

A Sculptor of Code Python My Clay

Brett Paufler
www.paufler.net

3 Years Coding
2 Years Python
25 Years Sales

OOP, ABM's
Genetic Algo's
Mucho Theory

Numpy
Pandas
NetworkX

And Really, So Much More

But, I'd Rather Be Coding

I like writing, don't get me wrong. But rather than composing (yet another) resume, I'd rather be working on my latest project. It's sort of a self-playing game, AI thingy, wherein Genetic Algorithms are used to discover the best strategy for competing teams in a fairly elaborate Agent Based Model (a football sim, to be exact). Anyhow, I was all sophisticated in the first go round of that project and happily passed named tuples between my objects to achieve my goals, but the overhead on that was too intense (for my poor brain, much less my laptop) to extend it past a two-player team. So, for the second go round, I used a flat list of the form [str, float, float, int...], but slicing that sucker required complex stride formations. So now, it's back to the drawing board. And I've got high hopes for nested lists. But then, I wonder if I'm devoting too much time on this part of the project. But then, yet again, unless you hire me, I've got time on my hands. And so, will be with this code for a while to come. And frankly, the interfaces I've tried thus far feel too complicated to be maintainable in the long term. Whatever. Perhaps I bore. Long story short, I've got the pass-receiver two-man-team MutaGenetic Football™ demo ready to go, so proof of concept is there. And I'd be more than happy to shoot you off a gif. Or heck-golly-jeepers, maybe I already have. So, to sum it all up, that demo gif and this paragraph on my current project pretty much sums up my programming experience, where I am, and where I wish to go. Believe it or not, the rest of the resume is fluff designed to get past the auto-scans. I like to code. Formatting lists excites me. Do you really need to know more?

Words: The Stuff of Languages

As I said, from here on down, but especially this section, is specifically designed to include keywords for the auto-scan. Because, guess what? I have 25 years experience in the industry... the construction industry, selling fabricated metals, and running a small office, which I know how to do, inside and out. But, of course, all of that has nothing to

do with Programming in Python. And neither do my certificates in Substance Abuse, Electronics, or my BS in Psychology. Still, I like to mention them... along with the few dozen novels I've written. But since they're not published, please be the first to check them all out at www.paufler.net. But in all seriousness, all of that is neither here nor there. For the meat of this section would be an overview of languages that I've reviewed over the years, enjoyed for their part, and in the end, decided that they were inferior to Python for my particular use case. The rational behind including the following, of course, would be to try and convince you that I am a serious programmer, enjoy code for code's sake, and have some hope of understanding the concerns of my CS Degree holding betters whilst standing around the water cooler. Lisp, for example, is a fine language, but I don't think in prefix notation. And quite frankly, I spent too much time trying to get all those parentheses to match up correctly to feel it was efficient. Haskell has more than a little intellectual allure for me. Yes, I'd love to be a language snob. I even solved 75+ Project Euler problems using Haskell a year or so back. But being no Category Theorist, I decided to leave that language to the math whizzes of the world. Hadoop, though all the rage, is practical only for data of terabit size that requires 10k+ nodes to map and then reduce. Um, though I may be more that a bit wordy at times, my personal trove of data hasn't quite grown that far. So, Hadoop would be overkill. But hey, I'd be happy to learn it as I go, as it would fit right in there with Linux (what I'm currently studying on the side). And from that present, let's jump back to the past, back to JavaScript, my first love and the language upon which I cut my programming chops. Sadly (on the client side, at least; but don't ask me about that node.js, for on that I know nothing), JavaScript reveals all to the world. And should we (or I, but more hopefully we) be so lucky as to strike gold, I can't conceive that it would be prudent to post a map to the mine on the web for all to see, which JavaScript basically does. For if I had no desire to get paid for my time, and was quite content to consider my small terminal the whole of the world, SmallTalk might have held greater allure. But then, that image size thing... don't get me started. Of course, the mention of images is really only a literary device to transfer the discussion back to photographic images, wrap it around, and make it, oh, so very clear, that over the years, my favorite personal projects have involved simple image manipulations, custom filters, or if we are going to be pretentious, something that might rub up against and right into signal code work. And for that, for me, I find Python to be the perfect programming tool.

What Do I Know About Python

Three years experience? Is that enough? Python has something like 300 libraries in the base with over 10,000 commands. Does anybody know them all? I use numpy all the time for my image manipulation work and tracking the position of players in my current football sim. Pandas is great for crunching data. I played with PRAW (a reddit API library), until I got bored. Networkx, yeah, that one was fun, with its networks. Someday I will delve into something having to do with dynamic networks. But not today. And here we get to the nub. How does one express competence? The level of their ability? For some reason, I wish to mention that Python's wrapping decorators remind me of Haskell's monads, going in and out as they both do (or can, for example: converting from radians to degrees, apply a function, and then converting back from degrees to radians all in a wrapper at the head of a function). Nice and complicated. And

I'm sort of proud of the understanding (new as it is). But it's hard to believe that's a good use case (probably better as an anti-pattern). Anyway, rambling on, I feel I should discuss at least some of my projects. One of the more complex ones was the complete mapping of www.Paufler.net (tons of writing, tons of code, tons of annotated samples all there for your viewing pleasure) in which I pulled the pages of my site live from the web and graphed the output as nodes on a tree. So, a website scanner of sorts, which even managed to alert me of a security breach. That was sort of cool. Very fulfilling. On another occasion, I made a receipt scanner that could bin receipts into categories correctly about 80% of the time. Of course, the entire code base was a hack, painfully poor workmanship. So, on to the things of which I am more proud, dozens of image filters; but then, they're all quite the same. And data munging, string formatting, graphical reports, they're really not interesting to me anymore. Sort of like how I tire of writing this long-winded tome of a resume. So, with that all said, I shall end it all here. In many ways I'm a good programmer, perhaps among the best you will find. No, that's fairly obvious an empty bit of bragadillo. But I'm certainly not the worst of the lot. Yeah, there you go. There's a tagline I can live with. Brett Paufler: not the worst of the lot. Or no, that's not going to work either. I admit, there are holes in my knowledge-base that you could drive entire SQL databases through ('cause, like, I've never found the use case for one yet). But the simple truth is, now that I'm done writing this and putting some token effort into finding a job for the day, I shall simply fire up the ole IDE (Eclipse with PyDev) and go play with my code, because after 25 years in the business, actually working is an option... one that I choose. Bottom line, I like to code. Writing standard resumes or taking a job search seriously, perhaps not so much.

Anyway, if you think you might have a spot on your team for someone like me, please give me a ring.

Brett Paufler
www.paufler.net

Python Programmer
Content Creator

I've got standard resumes if you want one, but they're really quite boring. Besides, what else do you really need to know? I have about three years experience programming. And next year, I'll have one more.