



# Двухдневный интенсив по разработке на языке Python. День 1

Первое знакомство с языком Python

# Вы когда-нибудь писали на Python?

1. Нет. В первый раз
2. Пробовал
3. Имею хороший опыт
  1. Учусь в школе
  2. Учусь в ВУЗе/колледже
  3. Работаю

**Ответ: 2 – 2, 1 – 2 и т.д.**



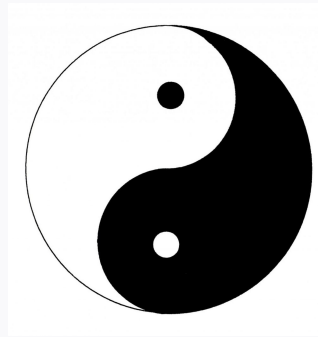
# План урока

1. Особенности языка Python.
2. Создание игры с графическим интерфейсом на Python.





Универсальность



Простота



Кроссплатформенность



Хорошая документация



Популярность



Много библиотек

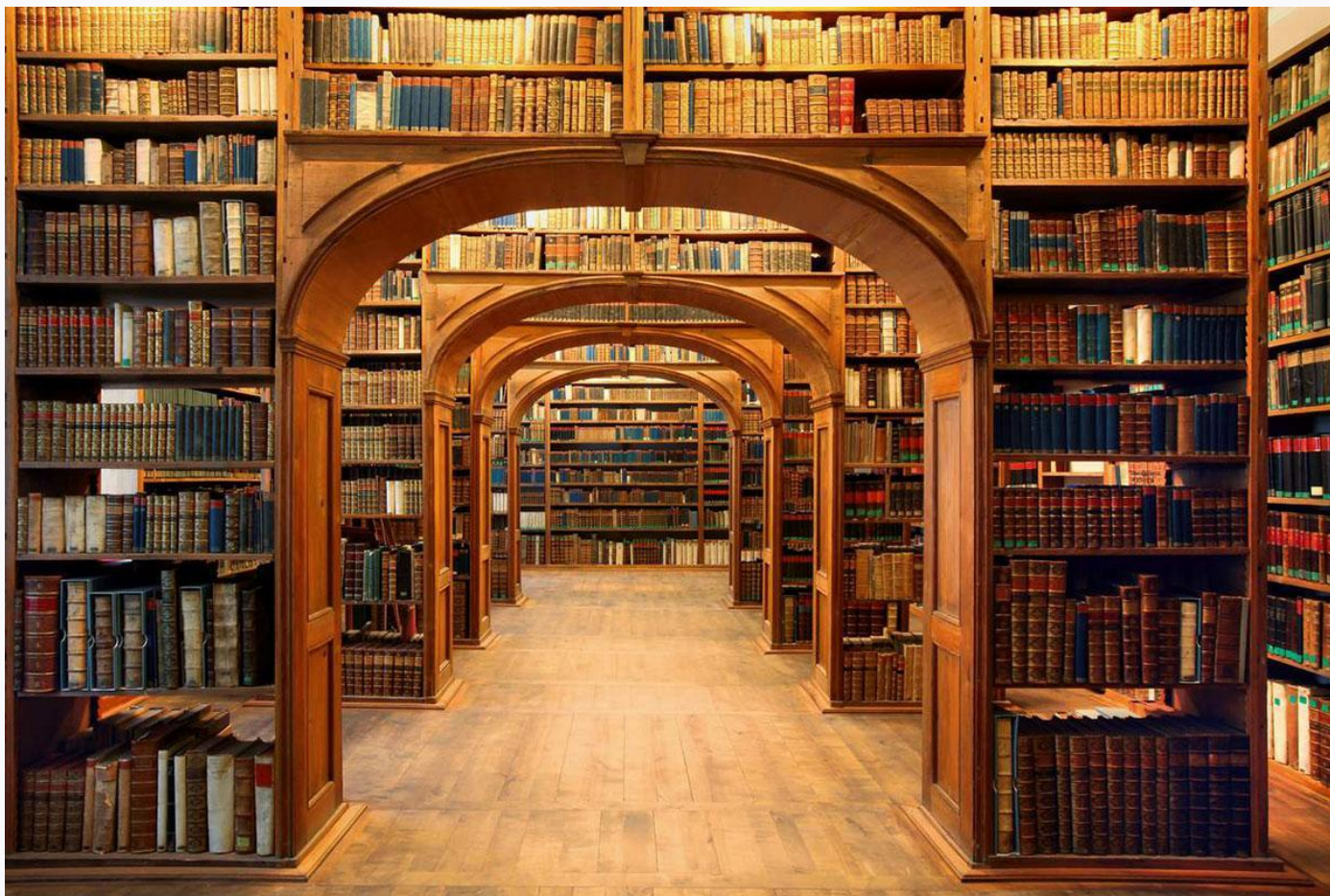


## Рейтинг ТЮВЕ (январь 2018)

Jan 2018	Jan 2017	Change	Programming Language	Ratings	Change
1	1		Java	14.215%	-3.06%
2	2		C	11.037%	+1.69%
3	3		C++	5.603%	-0.70%
4	5	▲	Python	4.678%	+1.21%
5	4	▼	C#	3.754%	-0.29%
6	7	▲	JavaScript	3.465%	+0.62%
7	6	▼	Visual Basic .NET	3.261%	+0.30%
8	16	▲▲	R	2.549%	+0.76%
9	10	▲	PHP	2.532%	-0.03%
10	8	▼	Perl	2.419%	-0.33%
11	12	▲	Ruby	2.406%	-0.14%
12	14	▲	Swift	2.377%	+0.45%

Рейтинг языков программирования <https://www.tiobe.com/tiobe-index/>





Много библиотек



# Много библиотек

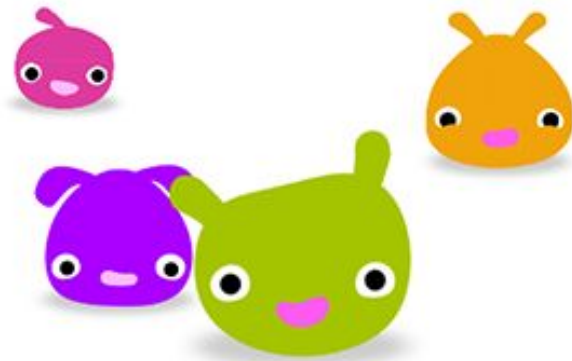
Центральное хранилище

Список Python



<https://pypi.org/pypi?%3Aaction=browse>

[python.com/](https://python.com/)





Хорошая документация





# Хорошая документация

## Module `time`

Time access and conversions.

### 8.1.1. Available Types

#### `class datetime.date`

An idealized naive date, assuming the current Gregorian calendar always was, and always will be, in effect. Attributes: `year`, `month`, and `day`.

