

# Modules and Functions in the Python Programming Language

**Author: Alimova Rayhon Abdugafforovna**

Student of the Faculty of Information Technologies of Termiz State University

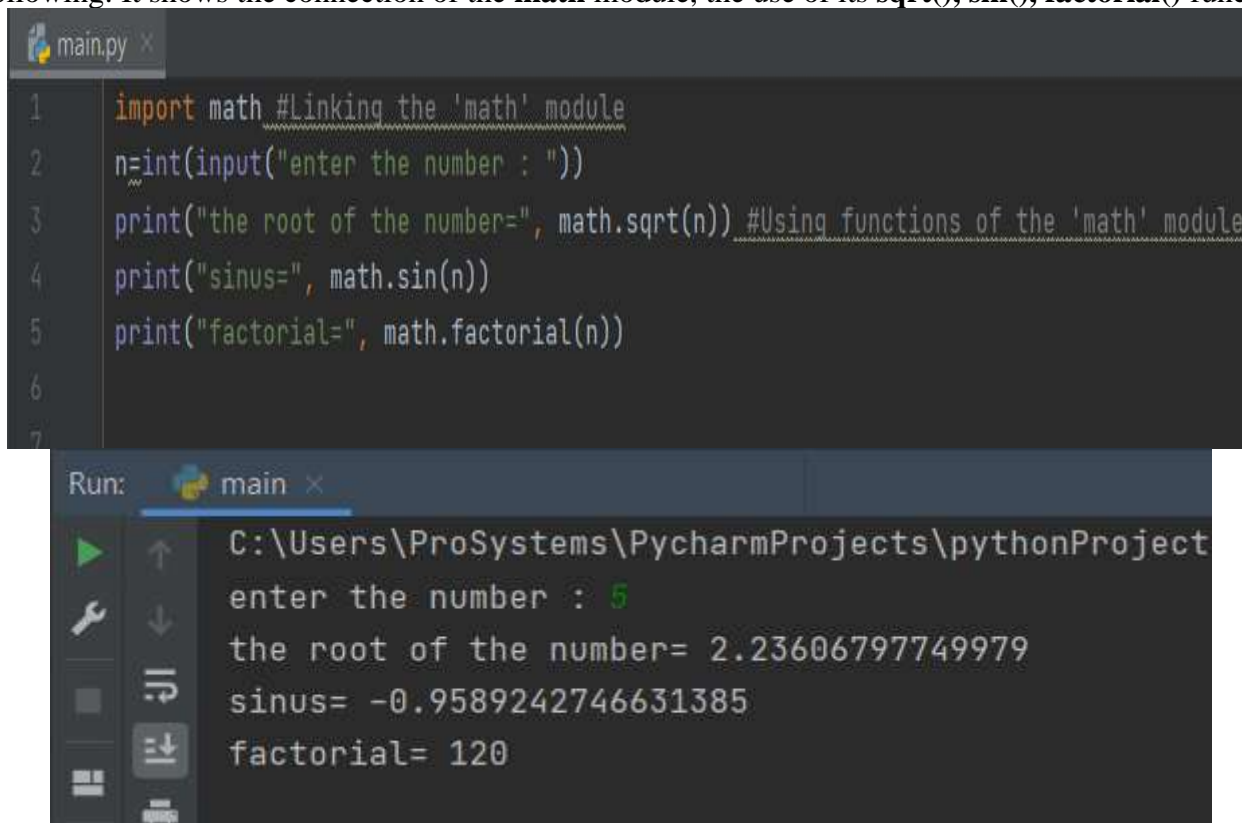
[alimovarayxon6@gmail.com](mailto:alimovarayxon6@gmail.com)

**Abstract:** This article describes some of the functions used in the python programming language, how to call them, and how to use them in a programming environment. Keywords related to them are defined.

**Keywords:** Function, parameters, arguments, math, abs(), pow(x, y), sqrt(), ceil(), floor(), min(), max(), range(), def, sin(), factorial(), turtle, random.

Another reason for the popularity of the Python programming language is its large number of library modules. Like other programming languages, python supports modules. More than a thousand library modules have been created for Python, which are designed for use in various fields. In addition, in Python, the program code can be divided into several modules and complex codes can be simplified.

