

Extracted from:

Learn to Program

This PDF file contains pages extracted from Learn to Program, published by the Pragmatic Bookshelf. For more information or to purchase a paperback or PDF copy, please visit <http://www.pragmaticprogrammer.com>.

Note: This extract contains some colored text (particularly in code listing). This is available only in online versions of the books. The printed versions are black and white. Pagination might vary between the online and printer versions; the content is otherwise identical.

Copyright © 2005 The Pragmatic Programmers, LLC.

All rights reserved.

No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form, or by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior consent of the publisher.

The
Pragmatic
Programmers

Learn to Program



Chris Pine

The Facets  of Ruby Series



Many of the designations used by manufacturers and sellers to distinguish their products are claimed as trademarks. Where those designations appear in this book, and The Pragmatic Programmers, LLC was aware of a trademark claim, the designations have been printed in initial capital letters or in all capitals. The Pragmatic Starter Kit, The Pragmatic Programmer, Pragmatic Programming, Pragmatic Bookshelf and the linking *g* device are trademarks of The Pragmatic Programmers, LLC.

Every precaution was taken in the preparation of this book. However, the publisher assumes no responsibility for errors or omissions, or for damages that may result from the use of information (including program listings) contained herein.

Our Pragmatic courses, workshops, and other products can help you and your team create better software and have more fun. For more information, as well as the latest Pragmatic titles, please visit us at

<http://www.pragmaticprogrammer.com>

Copyright © 2009 The Pragmatic Programmers LLC.

All rights reserved.

No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form, or by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior consent of the publisher.

Printed in the United States of America.

ISBN 0-9766940-4-2

Printed on acid-free paper.

Fourth printing, January 2009

Version: 2009-3-13

Chapter 2

Numbers

Now that you've gotten everything ready, it's time to write your first program! Open your text editor, and type the following:

```
puts 1+2
```

Save your program (yep, that's a complete program!) as `calc.rb`. Now run your program by typing `ruby calc.rb` into your command line. It should put a `3` on your screen. See, programming isn't so hard, now is it?

2.1 Did it work?

If it worked, that's great. But I get a lot of emails from people who are stuck right here. Did you see a window flash up and then disappear? Or nothing at all? If so, the problem is probably that you didn't run your program from the command line.

Don't just click your program's icon.

Don't just press `F5` in your text editor.

Run it by typing `ruby calc.rb` into your command line. Trust me. :)

2.2 Introduction to `puts`

So what's going on in that program? I'm sure you can guess what the `1+2` does; our program is basically the same as this:

```
puts 3
```

`puts` simply writes onto the screen whatever comes after it.

2.3 Integer and Float

In most programming languages (and Ruby is no exception) numbers without decimal points are called *integers*, and numbers with decimal points are usually called *floating-point numbers* or, more simply, *floats*.


```
puts 1.0 + 2.0
puts 2.0 * 3.0
puts 5.0 - 8.0
puts 9.0 / 2.0
```

This is what the program returns:

```
3.0
6.0
-3.0
4.5
```

(The spaces in the program are not important; they just make the code easier to read.) Well, that wasn't too surprising. Now let's try it with integers:

```
puts 1+2
puts 2*3
puts 5-8
puts 9/2
```

This is mostly the same, right?

```
3
6
-3
4
```

Uh...except for that last one! When you do arithmetic with integers, you'll get integer answers. When your computer can't get the "right" answer, it always rounds down. (Of course, `4` is the right answer in integer arithmetic for $9/2$. It just might not be the answer you were expecting.)

Perhaps you're wondering what integer division is good for. Well, let's say you're going to the movies but you only have \$9. When I lived in Portland a few years back, you could see a movie at the Bagdad for two bucks. (It was cheaper for two people to go to the Bagdad and get a pitcher of beer, *good* beer, than to go see a movie at your typical

theater. And the seats all had tables in front of them! For your beer! It was heavenly!) Anyway, nostalgia aside, how many movies could you see at the Bagdad for nine bucks? $9/2 \dots 4$ movies. You can see that 4.5 is definitely *not* the right answer in this case; they will not let you watch half of a movie or let half of you in to see a whole movie...some things just aren't divisible.

So now experiment with some programs of your own! If you want to write more complex expressions, you can use parentheses. For example:

```
puts 5 * (12-8) + -15
puts 98 + (59872 / (13*8)) * -51
```

```
5
-29227
```

2.5 A Few Things to Try

Write a program that tells you the following:

- How many hours are in a year?
- How many minutes are in a decade?
- How many seconds old are you? (I'm not going to check your answer, so be as accurate—or not—as you want.)

Here's a tougher question:

- If I am 1014 million seconds old (which I am, though I was in the 800 millions when I started this book), how old am I?

The Pragmatic Bookshelf

The Pragmatic Bookshelf features books written by developers for developers. The titles continue the well-known Pragmatic Programmer style, and continue to garner awards and rave reviews. As development gets more and more difficult, the Pragmatic Programmers will be there with more titles and products to help programmers stay on top of their game.

Visit Us Online

pragmaticprogrammer.com/titles/fr_ltp

Source code from this book, errata, and other resources. Come give us feedback, too!

Register for Updates

pragmaticprogrammer.com/updates

Be notified when updates and new books become available.

Join the Community

pragmaticprogrammer.com/community

Read our weblogs, join our online discussions, participate in our mailing list, interact with our wiki, and benefit from the experience of other Pragmatic Programmers.

New and Noteworthy

pragmaticprogrammer.com/news

Check out the latest pragmatic developments in the news.

Buy the Book

If you liked this PDF, perhaps you'd like to have a paper copy of the book. It's available for purchase at our store: pragmaticprogrammer.com/titles/fr_ltp.

Contact Us

Phone Orders:	1-800-699-PROG (+1 919 847 3884)
Online Orders:	www.pragmaticprogrammer.com/catalog
Customer Service:	orders@pragmaticprogrammer.com
Non-English Versions:	translations@pragmaticprogrammer.com
Pragmatic Teaching:	academic@pragmaticprogrammer.com
Author Proposals:	proposals@pragmaticprogrammer.com