, na kterou se spoléhal, není dostupná, neví, jak si s problémem poradit.

Tato kniha je první publikací na trhu, která v celém svém průběhu celistvě srovnává nejoblíbenější a na funkce nejbohatší implementace regulárních výrazů.

Cílová skupina

Jestli na počítači pracujete pravidelně s texty, měli byste si tuto knihu přečíst. Nezáleží na tom, zda se prohrabáváte hromadou dokumentů, pracujete s textem v textovém editoru, anebo vyvíjíte software, který v textu musí umět vyhledávat či s ním manipulovat. V těchto případech jsou regulární výrazy fantastickým nástrojem. *Regulární výrazy Kuchařka programátora* vás naučí vše, co o regulárních výrazech potřebujete vědět. Nemusíte mít žádné předešlé zkušenosti, protože vám vysvětlíme i ty nejzákladnější aspekty regulárních výrazů.

Jestli ještě s regulárními výrazy nemáte zkušenosti, naleznete v této knize studnici podrobností, které jiné knihy a články na internetu často přehlíží. Jestliže jste se již někdy dostali do situace, kdy regulární výraz v jedné aplikaci fungoval, ale v jiné ne, bude se vám podrobný a rovnocenný popis sedmi světově nejoblíbenějších regulárních výrazů velice hodit. Celou knihu jsme uspořádali do podoby kuchařky, takže můžete přeskakovat přímo na témata, o kterých se chcete něco dočíst. Budete-li ji číst od začátku do konce, stane se z vás prvotřídní šéfkuchař regulárních výrazů.

Tato kniha vás o regulárních výrazech naučí vše, co potřebujete znát, a ještě i něco navíc. Nezáleží na tom, zda jste programátor nebo ne. Chcete-li regulární výrazy použít při práci s textovým editorem, vyhledávacím nástrojem či jinou aplikací, která vyžaduje zadání regulárního výrazu, nemusíte mít k četbě této knihy s regulárními výrazy žádné zkušenosti. Většina z receptů v této knize zakládá řešení čistě na jednom či několika regulárních výrazech.

Jste-li programátor, 3. kapitola vám poskytne veškeré informace potřebné k implementaci regulárních výrazů ve zdrojovém kódu. Kapitola předpokládá, že znáte základní programovací funkce vámi vybraného programovacího jazyka, ale nepočítá s tím, že byste již někdy ve zdrojovém kódu regulární výrazy používali.

Popisované technologie

Hesla .NET, Java, JavaScript, PCRE, Perl, Python a Ruby nejsou na zadní straně přebalu uvedeny jen ze zvyku. Jedná se o oněch sedm příchutí regulárních výrazů, kterým se v knize budeme věnovat. Všemi se přitom zaobíráme se stejnou intenzitou. Zvláště se snažíme popsat všechny nesoulady, které by se mezi těmito příchutěmi mohly vyskytnout.

Kapitola pro programátory (Kapitola 3) pracuje s výpisy kódu v jazycích C#, Java, JavaScript, PHP, Perl, Python, Ruby a VB.NET. I zde je v každém receptu zahrnuto řešení a vysvětlení všech osmi jazyků. I když se tak v kapitole trochu opakujeme, můžete diskuze o jednotlivých jazycích, které vás nezajímají, jednoduše přeskočit, aniž byste přišli o něco, co byste o vámi vybraném jazyce měli vědět.

Jak je kniha členěna

První tři kapitoly knihy se věnují užitečným nástrojům a základním informacím, které vám poskytnou základy používání regulárních výrazů. Všechny podřízené kapitoly představují škálu regulárních výrazů, přičemž se vždy hlouběji věnují jedné oblasti zpracování textu.

Kapitola 1, *Úvod do regulárních výrazů*, osvětluje roli regulárních výrazů a uvádí množství nástrojů, s nimiž se výrazy snáze učí, vytváří a ladí.

Kapitola 2, *Základy při práci s regulárními výrazy*, se věnuje všem prvkům a funkcím regulárních výrazů, spolu s důležitými návody pro jejich efektivní použití.

Kapitola 3, *Programování s regulárními výrazy*, specifikuje techniky kódu a zahrnuje výpisy kódu pro práci s regulárními výrazy ve všech programovacích jazycích, jimiž se v knize zaobíráme.

Kapitola 4, *Ověřování a formátování*, obsahuje recepty na správu s běžnými uživatelskými vstupy, například daty, telefonními čísly a poštovními směrovacími čísly v různých státech.

Kapitola 5, *Slova, řádky a zvláštní znaky*, zkoumá nejběžnější úlohy při práci s textem, například hledání řádků obsahujících či neobsahujících konkrétní slova.

Kapitola 6, *Čísla*, ukazuje, jak lze detekovat celá čísla, čísla s plovoucí desetinnou čárkou a některé další formáty tohoto vstupního druhu.

Kapitola 7, *URL, cesty a internetové adresy*, vám ukáže, jak rozebrat a použít řetězce, se kterými se běžně pracuje na internetu a na systémech Windows při vyhledávání.

Rozlišení textu

V knize pracujeme v souladu s následujícími typografickými konvencemi:

Kurzíva

Značí nové výrazy, adresy URL, e-mailové adresy, názvy a přípony souborů.

`Písmo s konstantní šířkou znaků`

Toto písmo používáme ve výpisech programů, programových prvcích, jakými jsou například proměnné či názvy funkcí, hodnot vracených coby výsledky náhrad regulárních výrazů, a v zdrojovém či vstupním textu aplikovaném v regulárním výrazu. Může se jednat o obsah textového pole v aplikaci, souboru na disku, nebo obsahu řetězcové proměnné.

