

Содержание

Цикл while2

 Задания по while3

Массивы5

 Задания по массивам7

Цикл for9

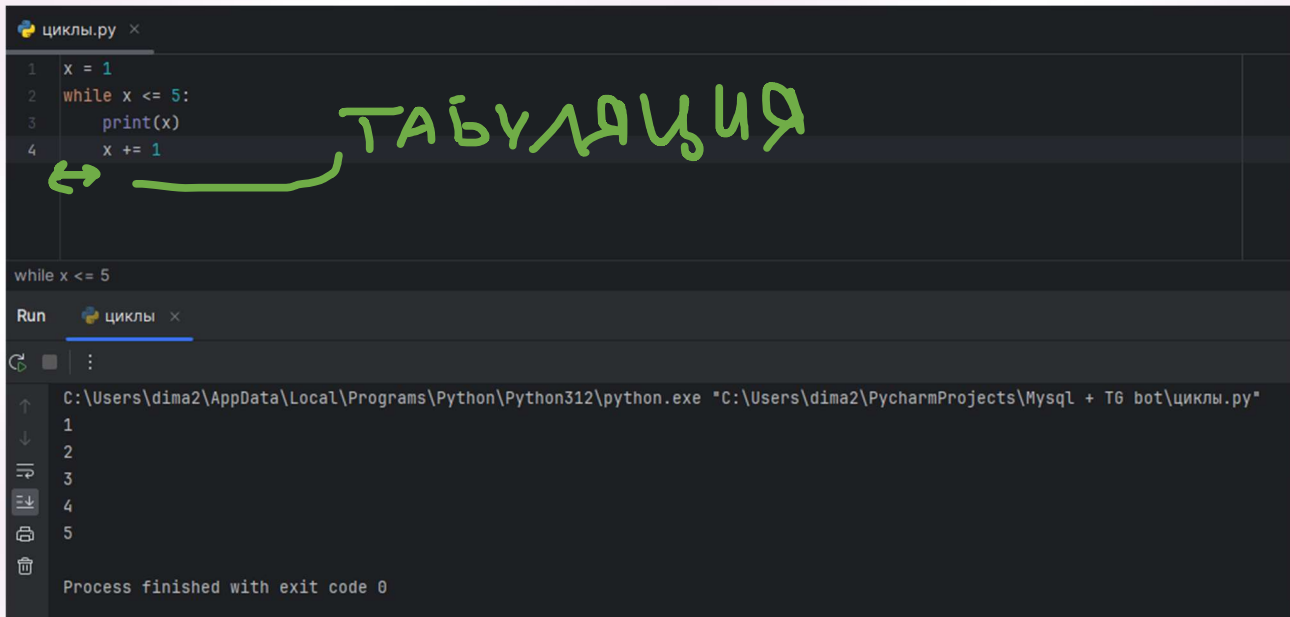
 Задания на оператор for11



Цикл while

Цикл **while** повторяет блок кода, пока условие остаётся **True**
Код цикла выделяется табуляцией (клавиша Tab или 4 пробела)

```
x = 1
while x <= 5:
    print(x)
    x += 1
```



The screenshot shows a PyCharm IDE window titled 'циклы.py'. The code editor contains the following Python code:

```
1 x = 1
2 while x <= 5:
3     print(x)
4     x += 1
```

A green handwritten arrow points from the word 'ТАБУЛЯЦИЯ' (TABULATION) to the indentation of the code block. Below the code editor, the 'Run' tab shows the execution output:

```
while x <= 5
1
2
3
4
5
```

The console output shows the numbers 1 through 5, each on a new line. At the bottom, it says 'Process finished with exit code 0'.

если условие будет **True**, получится **бесконечный цикл**

```
x = 1
while True:
    print(x)
    x += 1
```



```
циклы.py x
1 x = 1
2 while True:
3     print(x)
4     x += 1

while True
Run циклы x
4221908
4221909
4221910
4221911
4221912
4221913
```

Задания по while ♪•ω•^♪

1. Выведи числа от 1 до 10

Пример: вывод чисел от 100 до 111

```
Project i.py x
1 x = 100
2 while x <= 105:
3     print("x =", x)
4     x += 1

while x <= 105
Run циклы x
C:\Users\dima2\AppData\Local\Programs\Python
x = 100
x = 101
x = 102
x = 103
x = 104
x = 105

Process finished with exit code 0
```

2. Спроси у пользователя пароль. Повторяй ввод, пока не введёт "qwerty"

