



Jak przygotować się do rozmowy rekrutacyjnej na Python Developera



Wojciech Lichota - STX Next

Czerwiec 2016



Wojciech Lichota



STX Next

wojciech@lichota.pl

@wlichota

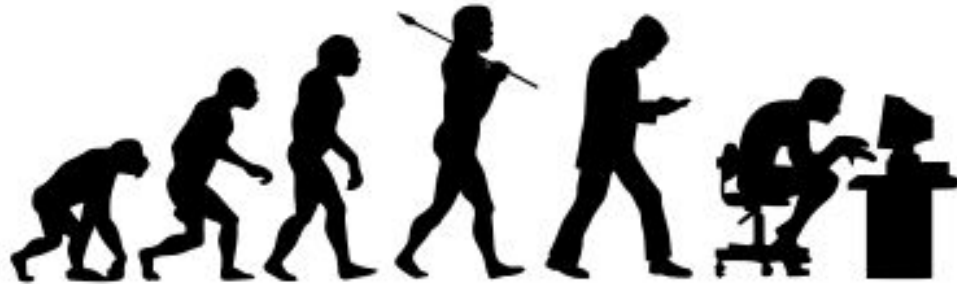
<http://lichota.pl>

OTWIERAMY
SIĘ NA
ŁÓDŹ

STXNEXT
agile / programming / teams

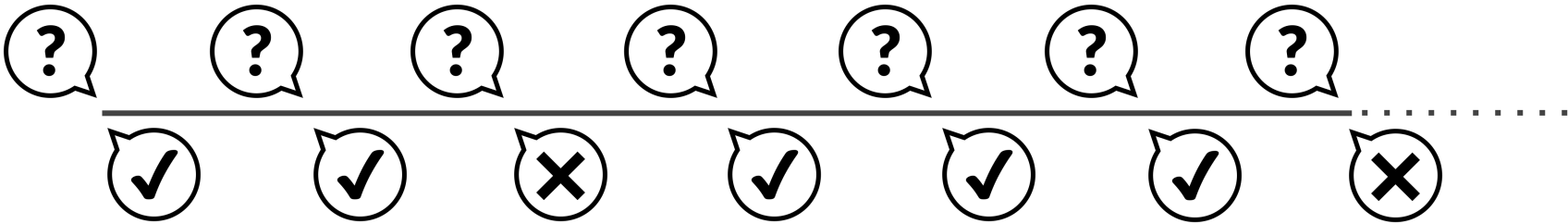
Why?

Evolution



<http://www.nicolasbize.com/blog/how-i-ended-up-conducting-the-most-successful-technical-interviews-with-a-single-question/>

Quiz



QUIZ



Idioms
&
Syntactic Sugar

- ★ slice
`my_list[2:-1]`
- ★ comprehensions
`list [n ** 2 for n in range(10) if n % 2 == 0]`
`dict {n: n ** 2 for n in range(10)}`
`generator (n ** 2 for n in range(10))`
- ★ `def my_func(*args, **kwargs):`
- ★ `a, b, c = my_tuple`
- ★ decorator (`@`)
- ★ generatory (`yield`)
- ★ context manager (`with`)
- ★ meta-klass (`__metaclass__`)

QUIZ

Quirks

- ★ mutable types
`a = [1, 2, 3]; b = a; b[1] = 0; print(a)`
- ★ immutable types
Co może być kluczem słownika?
- ★ `sort` VS. `sorted`
- ★ `''.join(my_list)`
- ★ `__getattr__` VS. `__getattribute__`
- ★ `_, _, _ = [1, 2, 3]`
- ★ `a, *b, c = [1, 2, 3, 4, 5]`
- ★ `a, b = b, a`
- ★ `True, False = False, True`
- ★ `..., Ellipsis`

QUIZ



Anti-patterns

- ★ `from module import *`
- ★ `exec, eval`
- ★ `except:`
- ★ `%` VS. `format`
- ★ `range` VS. `xrange`, `keys` VS. `iterkeys`
- ★ `def my_dunc(data=[]):`
- ★ `my_obj._MyClass__attr`
- ★ nadpisywanie built-in'ów

