

Python

#2



Встроенные типы данных



Feedback



ТЫ МОЛОДЕЦ!!!

не сомневайся




План

- Строки
- Списки
- Кортежи
- Словари
- Множества
- И их методы



The Zen of Python

`import this`

- 
- Красивое лучше, чем уродливое.
 - Явное лучше, чем неявное.
 - Простое лучше, чем сложное.
 - Сложное лучше, чем запутанное.
 - Плоское лучше, чем вложенное.
 - Разреженное лучше, чем плотное.
 - Читаемость имеет значение.
 - Особые случаи не настолько особые, чтобы нарушать правила.
 - При этом практичность важнее безупречности.
 - Ошибки никогда не должны замалчиваться.
 - Если не замалчиваются явно.
 - Встретив двусмысленность, отбрось искушение угадать.
 - Должен существовать один — и, желательно, *только* один — очевидный способ сделать это.
 - Хотя он поначалу может быть и не очевиден, если вы не голландец.
 - Сейчас лучше, чем никогда.
 - Хотя никогда зачастую лучше, чем *прямо* сейчас.
 - Если реализацию сложно объяснить — идея плоха.
 - Если реализацию легко объяснить — идея, *возможно*, хороша.
 - Пространства имён — отличная штука! Будем делать их побольше!



PEP 8

отступ



строка_кода...

79



~~булево == True~~

~~булево == False~~

is (is not)

None

~~==~~

~~имена: | 0 |~~

func1

...
...
...

} 2 строки

func2

1 пробел

a = b + c - d * e / f

между операторами

- <https://www.python.org/dev/peps/pep-0008/>
- <http://pep8.ru/doc/pep8/>
- <http://pythonworld.ru/osnovy/pep-8-rukovodstvo-po-napisaniyu-koda-na-python.html>



PEP 8

Один на строке

```
import standart      # 1
import third-party   # 2
import mymodule       # 3
```

имя модуля
Имя Класса
имя_функции
имя_метода
_внутр_метод
УХ_КОНСТАНТА
имя_переменной



.__doc__()

"""Краткое описание

Подробнее – PEP 257.

"""

Проверка типа
`isinstance(a, list)`



```
''.join(s1, s2)  # fast
s1 = s1 + s2     # slow
```

префикс

`.startswith()`
`.endswith()`

суффикс

~~срезы~~
~~`[:3]`~~



- <https://www.python.org/dev/peps/pep-0008/>
- <http://pep8.ru/doc/pep8/>

Типы данных

Изменяемые

Список (`list`) `# [1,2,3,4]`

Словарь (`dict`) `# {1:'a', 'b':True, None:112}`

Множество (`set`) `# {1, True, 'mystr'}`

Неизменяемые

Число (`int`, `double`) `# 1 2.5`

Строка (`str`) `# 'stroka'`

Булево (`True`, `False`)

`None`

Кортеж (`tuple`) `# (1,2,3,4)`

Неизменяемое множество `frozenset`



Строка (str)

Литералы строк:

```
>>> 'stroka'
'stroka'
>>> "tozhe stroka"
'tozhe stroka'
>>> ''' Multi-line
string
'''
' Multi-line\nstring\n'
```

Нет отдельного символьного типа:

`'s'` # строка из одного символа

Операции над строками:

+ *****

Строковые методы (некоторые):

`find, replace, upper, strip, join, format`



Список (list)

Изменяемая последовательность элементов

Списки:

```
>>> [] # пустой список
[]
>>> [1, 2, True, None, 'stroka', [3, 4, 5]] # разные типы в списке
[1, 2, True, None, 'stroka', [3, 4, 5]]
```

Операции над списками:

+ * in del

Некоторые методы списков:

append, extend, insert, pop, reverse, sort



Словари (dict)



- доступ по ключу;
- неупорядочены
- ключи — только неизменяемых типов

объявление словаря

```
d = {1:'str1', '2':0, True:None, False:10}
```

```
d = dict((1,2), (3,4))
```

```
d = dict.fromkeys(1, 2, 3)
```

```
d.keys()      # список ключей
```

```
d.values()    # список значений
```

```
d.items()     # список (ключ, значение)
```