Пример: спрашиваю у пользователя цвет, повторяю вводу, пока не введет "голубой"

```
Project .py x
1 while input("Введи цвет: ") != "голубой":
2     print("Не не этот")
3
4     print("Я так и думал...")

while input("Введи цвет: ") != ...

Run циклы x

C:\Users\dima2\AppData\Local\Programs\Python\Python312\
Введи цвет: aaa
Не не этот
Введи цвет: Зеленый
Не не этот
Введи цвет: Красный
Не не этот
Введи цвет: голубой
Я так и думал...

Process finished with exit code 0
```



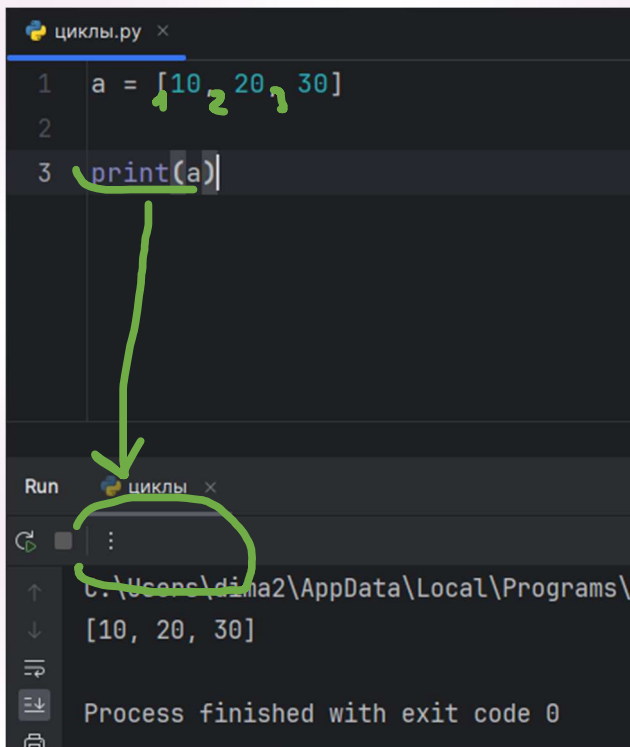
Массивы

Массив — это когда несколько значений можно записать в одну переменную

```
a = [10, 20, 30]
```

Данный массив содержит 3 элемента: 10, 20, 30

Вывод массива через **print**



The screenshot shows a Python IDE window titled 'циклы.py'. The code in the editor is:
1 a = [10, 20, 30]
2
3 print(a)
The numbers 1, 2, and 3 are written in green next to the lines. A green arrow points from the 'print(a)' line down to the Run button. The Run button is also circled in green. Below the Run button, the output is shown: '[10, 20, 30]'. At the bottom, it says 'Process finished with exit code 0'.

Чтобы взять какое-то конкретное значение из массива, обращаемся к нему через

имя_массива[индекс]

Причем **элементы нумеруются, начиная с нуля**

Пример:

```
a = [10, 20, 30]  
  
print("Первый элемент =", a[0])  
print("Второй элемент =", a[1])  
print("Третий элемент =", a[2])
```

Вывод этого кода

```
Run  циклы x
C:\Users\dima2\AppData\Local\Programs\Python\Python39-64\python.exe
Первый элемент = 10
Второй элемент = 20
Третий элемент = 30
Process finished with exit code 0
```

Индикс 0
1
2

По такому же принципу можно менять элементы массива

```
a = [10, 20, 30]
print("Начальный массив =", a)

a[0] = 500
a[1] = 1000
a[2] = "aaaa"
print("Измененный массив =", a)
```

Вывод:

```
Run  циклы x
C:\Users\dima2\AppData\Local\Programs\Python\Python39-64\python.exe
Начальный массив = [10, 20, 30]
Измененный массив = [500, 1000, 'aaaa']
Process finished with exit code 0
```

