

**A DEEPER UNDERSTANDING OF THE CONCEPT OF A FUNCTION IN A PYTHON PROGRAM****Gulhayo Panjiyeva***2nd year student of Chirchik State Pedagogical University***Mohinur Raupova***Teacher of Chirchik State Pedagogical University*

**Annotation.** *This article is about the concept of function in Python program, examples and problems related to it, and their solutions are considered in detail. In this article, we'll start with the basics of functions and learn how to use functions.*

**Keywords:** *print, return, function body, argument, parameter.*

A function is a reusable piece of software, that is, a sequence of instructions designed to perform a specific calculation. Functions allow you to save a certain block of commands with a specified name and execute this block anywhere in the program, any number of times [1]. A set of certain codes. Why do we use functions?

In the process of writing a program, we have to repeatedly write several lines of code. At this point, we can use functions, that is, we can put them in a single function, and then when we refer to the code, we can refer to this function by its name. Python has pre-existing functions that are used by all programmers, let's look at the "print" function for example: the print operator is called to print the result in the function body. For this, it is enough to include the print operator in the function body:

```
def sum_1
    a = int(input("a = "))
    b = int(input("b = "))
    sum_2=a+b;
    return sum_2;
    print(sum_1);
```

Another example of the functions we have is the int function. Its task is to convert the entered data into an integer. If this is not possible, it returns an error value.

Using functions to calculate mathematical functions in Python is very convenient. This is because the concepts of functions in Python and functions in mathematics are very close to each other. For this reason, we can also introduce mathematical problems and perform operations on them. For this, we tell the program to import math, so that it knows the mathematical functions. Then we can easily perform trigonometric and logarithmic examples:

```
import math
gradus = 45
radian =  $\frac{gradus}{360} * 2 * math.pi$ 
math.sin(radian)
>> 0.7071067811865476
```

Functions are the basic unit of work in python. A function in Python performs an operation and returns a result. Immutable functions are basic functions in python that can be used at any time by entering their name in the program field.

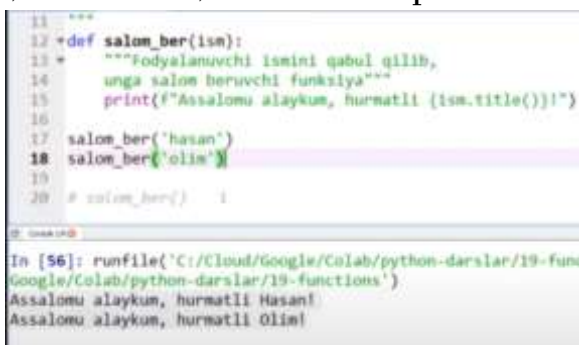
We only got acquainted with the functions available in all programs and used them, now we can create a new function ourselves. When creating a function, we proceed in the following sequence: Each function must be given a name, and this name begins with the keyword `def`, which is short for define in Python, followed by the name of the function, opening and closing parentheses. , we enter a parameter or list of parameters in parentheses. This first line is called the function header. After that, we create the body of the function. Variables and function conditions are given in it.

```
def add (x,y):
return x + y
```

It is required that the body portion of the function be written one letter lower than the header portion of the function. Both straightforward and intricate functions are possible. It has the capability of returning an object of any type, including a list, a tuple, or even a function from the list. During the process of constructing a function, once we have given the function a name, we may put the data for the function in three quote marks from the next line in order to make it more understandable to the user.

```
def salom_ber():
    """ Salom beruvchi funksiya """
    print( "Assalomu alaykum")
```

As of right now, the " salom\_ber " function will provide the output " Assalomu alaykum" when it is called at the console. In the event that we provide it with a variable value, for instance, when we also provide it with a name, it will greet us by name.



```
11 ***
12 *def salom_ber(ism):
13 *     """Fodiyalanuvchi ismini qabul qilib,
14 *     unga salom beruvchi funksiya"""
15 *     print(f"Assalomu alaykum, hurmatli {ism.title()}!")
16
17 salom_ber('hasan')
18 salom_ber('olin')
19
20 # salom_ber() 1
```

```
In [56]: runfile('C:/Cloud/Google/Colab/python-darslar/19-func
Google/Colab/python-darslar/19-functions')
Assalomu alaykum, hurmatli Hasan!
Assalomu alaykum, hurmatli Olin!
```

The execution of the function code is terminated when the return command is executed within the body of the function. The value that is stated right after the return command is the value that is returned as the result of the function.



A function may take any number of arguments, or none at all, depending on the context. In addition, there are functions that may take an arbitrary number of parameters, arguments that are positional and named, arguments that are necessary and required, and arguments that are optional. In the context of programming, a function parameter is a variable that stores a value that is provided through an argument [2].

