Python

- NIDHI VAISHNAV

Introduction

JAVASCRIPT — 2.3M
PYTHON — 1M
JAVA — 986K
RUBY — 870K
PHP — 559K
C++ — 413K
CSS — 335K
C# — 326K
GO — 285K
C — 239K
TYPESCRIPT — 207K
SHELL — 206K
SWIFT — 107K
SCALA — 99K
OBJECTIVE-C — 66K

GitHub

0   833,333   1,666,666   2,500,000

OF PULL REQUESTS

Python replaced Java as the second-most popular language on GitHub, with 40 percent more pull requests opened this year than last. Typescript was also on the rise in 2017, used in almost four times as many pull requests as last year.

Created by **Guido van Rossum**
(Benevolent Dictator For Life)
By Nidhi Vaishnav

2nd most popular language on the GitHub in 2017

2

What is python?

Multi-purpose

- software in a wide variety of application domains
  - Ex. Web, GUI, Scripting etc.

- Strongly typed and dynamic speed

- Focus on readability and productivity

Object Oriented

- Works with objects and class

- Abstraction
  - Show only relevant data

- Encapsulation
  - Everyone knows how to access it.
  - can be easily used regardless of implementation details.

- Inheritance
  - Acquiring properties of parents

- Polymorphism
  - Based on certain implementation, perform action

Interpreted

- Compiled vs Interpreted
  - Compiled Language: convert entire program into machine code in order to execute
    - Faster to execute
  - Interpreted: run through a program line by line and execute each command instead of converting it into machine code
    - Easy to code, test and change
    - Platform independence
  - Python: interpreted language

- Supported by most IDE (Integrated Development Environment)
  - Ex. Eclipse, NetBeans, IntelliJ IDEA

Features

- Everything is an object

- Interactive shell

- Strong introspection

- Cross platform

- Libraries like numpy, scipy, matplotlib, pandas, opencv, tensorflow
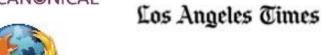
Core Philosophy of Python

- Beautiful is better than ugly
  - Write programs for human readers
  - Consistent sentence and behavior

- Readability counts
  - Easy to understand
  - Well-indented code
  - Minimal punctuation

- Sparse is better than dense
  - Don't try to stick too much code on one line

Core Philosophy of Python

- Explicit is better than implicit
  - Always mention the module whose method you are working with
  - Use of self inside methods to understand which object you are working with

- Simple is better than complex
  - Print statement
  - Interactive interpreter for experiments

Keywords

False	elif	lambda
None	else	nonlocal
True	except	not
and	finally	or
as	for	pass
assert	from	raise
break	global	return
class	if	try
continue	import	while
def	in	with
del	is	yield