

Python



План занятия

- "Сложные" типы данных
- Обработка исключений (ошибок)
- Функции
- Работа с командной строкой
- Пакеты и модули, как ими управлять
- Дебаг приложения (выполнение приложения по шагам)

"Сложные" типы данных

Неизменяемые типы (immutable):

- Кортежи (`tuple()`)

Изменяемые типы (mutable):

- Списки (`list()`, `set()`, `deque()`)
- Словари (`dict()`, `OrderedDict()` `defaultdict()`)

«Присваивание никогда не копирует
данные!».

—Знающие люди

Практика

`complex_types/*.py`

Какова сложность операций?

<https://wiki.python.org/moin/TimeComplexity>

Программы ломаются

```
s = int(some_value)
d = 3 / some_delta
l = len(some_var)
```

Обработка исключений -
еще один способ управления
логикой программы

Практика

`try_except.py`

Функция в Python

```
def my_function(input_var1, input_var2):  
    print(type(input_var1), type(input_var2))  
    # print type(input_var1), type(input_var2)  
    types_match = type(input_var1) == type(input_var2)  
    return types_match
```

Где:

`def` – ключевое слово для объявления функции

`my_function` – имя функции (бывают функции без имен)

`input_var1, input_var2` – входные параметры

`return` – ключевое слово, чтобы вернуть значение туда, где функция была вызвана

«Зачем нам нужны функции?»

– Не знающие люди

Практика

`course2/functions.py`