`Kurzíva s konstantní šířkou znaků`

Písmo používáme k zobrazení textu, který by měl uživatel nahradit svými vlastními hodnotami, nebo hodnotami vycházejícími z kontextu.

`<Regulární•výraz>`

Představuje regulární výraz stojící samostatně, anebo tak, jak byste ho zadávali do vyhledávacího pole aplikace. Mezery jsou v regulárních výrazech, s výjimkou případů, kdy se s nimi pracuje v režimu free-spacing (ignorování bílých znaků), značeny černými tečkami.

`«Náhradní•text»`

Představuje text, kterým se v rámci hledacích a nahrazujících operací nahradí shodné regulární výrazy. Mezery jsou v náhradních textech značeny černou tečkou.

`Shodný text`

Představuje část zdrojového textu, který je shodný s regulárním výrazem.

`...`

Šedá elipsa v regulárním výrazu značí to, že je třeba „vypsat prázdné pole“ dříve, než budete moci použít regulární výraz. Doprovodný text vysvětluje, co můžete do pole zadat.

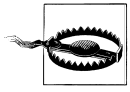
CR, LF a CRLF

CR, LF a CRLF v rámečcích představují v řetězci vlastní znaky pro zalomení řádku namísto znaků \r, \n a \r\n. Tyto řetězce vznikají v aplikaci třeba ve víceřádkovém editoru stisknutím klávesy Enter, nebo při práci s víceřádkovými řetězcovými konstantami ve zdrojovém kódu, například u doslovných řetězců, či v Pythonu u řetězců s trojitými uvozovkami.

↵

Zpětná šipka, jakou vidíte na klávesnici na tlačítku Enter, značí, že bylo třeba řádek zalomit, aby se vešel na šířku zobrazené stránky. Při zadávání textu do zdrojového kódu byste neměli používat klávesu Enter, ale naopak byste měli všechno zapisovat na jeden řádek.

Tato ikona značí tip, návrh či obecnou poznámku.



Tato ikona předznamenává varování či upozornění.

Příklady kódu

Tato kniha by vám měla pomoci při práci. Kód uvedený v knize můžete ve svých programech a aplikacích volně používat. Jestliže nebudete kopírovat velkou část kódu, nemusíte nás kontaktovat a žádat o svolení. Například k programu, ve kterém použijete několik úryvků kódu z této knihy, naše svolení potřebovat nebudete. K prodeji či redistribuci CD s příklady z knih nakladatelství O'Reilly byste již svolení potřebovali. Zodpovězení otázky citováním této knihy a příkladem kódu si svolení vlastníka práv nežádá. K zařazení většího množství vzorového kódu obsaženého v této knize do dokumentace produktu je však již svolení třeba.

Jakkoliv to nevyžadujeme, byli bychom rádi, kdybyste nám za naši práci přiznali zásluhy. V anotaci se obvykle objevuje titul, autor, nakladatel a číslo ISBN. Například „Regulární výrazy Kuchařka programátora od Jana Goyvaertse a Stevena Levithana. Copyright 2009 Jan Goyvaerts a Steven Levithan, 978-0-596-2068-7.“.

Máte-li pocit, že se chystáte příklady kódu použít nad rámec zde povolených situací, neváhejte nám napsat na adresu *permissions@oreilly.com*.

Safari® Books Online

Když na přebalu své oblíbené technické knihy uvidíte ikonu Safari® Books Online, znamená to, že knihu lze dohledat i v knihovně O'Reilly Network Safari Bookshelf.

Safari nabízí lepší řešení než elektronické knihy. Poskytuje virtuální knihovnu, s níž budete moci snadno prohledávat tisíce nejlepších technicky zaměřených knih, kopírovat vzorky kódu, stahovat kapitoly a hledat rychle odpovědi, když potřebujete nejpřesnější a nejaktuálnější informace. Službu si můžete zdarma vyzkoušet na adrese *http://my.safaribooksonline.com*.

Jak se s námi spojit

Komentáře a otázky týkající se této knihy prosím směřujte na nakladatele:

O'Reilly Media, Inc.

1005 Gravenstein Highway North

Sebastopol, CA 95472

800-998-9938 (ve Spojených státech amerických či v Kanadě)

707-829-0515 (mezinárodní nebo místní)

707-829-0104 (fax)

K této knize provozujeme i webovou stránku, na které uvádíme seznam chyb, příkladů a další informace. Stránku naleznete na adrese:

http://www.regexcookbook.com

nebo na adrese:

http://oreilly.com/catalog/9780596520687

Chcete-li knihu komentovat, anebo položit technický dotaz, který se jí týká, zašlete e-mail na adresu:

bookquestions@oreilly.com

Další informace o našich knihách, konferencích, vzdělávacích zařízeních a službě O'Reilly Network hledejte na adrese:

http://www.oreilly.com

Poděkování

Děkujeme Andymu Oramovi, našemu redaktorovi ve společnosti O'Reilly Media, Inc, za pomoc při realizaci celého projektu. Za důkladné překontrolování technické stránky knihy patří naše díky také Jeffreymu Friedlovi, Zaku Greantovi, Nikolaji Lindbergovi a Ianu Morseovi. Kniha je díky nim srozumitelnější a přesnější.

Poznámka redakce českého vydání

I nakladatelství Computer Press, které pro vás tuto knihu přeložilo, stojí o zpětnou vazbu a bude na vaše podněty a dotazy reagovat. Můžete se obrátit na následující adresy:

Computer Press
redakce PC literatury
Holandská 8
639 00 Brno

nebo

knihy@cpress.cz

Další informace a případné opravy českého vydání knihy najdete na internetové adrese *http://knihy.cpress.cz/K1577*. Prostřednictvím uvedené adresy můžete též naší redakci zaslat komentář nebo dotaz týkající se knihy. Na vaše reakce se srdečně těšíme.