

# Для кого наш курс?

Для новичков

# Чему мы научимся?

- Синтаксису и основам Python
- Работе со стандартной библиотекой
- Работе с зависимостями
- Работе с популярными инструментами

<https://jupyter.org/try>

```
print('Hello, world!')
```

print()

# Типы данных

«Все в Python - объект».

–Знающие люди (когда-нибудь и мы все поймем!)

# Числовые типы данных в Python

- `int()`, например: 4
- `float()`, например: 3.14
- `complex()`, например:  $1j$ , такая что  $1j^2 = -1$



# Математика и логика

- Математические операторы: + - \* / ^ %
- Логические операторы: > < == != >= <=

# Практика

Файл "types/numeric\_types.py"

# bool()

- В Python истина обозначается как `True`
- Ложь обозначается как `False`

```
print(True) # True
print(False) # True
print(1 < 3) # True
print(3 == 3.0) # True
print(4 - 3.0 == 1.0) # True
print(4 >= 5) # False
```

# Математика и логика

- Логические операторы: **and or not**

# Практика

Файл "types/bool\_type.py"

# Строки

- Представляют текстовые данные

```
print('Hello world')  
print("Hello world")  
print("""
```

```
print("a" == 'a') # True
```

# Кодировки

Unicode - стандарт кодирования символов, позволяющий представить знаки почти всех письменных языков в виде числовых кодов.

ASCII - название таблицы сопоставления популярных символов с числовыми кодами внутри Unicode (подмножество Unicode символов).

Варианты представления Unicode:

- UTF-8 (Unicode transformation format)
- UTF-16
- UTF-32

Существуют не-Unicode кодировки: windows-1251, koi8-r и другие

# Практика

Файл "string\_types.py"



# None

Специальный тип данных для обозначения "ничего", "пустоты".

```
print (None * 4) # Oops!
```

# Практика

Файл "types/None\_type.py"

# input()

```
input('What is your name?')
```

# Переменные

Нам нужна ВОЗМОЖНОСТЬ  
называть наши "штуки"!

```
my_name = 'Ivan'
```

# Практика

Файл "vars.py"

Сравнение: **is** / **==**

# Практика

Файл "compare.py"



# Логика

# Два основных способа управления логикой программы

Условия

Циклы

# УСЛОВИЯ

УСЛОВИЯ ВЫГЛЯДЯТ ТАК:

```
if condition1:  
    print('condition1')  
elif condition2:  
    print('condition2')  
else:  
    print('other condition')
```

# Тернарный оператор

```
print('some' if condition1 else 'other')
```

# Практика

Файл "ifs.py"

# ЦИКЛЫ

Какие бывают циклы?

- С пред-условием: `while`
- "Перебирающие" итераторы: `for`

# Практика

Файл "loops.py"

# На следующем занятии

- Составные типы данных
- Простейшие алгоритмы и их сложность
- Функции



# Полезные материалы

<http://pythontutor.com/>

[pep8.ru/doc/pep8](http://pep8.ru/doc/pep8)

# Помоги себе сам

Не попадайте в проблему  
XY.

<http://meta.stackexchange.com/questions/66377/what-is-the-xy-problem>



# Как мне теперь жить?

Читать:

- <http://habrahabr.ru/feed/> Хабы: Python, Django, Python3, Flask
- <https://www.reddit.com/r/Python>
- <https://www.reddit.com/r/learnpython>
- <https://changelog.com/>

Общаться:

- <https://pythondev.slack.com>
- <https://python-ru.slack.com>
- <https://chat.stackoverflow.com/rooms/6/python>
- @moscowpythonconf в telegram

Посещать:

- <http://www.moscowpython.ru/>