

FUN WITH LANGUAGES

PYTHON3 VS PHP7 – A COMPARISON

AGENDA

- ▶ /me
- ▶ Python3
- ▶ PHP7
- ▶ Syntax
- ▶ OOP
- ▶ Package Manager and Libraries
- ▶ Performance

/ME

- ▶ Lars Eler
- ▶ Team Lead @moebel
- ▶ 14 years work experience
 - ▶ Team Lead @Notebooksbilliger,
 - ▶ Team Lead @Lamudi (Rocket Venture),
 - ▶ CTO @Bringmeister (Kaiser's Tengelmann)
- ▶ Passioned about automation, open source, delivering good software for the customer and fixing problems @root cause

DISCLAIMER: MY EXPERIENCE

- ▶ 14 years PHP
- ▶ 1 year Python
- ▶ Fair comparison?

PYTHON3

- ▶ Python is an interpreted high-level programming language for general-purpose programming
- ▶ Created by Guido van Rossum and first released in 1991
- ▶ Has a design philosophy that emphasizes code readability, notably using significant whitespace
- ▶ Python features a dynamic type system and automatic memory management.
- ▶ It supports multiple programming paradigms, including object-oriented, imperative, functional and procedural, and has a large and comprehensive standard library
- ▶ Major rewrite 2 to 3, unicode, slow adoption - <https://devopedia.org/python-2-vs-3>

* [https://en.wikipedia.org/wiki/Python_\(programming_language\)](https://en.wikipedia.org/wiki/Python_(programming_language))

PHP7

- ▶ PHP is a server-side scripting language designed for Web development, and also used as a general-purpose programming language
- ▶ Created by Rasmus Lerdorf in 1994
- ▶ The PHP language evolved without a written formal specification or standard until 2014, with the original implementation acting as the de facto standard which other implementations aimed to follow
- ▶ Since 2014 work has gone on to create a formal PHP specification (PSR)
- ▶ PHP features like Python a dynamic type system and automatic memory management. It supports object-oriented and procedural programming paradigms, and has also large and comprehensive standard library
- ▶ PHP7 major rewrite, static type hinting possible, performance improvements

* [https://en.wikipedia.org/wiki/Python_\(programming_language\)](https://en.wikipedia.org/wiki/Python_(programming_language))

COMMON FEATURES OF PHP AND PYTHON

- ▶ They can be used for general purpose programming and scripting.
- ▶ Both languages are interpreted, high-level languages that have dynamic typing.
- ▶ Both are Open Source languages supported by large developer communities.
- ▶ Both of them are easily understood and easy to learn.
- ▶ Both languages have support for method chaining.
- ▶ Both of them are easy to extend in Java, C and C++.
- ▶ Both have support for variable number of function arguments.
- ▶ They have single statements for all data types

CLEANLINESS AND UNDERSTANDABILITY OF CODE

- ▶ How does the syntax looks like?
- ▶ Is the code readable?
- ▶ Is the standard library consistent?

SYNTAX

- ▶ PHP uses traditional C-like programming style, enforces no code style at language level
- ▶ Python design philosophy emphasizes on code readability
 - ▶ No {} for control structures, instead indentation
 - ▶ No call terminator ;
 - ▶ for x in list
 - ▶ if/elif/else
 - ▶ No \$ sign for variables
 - ▶ no switch / case
 - ▶ not much () - if x=2:

SYNTAX

- ▶ PHP is well known for its powerful but inconsistent standard library
- ▶ Python has extensive and clean standard library
- ▶ Python has Decorators, which dynamically alter the functionality of a function, method, or class without having to directly use subclasses or change the source code of the function being decorated
 - ▶ Can be used for stuff like `@lru_cache()` or `@debug()`
- ▶

OOP

- ▶ PHP7 had addition of static type hinting and return type hinting
- ▶ Python3 has the same, but its not checked at runtime at all :(, but at least for you and the IDE
- ▶ Python does not have private/public/protected modifiers for class variables or constants
 - ▶ You can protect by certain `__init__.py` construct (afaik)
- ▶ Abstract classes are not part of language in python, you need to load module

OOP

- ▶ Multiple Inheritance is possible in python, while single one in PHP
 - ▶ You have still traits in PHP
- ▶ Generators, Closures etc. available for both
- ▶ In python, more or less everything is an object
 - ▶ `print(„Hello {}".format(„World“))`
 - ▶ `print(sprintf(„Hello %s“, „World“));`

PERFORMANCE

- ▶ <https://benchmarksgame-team.pages.debian.net/benchmarksgame/faster/php.html>
- ▶ My personal experience is that python is rather slow (at least standard cpython)
- ▶ For speedup, mostly used by ML, there are bindings for C and Fortran

PACKAGE / DEPENDENCIES MANAGER

- ▶ Composer defacto standard @php
- ▶ PIP(3) defacto standard @python
- ▶ Both work fine, same principles
- ▶ Basically same same but different
- ▶ Addition: Composer is able to run custom commands

SUMMARY

▶ Python's Advantages

- ▶ Programs are easy to read - Thanks to Python's elegant syntax it is much easier to read and write than a lot of other languages out there.
- ▶ Easy to use - Getting a program to work is simple and straightforward, which is why Python is widely used for prototype development.
- ▶ Easy to extend - You can always add new modules that are implemented and compiled in a language like C or C++.
- ▶ Supports object-oriented programming with multiple inheritance and classes.
- ▶ Advanced programming features - Python has list comprehensions, generators and many other useful features.

SUMMARY

▶ Python's Disadvantages

- ▶ Python is slow. Much slower than C++, PHP and a lot of other programming languages.
- ▶ Python is not suited for multi-processor/multi-core work.
- ▶ Has a lot of limitations with database access.
- ▶ Not a good language for mobile development.
- ▶ Has a lot of design restrictions. Python is a dynamically typed language, so it has more errors that only show up during runtime and thus requires a lot more testing.

END

- ▶ Questions, Remarks, Concerns?