Extracted from:

New Programmer's Survival Manual

Navigate Your Workplace, Cube Farm, or Startup

This PDF file contains pages extracted from *New Programmer's Survival Manual*, published by the Pragmatic Bookshelf. For more information or to purchase a paperback or PDF copy, please visit http://www.pragprog.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.

The Pragmatic Bookshelf

Dallas, Texas • Raleigh, North Carolina

The Pragmatic Programmers

New Programmer's Survival Manual

Navigate Your Workplace, Cube Farm, or Startup

Josh Carter

Edited by Susannah Davidson Pfalzer

New Programmer's Survival Manual

Navigate Your Workplace, Cube Farm, or Startup

Josh Carter

The Pragmatic Bookshelf
Dallas, Texas • Raleigh, North Carolina

Pragmatic Bookshelf

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, PragProg 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://pragprog.com.

The team that produced this book includes:

Susannah Pfalzer (editor)
Potomac Indexing, LLC (indexer)
Kim Wimpsett (copyeditor)
David J Kelly (typesetter)
Janet Furlow (producer)
Juliet Benda (rights)
Ellie Callahan (support)

Printed in the United States of America.
ISBN-13: 978-1-934356-81-4
Printed on acid-free paper.
Book version: P1.0—November 2011

For Daria and Genevieve.

It's day one on the job. You have programming chops, you've landed the job, you're sitting at your workstation…now what? Before you, a new jungle awaits:

- Programming at industry scale, with code bases measured in thousands (or hundreds of thousands) of lines of code. How do you get your bearings and start contributing quickly?
- Navigating an organization containing programmers but also people in many, many other roles. When you need guidance on a product feature, who do you ask?
- Building your portfolio of achievements each year. When performance reviews lurk on the horizon, do you know what your boss is looking for and how you'll be judged?

…and so much more. Your programming skills are only one part of what you'll need in these first years on the job.

The lucky among us have guides who already know the landscape. This book is a *virtual guide*. It'll get you oriented, point out the mountains and canyons ahead, and also save you from some nasty pitfalls.

Where I'm Coming From

You may find some similarity between your experience and where I stood in college in 1995: I started on a traditional path, a computer science and electrical engineering program at Duke University. I went to my advisor, asking about classes that would best prepare me for working in industry. He was a smart guy—a Rhodes scholar and rising star in the engineering school—and he responded, "I have no idea. I've never worked a day in industry in my life."

I was more than a little disillusioned. I wanted to build real, shipping products—not write research papers. So, that summer I managed to get my foot in the door at one of the hottest start-ups in Silicon Valley, General Magic. It was founded by some of the same guys who created the original Macintosh computer, Andy Hertzfeld and Bill Atkinson. My peers included some of the top players from Apple's System 7 (operating system) team and the guy who would later found eBay.

I learned more about programming in my two-month internship than I could have learned in two years of school. I called Duke and said I wasn't coming back. And so my wild ride in industry began.

And Now About You

Readers of this book will fall into a few broad categories:

- College students and recent graduates taking computer science classes and wondering, "Is this what programming is like in the real world?" (Short answer: no.)
- Professionals from other backgrounds who got into programming as a hobby or side job, now wanting to take it on full-time.
- Others who are considering a job in programming but want the skinny on what the books and classes aren't telling them.

Regardless of path, here you are: it's time to pay the bills with code. There are plenty of books out there on the code part. There's not so much on *everything else* that goes with the job—and that's where this book comes in.

For the professionals coming from other fields, some sections won't apply as much to you—you don't need *me* to tell you what marketing does if your background is marketing. However, you will still benefit from details about how things run within the engineering department and how code evolves from concept to release.

Structure of This Book

This book is written in small mini-chapters, called *tips*, that are designed to address a single topic within a few pages. Some are longer by necessity. Related tips are close together, but you can read them in any order. If you're going for the big picture, go ahead and read it from cover to cover. But feel free to flip around—when tips need to reference each other, that's stated explicitly in the text.

We start close to the code: Chapter 1, *Program for Production*, on page ? starts from your programming talent and gives you guidance on making it production-ready. Nobody wants to ship buggy code, but it's especially challenging on industrial-scale projects to ensure that your code is correct and well-tested.

Next, Chapter 2, *Get Your Tools in Order*, on page ? helps with your workflow. You'll need to coordinate with others, automate builds, and learn new technologies as you go. Plus, you'll need to hammer out a *ton* of code. It pays to invest in your tools up front.

Then we get into the squishier side of things. The one manager you'll have throughout your life is *you*, and Chapter 3, *Manage Thy Self*, on page ? gets you started on issues such as stress management and job performance.

No programmer is an island, so Chapter 4, *Teamwork*, on page ? focuses on working with others. Don't discount people skills—true, you were hired to be good at computers, but industry is a team sport.

Then we get to the bigger picture. Chapter 5, *Inside the Company*, on page ? considers all the moving pieces that make up a typical high-tech company and your part within the whole. It ultimately tries to answer, "What do all these people do all day?"

Closer to home is the business of software. Chapter 6, *Mind Your Business*, on page ? gets into who's paying your paycheck and why, the life cycle of a software project, and how your day-to-day programming changes with that life cycle.

Finally, Chapter 7, *Kaizen*, on page ? looks forward. The Japanese *Kaizen* is a philosophy of continuous improvement, and I hope to see you on that path before we part ways.

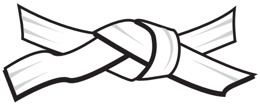
Conventions Used in This Book

I often use the Ruby programming language in tips that have example code. I chose Ruby simply because it's concise and easy to read. Don't worry if you don't know Ruby; the intent of the code should be self-evident. The examples are intended to demonstrate bigger-picture principles that may apply to any programming language.

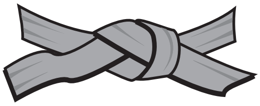
Throughout the book you'll encounter sidebars titled *industry perspective*. These are voices from industry pros: programmers and managers who have been down this road before. Each contributor has decades of experience, so consider their advice carefully.

White Belt to Black Belt (and Back)

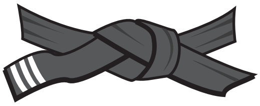
Throughout the book I use the notion of martial arts belts to signify when you'll need to apply a certain tip. The coloring of belts has a story behind it that is helpful beyond the martial arts. When a student begins, she starts with a white belt, signifying innocence. White-belt tips, likewise, apply from the very beginning.



Over years of practice, her belt becomes soiled. The brown belt is an intermediate step where the belt is, frankly, dirty. (We modern wimps just buy a new belt that's colored brown.) For this book, I expect brown-belt topics to become relevant between years two and five.



As the artist practices further, her belt becomes darker and darker until it's black. At this point, she dons the title *master*. For the book I draw the line rather early, where black-belt topics may apply around year five and onward. In real life, true mastery begins more around year ten.



What happens as the new master continues to use her belt? It becomes frayed and bleached from sunlight…it starts to become white again. The masters of old discovered something about expertise that psychologists have only recently studied: you need to get to a certain threshold before you can *know* what you *don't know*. And then you begin your learning anew.

Online Resources

This book's web page is located here:

http://pragprog.com/titles/jcdeg

From here you can participate in a discussion forum with me and other readers, check the errata for any bugs, and report any new bugs you discover.

Onward

Enough chatter about the book. You're sitting at your workstation wondering, "Now what?" And your boss is wondering why you're not working yet. So, let's get going!