The values that are provided are only altered within the function itself, and they do not have any impact on the values that are outside of the function. In many cases, a default value is assigned to the function argument. This means that if the function argument is explicitly specified, we make use of it by giving a value to it. When this is not the case, the default value is utilized instead. This is a default value; how do we input it? An equals sign, denoted by the symbol "=", is used to demarcate the default value from the parameter. A variety of parameters can be added to the function, as was mentioned before in this paragraph. Python function exercises may entail not just the usage of functions but also the use of primitives. Some of these exercises may contain a default value, while others may require the use of primitives. There are a great number of algorithms that have previously been researched and put into practice, despite the fact that the algorithms that are going to be implemented can be done without understanding the functions. For new students, however, it is of the utmost importance to comprehend and attain mastery of concepts such as function creation, call, parameter passing, return value, and others through the process of constructing functions in order to enhance their ability to reason logically [3].

```
def sum_1(a):
    sum_2=a+10;
    return sum_;
    a=int(input('a='));
    b=int(input('b='));
    print(sum_1(a));
    print(b);
```

So, let's look at the issues of creating a function and running it:

Example 1. Create a function called Daraja2 that calculates the rank of an arbitrary number. Create a program that calculates the degree of numbers a, b, c using the function Daraja2.

```
def Daraja2(a):
    a = a * a;
                                return a;

a = int(input('a = '));
b = int(input('b = '));
c = int(input('c = '));
print("a sonning darajasi = ",Daraja2(a));
print("b sonning darajasi = ",Daraja2(b));
```

`print("c sonning darajasi = ",Daraja2(c))`

Result:



Example 2. Create a function named `teskari_tartibda` that reverses the digits of a positive integer. Create a program that reverses the numbers a, b, c using this function.

`import math;`

`def teskari_tartibda(n):`

```

    while n > 0:
        i = n%10;
        n = math.floor(n/10);
        print(i, end = "");
    
```

`a = int(input('a = '));`

`b = int(input('b = '));`

`c = int(input('c = '));`

`teskari_tartibda(a);`

`print('\n');`

`teskari_tartibda(b);`

`print('\n');teskari_tartibda(c);`

Result:



When we talk about functions, the first thing that comes to mind is a set of certain codes. A function is a piece of software that can be used multiple times. Functions allow you to save a block of commands with a specified name and execute that block anywhere in the program, any number of times. It is reusable, improves reading and coding efficiency. A function is a block of code that only runs when it is called. In the



process of writing code, we have the option of using the function we create one or more times or not at all. We can save it under the name we want. We also learned how to pass parameters to functions and use the return keyword.

**REFERENCES:**

1. <https://fayllar.org/11-mavzu-python-da-funksiyalar.html>
2. <https://kompy.info/sh-a-mengliyev-o-a-abduganiyev-s-q-shonazarov-d-sh-torayev.html?page=53>
3. Deep Learning in Python Francois Chollet
4. Python programming language. Termiz-2021
5. TIOBE Index for March 2022- <https://www.tiobe.com/tiobe-index/>
6. Programming Language Trends - O'Reilly Radar". Radar.oreilly.com.
7. "The RedMonk Programming Language Rankings: tecosystems". Redmonk.com.
8. Kuhlman, Dave. "A Python Book: Beginning Python, Advanced Python, and Python Exercises". <https://stackoverflow.com>
9. PYPL Index. (2020).PYPL Popularity of Programming Language. Available at: <http://pypl.github.io/PYPL.html>.
10. Raupova, M. H., & Xasanova, M. F. (2024). PERFORMING COMPLEX MATHEMATICAL CALCULATIONS IN PYTHON USING THE SYMPY LIBRARY. *European Journal of Interdisciplinary Research and Development*, 25, 73-78.
11. Raupova, M. H. (2024). USING QUANTUM ALGORITHMS IN CRYPTOGRAPHY. *Web of Technology: Multidimensional Research Journal*, 2(3), 9-14.
12. Raupova, M. H., Xasanova, M. F., & Qurbonqulova, S. B. (2024). PYTHON DASTURLASH TILINING SYMPY VA SCIPY KUTUBXONALARIDAN FOYDALANGAN HOLDA INTEGRALGA DOIR MASALALARNI HAL QILISH. *PEDAGOG*, 7(2), 523-528.
13. Раупова, М. Х., & Урманов, Ш. М. (2024). ИЗМЕНЕНИЯ ПЕДАГОГИЧЕСКОГО ПРОЦЕССА В СОВРЕМЕННОЙ ШКОЛЕ: ВЫЗОВЫ И ПЕРСПЕКТИВЫ. *PEDAGOG*, 7(2), 515-522.