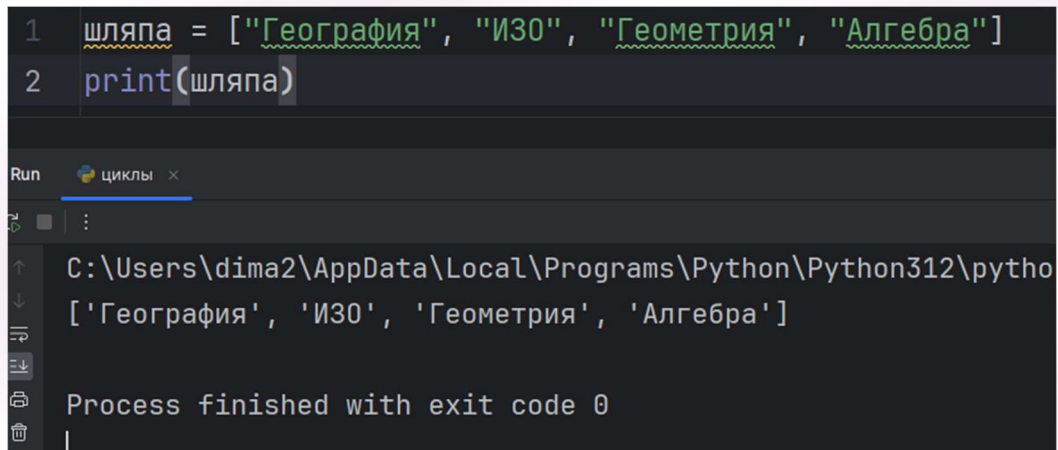


Задания по массивам (Λ">~<"Λ)

1. Создай массив из самых душных предметов в школе

Пример:

```
1 шляпа = ["География", "ИЗО", "Геометрия", "Алгебра"]
2 print(шляпа)
```

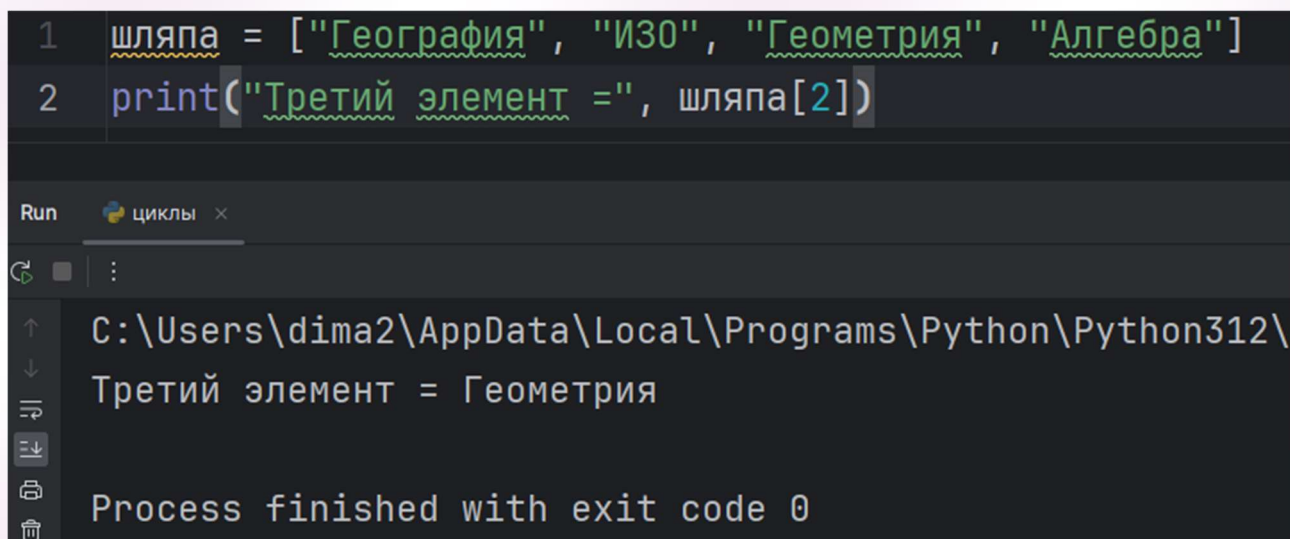


```
C:\Users\dima2\AppData\Local\Programs\Python\Python312\python.exe
['География', 'ИЗО', 'Геометрия', 'Алгебра']
Process finished with exit code 0
```

2. Выведи первый элемент из созданного массива

Пример: вывожу третий элемент из своего массива

```
1 шляпа = ["География", "ИЗО", "Геометрия", "Алгебра"]
2 print("Третий элемент =", шляпа[2])
```



```
C:\Users\dima2\AppData\Local\Programs\Python\Python312\python.exe
Третий элемент = Геометрия
Process finished with exit code 0
```

3. Создай числовой массив, в котором будет твой возраст, число, месяц и год твоего рождения.
Выведи оттуда месяц своего рождения

Пример: вывожу год своего рождения



```
1 шляпа = [23, 23, 8, 2002]
2 print("Год рождения =", шляпа[3])
```

Run циклы x

C:\Users\dima2\AppData\Local\Programs\Python

Год рождения = 2002

Process finished with exit code 0

4. Выведи созданный в предыдущем задании массив
Умножь каждый его элемент на 10
Выведи массив снова

