Лекция 7

Язык программирования Python.

Хайрулин Сергей Сергеевич

email: s.khairulin@g.nsu.ru, s.khayrulin@gmail.com

Ссылка на материалы

План

- Лекции/практические занятия
  - Тест
- Дифференцированный зачет в конце семестра
  - Защита задания

Литература

Начальный уровень

- Mark Pilgrim. Dive into Python - http://www.diveintopython.net/
- Марк Лутц. Изучаем Python, 4-е издание // Символ-Плюс 2011.
- ...

Стандарт/Документация

- PEP-8 - https://www.python.org/dev/peps/pep-0008/
- https://www.python.org/
- https://github.com/python/cpython

Экспертный уровень

- Лучано Рамальо: Python. К вершинам мастерства
- Mitchell L. Model. Bioinformatics Programming Using Python // O'Reilly 2010.

Версии Python

- Python 2 вышел 2010 году последняя версия 2.7.16 - исправлялись только баги(ошибки) с января 2020 года поддержка прекращена.
- Python 3 в появился в 2008, является актуальной версией языка. Текущая стабильная версия 3.8.5 -> в предрелиз 3.9, в разработке 3.10
  - Python 3 не гарантирует совместимости кода с Python 2

План занятия

- Генераторы

Определение

Generator functions allow you to declare a function that behaves like an iterator, i.e. it can be used in a for loop (https://wiki.python.org/moin/Generators).

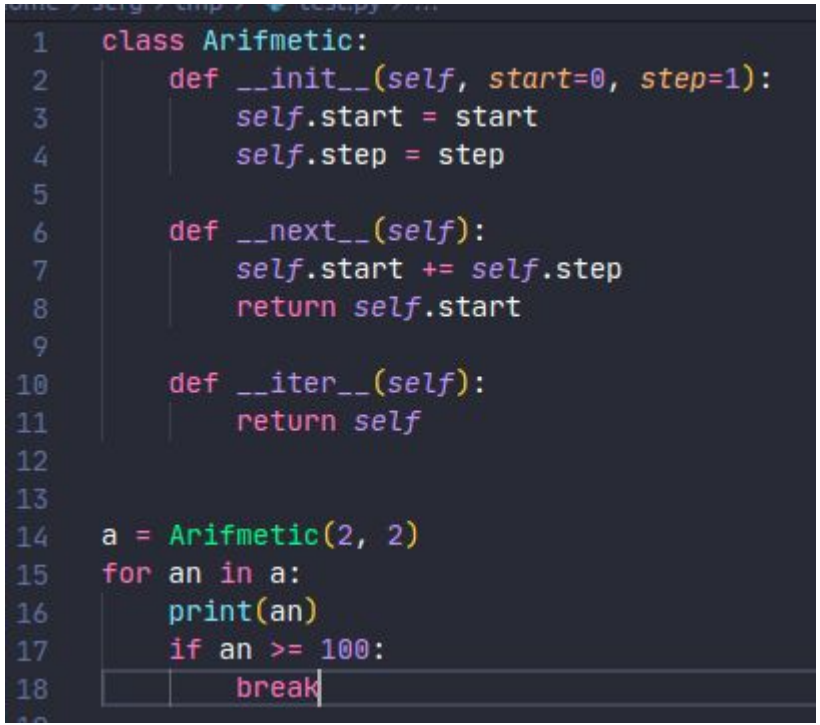
Создание

```
In [1]: gen = (i for i in range(10))

In [2]: type(gen)
Out[2]: generator

In [3]: for i in gen:
   ...:     print(i)
   ...:
0
1
2
3
4
5
6
7
8
9
```

Создание



```python
class Arifmetic:
    def __init__(self, start=0, step=1):
        self.start = start
        self.step = step

    def __next__(self):
        self.start += self.step
        return self.start

    def __iter__(self):
        return self


a = Arifmetic(2, 2)
for an in a:
    print(an)
    if an >= 100:
        break
```

Создание

```
class Arifmetic:
    def __init__(self, end, start=0, step=1):
        self.start = start
        self.n = end
        self.step = step
        self.cnt = 1

    def __next__(self):
        if self.cnt < self.n:
            self.cnt += 1
            self.start += self.step
            return self.start
        else:
            raise StopIteration()

    def __iter__(self):
        return self


a = Arifmetic(100, 2, 2)
print(sum(a))
```

Ключевое слово yield

Ключевое слово yield

```python
def gen(start, end, step):
    while start < end:
        yield start
        start += step


g = gen(1, 10, 1)
for i in g:
    print(i)
```

```
OUTPUT  TERMINAL  DEBUG CONSOLE  PROBLEMS

[13:10:56] serg :: serg-pc  → ~/tmp»
python test.py
1
2
3
4
5
6
7
8
9
[13:10:57] serg :: serg-pc  → ~/tmp»
```

Ключевое слово yield

```python
def gen(start, end, step):
    while start < end:
        yield start
        start += step


g = gen(1, 10, 1)
for i in g:
    print(i)

for i in g:
    print(i)
```

```
OUTPUT  TERMINAL  DEBUG CONSOLE  PROBLEMS

[13:09:18] serg :: serg-pc  → ~/tmp»
python test.py
1
2
3
4
5
6
7
8
9
[13:10:31] serg :: serg-pc  → ~/tmp»
```

Спасибо за внимание. Вопросы?