



python

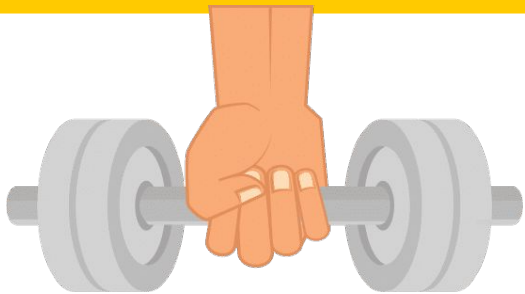
```
def dotwrite(ast):  
    nodename = getNodeName(ast[0],ast[0])  
    label=symbol.symbol(ast[0],ast[0])  
    print ' %s [%s] (%s)' % (ast[0], label),  
    if isinstance(ast[1],list):  
        if ast[1]:  
            print '  
            else:  
                print ''  
        else:  
            print ''  
            children = []  
            for n, child in enumerate(ast[1:]):  
                children.append(dotwrite(child))  
            print ' %s-> (%s)' % (nodename, children)  
            for name in children:  
                print ' %s' % name,
```

# Разбор домашнего задания



## Цели занятия

- понять, как запускается код
- научиться работать с питоном в консоли
- (заодно вспомнить, что такое консоль)



# Разминка



# Генератор определений

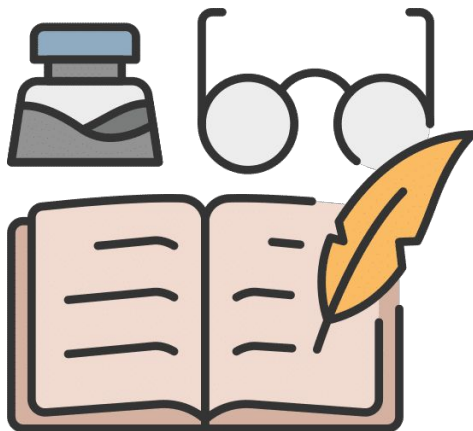
1. Делимся на команды
2. Получаем термин
3. Придумываем к нему определение  
(на вашу креативность)
4. Слушаем все определения
5. Голосуем за то, которое считаем правдивым

