

**Now is
better
than never**

Юрий Сыроецкий

Python 5

Совместимость

Уважая традиции,
ломаем совместимость
во имя прогресса!

Пример (Python 3)

```
def main():  
    print("Hello world")  
  
if __name__ == "__main__":  
    main()
```

Всегда начинаем с `main`

```
def main():  
    print("Hello world")
```

Убираем лишний `def`

```
main() =  
    print("Hello world")
```

...и лишние пустые скобки

```
main =  
    print("Hello world")
```

Цикл `for` (Python 3)

```
numbers = [2, 4, 6, 8]
product = 1
for number in numbers:
    product *= number
print("The product is", product)
```

```
The product is 384
```


Цикл `for` (Python 5)

```
let numbers = [2, 4, 6, 8]
product <- var 1
for numbers `by` \number ->
  product *= number
print("The product is", product)
```

```
The product is 384
```

Добавим немного кода в тело цикла

```
let numbers = [2, 4, 6, 8]
product <- var 1
for numbers `by` \number -> do
  print("Current sub-product is", product)
  product *= number
print("The product is", product)
```

```
Current sub-product is 1
Current sub-product is 2
Current sub-product is 8
Current sub-product is 48
The product is 384
```

Да! Многострочные лямбды!

```
for numbers `by` \number -> do
  print("Current sub-product is", product)
  product *= number
```

Модульная система (1)

```
namespace1/
```

```
    package2/
```

```
        package3/
```

```
            __init__.py
```

```
            module3_4.py
```

```
        package5/
```

```
            __init__.py
```

```
            module5_6.py
```

Модульная система (2)

```
namespace1/  
  __init__.py # fiction  
  package2/  
    __init__.py # fiction  
    package3/  
      __init__.py  
      module3_4.py  
  
  package5/  
    __init__.py  
    module5_6.py
```

Новая модульная система (1)

```
package1/  
  
  module2/  
  
    module3/  
      __init__.py  
      module3_4.py  
  
    module5/  
      __init__.py  
      module5_6.py
```

Новая модульная система (2)

```
package1/  
  
  module2/  
  
    module3.hs  
    module3/  
      module3_4.hs  
  
    module5.hs  
    module5/  
      module5_6.hs
```

hs = hacker's script

Инструменты: интерпретатор

```
$ ls  
hello.hs
```

```
$ python5 hello.hs  
Hello world!
```

Невалидная программа не запустится

Инструменты: компилятор

```
$ ls  
hello.hs
```

```
$ py5compile hello.hs
```

```
$ ls  
hello hello.hs
```

**AttributeError: 'NoneType' object
has no attribute 'attribute'**

**TypeError: can only concatenate list
(not "tuple") to list**

500 Internal Server Error

Статическая типизация

Найди ошибку до того, как ошибка найдёт тебя!

Ненавязчивая статическая ТИПИЗАЦИЯ

```
distance((x1, y1), (x2, y2)) =  
    sqrt((x2 - x1) ^ 2 + (y2 - y1) ^ 2)
```

Ненавязчивая статическая ТИПИЗАЦИЯ

```
distance :: (Double, Double), (Double, Double) -> Double
distance((x1, y1), (x2, y2)) =
    sqrt((x2 - x1) ^ 2 + (y2 - y1) ^ 2)
```

Ненавязчивая статическая ТИПИЗАЦИЯ

```
type Point = (Double, Double)
distance :: (Point, Point) -> Double
distance((x1, y1), (x2, y2)) =
    sqrt((x2 - x1) ^ 2 + (y2 - y1) ^ 2)
```


Связывание имён (Python 3)

```
>>> adders = [lambda x: x + i for i in range(3)]  
>>> [f(0) for f in adders]  
[2, 2, 2]
```


Связывание имён (Python 5)

```
>>> let adders = [\ x -> x + i | i <- range(3)]  
>>> [f(0) | f <- adders]  
[0, 1, 2]
```

C Preprocessor

```
#ifdef _DEBUG
    debug(x) = print(x, file:=stderr)
#else
    debug(_) = pass
#endif
```

Спасибо!

github.com/cblp/python5

Юрий Сыровецкий

cblp.su · fb.me/cblp.su · [@cblp_su](https://twitter.com/cblp_su)