

# Программирование на Python

Презентация занятия

## **Изучение возможностей и синтаксиса Python: Циклы.**

5 занятие

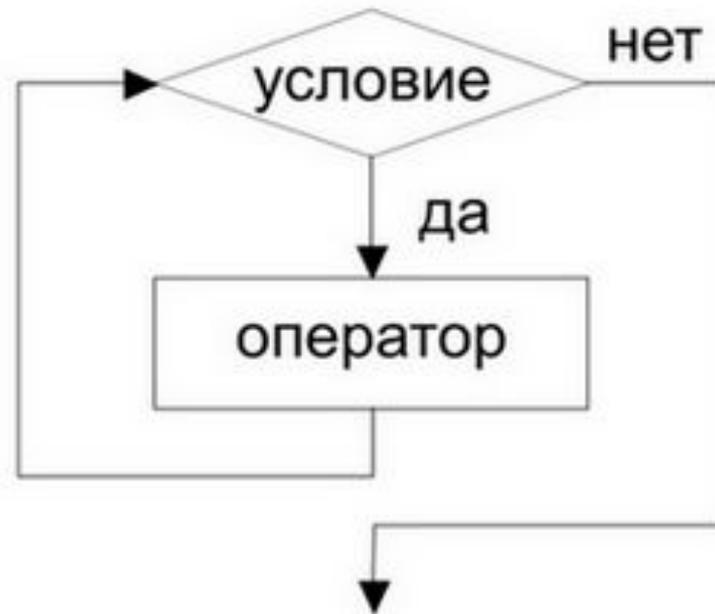


**инжинириум**<sup>®</sup>

МГТУ им. Н.Э. Баумана

## Тема: Изучение возможностей и синтаксиса Python: Циклы.

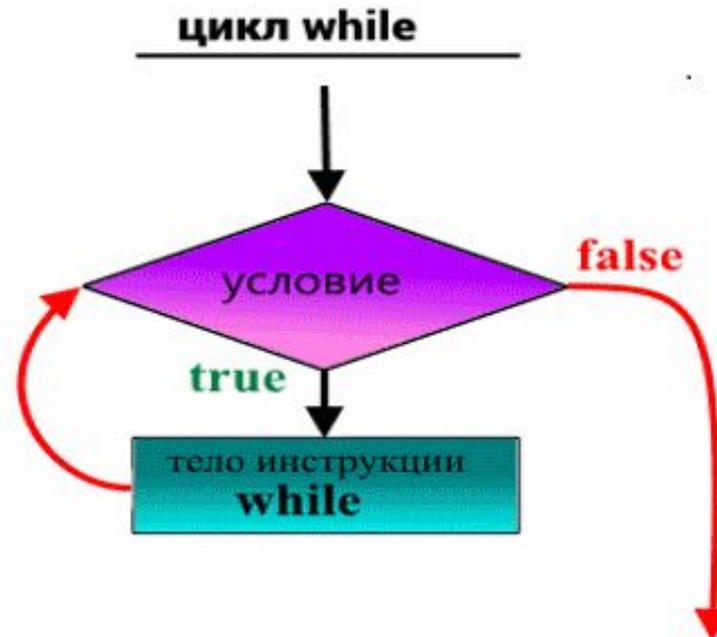
Циклы - это инструкции, выполняющие одну и ту же последовательность действий многократно.



## Тема: Изучение возможностей и синтаксиса Python: Циклы.

`while` <логическое выражение>:  
<действия, выполняемые, пока логическое  
выражение принимает значение True >

```
s = 0
n = 0
while s < 350:
    x = int(input ())
    s = s + x
    n = n + 1
print(n)
```



## Тема: Изучение возможностей и синтаксиса Python: Циклы.

Функция **range()** может принимать от одного до трех аргументов, при этом аргументами должны быть целые числа (int). По умолчанию старт равняется нулю, шаг единице.

**range**(старт, стоп, шаг)

```
range(1,8) # шаг = 1
[1, 2, 3, 4, 5, 6, 7]
range(0, 20, 5)
[0, 5,
range(0,-7, -1)
[0, -1, -2, -3, -
range(1,0)
[
]
```



## Тема: Изучение возможностей и синтаксиса Python: Циклы.

```
for i in range(a, b):  
    Тело цикла
```

```
n = int(input())  
summa = 0  
for i in range(1, n + 1):  
    summa = summa + i  
print(summa)
```

# Что выведет программа, если ввести n = 10 ?

```
line = 'hello world'  
for i in line:  
    print(i * 2, end="")  
hheelllloo wwoorrlldd
```

```
line = 'hello world'  
for i in range(0, len(line)):  
    print(i * 2, end=' ') # Здесь пробел  
0 2 4 6 8 10 12 14 16 18 20
```



## Тема: Изучение возможностей и синтаксиса Python: Циклы.

В Python оператор **break** позволяет прервать цикл при возникновении внешнего фактора.

```
for i in 'hello world':  
    if i == 'o':  
        break  
    print(i * 2, end="")
```

hheellll

```
for i in 'hello  
world':  
    if i == 'o':  
        continue  
    print(i * 2,  
end='')  
wwrrlldd
```