+1 очко, если угадали правдивое

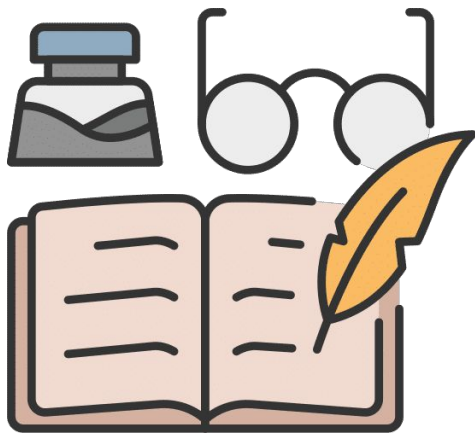
+1 очко, если за ваше  
проголосовали



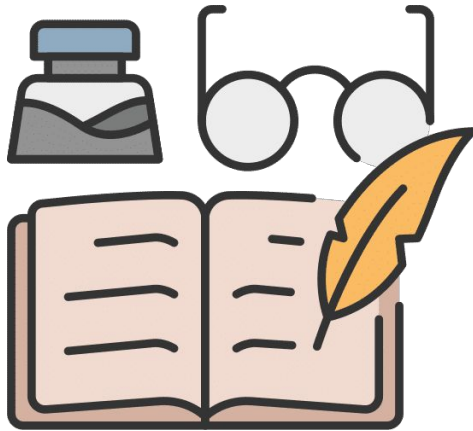
# БРУСБАРТ



# ЛОГИРОВАНИЕ



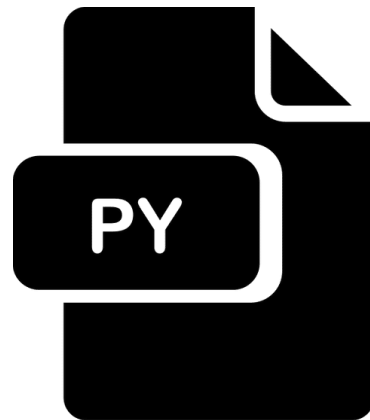
# ИНТЕРПРЕТАТОР



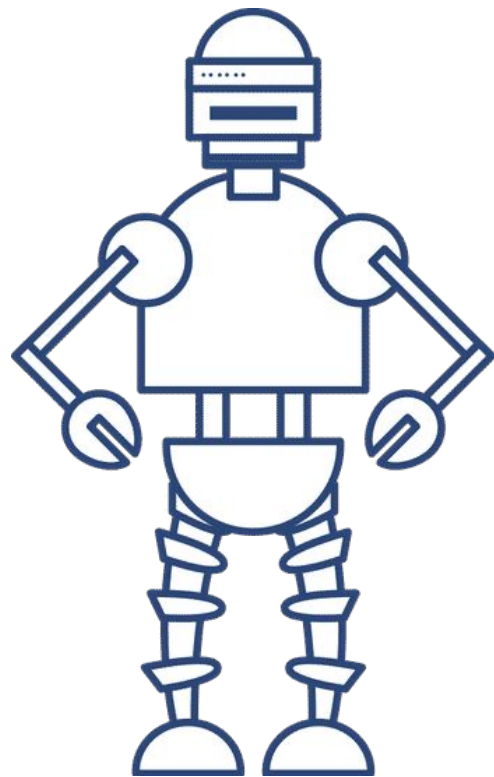


# ИНТЕРПРЕТАТОР

*Молодцы, кто вспомнил!*

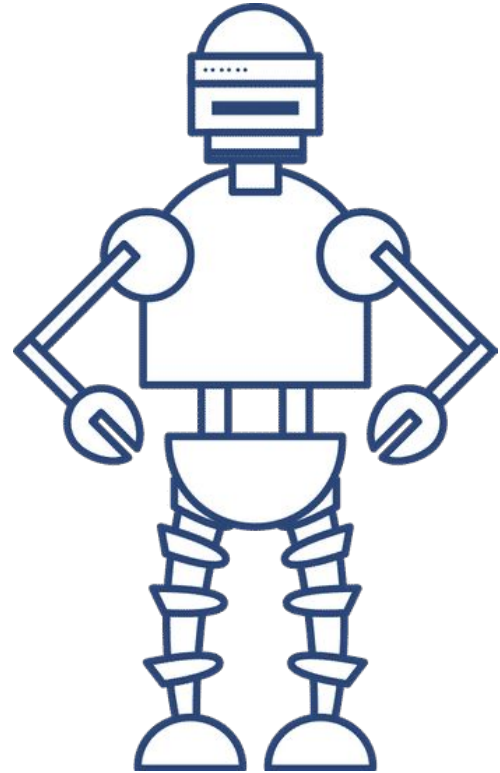


Как запускается наш код?