Множества (set, frozenset)

```
s = set (1, 2, 3, 4, 5)
```

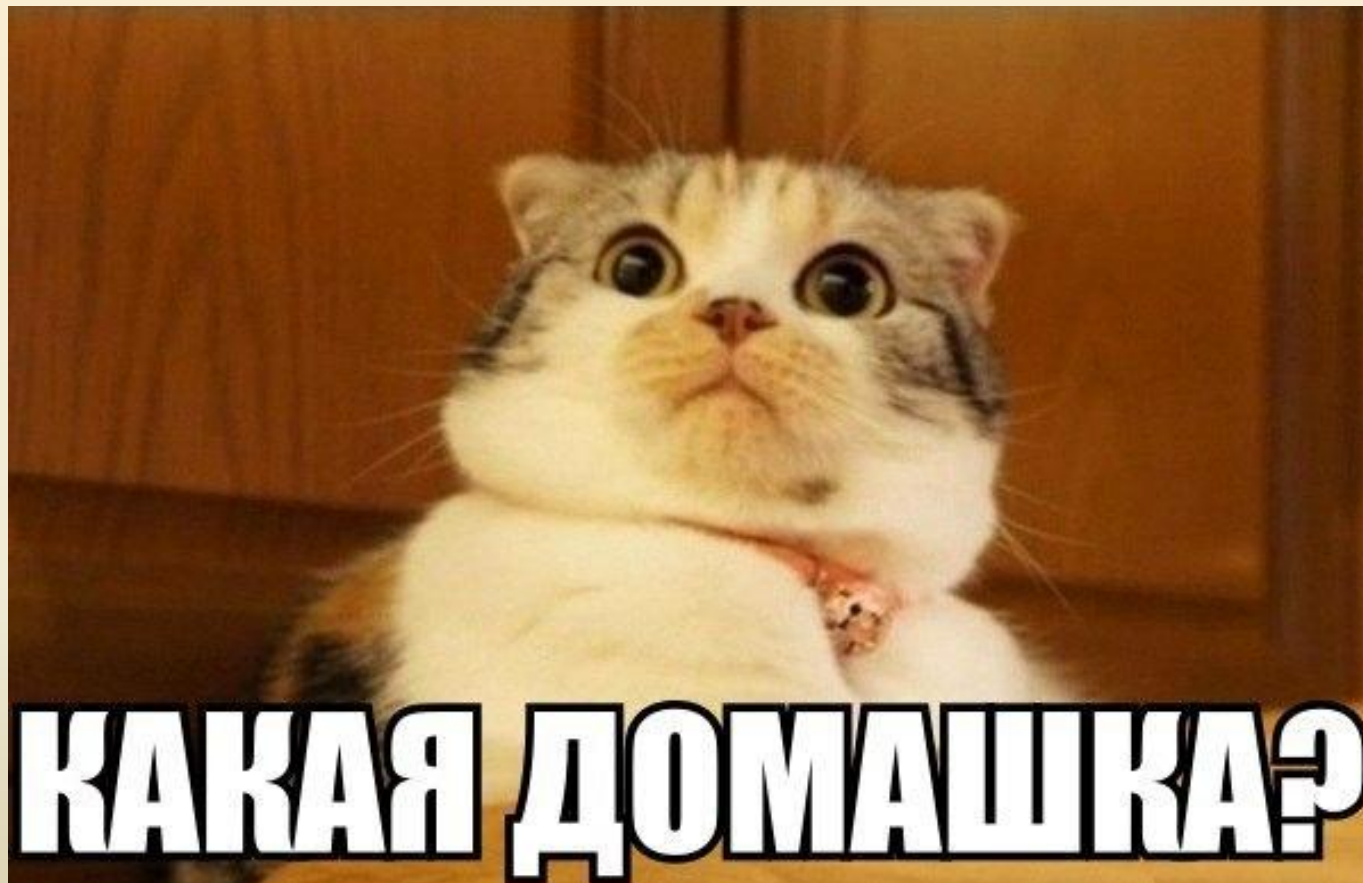
```
# изменяемое множество
```

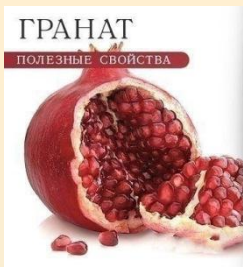
```
z = frozenset(3, 5, 6)
```

```
# неизменяемое множество
```



Домашка





Для пользы дела

- Python Tips, Tricks, and Hacks - http://idzaaus.org/static/files/articles/Python_Tips,_Tricks,_and%20Hacks_%28rus%29.pdf
- 100 вопросов по Python на собеседованиях - <https://www.dezyre.com/article/100-data-science-in-python-interview-questions-and-answers/188>
- IDA+Python - <https://xakep.ru/2011/06/23/55780/>
- PyTricks - <https://github.com/brennarm/PyTricks>



Видео-лекции

- <http://programming086.blogspot.ru/2015/12/python-2015.html> - Сергей Лебедев
- <http://www.youtube.com/playlist?list=PLPERlLqzuTQqXEIjjN6gwFzV1yRuwReR0> - Георгий Курячий



Для пользы дела

Реализация словаря в Python 2.7 - <http://habrahabr.ru/post/247843/>

Python Tips, Tricks, and Hacks -

[http://idzaaus.org/static/files/articles/Python Tips, Tricks, and%20Hacks %28rus%29.pdf](http://idzaaus.org/static/files/articles/Python%20Tips,%20Tricks,%20and%20Hacks%208rus%29.pdf)

100 вопросов по Python на собеседованиях - <https://www.dezyre.com/article/100-data-science-in-python-interview-questions-and-answers/188>

PyTricks - <https://github.com/brennerm/PyTricks>

ИНТЕРЕСНО...



ИНТЕРЕСНО...

ns0vach.ru

- Кодо-игры:

<http://www.checkio.org>

<http://www.pythonchallenge.com/>

