

The Daily Bull is probably not suitable for those under the age of 18 and should not be taken seriously...

-like The Onion, but shittier!

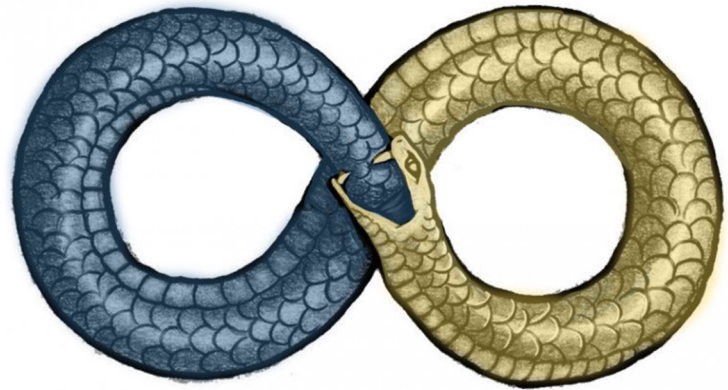
## Python Sucks Ass

DJ Benjamin

Recently, I embarked on what I expected to be a simple Python programming project. It was part of a group project and the overall project structure restricted us to either it or C++, and while I'm a cpp enjoyer I didn't feel like spending fifteen hours setting up a simple GUI that would need to be changed for every OS. Nevertheless, the project was broken into small parts, and none of them seemed like anything crazy on the face of it. I was assured by the more experienced Python programmers that there were tutorials and libraries for everything, and it would just be a matter of figuring out what to use and putting it together.

Naïve little me set out with much enthusiasm, first over Spring Break. I had interfaced only very briefly with Python before, and was trusting my group members that it was going to be simple. Thus, I began searching for tutorials, and was immediately taken by how simple and clear everything seemed. In eleven lines of code, I could get exactly what I wanted! Even better, four of those were import statements. This is going to be easy, I thought smugly, not yet aware of the horrors awaiting me.

I was introduced to the "pip" tool very early, which I was promised would just handle all the nitty-gritty installation details. That much was simple enough, but one of the libraries simply wouldn't work even after pip very positively assured me it was working. "That's weird" I thought, and began digging through tutorial pages. I learned there had been knowledge so far withheld from me—I needed to set up a Conda environment. What's a Conda Environment? I haven't the faintest clue. But I pledged away another ~100 MB of my hard disk



to all the libraries and this "Conda Environment", which promised to fix all my problems. It didn't, at least not at first.

Every tutorial I found had those few lines of code and unbridled optimism in my ability to have set up my libraries properly. The actual process of configuring the environment and setting up libraries is knowledge only afforded to those who have undergone initiation, apparently. Even the documentation for the library that needed it simply would say "Install Conda". StackOverflow was no help, as even when I found other beginners like me asking the exact questions I had, the only response they would receive would be from someone who had already suffered this onslaught of not having a single clue what the fuck was going on and what Python's single-line error messages mean. They were no help, as they dismissed my problems out of hand with some simple solution that A) was not relevant to me, or B) referenced a completely *new* thing I had never heard about.

Three hours of wading through the muck finally gave me some sort of grasp on my situation. Having read the tomes of the ancients on the Dark Web and documentation and tutorials from every troublesome library and every coder go-to, I was ready...*(Cont. on Back)*

# Python Sucks Ass

(Cont. From Front) With a sacrifice of my SSD, three hours of my time, and my desire to have a single clue of what was happening prepared, I settled in to begin a ritual to beseech the Python gods to let my code work please God please. I started on the command line, repeating cryptic lines of code in and out of the Python shell until, from pip invocations to bash navigation to environment configuration and creation. When I had finished the ritual, finally, no “ModuleNotFoundError” nor “DLL Import Error” was in sight. Finally, I could *start writing the actual program*.

My trials and tribulations were not over just yet, however, for little did I know setting up the environment isn't the hardest part. Not that the programming itself is, either—once you've prepared your environment just about all the code is truly available online in one form or another. The real problem is figuring out *which* code you actually need to use, and how you need to adapt it, because much like environment configuration, Python code itself is entirely cryptic and abstracted to Helheim and back. Hanging over your head the whole time as you decipher the process of prior initiates, who almost unilaterally refuse to actually comment a damn thing of value, is the understanding that your code is shit and will be torn to shreds the first time an experienced Python programmer looks at it, before they immediately insist on turning it into similarly cryptic and illegible code that “doesn't need comments”. At least the indentation is enforced, though whether you're supposed to use a tab or 4 spaces changes between which group of followers you ask.

Eventually, though, you survive the process and come up with *something* that works, even though it's hardly elegant and again will be executed and dismembered by a senior programmer should they deem it offensive enough. Then comes the cleaning and organization. Coming from the Church of C, and even Java, I was unprepared to deal with the most disgusting aspects of Python's soft-typing structure. Every variable, every object, every method, it's all *mushy*. You just gotta believe that it's going to work, with no way to enforce “hey, you need to pass in a *dictionary* here, *not* a List”. Also, don't try to figure out what kind of List it is—Python's abstracted that away for you :). It's a secret :). Just trust us, this is going to work if you follow the tutorials exactly—despite the fact that no central repository or agreement exists :). Global variables? Totally fine. In fact, they're customary. Forget everything else you learned from every other language. Don't worry about standard programming conventions. Just trust Python, enjoy Python, use Python for everything, let it handle everything for you. Become a Pythonite, it's so easy to learn Python. Become one of us. You'll never want to use another language ever again. Memory management? Variable Types? Let them go. Embrace the glory and majesty that is Python...



**BLIZZARD T. HUSKY**  
wants YOU



**TO JOIN  
THE DAILY BULL!**

Email [bull@mtu.edu](mailto:bull@mtu.edu) for more info!



**STUDIO PIZZA**

YOU'LL FALL IN LOVE AFTER YOUR FIRST BITE!

**We Deliver!**  
**482-5100**

[www.thestudiopizza.com](http://www.thestudiopizza.com)

The  
**Daily Bull**

Sophia Editor-in-chief  
Abraham  
Katie McIntosh Lorax VP  
Nick Franklin Local VP  
Rose Secretary  
Siebigteroth  
Ben Wittrup Treasurer  
Carl Blair Advisor

Staff writers: halfdim, DJ Benjamin, Wild Goose Chaser, Cambionical, Wendel J. Starkiller

Copyright © 2023 The Daily Bull, a non-profit organization. All rights reserved. Articles may be freely distributed electronically or on late night talk shows provided credit is given, and that this notice is included. The Daily Bull reserves the right to refuse any ads or guest articles without reason. All opinionated letters sent to the editor will be treated as material to be published unless expressly stated otherwise. Original works published remain property of the creator, however, The Daily Bull reserves the right to reprint any submission in future issues unless expressly stated other wise.

Advertising inquiries, question and comments should be directed to [BULL@MTU.EDU](mailto:BULL@MTU.EDU). Guest submissions are welcomed and encouraged. Guest contributors may write under a pen name to remain anonymous.

**DISCLAIMER:** The Daily Bull is a satirical publication for entertainment purposes only. All stories in The Daily Bull are works of fiction and any resemblance to real events or people are coincidental.

*Hi, my name is Big A\$, and I approve this message*