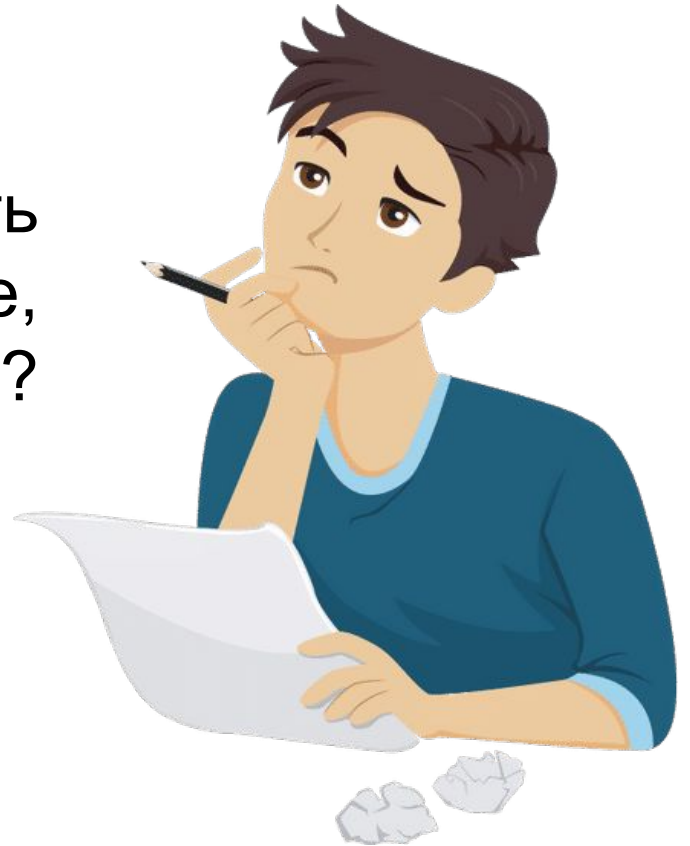


# Как запускается наш код?

**Интерпретатор** — это программа, которая построчно **преобразует текст** программы, написанной человеком, в программу на машинном языке, «понятную» компьютеру, и сразу **выполняет** указанные действия.

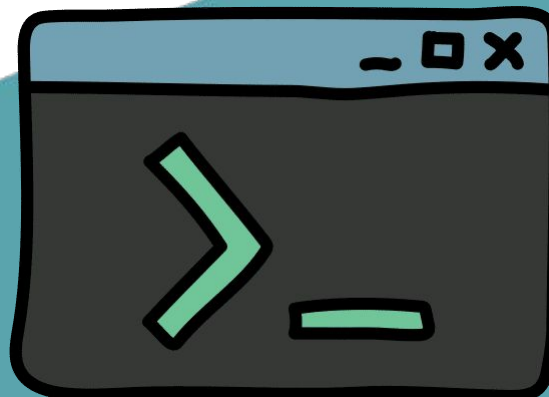


Можно ли запустить  
программу на Питоне,  
имея только её код?





# Работа с консолью



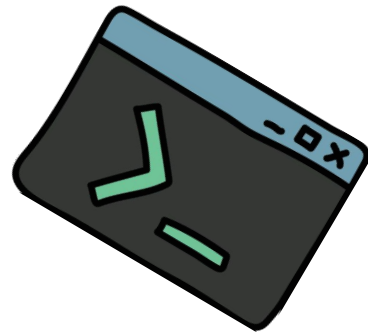


Как выглядит консоль в Windows?  
(командная строка)

Как её открыть?

Когда и зачем мы её уже использовали?

# Как открыть консоль?



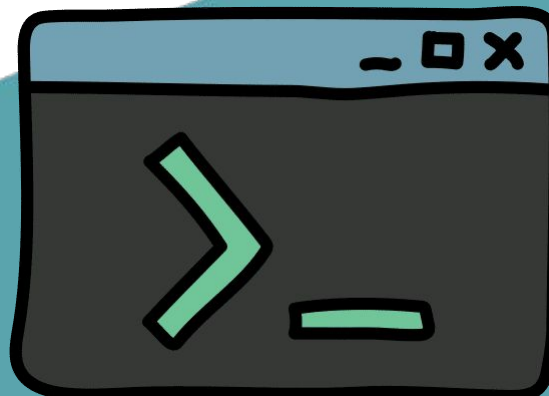
1. через поиск “командная строка”
2. нажать windows+R, затем написать cmd
3. в проводнике вместо пути к папке написать cmd



# Работа с консолью

1. Как писать и запускать код на питоне не открывая никаких специальных программ?

Открыть консоль  
Написать “python”





C:\windows\system32\cmd.exe - python

Microsoft Windows [Version 10.0.18362.900]

(c) Корпорация Майкрософт (Microsoft Corporation), 2019. Все права защищены.