There are two types of modules in Python: **standard modules** and **user-created modules**. There are more than 200 standard modules, including math, random, turtle, tkinter, http, os, sys and others. To use modules, we need to link them to our program code. If we want to connect the Math module, we write '**import math**'. If there are several modules, it can be written with a comma. For example, it can be understood through the following: It shows the connection of the **math** module, the use of its **sqrt()**, **sin()**, **factorial()** functions.



```
main.py x
1 import math #Linking the 'math' module
2 n=int(input("enter the number : "))
3 print("the root of the number=", math.sqrt(n)) #Using functions of the 'math' module
4 print("sinus=", math.sin(n))
5 print("factorial=", math.factorial(n))
6
7
```

```
Run: main x
C:\Users\ProSystems\PycharmProjects\pythonProject
enter the number : 5
the root of the number= 2.23606797749979
sinus= -0.9589242746631385
factorial= 120
```

It is more convenient to store frequently used program codes in one file in the form of a module and use them when needed. Modules are a simple python file ".py" in which functions, classes, lists are stored. We can use these files as needed in our Python programs.

Turtle - with the help of this module, we can draw various shapes and lines on the screen.

|          |      |          |      |
|----------|------|----------|------|
| Function | task | Function | task |
|----------|------|----------|------|

|            |                  |               |                                     |
|------------|------------------|---------------|-------------------------------------|
| Forward()  | walk forward     | Speed()       | Speed between 0 and 10              |
| Backward() | walk back        | Pendown()     | you can draw by lowering the pencil |
| Right()    | turn right       | Penup()       | pick up the pen                     |
| Left()     | turn left        | Pensize()     | line thickness                      |
| Bgcolor()  | background color | Pencolor()    | line color                          |
| Title()    | header name      | Clearscreen() | clear the screen                    |

The Turtle module has several other functions besides these. Now we will create a rectangle drawing program using this module.

```

main.py x
1 import turtle
2 t=turtle.Turtle()
3 t.forward(100)
4 t.right(90)
5 t.forward(100)
6 t.right(90)
7 t.forward(100)
8 t.right(90)
9 t.forward(100)
    
```

Random - this module is used to get random numbers or randomly mix list elements.

| Function           | task   | Function      | task  |
|--------------------|--|---------------|---|
| Random()           | takes one of 0 and 1                                       | Choice()      | gets a random element from the list             |
| Randint(a, b)      | takes a random number between a and b                      | Shuffle()     | shuffling the list                              |
| Randrange(a, b, c) | taking a random number from the interval a and b by step c | Uniform(a, b) | Picks a random number from the interval a and b |

As an example, using this module and its function in the program, we will create a program that displays a random number from 1 to 100.

```

main.py x
1 import random
2 print("random number: ", random.randint(1, 100))
3
    
```

A function is a piece of software that can be used multiple times. Functions allow you to save a certain block of commands with the specified name and execute this block anywhere in the program, any number of times. The names specified when the function is declared are called parameters, and the values given to it when calling the function are called arguments.

The Python programming language has ready-made modules and functions for performing mathematical operations, the name of this module is **math**. It helps us perform digital operations. Let`s look at some mathematical functions.

**abs()** – returns the positive value of the given value. Returns a positive value even if the input value is positive.

```
main.py x
1 x=float(input("Enter an arbitrary number: "))
2 y=abs(x)
3 print(y)
4
```

```
Run: main x
C:\Users\ProSystems\PycharmProjects\python
Enter an arbitrary number: -5.24
5.24
```

```
Run: main x
C:\Users\ProSystems\PycharmProjects\python
Enter an arbitrary number: 1.25
1.25
```

The **pow(x, y)** function calculates the y-rank of x. In this you will have the opportunity to calculate the numbers under the root.

```
main.py x
1 x=int(input("Enter the number x : "))
2 y=int(input("Enter the level of x : "))
3 z=pow(x, y)
4 print(z)
5
```

```
Run: main x
C:\Users\ProSystems\PycharmProjects\
Enter the number x : 12
Enter the level of x : 2
144
```

A program using this pow function can also calculate the largest powers of an arbitrary number. The function that calculates the square root of the given value is **sqrt()**.

```
main.py x
1 import math
2 x=float(input("Enter the number x : "))
3 y=math.sqrt(x)
4 print(y)
5
```

```
Run: main x
C:\Users\ProSystems\PycharmProjects\
Enter the number x : 81
9.0
```

```
Run: main ×
C:\Users\ProSystems\PycharmProjects\
Enter the number x : 1.44
1.2
```

