

PYTHON TYPE HINTS

SUNSCRAPERS TECH TALK

Przemek Lewandowski

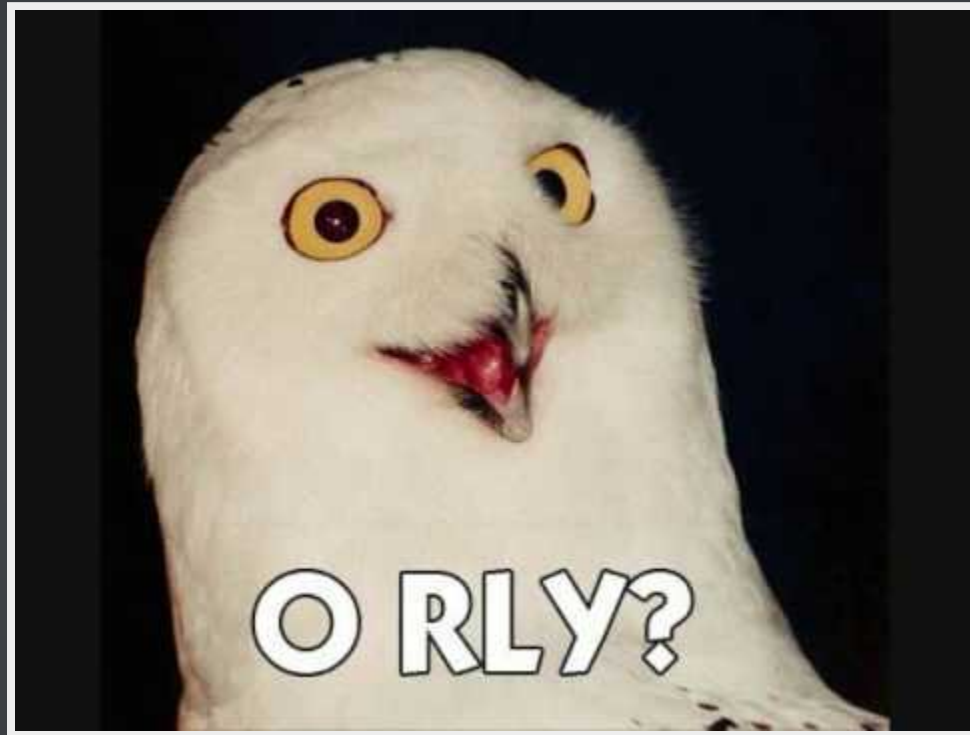
WHAT'S TYPE HINTS?

CODE COMPLETION

Do you use code completion
in your IDE / code editor?

Let's have a look on PyCharm and Python 2.7
(demo)

PyCharm parses docstrings



So PHP-ish

ADVANTAGES

- Productivity
- Better code completion
- Early bug findings

DISADVANTAGES

- More code
- Maintenance
- docstrings - orly?

ANNOTATIONS TO THE RESCUE!

Python 3 introduces
functions annotations syntax
(PEP 3107)

HOW DOES IT LOOK LIKE?

```
def compile(source: "something compilable",  
            filename: "where the compilable thing comes from",  
            mode: "is this a single statement or a suite?"):  
    ...
```

WHAT DOES IT GIVE US?

USE CASES (FROM PEP 3107):

- Providing typing information
 - Type checking
 - Let IDEs show what types a function expects and returns
 - Function overloading / generic functions
 - Foreign-language bridges
 - Adaptation
 - Predicate logic functions
 - Database query mapping
 - RPC parameter marshaling
- Other information
 - Documentation for parameters and return values

TYPE CHECKING

Jukka Lehtosalo creates **mypy** (2012)

mypy uses function annotation
for static type checking

mypy does **not**
run Python code

SOME EXAMPLES

BUILT-IN TYPES

```
def greeting(name: str) -> str:  
    return 'Hello ' + name
```

CUSTOM TYPES

```
from typing import TypeVar, Iterable, Tuple

T = TypeVar('T', int, float, complex)
Vector = Iterable[Tuple[T, T]]

def inproduct(v: Vector) -> T:
    return sum(x*y for x, y in v)
```

Python authors decides to built in
type hinting inspired by mypy

PEP 484

PEP 484 introduces a provisional module to provide these standard definitions and tools, along with some conventions for situations where annotations are not available.

PEP 484 will be introduced in Python 3.5

PYCHARM

PyCharm has preliminary support for type hinting using function annotations as specified by PEP 484

Demo - mypy & PyCharm

Resources:

- [PyCharm Type Hinting](#)
- [PEP 3107](#)
- [PEP 484](#)
- [mypy](#)

Slides and code:

<https://github.com/haxoza/talk-python-type-hints>

THANK YOU!

Q & A