C:\Users\SHP>python

Python 3.7.7 (tags/v3.7.7:d7c567b08f, Mar 10 2020, 10:41:24) [MSC v.1900 64 bit (AMD64)] on win32

Type "help", "copyright", "credits" or "license" for more information.

>>>

*Если видим такое сообщение, значит питон установлен.  
Ура!*

Cit. C:\windows\system32\cmd.exe - python

```
Microsoft Windows [Version 10.0.18362.900]
(c) Корпорация Майкрософт (Microsoft Corpora

C:\Users\SHP>python
Python 3.7.7 (tags/v3.7.7:d7c567b08f, Mar 10
Type "help", "copyright", "credits" or "lice
>>> 11 * 4
44
>>> print("Hello")
Hello
>>> a = 10
>>> b = 3
>>> print(a+b)
13
>>> █
```

*Можем писать и  
запускать код  
прямо здесь.*

*Удобно для  
быстрой проверки  
или подсчетов.*



# Работа с консолью

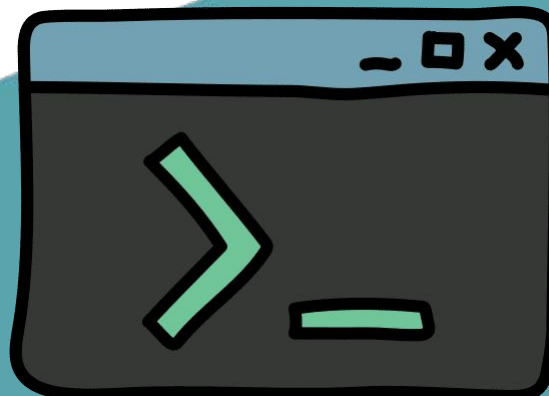
1. Как писать и запускать код на питоне не открывая никаких специальных программ?

Открыть консоль

Написать “python”

Для окончания работы  
с командами пишем:

```
>>> exit()
```



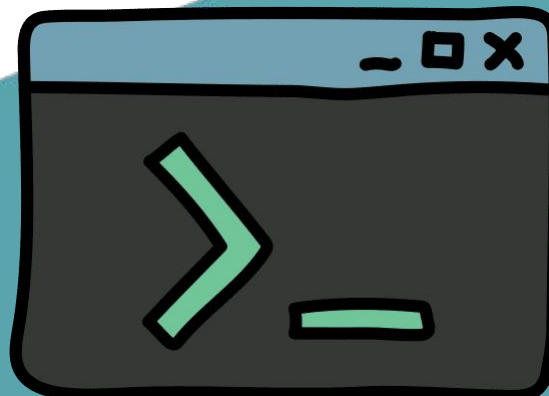


# Работа с консолью

2. Как запустить файл с кодом через консоль?

Открыть консоль

Написать “python путь/имяфайла.py”

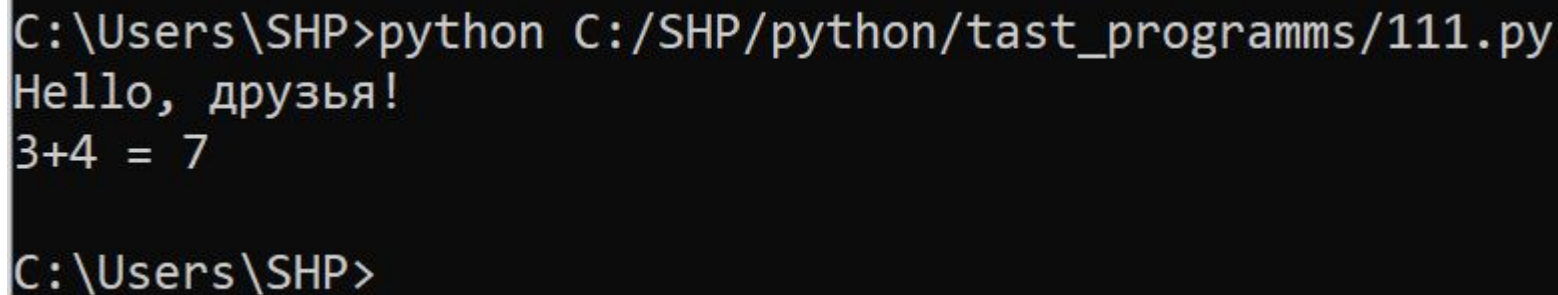


## Наша программа



```
111.py x 22.py x тест.py x game_of_a_student.py x
1
2 print("Hello, друзья!")
3 print("3+4 =", 3+4)
4
```