A function that rounds a given value up to the nearest integer is **ceil()**. Using it, we create the following program:

```
main.py ×
1 import math
2 x=float(input("Enter the number x : "))
3 y=math.ceil(x)
4 print(y)
5
Run: main ×
C:\Users\ProSystems\PycharmProjects\
Enter the number x : 1.44
2
```

A function that rounds the given value down to the nearest integer – **floor()**.

```
main.py ×
1 import math
2 x=float(input("Enter the number x : "))
3 y=math.floor(x)
4 print(y)
5
Run: main ×
C:\Users\ProSystems\PycharmProjects\
Enter the number x : 1.44
1
```

The functions used to find the smallest and largest numbers are **min()** and **max()**.

```
1 x=min(5,8,15)    5
2 y=max(5,8,15)   15
3 print(x)
4 print(y)
```

Now let's get acquainted with the **range()** function. If the program needs to output numbers consecutively, then the range() function is used. It creates a list based on arithmetic progression. For example:

```
1 x = range(1, 12, 2)
2 for n in x:
3     print(n)
```

```
1
3
5
7
9
11
```

The "def" keyword is used to create a function in the Python programming language. When calling a function, we use its name, if there are parameters, we send parameters, otherwise we use empty () brackets. See the following.

```
hello.py x
1 #Functions - Programmer UZ
2
3 def programmer():
4     print("Programmer UZ")
5
6 programmer()
```

```
hello x
↑ "P:\Python tutorials\python-
↓ Programmer UZ
```

Arguments to the function must be given in parentheses next to the function name. In this case, you will be able to send specified arguments to the function.

```
hello.py x
1 #Functions - Programmer UZ
2
3 def programmer(word):
4     print("> " + word)
5
6 programmer("Hello")
7 programmer("How are you")
8 programmer("Where are you going")
9 programmer("Bye!")
```

```
hello x
↑ "P:\Python tutorials\python-3.8.1-embed-amd64\python.exe"
↓ > Hello
> How are you
> Where are you going
> Bye!
```

We can create any number of functions (division, subtraction, multiplication, addition). Many examples can be given. For example, let's create a function called "addition". This is not difficult to understand from the code written below.

```
main.py x
1 def addition(a, b):
2     print(a+b)
3
4 addition(5, 4)
5 addition(10, 12)
```

```
main x
↑ C:\Users\ProSystems
↓ 9
22
```



**Conclusion:** Modules and functions are very important for programming, they make the process of creating a program easier. You can write a whole book about the standard modules and functions of the Python programming language. Through this article, I gave information about some frequently used modules and functions and tried to explain them through small programs. It is important not to spend a lot of time in the process of creating a program and to study the modules and functions in detail in order to increase the quality of work and productivity.

**References:**

1. Normurodov Ch. B, Mengliyev Sh. A. PHP7 programming language - Tutorial - Termiz: "Khamidi private firm", 2020.
2. Maxim Kuznetsov, Igor Simdyanov, Sergey Golyshev. PHP 5 by examples. Series: On examples. Publisher: BHV – Petersburg, 2005.
3. Boltayev B, Azamatov A, Askarov A, Sodikov M, Azamatova G. Fundamentals of computer science and computer technology. Textbook for the 9th grade of general secondary schools. Tashkent: NMIU named after "Cholpon", 2015.
4. Informatics and information technologies: textbook for the 9th grade of general secondary schools: M. R. Fayziyeva, D. M. Sayfurov, N. S. Khaitullayeva – Tashkent: Tasvir, 2020.
5. Eric Matthes. Python crash course: a hands-on, project-based introduction to programming. – San – Francisco: No starch Press, 2015.
6. K.Yu. Polyakov, V.M. Gurowitz. Python language in the school course of informatics - M.: MPEI Publishing House, 2011.
7. M. R. Fayziyeva, D. M. Sayfurov, N. S. Khaitullayeva. "Informatics and information technologies: a textbook for the 9th grade of general secondary schools": Tashkent - 2020.
8. [www.python.org](http://www.python.org)
9. [www.ziyonet.uz](http://www.ziyonet.uz)