AMERICAN JOURNALS
Open Access,
Peer Reviewed Journals

American Journal of Technology and Applied Sciences
ISSN (E): 2832-1766 Volume 38, July - 2025

SIMPLIFY CODE THROUGH PART PROGRAMS

Gulmurodov Muhriddin Rahmatulla o'g'li
Bukhara State University Student 1-1KIDTM-23

ABSTRACT

The article provides information about the advantages of using part programs when creating a program. The usefulness of the procedure and function of application is covered in specific examples.

KEYWORDS

Python, programming, part program, procedure, function, optimal code.

Introduction

During the creation of a practical program, it is important to write clean and understandable code. Because the ease of reading the code is of great importance when the program is processed or additional input is made into it. One of the possibilities that makes it easier to read the program code is to use part programs. Part prevents programs from duplicating a piece of code and ensures that the program is understandable. In Python, the part program is declared through the word def servant. Part program is divided into 2uring the creation of a practical program, it is important to write clean and understandable code. Because the ease of reading the code is of great importance when the program is processed or additional input is made into it. One of the possibilities that makes it easier to read the program code is to use part programs. Part prevents programs from duplicating a piece of code and ensures that the program is understandable. In Python, the part program is declared through the word def servant. Part program is divided into 2. These are functions and procedures.

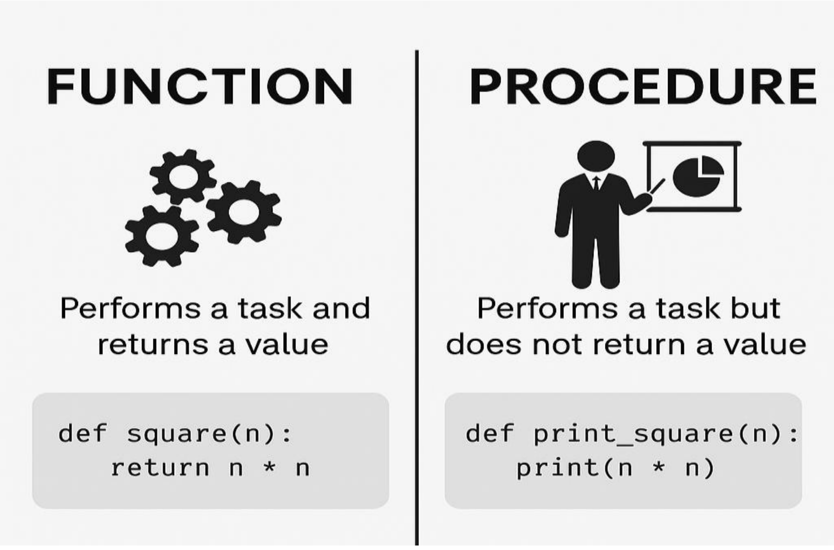
D.. Yu. We are not interested in what processes go inside it, since Federov likens the function to a baccalaureate, that is, a plastic card is inserted into it (with Pincode and sum), as a result of which the point is taken money, and so on that the ATM is insecure. [1] The function performs a similar function. That is, taking the value and returning a certain resu. Yu. We are not interested in what processes go inside it, since Federov likens the function to a baccalaureate, that is, a plastic card is inserted into it (with Pincode and sum), as a result of which the point is taken money, and so on that the ATM is insecure. [1] The function performs a similar function. That is, taking the value and returning a certain result.

According to the book "Язык программирования Python" written by G. Rossum and others, procedures are similar to functions but their output is of lesser significance [2].

Table 1. An example of a function and procedure

Part program	
Function	**Procedure**

def greater(a, b):
**   if a > b:**
**   return a**
**   else:**
**   return b**
 	
a = int(input("Enter first number: "))
b = int(input("Enter second number: "))
print("The greater number is", end=" ")
print(greater(a, b))
	def greater(a,b):
	    if a>b:
	        print(a)
	    else:
	        print(b)
	a=int(input("Enter first number: "))
	b=int(input("Enter second number: "))
	print("The greater number is", end=" ")
	greater(a,b)


FUNCTION
Performs a task and returns a value
PROCEDURE
Performs a task but does not return a value

```
def square(n):
    return n * n
```

```
def print_square(n):
    print(n * n)
```

Figure 1. Difference of function and procedure.

The main difference between the procedure and the function is that the function accepts a value and reproduces these values, returning the result to the main program through the word return. The procedure also processes them by taking values, but the result does not return to the main program.

Methods

Let's consider the practical implementation of the use of procedures and functions.

Let the file with information about students need a writer program. We carry it out without a procedure:

```
with open("students.txt", "a", encoding="utf-8") as f:
    f.write("Name: Ziyoda\n")
    f.write("Age: 18\n")
    f.write("Grade: 4.5\n")
    f.write("-" * 30 + "\n")

with open("students.txt", "a", encoding="utf-8") as f:
    f.write("Name: Bobur\n")
    f.write("Age: 20\n")
    f.write("Grade: 3.7\n")
    f.write("-" * 30 + "\n")

with open("students.txt", "a", encoding="utf-8") as f:
    f.write("Name: Asilbek\n")
    f.write("Age: 19\n")
    f.write("Grade: 5.0\n")
    f.write("-" * 30 + "\n")
```

We will observe the execution of this program using the function.

```
def write_to_file(name,age,grade):
    with open("students.txt", "a", encoding="utf-8") as f:
        f.write("Name: "+name+"\n")
        f.write("Age: "+age+"\n")
        f.write("Grade: "+grade+"\n")
        f.write("-" * 30 + "\n")

write_to_file("Ziyoda","18","4.5")
write_to_file("Bobur","20","3.7")
write_to_file("Asilbek","19","5")
```