Запуск через консоль



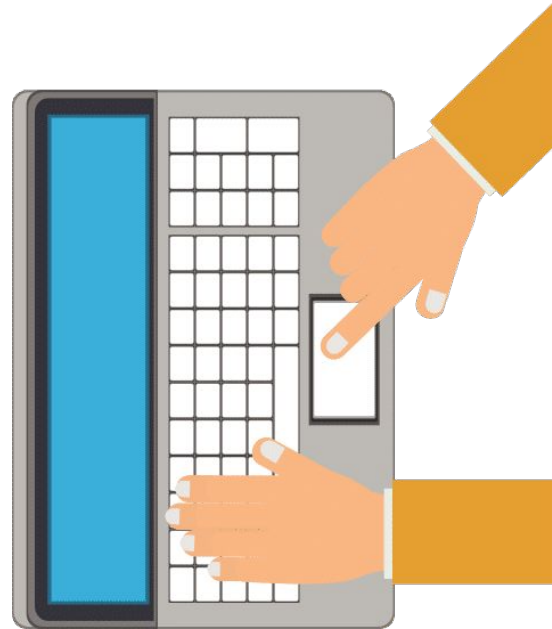
```
C:\Users\SHP>python C:/SHP/python/tast_programms/111.py
Hello, друзья!
3+4 = 7

C:\Users\SHP>
```

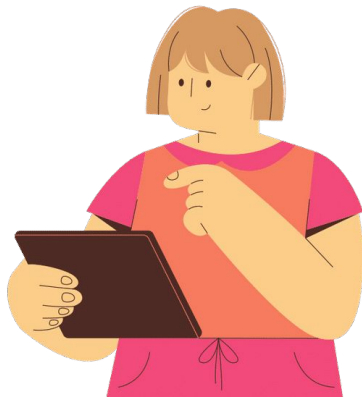
# Практика

Попробуйте написать в  
консоле

```
>>> import this
```



1 0 1 0  
0 1 0 1  
1 0 1 0

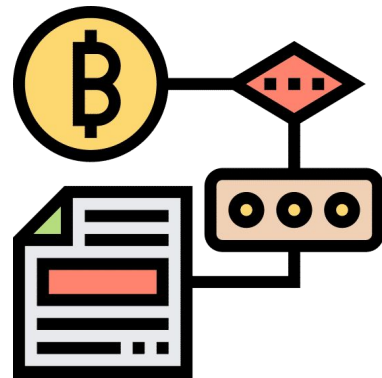


1 0 1 0  
0 1 0 1  
1 0 1 0

Вперёд в прошлое

# Алгоритмы

Какие алгоритмы мы изучили?

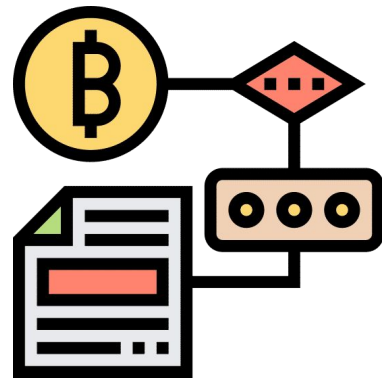




# Алгоритмы

Какие алгоритмы мы изучили?

- сумма, произведение
- минимум, максимум
- количество
- выделение цифр в числе





# Алгоритм суммы

Какие шаги?





# Алгоритм суммы

Какие шаги?

