

Python

Базовый тренинг

Ружин Алексей
ruzin@me.com

Цель

- Познакомиться с языком
- Научиться самостоятельно разрабатывать приложения

Занятие 1

Введение в Python

Дзен

- Красивое лучше уродливого
- Явное лучше неявного
- Простое лучше сложного
- ... (import this)
- Style Guide: PEP8

```
file = open('file.csv', 'r')  
for line in file:  
    print(line)  
file.close()
```

Достоинства и недостатки

- простой и понятный
- выразительный
- поддержка комплексных чисел
- интерпретируемый (+/-)
- динамическая типизация (+/-)
- сборщик мусора (+/-)
- большое количество библиотек
- GIL - ограничение на многопоточность (недостаток)

Где используется

- Научные исследования
- Системное администрирование
- Веб-разработка
- Google, Yandex и пр.

Hello, World

- Создание проекта в PyCharm
- Вызов консоли
- Hello, world
- Перезапуск консоли

«Кирпичи» программы

- переменные
- ветвления
- циклы
- функции

Переменные

- Переменные
- Типы данных: целые, вещественные, комплексные, булевы, строки, объекты, None
- `type(var)`, `isinstance(var, type)`

- `type(10)`
- `type(type(10))`
- `type(type(type(10)))`

Арифметика

- $+$, $-$, $*$, $/$, $**$

$$2 * 3 + 3$$

$$3 + 3 * 2$$

- изменение приоритетов с помощью $()$

Строки

- ' ', " ", """ """
, ,
- a[10:15]
- конкатенация
- format()

Условные операторы

- if a < b:
 print 'a < b'
elif a > b:
 print 'a > b'
else:
 print 'a == b'
- if my_var is None or my_var is not None:
 print True
- if 'key' in 'hockey':
 print 'key is in hockey'

Цикл while

- `a = 0`
`while a < 100:`
 `a += 1`
- `break`, `continue`

СПИСКИ

- `a = [1,2,4,'asdf']`
- `append / insert(2, 'Thomas')`
- срезы списков `[a:n]`
- конкатенация списков
- "двумерный массив" == "список списков"
- генератор списков `[x for x in a]`

Итерация по списку

- `a = [1, 2, 4, 8]`
`for x in a:`
 `print(x)`
- генератор списков `[x for x in a]`
- `for x in range(10):`
 `pass`

Задача 1

Дан список целых чисел, например:
[5, 13, 23, 32, 16, 1].

Требуется напечатать минимальное и
максимальное числа из этого списка.

Задача 2

Вывести все числа Фибоначи меньше 100.

Числа Фибоначи образуются путем сложения двух предыдущих чисел. При этом первые два числа равны единице: 1, 1, 2, 3, 5, 8, ...

Задача 3

Дана строка `a=«PYTHON»`, необходимо напечатать строку, где буквы идут задом наперед.

Задача 4

Напечатать все простые числа меньше 1000.

Задача 5

Напечатать все простые числа Фибоначи меньше 1000.

Задача 6

Дана произвольная строка. Найти и напечатать самый длинный фрагмент этой строки, который встречается больше одного раза (фрагменты не должны пересекаться).