Пример: прибавляю к каждому элементу число 1000

```
1 шляпа = [23, 23, 8, 2002]
2 print("Массив =", шляпа)
3 шляпа[0] = шляпа[0] + 1000
4 шляпа[1] = шляпа[1] + 1000
5 шляпа[2] = шляпа[2] + 1000
6 шляпа[3] = шляпа[3] + 1000
7 print("Новый массив =", шляпа)
```

Run циклы x

C:\Users\dima2\AppData\Local\Programs\Python

Массив = [23, 23, 8, 2002]

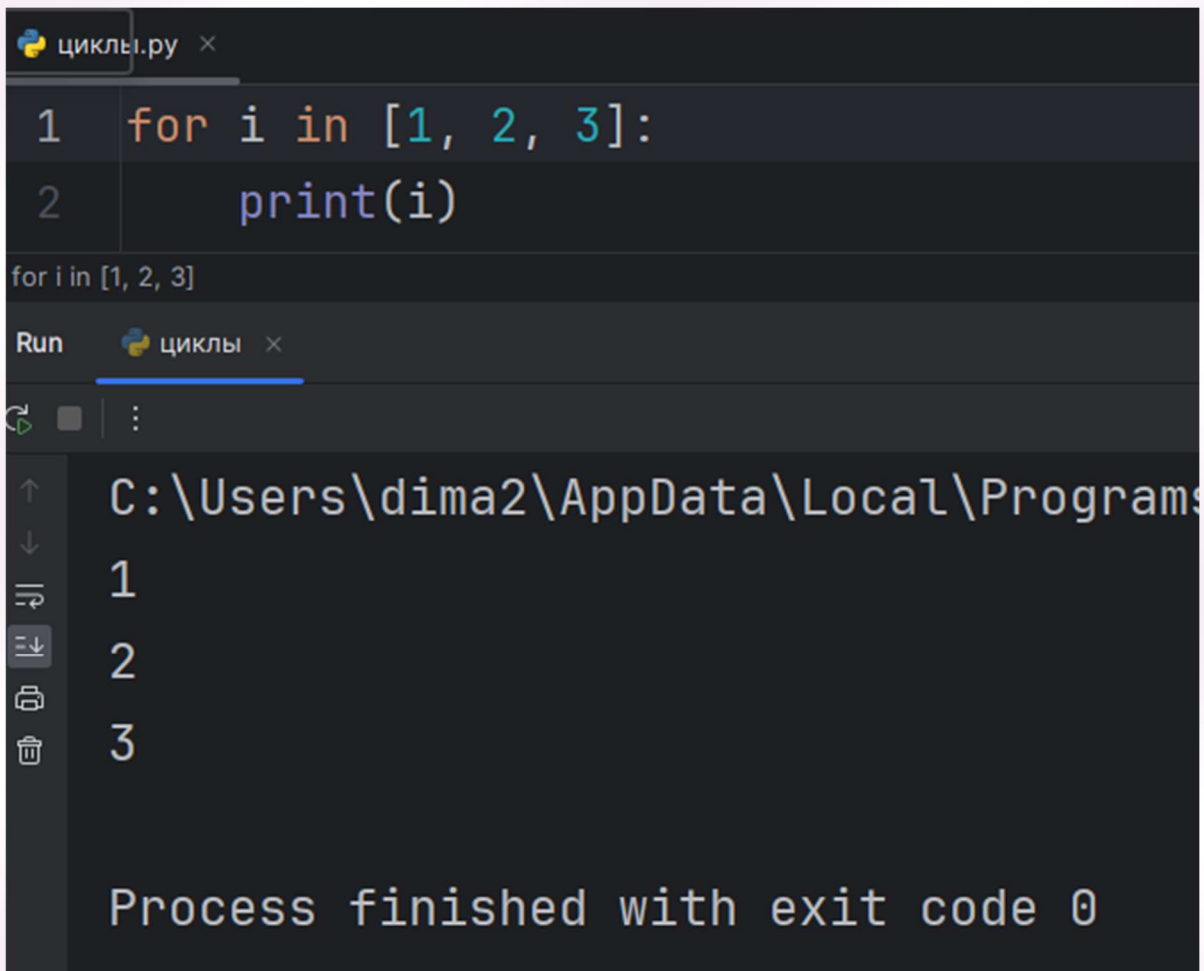
Новый массив = [1023, 1023, 1008, 3002]

Цикл for

Цикл for перебирает элементы (строки, массивы, диапазоны) – то есть **пробегаются по каждому элементу**

Пример с массивом:

```
for i in [1, 2, 3]:  
    print(i)
```

A screenshot of a Python IDE window titled 'циклы.py'. The code editor shows a for loop:

```
1 for i in [1, 2, 3]:  
2     print(i)
```

 Below the code, the output of the program is displayed:

```
for i in [1, 2, 3]  
1  
2  
3
```

 The output is shown in a terminal-like window with the title 'Run' and 'циклы'. The path to the file is visible: 'C:\Users\dima2\AppData\Local\Programs'. The message 'Process finished with exit code 0' is shown at the bottom.

