Python Syntax Enhancements: Adding Swift's External-Internal Function Argument Names & other Python Syntactic Sugar

By: Zak Ahmed, Pedro Oliveira, Piranaven Selvathayabaran

Department of Computing and Software, McMaster University
1280 Main St. W, Hamilton, Ontario, Canada L8S 4L8

April 17th 2019

Introduction

The purpose of the project was to apply the knowledge learned from the course CS 4TB3 in an interesting manner. Group 5, comprised of Pedro Oliveira, Piranaven Selvathayabaran & Zak Ahmed chose to apply what they learned to enhance a programming language we loved, Python.

Using cPython, the grammar was altered to enable 5 different syntactic enhancements. Firstly, adding a Swift-like External-Internal Function Argument Names. This allows a function to be called in an expressive, sentence-like manner, while still providing a function body that is readable and clear in intent. After this, we added four language constructs: Increment, Decrement, Do-While Loops, \& Switch-Case Statements. These enhancements, although not pythonic, would enhance the toolkit of developers. Lastly, Python function definitions were added with Pattern Matching as seen in functional programming. All additions were done to provide developers with more flexibility, enhance our knowledge of Python Virtual Machines and the cPython interpreter.

Process

In order to alter the grammar, we needed to take advantage of VM tools to safely work with cPython to add the four language constructs. We installed cPython on a Vagrant virtual machine and manipulated the grammar along with the lexer and parser. The vital make commands that regenerated the cPython grammar were:

- **Make regen-grammar**
- **Make regen-ast**

Python Grammar Modifications & AST Generation Code

Increment & Decrement

Do - While Loop

Until

Switch-Case Statements

Swift-like External-Internal Function Argument Names

Usage Results

Acknowledgments

We would like to take the opportunity to thank Dr. Sekerinski and the TA's , Erin Varey & Spencer Park who helped us define the project and navigate through various obstacles.

References

[1] Python Software Foundation. (n.d.). 24. Changing CPython's Grammar¶. Retrieved April 09, 2019, from https://devguide.python.org/grammar/

[2] Apple Inc. (2019). Swift.org. Retrieved April 08, 2019, from https://docs.swift.org/swift-book/LanguageGuide/Functions.html

[3] Python Software Foundation. (n.d.). 10. Full Grammar specification¶. Retrieved April 6, 2019, from https://docs.python.org/3/reference/grammar.html