Let's look at the program that finds more than 4 numbers[3]:

```
a = int(input("Enter 1st number: "))
b = int(input("Enter 2nd number: "))
c = int(input("Enter 3rd number: "))
d = int(input("Enter 4th number: "))

if a > b:
    k = a
else:
    k = b
```

```
if c > d:
    p = c
else:
    p = d
if k > p:
    print(k)
else:
    print(p)
```

We observe that this program is performed using the function

```
def write_to_file(name, age, grade):
    with open("students.txt", "a", encoding="utf-8") as f:
        f.write("Name: " + name + "\n")
        f.write("Age: " + age + "\n")
        f.write("Grade: " + grade + "\n")
        f.write("-" * 30 + "\n")

write_to_file("Ziyoda", "18", "4.5")
write_to_file("Bobur", "20", "3.7")
write_to_file("Asilbek", "19", "5")
```

Result

We will analyze the above codes. When writing information about students in a file, the procedure consisted of 15 lines when not used. And with the help of the procedure, the same code was executed in 9 rows. Now let this code need to be changed to itself. For example, let the information about the student need to be written in 1 row, not tagma-tad.e will analyze the above codes. When writing information about students in a file, the procedure consisted of 15 lines when not used. And with the help of the procedure, the same code was executed in 9 rows. Now let this code need to be changed to itself. For example, let the information about the student need to be written in 1 row, not tagma-tad. In this case, it is necessary to make changes to 6 lines of the program compiled without the procedure.and it is enough to change the 2 rows of the program compiled through the procedure.

A program that detects larger than 4 numbers can also be performed in several ways. if we analyze that the finished function was performed without using it, then in case 1, the program was executed in 16 rows, and using the function-in 10 rows. The number of rows in program 1 increases as the number of numbers that can also be compared to this program increases. We see the difference in programs in the table below:

Table 2 program that detects larger than 4 numbers can also be performed in several ways. if we analyze that the finished function was performed without using it, then in case 1, the program was executed in 16 rows, and using the function-in 10 rows. The number of rows in program 1 increases as the number of numbers that can also be compared to this program increases. We see the difference in programs in the table below:

Table 2. Advantage of the procedural program

Feature	Without procedure	With procedure
Number of code lines	15	9
Repeated code blocks	3	1
Readability	Low	Hight
Ease of modification	Difficult	Easy

Table 3. Utility of a functional program

Feature	Without procedure	With function
Number of code lines	16	10
Repeated code blocks	3	1
Readability	Low	Hight
Ease of modification	Difficult	Easy

Discussion

As you can see from the programs compiled above, with the help of part programs, it will be possible to break them into small and independent parts with an increase in the modularity of the compiled code. This in turn makes the code understandable and easily manageable. Prevents duplication of the same codes that are executed during the program. The fact that a function or procedure written once can be used repeatedly in different places of the program significantly reduces the size of the program. For example, by compiling a function that performs a mathematical calculation, it can be applied to the program in different places.

By correctly naming the function, it is possible to make the program much easier to read. For example, in a file, the task performed by this branch of the program is clearly displayed by naming exactly, such as_linking, numeric_linking. A properly named function also prevents over-annotation. If the error is only in one function, it is enough to fix it only in that function. This makes it much easier to detect and correct errors in large applications.y correctly naming the function, it is possible to make the program much easier to read. For example, in a file, the task performed by this branch of the program is clearly displayed by naming exactly, such as_linking, numeric_linking. A properly named function also prevents over-annotation. If the error is only in one function, it is enough to fix it only in that function. This makes it much easier to detect and correct errors in large applications. Since functions are independent, they can be tested separately. This increases the quality of the program and helps to detect errors early.

Another of the advantages of the function is that it hides its complex structures. for example, the sorted() function is widely used, but it is not important to know the complexity of its internal structure. By pre-preparing common tasks using functions, the process of writing an application is accelerated. In addition, in large projects, different programmers can write different functions. Since the functions are independent, each programmer focuses on his / her task, and the overall project is successfully unified.other of the advantages of the function is that it hides its complex structures. for example, the sorted() function is widely used, but it is not important to know the complexity of its internal structure. By pre-preparing common tasks using functions, the process of writing an application is accelerated. In addition, in large projects, different programmers can write different functions. Since the functions are independent, each programmer focuses on his / her task, and the overall project is successfully unified.

Conclusion

In the Python programming language, procedures and functions are important in effectively managing software projects, improving code quality, and simplifying the development process, supporting a modular programming paradigm. Allows you to divide the code into independent, reusable and easily understandable parts. Therefore, teaching programmers who are now learning programming to use them is an important skill. Due to Python's flexible and dynamic nature, functions and procedures allow programmers to simplify complex tasks, coordinate teamwork, and long-term project management. These features make Python one of the most important tools for beginner and professional programmers.

References

1. Федоров Д. Ю. Основы программирования на примере языка Python //Учебное пособие. – 2016.
2. Россум Г., Дрейк Ф. Л. Д., Откидач Д. С. Язык программирования Python //М.: Вильяме. – 2001.
3. Камалова Н. И. и др. МЕТОДИКА ОБУЧЕНИЯ ОПЕРАТОРУ IF В ЯЗЫКЕ ПРОГРАММИРОВАНИЯ PYTHON //International Conference on Economics, Finance, Banking and Management. – 2025. – С. 86-91.
4. Kamalova N. I. Methodology of teaching the subject of parameter repetition operator. – 2022.