Пример с текстом (**пробегаются по каждому символу**):

```
for i in "чупакабра":  
    print(i)
```

```
циклы.ру ×
1 for i in "чупакабра":
2     print(i)

for i in "чупакабра"
Run циклы ×
C:\Users\dima2\AppData\Local\Programs
ч
у
п
а
к
а
б
р
а

Process finished with exit code 0
```

Можно создавать массив не прописывая каждый элемент самому – с помощью функции

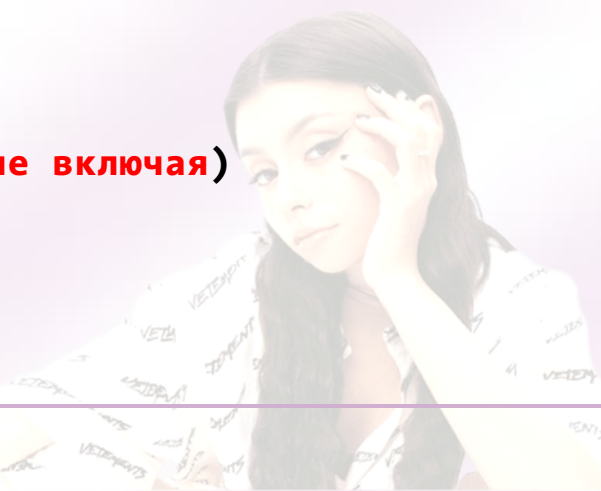
range

- **range(x)** – числа от 0 до n (не включая)

```
циклы.ру ×
1 for i in range(5):
2     print(i)

for i in range(5)
Run циклы ×
C:\Users\dima2\AppData\L
0
1
2
3
4
```

- **range(a, b)** – числа от a до b (не включая)



```
циклы.py x
1 for i in range(5, 10):
2     print(i)

Run циклы x
C:\Users\dima2\AppData\
5
6
7
8
9
```

Задания на оператор for ≧^" + " ^≦

1. Создай массив из своих любимых игр и выведи каждую игру **через for**

Пример: создаю массив из игр на своем компе и вывожу

```
циклы.py x
1 games = ["formula 1 2022", "Assetto Corsa Competizione", "CS2", "PUBG"]
2 print("Мои игры:")
3 for i in games:
4     print(" -", i)
5

Run циклы x
C:\Users\dima2\AppData\Local\Programs\Python\Python312\python.exe
Мои игры:
- formula 1 2022
- Assetto Corsa Competizione
- CS2
- PUBG

Process finished with exit code 0
```

2. Выведи числа от 1 до 10

Пример: вывожу числа от 0 до 8

```
циклы.py x
1 for i in range(0, 8):
2     print(i)

for i in range(0, 8)
Run циклы x
C:\Users\dima2\AppData\Local\Programs
0
1
2
3
4
5
6
7

Process finished with exit code 0
```

3. Выведи только чётные числа от 2 до 20

Пример: вывожу четные числа от 10 до 30

```
циклы.py x
1 for i in range(10, 31):
2     if (i % 2 == 0):
3         print(i)
4

for i in range(10, 31)
Run циклы x
C:\Users\dima2\AppData\Local\Programs\Py
10
12
14
16
18
20
22
24
26
28
30

Process finished with exit code 0
```



4. Посчитай сумму чисел от 1 до 100

Пример: считая произведение чисел от 5 до 10

```
циклы.py x
1 pr = 1
2 for i in range(5, 10):
3     pr = pr * i
4
5 print("Произведение =", pr)

Run циклы x
C:\Users\dima2\AppData\Local\Programs\Python\Python3
Произведение = 15120
Process finished with exit code 0
```

5. Выведи квадраты чисел от 1 до 10

Пример: вывожу кубы чисел от 2 до 8

```
циклы.py x
1 for i in range(2, 9):
2     print("Число =", i, " | Куб =", i * i * i)
3
4

Run циклы x
C:\Users\dima2\AppData\Local\Programs\Python\Python3
Число = 2 | Куб = 8
Число = 3 | Куб = 27
Число = 4 | Куб = 64
Число = 5 | Куб = 125
Число = 6 | Куб = 216
Число = 7 | Куб = 343
Число = 8 | Куб = 512
Process finished with exit code 0
```

