POWERHOUSE PERL

Mike Schilli, Perlmeister

"What is the sound of Perl? Is it not the sound of a wall that people have stopped banging their heads against?"

It is quotes like this by Perl creator Larry Wall that once got me hooked on Perl. Also, laziness, impatience, and hubris – the three virtues of the real programmer – made it clear that Perl was right up my alley.

Back in the 90s, a group of somewhat arrogant but determined people were running the Perl mailing lists. They would rigorously denounce confused newbies who dared to ask questions already answered in the FAQ. Curious young Perl acolytes like myself would spend countless hours studying manual pages and banging their heads over and over again against the wall of Perl's idiosyncratic syntax – just to make sure no one could publicly humiliate us by pointing out a section in the manual that clearly explained an odd behavior that didn't make sense.

Fast-forward 20 years later, and what's left of Perl? It seems like the cool kids have moved on, churning out code in cleaner languages. Perl's motto, "There's more than one way to do it," has been replaced by, "there's exactly one obvious way to do it," and, guess what, it caught on. Even in Perl, no one in their right mind writes obfuscated code anymore; everyone uses standard "best practices."

What has remained, though, is a fairly large and diverse group of dedicated Perl hackers churning out code and pushing it to CPAN, Perl's open package repository. In fact, CPAN is one of the main reasons Perl is still in widespread use. The CPAN archive is incredibly active; hundreds of new releases appear every week, and if you need a tool for a new project, chances are it's right there for you to use – free and more stable than anything you could whip up from scratch.

At least, that's what I've been doing while writing Perl columns for more than 10 years. With any type of idea that pops into my head, 90 percent of the time, all I need to do is combine a handful of CPAN modules, and I'll have something running in next to no time. But rather than only scratching my own itch, my intent has always been not just to automate annoying chores, dig up invisible data, or connect electronic gadgets in unusual ways, but also to inspire my readers to add to these initial ideas and come up with solutions I've never dreamed of.

This Powerhouse Perl special edition helps you sharpen your Perl skills by leading you through a series of real-life projects that show Perl at work on real problems, such as managing data, talking to devices, finding information, and automating network failover.

If you're new to Perl, you can get started with a comprehensive introduction to the Perl language by Randal Schwartz – the undisputed #1 Perl trainer in the world – before you dive head first into the various projects. I'm a big fan of Randal's clear and concise teaching style, presented in a manner that is so entertaining you can't stop reading. I know I thoroughly enjoyed learning Perl by reading Randal's work.

And so can you! Dig in.

Mike

Elena Elisseeva, 123RF