

The Ruby Way

Copyright © 2003 by Que Publishing

International Standard Book Number: 0672320835

Warning and Disclaimer

Every effort has been made to make this book as complete and as accurate as possible, but no warranty or fitness is implied. The information provided is on an "as is" basis. The author and the publisher shall have neither liability nor responsibility to any person or entity with respect to any loss or damages arising from the information contained in this book or from the use of the CD or programs accompanying it.

When reviewing corrections, always check the print number of your book. Corrections are made to printed books with each subsequent printing. To determine the printing of your book, view the copyright page. The print number is right-most number on the line below the "First Printing" line. For example, the following indicates the 4th printing of a title.

First Printing: December 2001

06 05 04 03 10 9 8 7 6 5 4

Misprint	Correction
Page 25 Third line of code in "A Sample Program" section: <code>exit if not str or not str[0]</code> Delete line <code>str.chomp!</code>	<code>exit if not str or str.chomp!.empty</code>
Page 32 Delete paragraph beginning with "As a crude example" through code line Repeat <code>(j<10) do { j+=1; print j, "\n" }</code>	

<p>Page 70</p> <p>Third line of code from bottom of page</p> <pre>puts "string contains lowercase charcters"</pre>	<p>Change to</p> <pre>puts "string contains lowercase characters"</pre>
<p>Page 71</p> <p>First line of code</p> <pre>puts "string contains uppercase charcters"</pre> <p>Third line of code</p> <pre>if string =~ /[A-Z]/ and string =~ /a-z</pre>	<p>Change to</p> <pre>puts "string contains uppercase characters"</pre> <p>Change to</p> <pre>if string =~ /[A-Z]/ and string =~ /[a-z]</pre>
<p>Page 105</p> <p>Line of code immediate before paragraph beginning with "The code shown here"</p> <pre>y = b.roundf(2) # 6.12</pre>	<p>Change to</p> <pre>y = b.roundf(2) # 6.14</pre>
<p>Page 116</p> <p>First paragraph, last line:</p> <p>by a sigma () is simply...</p>	<p>by a sigma () is simply...</p>
<p>Page 121</p> <p>5th paragraph, second sentence:</p> <p>Thus the U.S. time zones usually end in "ST" (standard time) or "DT" (daylight saving time).</p>	<p>Thus the U.S. time zones usually end in "ST" (standard time) or "DST" (daylight saving time).</p>
<p>Page 123</p> <p>Second line of code from bottom</p> <pre>day = time.strftime("%a") "Tuesday"</pre> <p>Last line of code on page</p> <pre>long = time.strftime("%A") #"Tuesday"</pre>	<p>Change to</p> <pre>day = time.strftime("%A") "Tuesday"</pre> <p>Change to</p> <pre>long = time.strftime("%a") #"Tue"</pre>

<p>Page 140</p> <p>The method length (or its alias size) will give the number of elements in an array. Note that this is one less than the index of the last item:</p>	<p>The method length (or its alias size) will give the number of elements in an array. Note that this is one more than the index of the last item:</p>
<p>Page 141</p> <p>Second paragraph, last sentence in "Comparing Arrays" section</p> <p>The methods == and != depend on this method.</p> <p>Code example at bottom of page, second line</p> <pre>def <=> other)</pre>	<p>Change to</p> <p>The methods == and != depend do not on this method.</p> <p>Change to</p> <pre>def <(other)</pre>
<p>Page 159</p> <p>Fifth line of code on page</p> <pre>musicians = letters.join</pre>	<p>Change to</p> <pre>musicians = letters.join(" ")</pre>
<p>Page 174</p> <p>Last sentence on page</p> <p>The corresponding instance methods in the Array class are called shift and unshift, respectively.</p>	<p>Change to</p> <p>The corresponding instance methods in the Array class are called unshift and shift, respectively.</p>
<p>Page 191</p> <pre>def search(x) if self.data == x return self else ltree = left != nil ? left.search(x) : nil return ltree if ltree != nil rtree = right != nil ? right.search(x) : nil end nil end</pre>	<p>Change to</p> <pre>def search(x) if self.data == x return self elsif x < self.data return left != nil ? left.search(x) : nil else return right != nil ? right.search(x) : nil end end</pre>

