



Historie a současnost

O autorovi

Jiří Konečný



Anaconda developer



- Twitter: @jiri_konecny
- GitHub: <https://github.com/jkonecny12>

Historie & Novinky

Historie

Historie

Počátek jazyka: 1989

První release: 1991

Python 2.0: 2000

Python 3.0: 2008

Historie



Historie



Historie

Python's Benevolent Dictator For Life

“Python is an experiment in how much freedom programmers need. Too much freedom and nobody can read another's code; too little and expressiveness is endangered.”

- Guido van Rossum



Historie

Python's Benevolent Dictator ~~For Life~~

“Python is an experiment in how much freedom programmers need. Too much freedom and nobody can read another's code; too little and expressiveness is endangered.”

- Guido van Rossum



Historie

- Mezera mezi C a Bash
- Kolem roku 2012 získal velkou pozornost
- Malé i velké projekty
- Rozšiřitelnost

Historie

- Mezera mezi C a Bash
- Kolem roku 2012 získal velkou podporu
- Malé i velké projekty
- Rozšiřitelnost

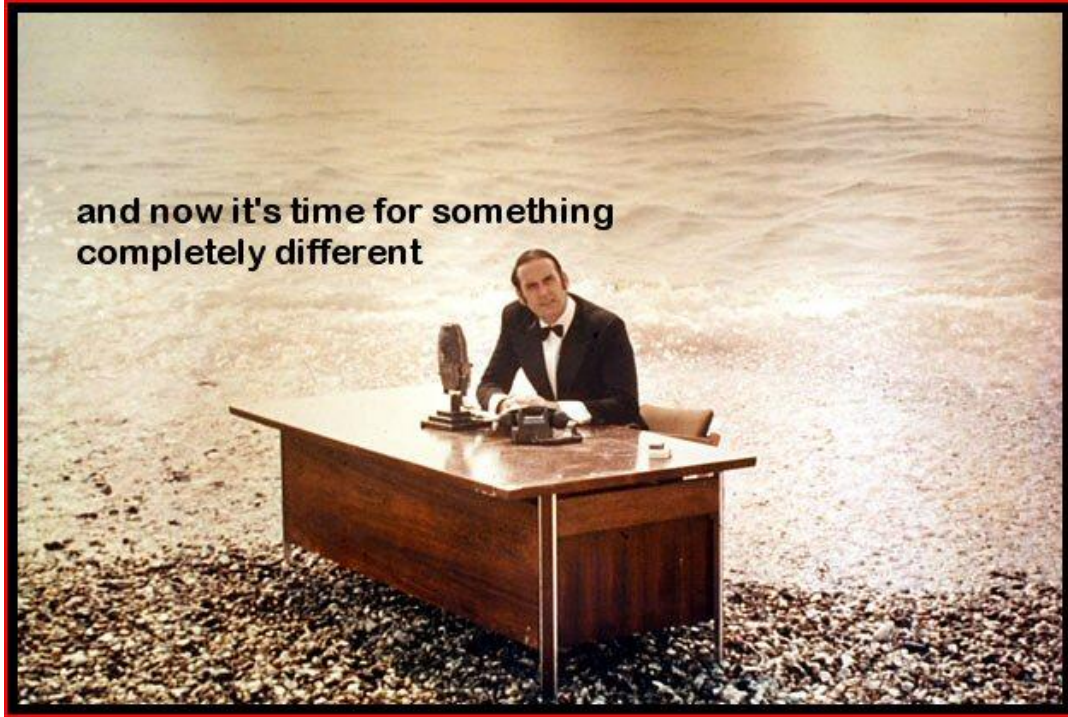


Historie - Python Zen (PEP 20)

- Beautiful is better than ugly.
 - Explicit is better than implicit.
 - Simple is better than complex.
 - Complex is better than complicated.
 - Readability counts.
- Krásné je lepší než ošklivé.
 - Explicitní je lepší než implicitní.
 - Jednoduché je lepší než složité.
 - Složité je lepší než komplikované.
 - Čitelnost se počítá.

Novinky

**and now it's time for something
completely different**



Novinky

Přednášející má vždy pravdu!

Novinky



Python 2 je mrtvý. Smiřte se s tím!

Data class

PEP 557

- `@data_class`
- Automatické generování metod

```
@dataclass
class InventoryItem:
    '''Class for keeping track of an item in inventory.'''
    name: str
    unit_price: float
    quantity_on_hand: int = 0

    def total_cost(self) -> float:
        return self.unit_price * self.quantity_on_hand
```


Mroží operátor

PEP 572

- Možnost nastavit proměnnou přímo ve výrazu

```
# A loop that can't be trivially rewritten using 2-arg iter()
while chunk := file.read(8192):
    process(chunk)
```

Novinky



Co se nepovedlo!

Co se nepovedlo

F-strings (PEP 498)

- Použitelnost
- Ideálně jenom jedno řešení

```
>>> f'The value is {value}.'  
'The value is 80.'
```

Co se nepovedlo

Positional only parameters (PEP 570)

- Proč?
- Odporuje to Python ZEN?
- Čitelnost
- Rozbili zpětnou kompatibilitu

Co se nepovedlo

Positional only parameters (PEP 570)

```
>>> help(pow)
...
pow(x, y, z=None, /)
...

>>> pow(x=5, y=3)
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
TypeError: pow() takes no keyword arguments
```

Co se nepovedlo

Positional only parameters (PEP 570)

```
>>> def kwd_only_arg(*, arg):  
...     print(arg)
```

Co se nepovedlo

Positional only parameters (PEP 570)

```
>>> def combined_example(pos_only, /, standard, *, kwd_only):  
...     print(pos_only, standard, kwd_only)
```

YOU HAVE ANY MONTY PYTHON MEMES AT ALL?



**NO SIR, I HAVE DELIBERATELY
BEEN WASTING YOUR TIME.**

Zdroje:

- https://en.wikipedia.org/wiki/Python_%28programming_language%29
- <https://www.youtube.com/watch?v=eaP4b2nAqm8>
- <https://www.python.org/dev/peps>