#### `class datetime.time`

An idealized time, independent of any particular day, assuming that every day has exactly 24\*60\*60 seconds (there is no notion of “leap seconds” here). Attributes: `hour`, `minute`, `second`, `microsecond`, and `tzinfo`.

#### `class datetime.datetime`

A combination of a date and a time. Attributes: `year`, `month`, `day`, `hour`, `minute`, `second`, `microsecond`, and `tzinfo`.

#### `class datetime.timedelta`

A duration expressing the difference between two `date`, `time`, or `datetime` instances to microsecond resolution.

Онлайн-документация <https://docs.python.org/3/library/index.html>

Встроенная документация – функция `help()`



# Что пишут на Python?



Системные утилиты



Веб-сайты (Django, Flask и другие)



Научные расчёты (NumPy, SciPy)



Игры (Pygame)



Desktop-приложения (tkinter, PyQt, wxPython)

Python                      C/C++



Прототипы



Мобильные приложения (kivy)



# Начинаем разработку!



# От простого к сложному

```
>>>  
>>> print("Hello World!")  
Hello World!  
>>> |
```

```
66  
67 def grid_x(curr, near):  
68     ''' Отрисовка расположения двух клеток  
69     ...  
70     if near is not None:  
71         curr.grid(row=curr.row, column=curr.column)  
72         near.grid(row=near.row, column=near.column)  
73  
74  
75 def exchange(curr, near):  
76     ''' Обмен местами клеток в общем списке  
77     ...  
78     if near is not None:  
79         ci = curr.row * SIDE + curr.column  
80         ni = near.row * SIDE + near.column  
81         labels[ci], labels[ni] = labels[ni], labels[ci]  
82
```



Спасибо за работу!  
До завтра.