<p>Page 192</p> <pre>temp += left.to_a if left temp += right.to_a if right str = tree.to_s # "," # str is now "bongo,grimace,jewel,monoid,nexus,plover,synergy" arr = tree.to_a # arr is now: # ["bongo",["grimace",[["jewel"], ,"monoid",[["nexus"],"plover", # ["synergy"]]]]]</pre>	<p>Change to</p> <pre>temp << left.to_a if left temp << right.to_a if right str = tree.to_s # str is now: # "[bongo,[grimace,[["jewel"],monoid ,[["nexus"],plover,[synergy]]]]]" arr = tree.to_a # arr is now: # ["bongo",["grimace",[["jewel"], monoid",[["nexus"],"plover", # ["synergy"]]]]]]"</pre>
<p>Page 211</p> <p>Last line of code on page</p> <pre>perms = File.stat.mode("somefile")</pre>	<p>Change to</p> <pre>perms = File.stat("somefile").mode</pre>
<p>Page 237</p> <p>Second paragraph in "Interfacing to PostgreSQL" section</p> <p>Assuming you already have PostgreSQL installed and set up (and you have a table named testdb),</p>	<p>Change to</p> <p>Assuming you already have PostgreSQL installed and set up (and you have a database named testdb),</p>
<p>Page 247</p> <p>Third line in third code example</p> <pre>attr_writer :b1, :b2, # Creates @b1, a1=, @b2, b2=</pre>	<p>Change to</p> <pre>attr_writer :b1, :b2, # Creates @b1, b1=, @b2, b2=</pre>

<p>Page 259</p> <p>First line of code</p> <pre>def makeWithdrawal(amount)</pre>	<p>Change to</p> <pre>def makeWithdrawal(amount)</pre>
<p>Page 264</p> <p>First line of code in Listing 5.7</p> <pre>class MyMod</pre>	<p>Change to</p> <pre>module MyMod</pre>
<p>Page 310</p> <p>Last sentence on page</p> <p>For instance, the variables \$tlab and \$temp have to be used inside the blocks passed to various constructors; this implies they can't be local variables.</p>	<p>Change to</p> <p>We don't advocate the careless use of globals in real-life programs.</p>
<p>Page 335</p> <p>Second paragraph</p> <p>This simple application has one widget in its main window--an instance of FXButton displaying the text Hello, world!:</p> <pre>FXButton.new(main, "&Hello, world!", nil,</pre>	<p>Change to</p> <p>This simple application has one widget in its main window--an instance of FXButton displaying a text string:</p> <pre>FXButton.new(main, str, nil,</pre>
<p>Page 366</p> <p>Next to last code line in example before "Dealing with Exceptions" section</p> <pre>puts "right."</pre>	<p>Change to</p> <pre>puts "wrong."</pre>
<p>Page 495</p> <p>Second paragraph in "Blocks" section, first sentence, remove</p> <p>(otherwise, it is treated as a hash literal)</p>	

<p>Page 506</p> <p>Third paragraph, second line</p> <p>Actually, the self notation can be used in Ruby as well.)</p>	<p>Change to</p> <p>Actually, the self notation can be used for Ruby methods, though not for attributes.)</p>
<p>Page 514</p> <p>Third paragraph in "Arrays" section</p> <p>Use <code><code>include?</code></code> where you would have used a conditional <code><code>in</code></code> in Python.</p>	<p>Change to</p> <p>Use <code>include?</code> where you would have used a conditional <code>in</code> in Python.</p>

This errata sheet is intended to provide updated technical information. Spelling and grammar misprints are updated during the reprint process, but are not listed on this errata sheet.