QUIZ

★
Tools

- ★ pep8, pylint, flake8
- ★ unittest, nosetests, pytest
- ★ pdb, ipdb, pdbpp
- ★ ipython
- ★ sphinx
- ★ git, hg
- ★ IDE
- ★ Jenkins, Travis



- ★ dekoratory - <http://stackoverflow.com/a/1594484/2342911>
- ★ yield - <http://stackoverflow.com/a/231855/2342911>
- ★ meta-klasy - <http://stackoverflow.com/a/6581949/2342911>
- ★ <http://stackoverflow.com/questions/101268/hidden-features-of-python>
- ★ <https://www.quantifiedcode.com/knowledge-base/>
- ★ <https://github.com/faif/python-patterns>
- ★ książka “Writing Idiomatic Python”, Jeff Knupp

Code



CODE

★
Algorithms

- ★ FizzBuzz
- ★ [Fibonacci](#)
- ★ [Palindrom](#)
- ★ isPrime
- ★ [Histogram](#)
- ★ [Problem wydawania reszty](#)
- ★ Sortowanie

- ★ podział na wiersze

```
>>> print(get_rows([1, 2, 3, 4, 5]))  
[[1, 2, 3], [4, 5]]
```
- ★ [GROT](#)
- ★ [WeirdText](#)

CODE

★
Others

- ★ [Testy jednostkowe](#)
- ★ “Coś” nie działa!
Jak dojdiesz do tego co jest problemem?
- ★ Garbage Collector (gc)
- ★ Global Interpreter Lock (GIL)
- ★ optymalizacja (profile)
- ★ Big O notation
- ★ [MapReduce](#)



- ★ [Project Euler](#)
- ★ [CheckIO](#)
- ★ [CodinGame](#)
- ★ [CodeCombat](#)
- ★ [CodeWars](#)
- ★ [exercism.io](#)
- ★ [GROT](#), [GROT server](#)

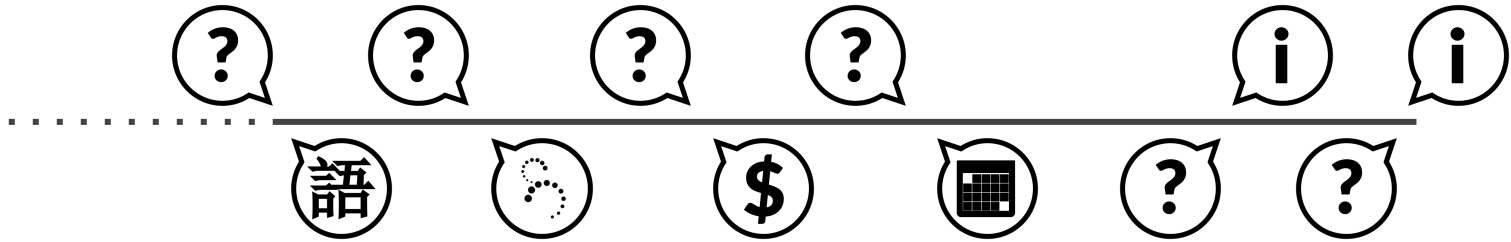
- ★ on-call duty - nauka debugowania złożonych problemów

- ★ <http://pythonfasterway.cf/>

Best project



End





- ★ Przeanalizuj ogłoszenie o pracę
- ★ Zaktualizuj CV (szczególnie “agencyjne”)
- ★ Ustal dostępność (np. sprawdź obecną umowę)
- ★ Zastanów się nad pensją i rodzajem umowy
- ★ Poproś o spotkanie “rano”
- ★ Nie przesadź z ubiorem

- ★ Spytaj czy możesz “googlować”
- ★ Nie rób ctrl+c / ctrl+v

- ★ Poproś o feedback
- ★ Podziękuj

- ★ książka “Cracking the Coding Interview”, Gayle L. McDowell

Podsumowanie

- ★ Ćwicz!
- ★ Trenuj!
- ★ Doksztalcaj się!



[Monty Python Job Interview](#)