

THE ZEN OF PYTHON

By: Umar Elizabeth Aisha

The word 'zen' literally means 'meditation or contemplation'. In python, they are aphorisms that sink deep to our hearts to prepare us ahead of the journey we embarking on. It will guide us in setting our priorities and how to make choices when coding. In python, we have nineteen (19) zens, of which I will delve into few of which interest me.

'Beautiful is better than ugly'; is a core zen in python which emphasizes on the readability and aesthetics of the code, which is better than a convoluted and scattered code.

'Simple is better than complex'; this explains how it is better to write your codes using simple and easy methods, rather than using complex and hard methods, which tends to make the code unattractive to any reader.

'Flat is better than nested'; flat codes are easier to read, reduce cognitive load on developers, easier to debug and test, and easier to handle when errors occurs. Whereas, nested' codes will make it difficult to read, amend, and correct mistakes when made.

'Readability counts'; any code that gives a reader hard time to read is a bad code. For readability to be possible, we ensure that our code is well spaced and clear, we also use simple terms and algorithms.

'Errors should never pass silently'; when a programmer encounters an error, he or she should not simply ignore it or hide it, but rather, make it visible to the user or the developer. When an error is ignored, the program might continue to operate with incorrect data or in an unexpected state, potentially leading to further issues or incorrect results down the line.

Other zens include; now is better than never, although never is often better than *right now*, in the face of ambiguity, refuse the temptation to guess and Special cases aren't special enough to break the rules.

In conclusion, we have more zens of python and all are very important principles that helps to shape developers to be very good at coding. I see them as moral values I need to practice to be a better version of me. Thank you Black Python development for making this mentorship possible!