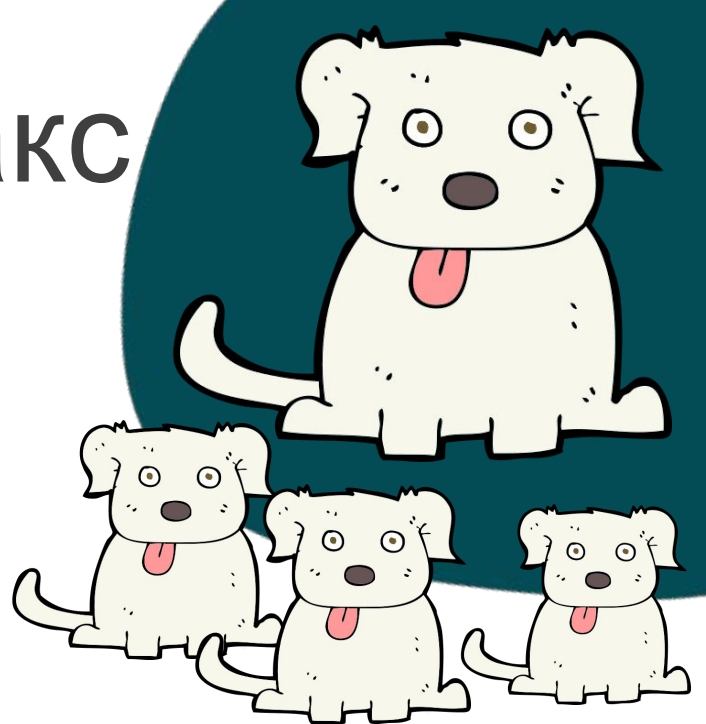
1. создать копилку  $s = 0$
2. считать данные
3. добавить числа в переменную копилку
4. печать результата





# Алгоритм мин/макс

Какие шаги?





# Алгоритм мин/макс

Какие шаги?



1. считать данные
2. создать хранилище  
 $\text{mini} = \text{первому числу}$
3. сравнить каждое число с  $\text{mini}$ , если найден  
новый, то переопределить  $\text{mini}$
4. печать результата

# Алгоритм подсчёта

Как работает?

Как написать код?

# Цикл for

Зачем нужны циклы?

Как записать его на Питоне?



# Цикл for

## Зачем нужны циклы?

Чтобы делать код короче и не дублировать повторяющиеся действия.

## Как записать его на Питоне?

```
for i in range(10):
```





# Синтаксис for

*счетчик*

*сколько раз повторить*

```
for i in range(5):  
    print("Привет!")  
    print("Пока!")
```

*что повторяем?*

*range делает ряд чисел:*  
*0 1 2 3 4*

# Алгоритм выделения цифр

Как работает?

Какие виды деления нужны?

Как написать код?

# Алгоритм вычисления цифр

```
n = 12345
for i in range(5):
    num = n % 10
    print(num)
    n = n // 10
```

Печатаем  
цифру

Длина числа

Выделяем  
последнюю  
цифру

Отрезаем  
последнюю  
цифру

# Практика

Задания в EduApp



# Цикл while

## Работа с mind-картой

Заполните пропуски и ответьте на вопросы про цикл while.  
В качестве источников информации можно использовать материалы из ЕдуАппа.



Можно в парах :)



переводится с англ.  
как \_\_\_\_\_



нарисуйте  
Мистера Байна:



While - это  
определённый / НЕ определён

# Обсудим ответы



Что это?  
☐ выступ ☐ отступ  
Зачем он нужен?

Пример:



- Тут будет:
- ☐ вопрос
  - ☐ ответ
  - ☐ условие
  - ☐ числа

Какие математические знаки  
помогают нам написать while .... :  
> больше

Автор:

группа:

# Практика

Задания в EduApp



# Итоги занятия





# Итоги занятия

- научились запускать код в консоли
- вспомнили и поработали с циклами
- поработали с разными алгоритмами



# Домашнее задание в ЕдуАпп!

